Systems & Control: Foundations & Applications

Tamer Başar
Editor

Uncertainty in Complex Networked Systems

In Honor of Roberto Tempo

Birkhäuser

Birkhäuser

Systems & Control: Foundations & Applications

Series Editor
Tamer Başar, University of Illinois at Urbana-Champaign, Urbana, IL, USA

Editorial Board
Karl Johan Åström, Lund University of Technology, Lund, Sweden
Han-Fu Chen, Academia Sinica, Beijing, China
Bill Helton, University of California, San Diego, CA, USA
Alberto Isidori, Sapienza University of Rome, Rome, Italy
Miroslav Krstic, University of California, San Diego, CA, USA
H. Vincent Poor, Princeton University, Princeton, NJ, USA
Mete Soner, ETH Zürich, Zürich, Switzerland;
Swiss Finance Institute, Zürich, Switzerland

Former Editorial Board Member
Roberto Tempo, (1956–2017), CNR-IEIIT, Politecnico di Torino, Italy

More information about this series at http://www.springer.com/series/4895

Tamer Başar
Editor

Uncertainty in Complex Networked Systems

In Honor of Roberto Tempo

Birkhäuser

Editor
Tamer Başar
Department of Electrical and Computer
Engineering
University of Illinois at Urbana-Champaign
Urbana, IL, USA

ISSN 2324-9749 ISSN 2324-9757 (electronic)
Systems & Control: Foundations & Applications
ISBN 978-3-030-04629-3 ISBN 978-3-030-04630-9 (eBook)
https://doi.org/10.1007/978-3-030-04630-9

Library of Congress Control Number: 2018962123

© Springer Nature Switzerland AG 2018
This work is subject to copyright. All rights are reserved by the Publisher, whether the whole or part of the material is concerned, specifically the rights of translation, reprinting, reuse of illustrations, recitation, broadcasting, reproduction on microfilms or in any other physical way, and transmission or information storage and retrieval, electronic adaptation, computer software, or by similar or dissimilar methodology now known or hereafter developed.
The use of general descriptive names, registered names, trademarks, service marks, etc. in this publication does not imply, even in the absence of a specific statement, that such names are exempt from the relevant protective laws and regulations and therefore free for general use.
The publisher, the authors and the editors are safe to assume that the advice and information in this book are believed to be true and accurate at the date of publication. Neither the publisher nor the authors or the editors give a warranty, express or implied, with respect to the material contained herein or for any errors or omissions that may have been made. The publisher remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

This book is published under the imprint Birkhäuser, www.birkhauser-science.com by the registered company Springer Nature Switzerland AG
The registered company address is: Gewerbestrasse 11, 6330 Cham, Switzerland

Preface

This book on *Uncertainty in Complex Networked Systems* is a collection of chapters compiled in memory of Roberto Tempo, who was a member of the Editorial Advisory Board of these *Series* until his untimely death on January 14, 2017, during a skiing excursion at the Alps near his home town in Northwestern Italy. The volume consists of 17 chapters written by world experts on uncertainty in systems, robustness, networked and network systems, social networks, distributed and randomized algorithms, and multi-agent systems—topical areas Roberto Tempo has contributed to profusely during his prolific research career. A salient common feature of all the chapters is that, besides they all addressing the general broad field of complex systems, networks, and uncertainty, at least one author on each chapter was a research collaborator of Dr. Tempo.

Before describing briefly the contents of the chapters comprising this volume, I will provide a brief account of Roberto Tempo's life story. More details can be found in the obituary that appeared in volume 78, pages 341–342, of the IFAC journal *Automatica* in April 2017. He was the Editor-in-Chief of *Automatica* at the time of his death.

Roberto Tempo was born in Cuorgnè, Italy, in 1956. In 1980, he graduated in Electrical Engineering from Politecnico di Torino, Italy. After a period spent at Politecnico di Torino, he joined the National Research Council of Italy (CNR) at the research institute IEIIT, Torino, where he had been a Director of Research of Systems and Computer Engineering since 1991. He held visiting and research positions at Tsinghua University and the Chinese Academy of Sciences in Beijing, Kyoto University, The University of Tokyo, University of Illinois at Urbana-Champaign, German Aerospace Research Organization in Oberpfaffenhofen, and Columbia University in New York. He was elected a Fellow of the IEEE (2000), a Distinguished Member of the IEEE Control Systems Society (2005), a Fellow of IFAC (2007), and a Corresponding Member of the Academy of Sciences, Institute of Bologna, Italy, Class Engineering Sciences (2011). He served as President of the IEEE Control Systems Society (2010), as General Co-Chair for the IEEE Conference on Decision and Control, Florence, Italy (2013), as Program Chair of the first joint IEEE Conference on Decision and Control and European Control

Conference, Seville, Spain (2005), as Editor for Technical Notes and Correspondence of the IEEE *Transactions on Automatic Control* (2005–2009), and as a Senior Editor of the same journal (2011–2014). He had a long association with *Automatica*, starting in 1992 as an Associate Editor, then (from 1996) as the Editor of the subject area "System and Control Theory", and for 11 years as Deputy Editor-in-Chief, before becoming Editor-in-Chief in 2015.

Roberto Tempo's research activities were initially focused on the analysis and design of complex systems subject to uncertainty. He pioneered the development of randomized algorithms for robust control, generating in this topical area trend-setting papers which appeared in international journals, books, and conferences, culminating in the publication, with co-authors, of the monograph *Randomized Algorithms for Analysis and Control of Uncertain Systems*, Springer, London, which appeared in two editions in 2005 and 2013, and became a standard reference in the field. His research then evolved toward the analysis and control of complex networked uncertain systems. In this area, he contributed to a number of important areas of application, among which the most relevant and prominent were his work on algorithms for PageRank computation in the Google search engine, and distributed localization of wireless sensor networks. More recently, he had focused his research on algorithmic approaches toward understanding how individuals in a group influence each other to reach a consensus—an activity that culminated in the publication of a *Science* article. Over the years, several of his publications received paper awards, including the "IEEE Control Systems Magazine Outstanding Paper Award" for the 2014 paper "The PageRank Problem, Multi-agent Consensus and Web Aggregation: A Systems and Control Viewpoint", and an "Automatica Outstanding Paper Prize Award" for the 1990 paper "The Robust Root Locus".

Now, coming back to the contents of this volume, the 17 chapters comprising this volume have been organized into three parts: Robustness (Part I), Randomization and Probabilistic Methods (Part II), and Distributed Systems and Algorithms (Part III).

Part I is comprised of four chapters. The first chapter, titled "Uncertain Systems: Time-Varying Versus Time-Invariant Uncertainties" by F. Blanchini and P. Colaneri, provides a survey of several decades of robustness investigation for uncertain systems with a critical view. The second chapter, titled "Cooperative Resilient Estimation of Uncertain Systems Subjected to a Biasing Interference" by V. Ugrinovskii, surveys some recent results on the analysis and design of networks of robust filters which cooperate to produce high fidelity estimates for uncertain plants, with application to the problem of detecting and neutralizing biasing attacks on distributed observer networks. The third chapter, titled "Robust Static Output Feedback Design with Deterministic and Probabilistic Certificates" by D. Arzelier, F. Dabbene, S. Formentin, D. Peaucelle, and L. Zaccarian, addresses, using a new bilinear matrix inequality (BMI) formulation, the problem of static output design for uncertain linear systems by iterative optimization procedures with either deterministic or probabilistic viewpoints, exploiting the fact that Lyapunov certificates are separated from the control gain design variables. The fourth, and last, chapter of this part, titled "Robust Control Against Uncertainty Quartet: A Polynomial Approach" by D. Zhao, C. Chen, S. Z. Khong, and L. Qiu, introduces a unifying framework to address different types of uncertainty in systems modeling and control, the uncertainty quartet, which combines the additive, multiplicative, subtractive and divisive uncertainties, and in this framework it develops an elementary robust control theory, involving mostly polynomial manipulations.

Part II of the volume is comprised of six chapters. The first one, titled "Randomization in Robustness, Estimation, and Optimization" by B. Polyak and P. Shcherbakov, addresses the question of when a random choice (in various decision-making scenarios, such as optimization) would be better than a deterministic one, and provides a survey of some recent results in this domain. The second one, titled "Stabilization of Deterministic Control Systems Under Random Sampling: Overview and Recent Developments" by A. Tanwani, D. Chatterjee, and D. Liberzon, addresses the problem of stabilizing continuous-time deterministic control systems via a sample-and-hold scheme under random sampling using the framework of piecewise deterministic Markov processes. The third one, titled "Robust Design Through Probabilistic Maximization" by T. Alamo, J. M. Manzano, and E. F. Camacho, studies randomized maximization methods for robust design under uncertainty, providing in this context concentration inequalities that lead to probabilistic guarantees on the obtained design parameters. The fourth one, titled "Compressive Sensing and Algebraic Coding: Connections and Challenges" by M. Vidyasagar and M. Lotfi, provides an overview of known results on compressive sensing using both probabilistic and deterministic approaches, followed by some new constructions of sparse binary measurement matrices, based on LDPC (low density parity check) codes, and a description of the authors' selected recent results that lead to the fastest available algorithms for compressive sensing in specific situations. The fifth chapter of this part, titled "Stochastic Optimization for Energy Storage Allocation in Smart Grids in the Presence of Uncertainty" by M. Bucciarelli, S. Paoletti, and A. Vicino, deals with the application area of smart grids, focusing on the problem of optimal siting and sizing of energy storage systems in a distribution network. The sixth, and last, one of this part, titled "A Data-Driven Basis Function Approach in Nonparametric Nonlinear System Identification" by E.-W. Bai and C. Cheng, proposes a data driven orthogonal basis function approach for non-parametric finite impulse response (FIR) nonlinear system identification, where the basis functions are not fixed *a priori* and match the structure of the unknown system automatically.

The last part of the volume, Part III, is comprised of seven chapters. The first one, titled "Perspectives on Network Systems and Mathematical Sociology" by F. Bullo and N. E. Friedkin, provides an overview of a large group of literature on the mathematics of network systems and its application to the study of dynamical models for the evolution of opinions and influence systems, presenting both mathematical results and empirical findings. The second one, titled "Distributed Randomized Algorithms for PageRank Computation: Recent Advances" by H. Ishii and A. Suzuki, provides an overview of recent studies on distributed algorithms for PageRank computation that have been developed in the systems control area, and following that introduces a new class of distributed algorithms based on a simple but novel interpretation, further demonstrating its advantages over existing ones via analysis and numerical simulations. The third one, titled "Distributed Optimization in Multi-agent Networks Using One-bit of Relative State Information" by J. Zhang and K. You, is concerned with the design of distributed discrete-time algorithms to cooperatively solve an additive cost optimization problem in multiagent networks, with the striking feature that the algorithms use only the sign of relative state information between neighbors. The fourth one, titled "Analysis of a Distributed Consensus Based Economic Dispatch Algorithm" by R. Mudumbai, S. Dasgupta, and M. M. M. U. Rahman, presents a consensus-based approach to the optimal economic dispatch of power generators in a smart microgrid, where the generators independently adjust their power-frequency primary controller setpoints using three pieces of information as delineated in the chapter. The fifth one, titled "Impact of Quantized Inter-agent Communications on Game-Theoretic and Distributed Optimization Algorithms" by E. Nekouei, T. Alpcan, and R. J. Evans, addresses the issue of handling the impact of the uncertainty that is generated by quantized inter-agent communications in game-theoretic and distributed optimization algorithms, and uses the information-theoretic notion of differential entropy power to establish universal bounds on the maximum exponential convergence rates of primal-dual and gradient-based Nash seeking algorithms under quantized communications. The sixth one, titled "Fault Diagnosis for Uncertain Networked Systems" by F. Boem, C. Keliris, T. Parisini, and M. M. Polycarpou, provides an overview of results on a model-based distributed fault diagnosis approach to uncertain nonlinear large-scale networked systems to specifically address the presence of measurement noise, modeling uncertainty, and the presence of delays and packet dropouts when viewed as a networked system. The seventh, and last, one, titled "Networked Quantum Systems" by I. R. Petersen, considers the modelling and realization of quantum networks from a control theory point of view, focusing particularly on quantum linear systems.

I thank all authors referenced above for their contributions to this book, where each chapter has maintained a wonderful balance between being expository and providing new results and identifying fruitful future directions in research—all on topics that were dear to Roberto. I am confident that the book will prove to be a high-demand reference volume to a broad community of researchers interested in uncertainty, complexity, robustness, optimization, algorithms, and networked systems, for many years to come—as a real tribute to the memory of Roberto Tempo.

Urbana, USA
September 2018

Tamer Başar

Contents

Part I Robustness

Part II Randomization and Probabilistic Methods

Part III Distributed Systems and Algorithms

Part I
Robustness

Uncertain Systems: Time-Varying Versus Time-Invariant Uncertainties

Franco Blanchini and Patrizio Colaneri

Abstract In this chapter, we survey a few decades of robustness investigation for uncertain systems. We aim at embracing most of the robustness literature, starting from the Lyapunov approach of the '70s, which involved both quadratic and non-quadratic Lyapunov functions, until recent developments on polynomial techniques for robustness. We consider both time-varying and time-invariant uncertainties, in an inclusive way: important techniques are presented, such as the Lur'e systems framework, qualitative feedback theory, parametric robustness analysis, linear matrix inequalities, parameter-dependent Lyapunov functions, H-infinity, small-gain theorems, non-quadratic Lyapunov functions and Lyapunov–Metzler inequalities. The chapter proposes a critical view on all these techniques, highlighting both advantages and limitations. Illustrative examples and applications are proposed. Technicalities are kept to the least possible level to render the chapter accessible to a broad, possibly interdisciplinary, audience. The chapter is written with a historic view. Nonetheless, future perspectives are emphasized, and several open problems and future research directions are pointed out. The chapter is inspired by the spirit, attitude and fairness of our great friend Roberto Tempo and is written following his invaluable teaching.

F. Blanchini
Dipartimento di Scienze Mathematiche, Informatiche e Fisiche, Università di Udine,
33100 Udine, Italy
e-mail: blanchini@uniud.it

P. Colaneri (✉)
Dipartimento di Elettronica, Informazione e Bioingeneria, Politecnico di Milano,
20133 Milano, Italy
e-mail: patrizio.colaneri@polimi.it

P. Colaneri
CNR-IEIIT, Turin, Italy

© Springer Nature Switzerland AG 2018

T. Başar (ed.), *Uncertainty in Complex Networked Systems*,
Systems & Control: Foundations & Applications,
https://doi.org/10.1007/978-3-030-04630-9_1

1 Introduction

In this chapter, we propose a survey of several fundamental concepts in robustness theory for control systems. Although it is impossible to be exhaustive, we wish to propose to the reader a brief journey in the different approaches coping with the analysis and control design of uncertain systems.

Quite unusually, we do not follow a specific philosophy, but our main effort is to present several approaches, ranging from the classical Lyapunov methods for robustness [18, 48, 101], to the frequency-domain approach [105, 121] and the parametric approach [7]. The main conclusion we draw is that there is no a best one, but all these theories reveal strength as well as weakness, depending on the type of model we are considering, in particular linear or nonlinear, on the type of uncertainty, constant or time-varying, and on the type of goal, stability or optimality.

The chapter has been thought to be a guide for further readings, so not too many details are reported. The idea is to give the main flavour of a field that is so massive in its scientific results that it would occupy the space of an encyclopedia rather than a book. For more formal results, proofs and examples, the reader is referred to the mentioned literature, which is by no means complete, and we sincerely apologize with many Authors, since we have been forced to limit the bibliography to a reasonable extent.

The reason why the robust control area has been, is and will be so prolific is twofold. First of all, robustness in control theory is a must. No reasonably designed control system can fail to be robust. Second, no other area in science has been so long concerned with robustness. It is absolutely true that nowadays, ‘robustness’ is a common keyword in several disciplines, including (beside control and dynamical systems) optimization, computer science, systems biology, game theory, management statistics, but ‘we control theoreticians’ still have the required expertise to be leaders in the topic. This means that many problems coming from other fields have found in our conferences and journals the proper venues to be fruitfully discussed.

This survey chapter follows the survey paper [98], co-authored by Roberto Tempo, recently passed away. We propose it to the community in memory of our great friend and scientist, having in mind his attitude in communicating as well in listening. So any comments or concerns regarding the contents of the chapter will be gratefully appreciated and taken into account in future work.

1.1 A General View of Robustness

The term robustness is deeply known in control theory since any real system is affected by uncertainties. Uncertainties may be of different nature and they can be essentially divided into the following categories:

- Unpredictable events.
- Unmodelled dynamics.

- Unavailable data.

Unpredictable events are typically due to factors that perturb the system and depend on the external environment (e.g. the air turbulence for an aircraft). Unmodelled dynamics is due to the unavoidable simplifications that are needed to represent a system with a tractable model (for instance, if we consider the air pressure inside a plenum in dynamic conditions, we often neglect the fact that the pressure is not in general uniform in space). Unavailable data is a very frequent problem in practice since in many cases some quantities are known only when the system operates (e.g. how much weight will be carried by a lift?).

Therefore, during the design stage, we cannot consider a single system but a family of systems. Formally the concept of robustness can be stated as follows.

Definition 1 A property $\mathscr{P}$ is said to be **robust** for the family $\mathscr{F}$ of dynamic systems if any member of $\mathscr{F}$ satisfies $\mathscr{P}$.

The family $\mathscr{F}$ and the property $\mathscr{P}$ must be properly specified. For instance, if $\mathscr{P}$ is 'stability' and $\mathscr{F}$ is a family of systems with uncertain parameters ranging in a set, we have to specify if these parameters are constant or time-varying. In the context of robustness the family $\mathscr{F}$ represents the uncertainty in the knowledge of the system. There are basically two categories of uncertainties. Precisely:

Parametric uncertainties: dealing with a class of models depending upon parameters which are unknown; in this case the typical available information is given by bounds on these parameters.

Non-parametric uncertainties: dealing with systems in which some of the components are not modelled; the typical available information is provided in terms of the input–output-induced norm of some operator.

1.2 A Brief History

Design for uncertain systems is a very wide topic in control theory, and robust control has always been a main paradigm in the analysis and design of a control system. The main motivation that spurred the research activity was the study of performances over finite and bounded variation of parameters, whereas in classical multivariable control the designer was only able to ensure robustness at the face of small parameter variations. Indeed, quantitative opposite to qualitative robustness is a watershed between classical and modern design methodologies.

A mainstream of research for quantitative robustness is the so-called H_∞ control theory, which dominated the scenario starting from the '80s, being able to bridge classical frequency-domain and state-space techniques in an elegant unified mathematical framework. This is certainly one of the most interesting historical merits of the H_∞ approach, whose versatile nature permitted to incorporate in the same mathematical framework historically different problems such as filtering, factorization, interpolation and conjugation. The idea underlying the H_∞ theory is rather simple: minimize a worst-case measure of the input–output map between disturbances and performance variables. When reduced to robustness of stability, this worst-case paradigm leads to the celebrated small-gain theorem due to George Zames [118], who was also the first who proposed an input–output setting to the theory in the pioneering papers [117, 119]. At those times, two main technical paths were undertaken: the Nevanlinna–Pick interpolation technique [47, 89, 99] and the AAK method [1], mainly based on the Nehari extension problem [60] and on the unitary dilation technique in operator theory [42]. The interpolation theory was originally part of the circuit theory [67, 116], and only in later years became the object of investigation by control theorists, for the solution of the disturbance reduction problem [31] and the robust stabilization problem [61, 77]. The state-space counterpart of the interpolation theory was worked-out in [78] via the notion of J-lossless conjugation, where the role of the Pick matrix was translated in terms of the solution of a Riccati equation. Successive developments of the H_∞ control theory were in the frameworks of the so-called J-spectral factorization approach [63] and chain-scattering representation of the plant [115]. Also the almost disturbance decoupling problem [78], which has an extensive literature behind it and is an important problem per se, was cast in the general H_∞ formulation. In the late 1980s the time became mature for the development of a state-space technique for the solution of the general multi-input multi-output H_∞ control problem [44]. The main drawback was that it was required to compute the solution of a high-order Riccati equation. This difficulty was removed later, and this was due to many contributors [45]. In [83], a comprehensive picture is traced and a complete solution is given of the robust and perfect tracking problem. The connections between differential games and H_∞ theory can be found in the book by Basar and Bernhard [11]. The robust stabilization problem via quadratic functions has been deeply investigated by Barmish [6], Khargonekar et al. [75] and Haddad and Bernstein [66]. The design of H_2 filters in the presence of uncertainties has been studied by Petersen and McFarlane [97] and Bolzern et al. [27], whereas the same problem in the H_∞ context was tackled by De Souza et al. [43] and Fu et al. [49]. The paper by Safonov and Limebeer [104] provides the so-called *loop shifting* approach for the solution of the H_∞ problem for plants with general structure. The H_∞ control problem was dealt with in many books after the monograph of Francis [46], see, e.g. [33, 39, 62, 109, 110], where the so-called singular problem is dealt with.

Hereto neglected approaches which rely on the gap metric and the polynomial framework are exploited in the paper by Georgiou and Smith [50] and by Kwakernaak [79].

A success of H_∞ control theory is in the easy way the various specifications on the closed-loop system that can be incorporated through appropriate shaping functions which reflect the desired dynamic behaviour [84]. Notably, a posteriori it was seen that, in the state-space context, the theory has some structural similarities with the classical LQG theory, the latter being recovered in the case where a design parameter (the so-called attenuation level) goes to infinity. It is a fact that the arguments underlying H_∞ control and related problems constitute a solid scientific background for the new researchers entering the field. We can count hundreds of papers, many books and regular university courses at all levels in this subject. Interestingly, the success of H_∞ control in a number of important applications contributes to reduce the historical gap between theory and application, despite the significant inherent mathematical sophistication required to understand the underlying theory.

An important aspect concerns the solution of robust problems via convex optimization. The various aspects of convex analysis, from the basic facts to most important and deep results, are included in the seminal book by Rockafellar [103]. The book [29] exhaustively treats the most important topics related to systems and control theory in the framework of linear matrix inequalities so that the problems to be handled are convex. In addition, this book is also an important source of references to those interested in a deeper view of optimal control design using convex programming techniques. Important papers for design problems of uncertain systems are [13, 57, 74, 91, 94] with connections to the results of [12] dealing with Riccati equation approach and Nash game for mixed H_2–H_∞ problems [82]. The importance of convex optimization for design of uncertain systems via linear matrix inequalities (LMI) and sum of squares (SOS) programming is difficult to underestimate, see the important monograph by G. Chesi [34] and the Ph.D. thesis of P. Parrillo [90]. In recent years, C. Scherer gave a push forward to the use of convex tools for robust design problems. In the paper [107], the classical μ-synthesis tools are generalized to the integral quadratic constraint (IQC) framework, enabling to perform robust controller synthesis for a significantly large class of uncertainties, like sector-bounded and slope restricted nonlinearities, time-varying parametric uncertainties and uncertain time-varying time delays, both with bounds on the rate-of-variation.

Lyapunov approach to robustness has played a major role. It started with the seminal work of the late ’70s [65, 69, 80]. It was initially shown to be effective for some classes of systems, in particular the mechanical ones satisfying the so-called matching conditions. Later on, the investigation of quadratic stability [5, 9] of uncertain systems became a popular subject of research. The concept of quadratic stability had intense years of considerations as long as its deep relationship with H_∞ theory was discovered [75]. Subsequently, the latter was substantially preferred to the classical theory based on Lyapunov quadratic functions. Quadratic Lyapunov functions are known to be conservative. In simple words, there exist Lyapunov stable uncertain systems for which no quadratic functions can be found. This was quite clear in both the Russian literature [85–87] and the western literature [30]. Conversely, there are other classes, for instance, the polyhedral Lyapunov functions, which provide necessary and sufficient conditions for stability [85] and stabilizability [15] of uncertain linear systems.

Concerning stability by means of linear state-feedback control, the paper of Barmish [5] is important since; for the first time, the author proposes an effective and simple way to handle uncertainties acting on both the state and input system matrices. The notion of guaranteed cost has been introduced by Chang and Peng in [32] related to a simple LQ problem. The main idea was to get an (nonlinear) upper bound to the solution of the associated Riccati equation. They have been solved so as to cope with parameter uncertainties. Two of the most important classes of uncertainty have been considered and compared, namely, polyhedral convex bounded and norm bounded uncertainty. For the first type, results have been reported in [92, 120]. With this last paper, the reader can go deeper into the comparison of these two types of parameter uncertainty models just mentioned. H_2 and H_∞ guaranteed cost control problems have been introduced in [58, 93], respectively. The stability and guaranteed cost control of dynamic linear systems subject to actuators failure has been analysed in [54]. Once again, the convexity plays a central role, and it is possible to verify that the uncertainty description by means of a convex domain leads in many instances to better results. For nonlinear uncertainties, we will address the so-called Persidiskii design, based on papers [51, 53, 73]. The former paper also provides many other and more general results and is an excellent reference on this topic. Another control design procedure is called Lur'e design, which is based on the classical results reported in the important book [114], where the notions of passivity and strictly positive real transfer functions are addressed in a general and complete setting.

Parametric approach to robustness is also a traditional topic [70, 71]. Basic concepts such as the frequency analysis based on value set were well established when, suddenly, the parametric approach in robustness had a scientific explosion with the famous Kharitonov theorem [76]. Subsequently, the edge theorem [4] and extensions of traditional tools like the root-locus approach [8], reinforced the attractiveness of the subject. For several years, it was among the most favourite lines of research in the control community. We refer to the book [7] and the tutorial manuscript [112] for references. Recently, the research activity has been diverted to a stochastic approach to deal with uncertain systems and optimization problems. The so-called *randomized approach* has its seeds in the monograph [113], where Roberto Tempo and co-authors lay the foundations of probabilistic methods in the analysis and design of systems affected by deterministic and stochastic uncertainties. Although the parametric approach is today considered mature enough, still it has several potential interesting applications.

2 Examples and Motivations

In this section, we discuss several examples of different nature, to motivate the theory.

2.1 Magnetic Levitation System

Consider the magnetic levitator system depicted in Fig. 1. A commonly accepted model for this system is

$$\begin{cases} \dot{x}_1(t) = x_2(t) \\ \dot{x}_2(t) = g - \frac{k}{m}\frac{x_3^2(t)}{x_1^2(t)} \\ \dot{x}_3(t) = -\frac{R}{L}x_3(t) + \frac{1}{L}u(t) \end{cases}$$

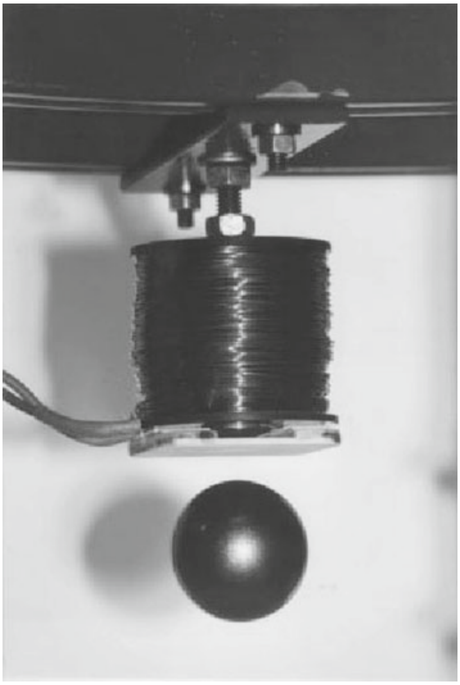


Fig. 1 Magnetic levitator

where x_1 is the distance of the sphere from the magnet (oriented downwards), x_2 is its speed and x_3 is the current in the coil. The control input u is the voltage supplied by an amplifier. The following constants are involved: g is the gravity, R is the coil resistance, L is the inductance, m is the ball mass and k is the levitator constant. Note that the electromagnetic force is assumed to obey the relation

$$f_{EM} = k\frac{x_3^2}{x_1^2}$$

which is a rough approximation of the true situation.

Denoting by $\bar{x}_1 = \xi$ the desired equilibrium value of the position, the equilibrium value of the current is $\bar{x}_3 = \sqrt{mg/k}\xi$ and the equilibrium value of the input voltage is $\bar{u} = R\sqrt{mg/k}\xi$. The equilibrium speed is clearly $\bar{x}_2 = 0$. The linearized system has equations

$$\begin{bmatrix} \dot{z}_1(t) \\ \dot{z}_2(t) \\ \dot{z}_3(t) \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 \\ 2\frac{k}{m}\frac{\bar{x}_3^2}{\bar{x}_1^3} & 0 & -2\frac{k}{m}\frac{\bar{x}_3}{\bar{x}_1^2} \\ 0 & 0 & -\frac{R}{L} \end{bmatrix} \begin{bmatrix} z_1(t) \\ z_2(t) \\ z_3(t) \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ \frac{1}{L} \end{bmatrix} v(t) \quad y = \begin{bmatrix} 1 & 0 & 0 \end{bmatrix} \begin{bmatrix} z_1(t) \\ z_2(t) \\ z_3(t) \end{bmatrix}$$

where z is the translated state variable vector. The transfer function of this system is

$$F(s) = \frac{-c}{(s^2 - a)(s + b)}$$

where

$$a = 2\frac{k}{m}\frac{\bar{x}_3^2}{\bar{x}_1^3}, \quad b = \frac{R}{L}, \quad c = 2\frac{k}{m}\frac{\bar{x}_3}{\bar{x}_1^2}\frac{1}{L}$$

Now the issue is that these parameters are highly uncertain. The essential reasons are that:

- the expression adopted for f_{EM} is approximated;
- k is roughly known even at the equilibrium;
- there is a strong dependence upon the desired position ξ;
- the resistance (affecting b) depends on the temperature.

By means of experiments, performed in the Dynamic System Laboratory in Udine, it has been seen that the range of uncertainty for the parameters are

$$a = 1400 \pm 400, \quad b = 350 \pm 10, \quad c = 1500 \pm 500$$

If we ignore the uncertainties and we consider the nominal values $a = 1400$, $b = 350$ and $c = 1500$, then the following compensator

$$v(s) = G(s)y(s)$$

with

$$G(s) = \kappa\frac{s + \beta}{s + \alpha}$$

and $\kappa = 3000$, $\alpha = 50$, $\beta = 5$ stabilizes the system. Now the question is

- is stability preserved for all possible parameter values?

For the moment we just observe that the parameters can be regarded as uncertain but constant (once the equilibrium value has been fixed). Note also that the system works properly also for slow variations of the parameters and therefore the state remains close to the nominal point. It is legitimate to consider this as *an uncertain system with constant parameters*.

2.2 Inverted Pendulum

Consider the cart pendulum system schematically illustrated in Fig. 2. Denoting by y the cart position and θ the pendulum angle, for small values of the angle, the equations are

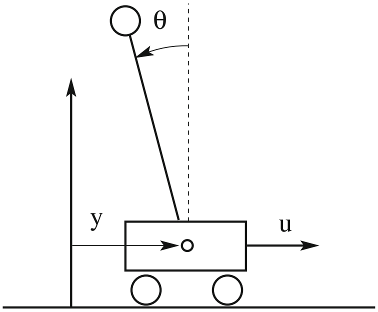


Fig. 2 Cart pendulum

$$M\ddot{y}(t) = -\alpha\dot{y}(t) + u(t)$$
$$ml\ddot{\theta}(t) = mg\theta(t) + m\ddot{y}(t)$$

where M is the cart mass, m is the pendulum top mass (the pole mass is negligible), g is the gravity, l is the length of the pendulum, and u is the applied force. The parameter α is, in principle, just a friction coefficient. The equations are valid for small angles which allow for the approximations $\sin(\theta) \approx \theta$ and $\cos(\theta) \approx 1$. Moreover, the considered experimental set-up is such that the cart mass is much bigger than the mass on the pole top and this is why no pole effects are in the equation of the cart. We now manipulate the second equation replacing the term $M\ddot{y}(t)$ from the first as follows:

$$ml\ddot{\theta}(t) = mg\theta(t) + \frac{m}{M}M\ddot{y}(t) = mg\theta(t) - \frac{m}{M}\alpha(t)\dot{y}(t) + \frac{m}{M}u(t)$$

Let $\beta = \frac{1}{Ml}$, $\gamma = \frac{g}{l}$ and $\delta = \frac{1}{M}$. Further let $x_1 = \theta$, $x_2 = \dot{\theta}$, $x_3 = y$, $x_4 = \dot{y}$, to get

$$\begin{bmatrix} \dot{x}_1(t) \\ \dot{x}_2(t) \\ \dot{x}_3(t) \\ \dot{x}_4(t) \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 & 0 \\ \gamma & 0 & 0 & -\beta\alpha(t) \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & -\delta\alpha(t) \end{bmatrix} \begin{bmatrix} x_1(t) \\ x_2(t) \\ x_3(t) \\ x_4(t) \end{bmatrix} + \begin{bmatrix} 0 \\ \beta \\ 0 \\ \delta \end{bmatrix} u(t)$$

Now we have to face the main problem: the friction α. The friction varies continually passing from the static to dynamic regime and it abruptly changes its value. There are many investigations of this phenomenon with many approximated models. Here, we assume that

$$\alpha^- \le \alpha(t) \le \alpha^+$$

for some given bounds $0 < \alpha^- < \alpha^+$, and we assume that $\alpha(t)$ can have any functional dependence on t that we ignore. Considering α uncertain but constant would be a major mistake, because the switch static/dynamic friction occurs close to the equilibrium. Note also that it is virtually impossible to measure the value of $\alpha(t)$ online. So a scheduling procedure for control design would be infeasible.

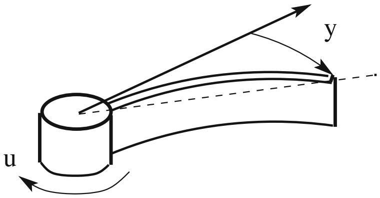


Fig. 3 Flexible beam

2.3 Flexible Systems with Parasitic Dynamics

Consider the problem of controlling a flexible beam such as that illustrated in Fig. 3. A torque u is applied to the flywheel, and the problem is to control the angle of the top of the beam. In principle this system is infinite-dimensional. Its transfer function would be of the form

$$\frac{y(s)}{u(s)} = \sum_{k=0}^{\infty} \frac{Q_k}{s^2 + 2\xi_k s + (\xi_k^2 + \omega_k^2)}$$

achieved by considering the contribution of all flexible modes. Considering this model would be most impractical. The typical realistic approximation is to consider a finite number of modes, say N. The function is then written in the form

$$\frac{y(s)}{u(s)} = \sum_{k=0}^{N} \frac{Q_k}{s^2 + 2\xi_k s + (\xi_k^2 + \omega_k^2)} + \Delta_N(s) = C_N(sI - A_N)^{-1}B_N + \Delta_N(s)$$

where (A_N, B_N, C_N), is a finite-dimensional realization of order $n = 2N$ and $\Delta_N(s)$ the frequency-dependent approximation. This choice is motivated also by the fact that the low frequency modes are typically faithfully represented, while the high frequency ones are not. Moreover, high-order nominal models lead to high-order nominal compensators.

Now the issue is to achieve bounds for $\Delta_N(s)$. This is typically done by assuming a bound of the form

$$|\Delta_N(j\omega)| \le \phi(\omega)$$

on the frequency response magnitude, determined by means of experimental data. This is a *dynamic uncertainty*, accounting for the unmodelled system dynamics. For instance, retaining two modes only, we have the nominal transfer function

$$F(s) = \frac{Q_0}{s^2} + \frac{Q_1}{s^2 + 2\xi_1 s + (\xi_1^2 + \omega_1^2)}$$

(we assume that the shaft friction is negligible). Then the nominal model would be

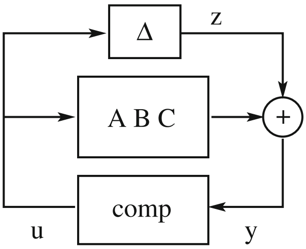


Fig. 4 Control diagram

$$\begin{bmatrix} \dot{x}_1(t) \\ \dot{x}_2(t) \\ \dot{x}_3(t) \\ \dot{x}_4(t) \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & -\xi_1 & \omega_1 \\ 0 & 0 & -\omega_1 & -\xi_1 \end{bmatrix} \begin{bmatrix} x_1(t) \\ x_2(t) \\ x_3(t) \\ x_4(t) \end{bmatrix} + \begin{bmatrix} 0 \\ 1 \\ 0 \\ 1 \end{bmatrix} u(t)$$

and

$$y(t) = \begin{bmatrix} Q_0 & 0 & \frac{Q_1}{\omega_1} & 0 \end{bmatrix} \begin{bmatrix} x_1(t) \\ x_2(t) \\ x_3(t) \\ x_4(t) \end{bmatrix} + \varDelta \circ u$$

This model can then be used for control synthesis. If we adopt a controller based on LQG or H_∞ theories, we end up with a fourth-order compensator. The compensator must be designed taking into account the presence of $\varDelta$. The overall paradigm corresponds to the diagram in Fig. 4.

In designing the compensator, based on the nominal model, we must make sure that the control input action is not too strong, because this could potentially excite the neglected dynamics $\varDelta$, causing instability.

If we consider the nominal part only (i.e. $\varDelta = 0$) the input output relation from z to u is

$$z(s) = \frac{K(s)}{1 - K(s)G(s)} u(s) = V(s)u(s)$$

where $G(s) = C(sI - A)^{-1}B$ and $K(s)$ are the plant and compensator transfer functions. The presence of $\varDelta$ implies that this transfer function is actually in a loop with

$$u(s) = \varDelta(s)z(s)$$

Since $\varDelta$ is assumed unknown, we cannot design this loop. Only a bound of the form $\|\varDelta\| \leq \delta$ is reasonably available. A robust strategy can be found by keeping the gain of the transfer function V small enough, precisely $\|V\|\delta < 1$, in such a way that this second loop does not destroy stability. This is the small-gain principle which will be discussed in detail later on.

2.4 Robust Control of an Engine Test Bench

The engine test bench system is illustrated in Fig. 5. There are important components:

- Dynamometer.
- Connection shaft.
- Combustion engine.

The task of the test bench control problem is to stabilize the engine torque and the engine speed. Considering the torque of the combustion engine and the air gap torque of the dynamometer as the inputs to the mechanical part of the engine test bench system, the overall system can be described as a two-mass oscillator.

$$\begin{aligned}
\Delta\dot{\varphi} &= \omega_E - \omega_D \\
\dot{\omega}_E &= \frac{1}{\theta_E}\left(T_E - c\Delta\varphi - d\left(\omega_E - \omega_D\right)\right) \\
\dot{\omega}_D &= \frac{1}{\theta_D}\left(c\Delta\varphi + d\left(\omega_E - \omega_D\right) - T_{DSet}\right)
\end{aligned}$$

where θ_E is the inertia of the combustion engine, θ_D the inertia of the dynamometer, ω_E and ω_D are the engine and the dynamometer speed, c is the spring constant and d the damping constant. T_E and T_{DSet} are the torque of the combustion engine and the air gap torque of the dynamometer, respectively.

For most engine test benches the dynamometer is a very fast induction machine with a subordinate air gap torque control loop. Since the dynamics of the subordinate air gap torque control are very fast, it is possible to neglect these dynamics and to consider the air gap torque as an input to the system $T_{DSet} = T_D$.

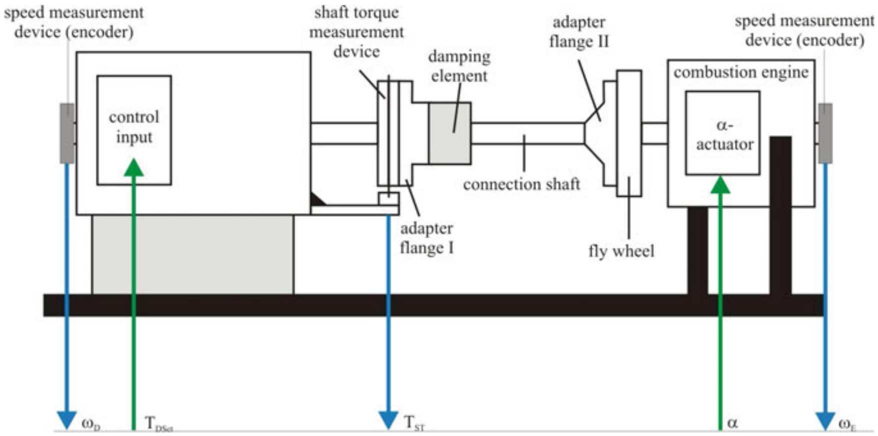


Fig. 5 Engine test bench system

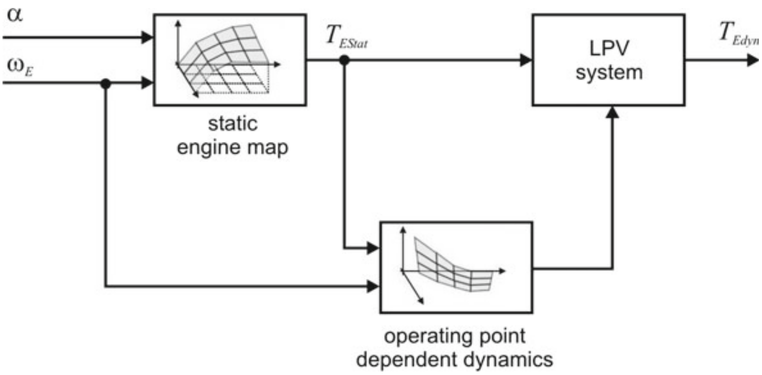


Fig. 6 Engine test bench system

The most critical part of the system is the engine behaviour, for which only a rough system description is possible, including uncertainties.

The input of the combustion engine is the accelerator pedal signal α, and the interesting output for control purposes is the engine torque. This engine torque can be split into two parts: the mean value engine torque and the oscillating torque caused by the combustion oscillations. The oscillating torque is in a frequency range where it is sufficiently damped by the test bench system, and therefore, only the mean value torque is considered. The system structure is illustrated in Fig. 6. Following this structure, the engine model consists of a static nonlinear map and a dynamical system which is also nonlinear. The dynamical system is restricted to be a first-order system. Hence, the dynamical part can be described by

$$\begin{aligned} \dot{T}_E &= -\rho\left(T_{Estat}, \omega_E\right) T_E + \rho\left(T_{Estat}, \omega_E\right) T_{EStat} \\ T_{Edyn} &= T_E \end{aligned}$$

where $T_{Estat} = T_{Estat}(\alpha, \omega_E)$ is the output of the static engine map and $\rho\left(T_{Estat}, \omega_E\right)$ is the nonlinear, state and input-dependent time constant. The parameters of the system model are identified locally for a sufficient number of operating points. Between these operating points, the parameters are calculated by linear interpolation.

Now, define the normalized state variables

$$x_1 = \frac{T_E - T_{E0}}{\Delta T_E}, \qquad x_2 = \frac{\Delta\varphi - \Delta\varphi_0}{\max\left(\Delta\varphi\right)}$$
$$x_3 = \frac{\omega_E - \omega_{E0}}{\Delta\omega_E}, \qquad x_4 = \frac{\omega_D - \omega_{D0}}{\Delta\omega_D}$$

and the normalized input

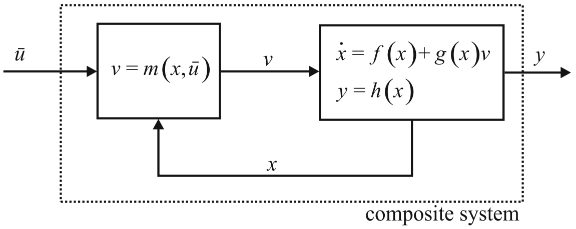


Fig. 7 Structure of the considered system class

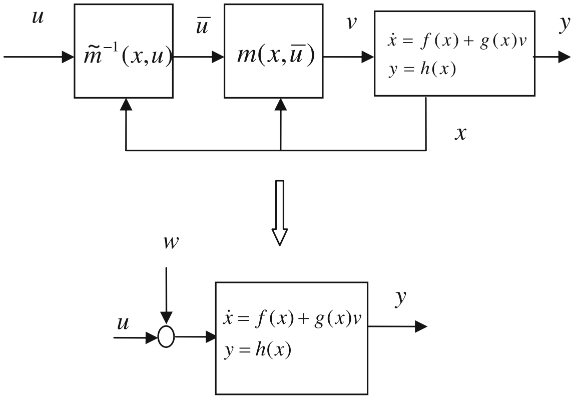


Fig. 8 Error model of the system with approximated inversion

$$\tilde{T}_{DSet} = \frac{T_{Dset} - T_{E0}}{\Delta\omega_E}$$

where T_{E0}, $\Delta\varphi_0$, ω_{E0} and ω_{D0} defines the operating point and ΔT_E, max $(\Delta\varphi)$, $\Delta\omega_E$ and $\Delta\omega_D$ the maximum expected distance from the equilibrium point.

The composite model of the engine test bench matches the structure in Fig. 7, i.e. it is an *Extended Hammerstein System*. Here, input $\bar{u}$ consists of α and $\tilde{T}_{DSet}$ and y consists of two outputs (engine speed x_3 and engine torque x_1), and the static map m comes from the approximation of the nonlinear map ρ in a polynomial fashion.

To solve the control problem, the nonlinear static map is locally inverted, and the approximation error affects the system in the same direction as the input, see Fig. 8.

A nonlinear H_∞ control law can be designed so as to ensure robust stability and performance at the face of the uncertainties due to the imperfect inversion of the static map, see [64] for details.

2.5 *Semi-active Suspension System*

The problem consists of designing a switching control strategy for comfort-oriented semi-active suspensions in road vehicles [26, 55]. The model, see Fig. 9, is as follows:

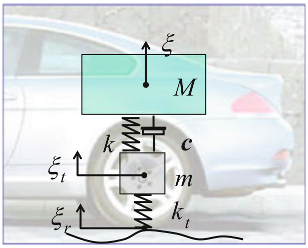


Fig. 9 Quarter-car system

$$\begin{aligned}
M\ddot{\xi}(t) &= -c(t)(\dot{\xi}(t) - \dot{\xi}_t(t)) - k(\xi(t) - \xi_t(t)) + k\Delta_s - Mg \\
m\ddot{\xi}_t(t) &= c(t)(\dot{\xi}(t) - \dot{\xi}_t(t)) + k(\xi(t) - \xi_t(t)) - k_t(\xi_t(t) - \xi_r(t)) - k\Delta_s + k_t\Delta_t - mg \\
\dot{c}(t) &= -\beta c(t) + \beta c_{in}(t)
\end{aligned}$$

where $\xi(t)$, $\xi_t(t)$ and $\xi_r(t)$ are the vertical position of the body, the unsprung mass and the road profile, respectively. The coefficients M and m are the quarter-car body mass and the unsprung mass (tyre, wheel, brake, etc.), respectively. The coefficients β, k and k_t are the bandwidth of the active shock absorber, the stiffness of the suspension spring and of the tyre, respectively. The coefficients Δ_s and Δ_t are the length of the unloaded suspension spring and of the tyre. Finally, $c(t)$ and $c_{in}(t)$ are the actual and requested damping coefficients of the passive shock absorber. In order to simplify the computations we assume that β is large enough so that $c(t) \approx c_{in}(t)$. Moreover, we consider a genuine switching strategy, so that $c(t)$ can assume only two values, namely, c_{min} and c_{max}, to be specified later on.

The control strategy consists of minimizing the chassis vertical acceleration $\ddot{\xi}(t)$ by a suitable choice of the control variable $c(t) \in \{c_{min},\ c_{max}\}$.

In order to fit this example in the framework of this chapter, let us take the variations $\delta\xi(t)$ and $\delta\xi_t(t)$ of $\xi(t)$ and $\xi_t(t)$ around an equilibrium point associated with zero road profile, arriving at the system

$$\begin{aligned}
\dot{\bar{\xi}}(t) &= A_\sigma\bar{\xi}(t) + B_r\xi_r(t) \\
y(t) &= C_\sigma\bar{\xi}(t) + d(t) \\
z(t) &= E_\sigma\bar{\xi}(t)
\end{aligned}$$

where $d(t)$ is the measurement noise and

$$A_1 = \begin{bmatrix} 0 & 1 & 0 & 0 \\ -\frac{k}{M} & -\frac{c_{min}}{M} & \frac{k}{M} & \frac{c_{min}}{M} \\ 0 & 0 & 0 & 1 \\ \frac{k}{m} & \frac{c_{min}}{m} & -\frac{(k+k_t)}{m} & -\frac{c_{min}}{m} \end{bmatrix}$$

$$A_2 = \begin{bmatrix} 0 & 1 & 0 & 0 \\ -\frac{k}{M} & -\frac{c_{max}}{M} & \frac{k}{M} & \frac{c_{max}}{M} \\ 0 & 0 & 0 & 1 \\ \frac{k}{m} & \frac{c_{max}}{m} & -\frac{(k+k_t)}{m} & -\frac{c_{max}}{m} \end{bmatrix}$$

$$E_1 = \begin{bmatrix} -\frac{k}{M} & -\frac{c_{min}}{M} & \frac{k}{M} & \frac{c_{min}}{M} \end{bmatrix}$$

$$E_2 = \begin{bmatrix} -\frac{k}{M} & -\frac{c_{max}}{M} & \frac{k}{M} & \frac{c_{max}}{M} \end{bmatrix}$$

$$B_r = \begin{bmatrix} 0 \\ 0 \\ 0 \\ \frac{k_t}{m} \end{bmatrix}$$

and C_σ depends on the choice of the measured variables. The state vector $\bar{\xi}(t)$ contains the chassis displacement $\delta\xi(t)$, its derivative, the tyre displacement $\delta\xi_t(t)$ and its derivative. Again, the disturbance vector $\xi_r(t)$ is the road profile.

A reasonable set of measurements is given by

$$C_1 = \begin{bmatrix} -\frac{k}{M} & -\frac{c_{min}}{M} & \frac{k}{M} & \frac{c_{min}}{M} \\ 0 & 1 & 0 & -1 \end{bmatrix}$$

$$C_2 = \begin{bmatrix} -\frac{k}{M} & -\frac{c_{max}}{M} & \frac{k}{M} & \frac{c_{max}}{M} \\ 0 & 1 & 0 & -1 \end{bmatrix}$$

that corresponds to measuring the body acceleration and the stroke derivative.

The problem is to minimize the $\mathscr{L}_2$ norm

$$\sqrt{\int_0^\infty \ddot{\xi}(t)^2 dt}$$

of the chassis acceleration $\ddot{\xi}(t)$ with respect to impulsive signals on the road profile acceleration $\ddot{\xi}_r(t)$ (or a white noise). This is indeed a realistic situation including road profiles described by ramps, in the deterministic setting, or double integral of a white noise, in the stochastic case. To do this, the model is rewritten as follows. Let

$$w(t) = \begin{bmatrix} \ddot{\xi}_r(t) \\ d(t) \end{bmatrix}, \quad z(t) = \ddot{\xi}(t)$$

and define

$$\begin{aligned} x_1(t) &:= \delta\xi(t) - \xi_r(t) \\ x_2(t) &:= \delta\dot{\xi}(t) - \dot{\xi}_r(t) \\ x_3(t) &:= \delta\xi_t(t) - \xi_r(t) \\ x_4(t) &:= \delta\dot{\xi}_t(t) - \dot{\xi}_r(t) \end{aligned}$$

With these new variables, the system can equivalently be rewritten as

$$
\begin{aligned}
\dot{x}(t) &= A_\sigma x(t) + Bw(t)\\
y(t) &= C_\sigma x(t) + Dw(t) + C_\sigma(\bar{\xi}(t) - x(t))\\
z(t) &= E_\sigma x(t) + E_\sigma(\bar{\xi}(t) - x(t))
\end{aligned}
$$

where A_1, A_2, C_1, C_2, E_1, E_2 have already been defined and

$$
B = \begin{bmatrix} 0 & 0 & 0\\ -1 & 0 & 0\\ 0 & 0 & 0\\ -1 & 0 & 0 \end{bmatrix}, \quad D = \begin{bmatrix} 0 & r_1 & 0\\ 0 & 0 & r_2 \end{bmatrix}
$$

where the tuning parameters r_1 and r_2 reflect the measurements uncertainties.

Both H_2 and H_∞ output feedback control problems have been tackled via Lyapunov–Metzler inequalities and compared with the traditional semi-active suspension control law, called SH (Sky-Hook), where the system is switched according to the sign of $\dot{\xi}(t)(\dot{\xi}(t) - \dot{\xi}_t(t))$, and the ADD (Acceleration-driven damper) strategy, where the switching law depends on the sign of $\ddot{\xi}(t)(\dot{\xi}(t) - \dot{\xi}_t(t))$. It can be shown (see Sect. 6.1, Example 12), that Lyapunov–Metzler switching improves over other methods, and, needless to say, over passive suspension with constant coefficients.

2.6 Systems Biology

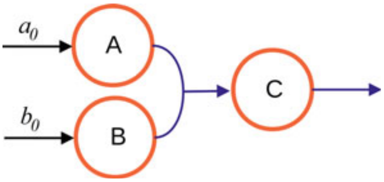
A field in which robust control can play a major role is the modelling and analysis of biological systems. There are many available models of natural phenomena, and mathematical biology is receiving considerable attention. Yet there are major problems in modelling biological phenomena compared to other more classical contexts such as physics, engineering, computer science. The models are deeply uncertain and sometimes very complex. Even when a reasonable model is available, typically its parameters are widely unknown and ranging in huge intervals. Variation of orders of magnitude in the parameters is a common circumstance. Far from attempting to present a general view, in this section, we wish to present some specific problems, especially concerning biochemical models. We will adopt the so-called BDC decomposition framework [16, 17, 59] which is a formal set-up in which several problems can be successfully framed.

Consider the general model of a biochemical reaction network of the form

$$
\dot{x} = Sg(x) + g_0, \tag{1}
$$

where the state $x \in R^n_+$ (the positive orthant) typically represents the concentration of biochemical species, $g(x) \in R^m$ is a vector of functions representing the reaction

Fig. 10 A simple chemical reaction



rates and $g_0 \geq 0$ is a vector of constant influxes; $S \in Z^{n\times m}$ (matrices with integer entries) is the stoichiometric matrix of the system, whose entries s_{ij} represent the net amount of the ith species produced or consumed by the jth reaction, excluding the contribution of constant influxes.

For isolated systems ($g_0 = 0$), the solution is forced to stay in the *stoichiometric compatibility class* $\mathscr{C}(x(0))$:

$$x(t) \in \mathscr{C}(x(0)) = \{x(0) + \text{Ra}[S]\} \cap R^n_+.$$

Typical assumptions for this model are the following.

Assumption 1 All the component functions of vector $g(x)$ are non-negative and continuously differentiable. All their partial derivatives are positive in the positive orthant.

Assumption 2 Each component function of vector $g(x)$ is zero if and only if at least one of its arguments is zero. Moreover, if $s_{ij} < 0$, then g_j must depend on x_i.

The latter assumption assures that for $x_i = 0$ we have $\dot{x}_i \geq 0$ and is required in order for (1) to be a positive system.

Biochemical reaction networks can be visually represented by graphs, as shown in Fig. 10: nodes are associated with biochemical species, while arcs represent interactions among them.

Example 1 Consider, for instance, biochemical reaction network The chemical reaction network shown in Fig. 10 which is associated with the ODE system

$$\begin{aligned} \dot{a} &= a_0 - g_{ab}(a,b) \\ \dot{b} &= b_0 - g_{ab}(a,b) \\ \dot{c} &= g_{ab}(a,b) - g_c(c) \end{aligned} \tag{2}$$

corresponds to the general model (1) with $x = [a\ b\ c]^\top$,

$$S = \begin{bmatrix} -1 & 0 \\ -1 & 0 \\ 1 & -1 \end{bmatrix}, \quad g(x) = \begin{bmatrix} g_{ab}(a,b) \\ g_c(c) \end{bmatrix}, \quad g_0 = \begin{bmatrix} a_0 \\ b_0 \\ 0 \end{bmatrix}.$$

According to [59], it is possible to absorb the system in a differential inclusion as follows.

Consider any component g_i of function g, depending on r variables $x_{k_1}, x_{k_2}, \ldots, x_{k_r}$. Take any point $\bar{x}$ with positive components (typically an equilibrium). We have

$$g_i(x_{k_1}, x_{k_2}, \ldots, x_{k_r}) = g_i(\bar{x}_{k_1}, \bar{x}_{k_2}, \ldots, \bar{x}_{k_r}) + \sum_{j=1}^{r} \left[\int_0^1 \frac{\partial g_i(\bar{x} + \sigma(x - \bar{x}))}{\partial x_{k_j}} d\sigma \right] (x_{k_j} - \bar{x}_{k_j})$$

We rewrite this formula as

$$g_i(x_{k_1}, x_{k_2}, \ldots, x_{k_r}) = g_i(\bar{x}_{k_1}, \bar{x}_{k_2}, \ldots, \bar{x}_{k_r}) + \sum_{j=1}^{r} \left[d_j(x) \right] (x_{k_j} - \bar{x}_{k_j})$$

Assume that an equilibrium $\bar{x}$ exists and let $z \doteq x - \bar{x}$. Since $0 = S\ g(\bar{x}) + g_0$, we have

$$\dot{z}(t) = S\ [g(z(t) + \bar{x}) - g(\bar{x})]$$

hence (1) can equivalently be written as

$$\dot{z}(t) = BD(z(t))C\ z(t),$$

where matrix $B \in Z^{n \times q}$ is formed by a selection of columns of S, $C \in Z^{q \times n}$ and $D(z)$ is a diagonal matrix with non-negative diagonal entries which represent the partial derivatives, so q is the number of all possible partial derivatives with respect to all arguments.

Example 2 For the reaction network (2) in Example 1, let $\alpha = \partial g_{ab}(a, b)/\partial a$, $\beta = \partial g_{ab}(a, b)/\partial b$ and $\gamma = \partial g_c(c)/\partial c$ be positive parameters. Then $D = \mathrm{diag}(\alpha, \beta, \gamma)$,

$$B = \begin{bmatrix} -1 & -1 & 0 \\ -1 & -1 & 0 \\ 1 & 1 & -1 \end{bmatrix}, \qquad C = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}.$$

Denoting by b_i the ith column of B and by $c_i^\top$ the ith row of C, the system can also be written as follows:

$$\dot{z}(t) = \sum_k\ d_k(z(t))\ [b_k c_k^\top]\ z(t),$$

so the matrix is a positive combination of rank-one matrices $[b_k c_k^\top]$.

Example 3 The reaction network in Fig. 11 has equations

$$\begin{aligned} \dot{a} &= a_0 - g_a(a) \\ \dot{b} &= g_a(a) - g_{bc}(b, c) \\ \dot{c} &= c_0 - g_{bc}(b, c) - g_c(c) \end{aligned}$$

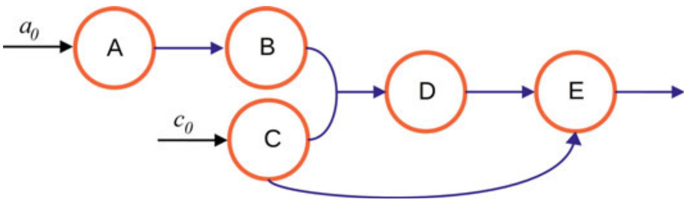
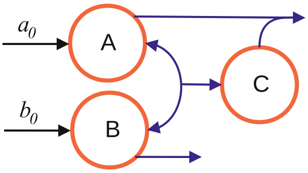


Fig. 11 A simple chemical reaction

Fig. 12 Graph of the dynamical network in Example 4



$$\dot{d} = g_{bc}(b,c) - g_d(d)$$
$$\dot{e} = g_d(d) - g_e(e) + g_c(c)$$

and BDC decomposition

$$BDC = \underbrace{\begin{bmatrix} -1 & 0 & 0 & 0 & 0 & 0 \\ 1 & -1 & 0 & -1 & 0 & 0 \\ 0 & -1 & -1 & -1 & 0 & 0 \\ 0 & 1 & 0 & 1 & -1 & 0 \\ 0 & 0 & 1 & 0 & 1 & -1 \end{bmatrix}}_{=B} \underbrace{\begin{bmatrix} \alpha & 0 & 0 & 0 & 0 & 0 \\ 0 & \beta & 0 & 0 & 0 & 0 \\ 0 & 0 & \gamma & 0 & 0 & 0 \\ 0 & 0 & 0 & \delta & 0 & 0 \\ 0 & 0 & 0 & 0 & \psi & 0 \\ 0 & 0 & 0 & 0 & 0 & \varphi \end{bmatrix}}_{=D} \underbrace{\begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix}}_{=C},$$

where $\alpha = \partial g_a/\partial a$, $\beta = \partial g_{bc}/\partial a$, $\gamma = \partial g_{bc}/\partial c$, $\delta = \partial g_d/\partial d$, $\psi = \partial g_c/\partial c$, $\varphi = \partial g_e/\partial e$ are again positive functions.

Example 4 The chemical reaction network associated with the graph in Fig. 12, is represented by the equations

$$\dot{a} = a_0 - g_{ab}(a,b) - g_{ac}(a,c) + g_c(c),$$
$$\dot{b} = b_0 - g_{ab}(a,b) - g_b(b) + g_c(c),$$
$$\dot{c} = g_{ab}(a,b) - g_{ac}(a,c) - g_c(c),$$

which can be rewritten as system (1) with $x = [a \;\; b \;\; c]^\top$,

$$S = \begin{bmatrix} -1 & 1 & -1 & 0 \\ -1 & 1 & 0 & -1 \\ 1 & -1 & -1 & 0 \end{bmatrix}, \; g(x) = \begin{bmatrix} g_{ab}(a,b) \\ g_c(c) \\ g_{ac}(a,c) \\ g_b(b) \end{bmatrix}, \; g_0 = \begin{bmatrix} a_0 \\ b_0 \\ 0 \end{bmatrix}.$$

Its BDC decomposition is characterized by the matrices,

$$BDC = \begin{bmatrix} -1 & -1 & 1 & -1 & -1 & 0 \\ -1 & -1 & 1 & 0 & 0 & -1 \\ 1 & 1 & -1 & -1 & -1 & 0 \end{bmatrix} \begin{bmatrix} \alpha & 0 & 0 & 0 & 0 & 0 \\ 0 & \beta & 0 & 0 & 0 & 0 \\ 0 & 0 & \gamma & 0 & 0 & 0 \\ 0 & 0 & 0 & \delta & 0 & 0 \\ 0 & 0 & 0 & 0 & \psi & 0 \\ 0 & 0 & 0 & 0 & 0 & \varphi \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \\ 1 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & 1 & 0 \end{bmatrix}$$

where $\alpha = \partial g_{ab}/\partial a$, $\beta = \partial g_{ab}/\partial b$, $\gamma = \partial g_c/\partial c$, $\delta = \partial g_{ac}/\partial a$, $\psi = \partial g_{ac}/\partial c$, $\varphi = \partial g_b/\partial b$ are again positive functions.

A remarkable fact is the following [16, 17].

Proposition 1 *The Jacobian of the system has the same BDC structure. Hence, the linearization of a system can be studied by considering the matrix of the form BDC, where D is a diagonal matrix with positive diagonal entries (related to the system partial derivatives).*

Now the problem is that the functions g_* are widely unknown. A standard approximation is to take them monomials of the form

$$g_{abc} = ka^p b^q c^q$$

(Mass Action Kinetics models). For instance, the reactions $2A + 3B \rightarrow C$ would have reaction speed ka^2b^3. This approximation holds only under several restrictive assumptions, and even in this case, the coefficient k is deeply uncertain and depends on several factors, for instance, the temperature.

A possible way to approach the robust investigation of chemical reaction networks is to assume bounds on the average values of the derivatives represented by the introduced expression

$$d_k = \left[\int_0^1 \frac{\partial g_i(\bar{x} + \sigma(x - \bar{x}))}{\partial x_h} d\sigma \right]$$

or bounds on the derivatives in an equilibrium point in the case of a study based on linearization. Therefore,

$$d_k^- \leq d_k(\cdot) \leq d_k^+$$

There are two possible approaches.

- Robust approach: d_k^- and d_k^+ are assumed to be assigned bounds.
- Structural approach: d_k are assumed to be arbitrary positive functions.

The second approach is very demanding yet appealing in biology, where even the values of the bounds can be unclear.

As a final consideration, we can say that the model absorbing procedure is a general tool for investigating nonlinear systems in any context.

3 Lyapunov Approach in Robustness

Lyapunov functions have been used in the mathematical literature to investigate robustness, although paradigms are typically different. In particular, classical theorems of the beginning of the past century show that any system of differential equations having an equilibrium point whose asymptotic stability has been established by means of a Lyapunov function remains stable under perturbations as long as they are small enough.

In robustness theory, the point of view is slightly different, the main concern being the evaluation of the size of the disturbances which is compatible with stability. In any case, Lyapunov theory plays a fundamental role in robustness which has remained unchanged, and even reinforced, through the years.

3.1 Control Lyapunov Functions and Gradient-Based Control

In this section, we present a classical yet effective approach to deal with uncertain systems based on Lyapunov functions. Consider a system of the form

$$\dot{x}(t) = f(x(t), w(t)) \qquad w(t) \in \mathscr{W} \tag{3}$$

where $w(t) \in \mathscr{W}$ is an unknown signal evolving in the compact set $\mathscr{W}$, $x(t) \in R^n$ is the state. Assume that 0 is an equilibrium state for all w.

$$0 = f(0, w) \qquad w(t) \in \mathscr{W}$$

Remark 1 It is worth pointing out that the previous assumption is restrictive since it requires that the equilibrium is invariant with respect to w. We accept this assumption leaving the general case to more specialized literature [18, 48, 101].

First we provide a notion of robust stability.

Definition 2 The 0 state of system (3) is robustly Globally Uniformly Asymptotically Stable (GUAS), if for any $\varepsilon > 0$ and $\mu > 0$ there exists $T(\varepsilon, \mu) > 0$ such that if $\|x(0)\| \leq \mu$, then $\|x(t)\| \leq \varepsilon$ for all $t > T(\varepsilon, \mu)$.

The word uniform refers to the fact that the convergence to 0 is uniform with respect to w and the ball of initial conditions. In the literature there are examples of systems whose trajectories all converge to 0, but some require much more time than others, namely, T depends on $x(0)$ and w. The following theorem holds.

Theorem 1 *Assume that there exist a positive definite and radially unbounded*[1] *function $V(x)$, $V : R^n \to R_+$ with $V(0) = 0$ which is continuously differentiable,*

[1] That is, $V(x) \to +\infty$ as $\|x\| \to +\infty$.

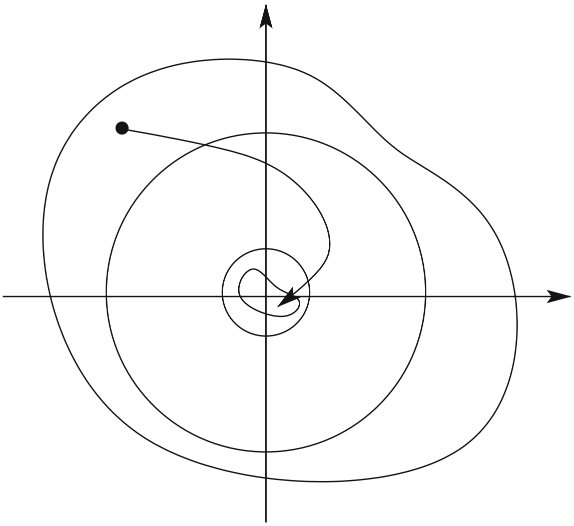


Fig. 13 Lyapunov function: the four sets $\{V(x) \le \bar{\varepsilon}\} \subseteq \{\|x\| \le \varepsilon\} \subseteq \{\|x\| \le \mu\} \subseteq \{V(x) \le \bar{\mu}\}$

and a continuous positive definite and radially unbounded function $\phi(x)$ such that, for any $w \in \mathcal{W}$,

$$\dot{V}(x, w) \doteq \nabla V(x) f(x, w) \le -\phi(x).$$

Then, the solution $x \equiv 0$ is robustly globally uniformly asymptotically stable.

Sketch of the proof. Letting $x(t) = x$ and $w(t) = w$, we have that

$$\frac{d}{dt} V(x(t))\Big|_{x(t)=x, \quad w(t)=w} \doteq \dot{V}(x, w) = \nabla V(x) f(x, w)$$

(the Lyapunov derivative depends only on the current state x and disturbance w). Hence by integration

$$V(x(t)) = V(x(0)) + \int_0^t \frac{d}{d\tau} V(x(\tau)) d\tau \le - \int_0^t \phi(x(\tau)) d\tau$$

Since the function in the integral is positive, $V(x(t))$ is not increasing.

The next step is to notice that, in view of the radial unboundedness, given $\varepsilon > 0$ and $\mu > 0$ (as in the definition), there exist $\bar{\varepsilon}$ and $\bar{\mu}$ such that the following set inclusions hold (see Fig. 13)

$$\{V(x) \le \bar{\varepsilon}\} \subseteq \{\|x\| \le \varepsilon\} \subseteq \{\|x\| \le \mu\} \subseteq \{V(x) \le \bar{\mu}\}$$

Moreover, if $x(T) \in \{V(x) \le \bar{\varepsilon}\}$ at some $T > 0$, then $x(t) \in \{V(x) \le \bar{\varepsilon}\}$ for $t > T$. This means that, if we prove that for $V(x(0)) \le \bar{\mu}$ there exists $T > 0$ such that $V(x(T)) < \bar{\varepsilon}$, then this T is that requested in the definition and the proof is complete.

Assume by contradiction that $x(t)$ remains in the closure of the complement $\{V(x) \geq \bar{\varepsilon}\}$. ϕ being positive definite and radially unbounded, we have that it reaches a minimum, $\varphi > 0$, in such a set. Therefore

$$V(x(t)) \leq V(x(0)) - \int_0^t \phi(x(\tau))d\tau \leq V(x(0)) - \varphi t$$

which would imply that $V(x(t))$ becomes negative for $t > V(x(0))/\varphi$, which is not possible. Hence in time $T = V(x(0))/\varphi$, the set $\{V(x) \leq \bar{\varepsilon}\}$ is reached.

The result above is very powerful. It ensures that, no matter how $w(t)$ evolves, the origin is asymptotically reached. The weakness is that it is not always clear how to find a proper function V having the property required by the theorem. If such a function exists, it is called *Lyapunov function.*

Let us now consider a controlled system. For brevity, we consider a special, yet quite common case in which the control enters linearly in the equation as follows:

$$\dot{x}(t) = f(x(t), w(t)) + Bu(t) \qquad w(t) \in \mathscr{W}$$

We then say that the positive definite continuously differentiable function $V(x)$ is a *control Lyapunov function* if there exists a, possibly nonlinear, control function

$$u(t) = K(x(t))$$

having the property that the resulting closed-loop system

$$\dot{x}(t) = f(x(t), w(t)) + BK(x(t)) \qquad w(t) \in \mathscr{W}$$

is well posed, namely, that it admits a unique globally defined solution, and that it admits V as Lyapunov function.

Remark 2 In some mathematical literature the control Lyapunov function is defined in a much more general way requiring that for any x and $w \in \mathscr{W}$ the equality

$$\dot{V}(x, w, u) \doteq \nabla V(x)[f(x, w) + Bu] \leq -\phi(x) \tag{4}$$

is pointwise feasible for some u (i.e. for all x there exists u such that the inequality is satisfied for all $w \in \mathscr{W}$). The fact that u is a function $u = K(x)$, with some regularity properties is not always granted, and proper assumptions have to be made. For instance, consider the scalar system

$$\dot{x} = x + |x|u$$

(which is not of the previous form since $B = B(x) = |x|$) and $V(x) = x^2$. Then

$$\dot{V} = 2x^2 + 2x|x|u < 0$$

requires $u < -1$ for $x > 0$ and $u > 1$ for $x < 0$ so no feedback control can be continuous in $x = 0$. The existence of a solution $x(t)$ is therefore (in general, and not only in this simple case) an issue.

Now we introduce the fundamental concept of *gradient-based control*. Assume that a control Lyapunov function is given, associated with a control law $u = K^*(x)$, which is not necessarily known. We assume that K^* is continuous. Let us rewrite (4) as

$$\nabla V(x)Bu \leq -\phi(x) - \nabla V(x)f(x, w)$$

This inequality is satisfied if

$$\nabla V(x)Bu \leq -\psi(x) \tag{5}$$

where we have defined

$$\psi(x) \doteq \max_{w \in \mathscr{W}} \{\phi(x) + \nabla V(x)f(x, w)\}$$

The inequality (5) is linear in u. We just need to find a function $K(x)$, such that $u = K(x)$ satisfies this inequality for all x. A possibility is to consider the minimum effort control [95], given by

$$u = K_{ME}(x) \doteq \begin{cases} 0 & \text{if } \psi(x) \leq 0 \\ -\frac{\psi(x)}{\|\nabla V(x)B\|^2} B^\top \nabla V(x)^\top & \text{if } \psi(x) > 0 \end{cases}$$

which is the control value of minimum norm which satisfies the inequality. This control has been shown to be continuous (under suitable assumptions) [95], and it is of the form

$$u = -\gamma(x)B^\top \nabla V(x)^\top$$

where γ is a sufficiently regular non-negative function. This is the general expression of what we call a *gradient-based controller*.

A remarkable property of the gradient-based control is that it has an infinite gain margin.

Proposition 2 *Assume that the gradient-based control $u = -\bar{\gamma}(x)B^\top \nabla V(x)^\top$ associated with the control Lyapunov function V is robustly stabilizing with some function $\bar{\gamma}(x) > 0$. Then the control is also robustly stabilizing for any $\gamma(x) > \bar{\gamma}(x)$.*

The proof is easy, because if $u = -\bar{\gamma}(x)B^\top \nabla V(x)^\top$ satisfies (5), namely, if

$$-\bar{\gamma}(x)\nabla V(x)BB^\top \nabla V(x)^\top = -\bar{\gamma}(x)\|\nabla V(x)B\|^2 \leq -\psi(x)$$

then, if we increase γ, the inequality remains satisfied. The concept of gradient-based control is interesting, because if we know a control Lyapunov function, then we have an expression for the control.

Clearly, the true problem is finding a control Lyapunov function which is in general hard. There are special classes of systems for which a solution can be found and some will be discussed next.

3.2 Some Special Classes of Systems

One remarkable class of systems of practical interest in which the theory presented before can be successfully applied is the class of systems with matched uncertainties [9, 80]. Remarkable examples are fully actuated robots and several types of mechanical systems.

Consider a system of the form

$$\dot{x}(t) = f(x(t)) + B\Delta(x,t) + Bu(t)$$

where we assume

$$\|\Delta(x,t)\| \leq \delta \|x\|$$

for some positive δ. Assume that the nominal part is stable and admits a Lyapunov function $V(x)$ for which

$$\nabla V(x) f(x) \leq -\alpha^2 \|x\|^2$$

In practice this condition is satisfied if f is a pre-stabilized nominal system. The idea is that we may use the Lyapunov function of the nominal system as control Lyapunov function for the uncertain system. Consider the gradient-based control $u = -\gamma B^\top \nabla V(x)^\top$. Write the Lyapunov derivative, add and subtract the term $\|\Delta\|^2/(4\gamma)$ and complete the square, to get

$$\begin{aligned}\dot{V} &= \nabla V(x) f(x) + \nabla V(x) B\Delta - \gamma \nabla V(x) B B^\top \nabla V(x)^\top + \frac{\Delta^\top \Delta}{4\gamma} - \frac{\Delta^\top \Delta}{4\gamma} = \\ &= \nabla V(x) f(x) - \left\| \sqrt{\gamma}\, B^\top \nabla V(x)^\top - \frac{\Delta}{2\sqrt{\gamma}} \right\|^2 + \frac{\Delta^\top \Delta}{4\gamma} \leq \\ &\leq -\alpha^2 \|x\|^2 + \frac{\Delta^\top \Delta}{4\gamma} \leq -\left[\alpha^2 - \frac{\delta^2}{4\gamma}\right] \|x\|^2 < 0\end{aligned}$$

The last inequality holds for

$$\gamma > \frac{\delta^2}{4\alpha^2}$$

This result means that under the matching assumption we can counteract uncertainties using a sufficiently strong gain.

Example 5 An interesting real experiment was done at the Laboratory of System Dynamics in Udine. Consider the inverted pendulum presented in the motivation section, which corresponds to a system of the form

$$\dot{x}(t) = Ax(t) + B\Delta(\cdot) + Bu(t)$$

where we can assume that $\Delta(\cdot)$ is an uncertainty due to friction and the friction depends (in a nasty way) on the cart speed $\|\Delta(\cdot)\| \leq \delta|x_4|$.

All attempts to stabilize this system adopting pole placement resulted in frustrating failures.

On the other hands, Lyapunov design has been successful. Indeed, we have considered a gradient-based control associated with a quadratic Lyapunov function $V(x) = x^\top Px$ which ensures

$$\dot{V}(x) = 2x^\top xP[A + BK^*]x(t) = x^\top P[A + BK^*] + [A + BK^*]^\top P \leq -x^\top Qx$$

for some gain K^*. We have actually applied the gain $-\gamma B^\top Px$. The matrix P was found adopting an LQ control approach. The optimal control is indeed a gradient-based control.

In Fig. 14 we report the simulation (including figures) with the system equipped with a pole assignment-based control and with an optimal linear quadratic LQ control. The simulations are realistic and confirmed by the experiments.[2] Due to the amplitude of the oscillations, the real system controlled with pole assignment does not work while the system controlled via LQ control does.

Of course, the matching conditions are quite strong. An interesting generalizations are possible for certain classes of systems. As an example, consider the very simple model

$$\begin{aligned}\dot{x}_1 &= \phi(x_1) + x_2\\ \dot{x}_2 &= u\end{aligned}$$

where ϕ satisfies the sector-bound conditions

$$\alpha \leq \phi(x_1)/x_1 \leq \beta$$

Note that we can rewrite the system as

$$\begin{aligned}\dot{x}_1 &= wx_1 + x_2\\ \dot{x}_2 &= u\end{aligned}$$

[2] The material is reported in the thesis: Roberta Ribis, Analisi sperimentale sul sistema carrello-pendolo, undergraduate thesis, 2005 (in Italian).

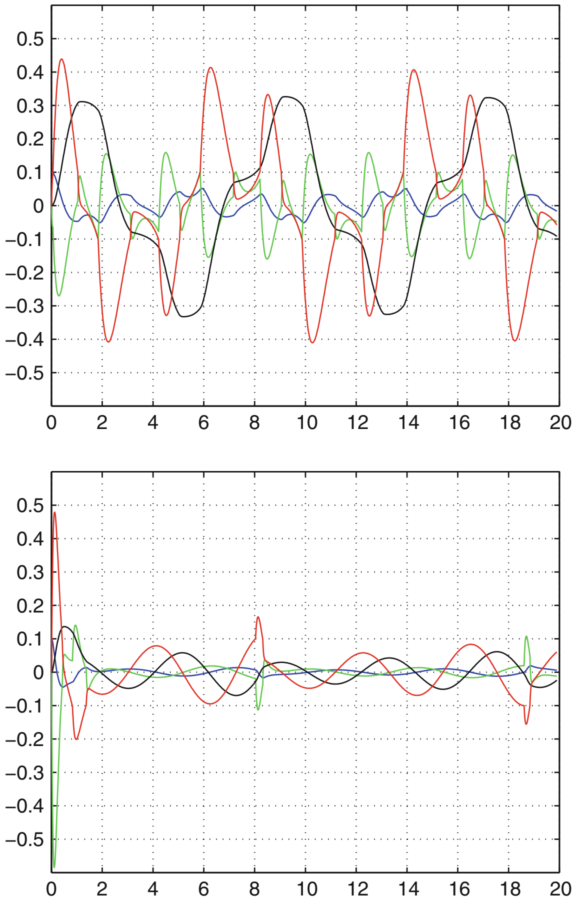


Fig. 14 Simulations of the system equipped with a pole placement based control (top) and equipped with an LQ control (bottom). Time is measured in seconds

where $w = \phi(x_1)/x_1$ and $\alpha \leq w \leq \beta$. The uncertainties are not matched. However, we can apply the so-called backstepping procedure [48]. Let us (initially) cheat by pretending that x_2 is a control variable, which is not. Then $x_2 = -k_1 x_1$ would stabilize the x_1-subsystem if $k_1 > \beta$. Now we use the true control in such a way that x_2 tracks the signal $-k_1 x_1$, i.e. $u = -k_2(x_2 + k_1 x_1)$ with k_2 large enough. Under the variable transformation

$$z_1 = x_1, \qquad z_2 = x_2 + k_1 x_1,$$

and $u = -k_2(x_2 + k_1 x_1) - k_2 z_2$, it turns out that

$$\begin{bmatrix} \dot{z}_1 \\ \dot{z}_2 \end{bmatrix} = \begin{bmatrix} -(k_1 - w) & 1 \\ k_1 w - k_1^2 & k_1 - k_2 \end{bmatrix} \begin{bmatrix} z_1 \\ z_2 \end{bmatrix}$$

The derivative of the Lyapunov function $V(z_1, z_2) = (z_2^2 + z_1^2)/2$ is

$$\dot{V}(z_1, z_2) = -(k_1 - w)z_1^2 + (1 + k_1 w - k_1^2)x_1 x_2 - (k_2 - k_1)z_2^2$$

Since $k_1 > \beta \geq w$, if we take k_2 large to ensure $(k_2 - k_1)(k_1 - w) > (1 + k_1 w - k_1^2)$, it is not difficult to see that $\dot{V}(z_1, z_2) < 0$ $z \neq 0$. The backstepping procedure is successful for systems in the so-called feedback form [48]. For more general cases, we need to resort to other methods.

3.3 Quadratic Stability and LMI

In this subsection, we consider linear uncertain systems. Before introducing the technical results, we would like to provide some motivations. In particular, we wish to justify why linear uncertain systems with time-varying parameters are important. Consider the very general problem of analysis of control of a nonlinear system of the form

$$\dot{x}(t) = f(x(t)) + Bu(t) \tag{6}$$

where f can be uncertain. The Lyapunov approach suffers a major trouble: finding a suitable V. One possible way is to adopt numerical techniques which are, unfortunately, effective only for linear uncertain systems.

One possible way to proceed is based on model–absorbing (see, for instance, [18]). Consider again the formula (similar to that presented in Sect. 2.6)

$$f(x) = f(\bar{x}) + \left[\int_0^1 \frac{\partial f(\bar{x} + \sigma(x - \bar{x}))}{\partial x} d\sigma\right](x - \bar{x}) = J(x)(x - \bar{x})$$

Assume that $\bar{x}$ is an equilibrium state, namely, $f(\bar{x}) = 0$. Then we can write

$$\dot{z}(t) = J(z(t))z(t)$$

with $z(t) = x(t) - \bar{x}$.

Now assume that the Jacobian belongs to a convex and compact set

$$\frac{\partial f(\bar{x} + \sigma(x - \bar{x}))}{\partial x} \in \mathscr{A}.$$

Then it is not difficult to show that also $J(z) \in \mathscr{A}$, namely, also the averaged values belong to the set. Hence, we have to deal with an uncertain linear system. Typically, it is possible to find a parametrization $A(w)$, with $w \in \mathscr{W}$. We have that any trajectory of the nonlinear system is also a trajectory of the following *Linear Differential Inclusion*:

$$\dot{z}(t) = A(w(t))z(t), \quad w(t) \in \mathscr{W} \tag{7}$$

(and not the other way). Therefore, if we are able to find Lyapunov functions, or control Lyapunov functions for (7) these will serve as control Lyapunov functions for (6). A polytope is a special (yet important) case of convex set, and hence, we consider the case in which $\mathcal{W}$ is a polytope. Note that any convex set can be approximated by a polytope.

Example 6 Consider the simple mechanical model of a pendulum in the upper or lower position, depending on the sign of α. We know only that $|\alpha| \leq \bar{\alpha}$, for some given bound $\bar{\alpha} > 0$. Then

$$\begin{aligned} \dot{x}_1 &= x_2 \\ \dot{x}_2 &= \alpha \sin(x_1) + 1 = \left[\alpha \frac{\sin(x_1)}{x_1}\right] x_1 + u \end{aligned}$$

Since $|\alpha \sin(\theta)/\theta| \leq \bar{\alpha}$ the polytopic differential inclusion is

$$\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ w(t) & 0 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + \begin{bmatrix} 0 \\ 1 \end{bmatrix} u \qquad |w(t)| \leq |\alpha|$$

If we are able to stabilize this system, for all possible $w(t)$, then we can also stabilize the original system.

A polytopic system is a system of the form $\dot{x} = A(w)x$ with

$$\dot{x} = \left[\sum_{i=1}^{N} w_i A_i\right] x, \quad w \in \mathcal{W} \tag{8}$$

where $\mathcal{W}$ is the simplex

$$\mathcal{W} := \left\{ w \in R^N \ : \ \sum_{i=1}^{N} w_i = 1, \ w_i \geq 0 \right\} \tag{9}$$

In the case of a controlled system, we will consider the expression

$$\dot{x} = A(w)x + B(w)u,$$

where $B = \sum_{i=1}^{N} w_i B_i$ has the same structure as A.

Note that a family of the form

$$\sum_{i=1}^{M} q_i A_i, \quad q_i^- \leq q_i \leq q_i^+$$

can be reduced to the polytopic case by considering all the vertex matrices $\hat{A}_k = \sum_{i=1}^{M} \hat{q}_i A_i$ where $\hat{q}_i \in \{q_i^-, q_i^+\}$, are taken on the extrema. For instance in the case $M = 2$, the extrema are

$$\hat{A}_1 = A_1 q_1^- + A_2 q_2^-,\ \hat{A}_2 = A_1 q_1^+ + A_2 q_2^-,\ \hat{A}_3 = A_1 q_1^- + A_2 q_2^+,\ \hat{A}_4 = A_1 q_1^+ + A_2 q_2^+$$

We derive conditions ensuring asymptotic stability for all $w(t) \in \mathscr{W}$. As a first step, we note that robust stability implies Hurwitz stability of all $A(w)$ for w constant, because $w = \text{const} \in \mathscr{W}$ is a possible realization of the function w. In particular, all the vertex matrices A_i must be Hurwitz. As already mentioned, stability under arbitrary time-varying uncertainties is a much stronger condition than Hurwitz stability for all constant parameters.

To ensure time-varying asymptotic stability, we seek for a common quadratic Lyapunov function, namely, a positive definite function of the form

$$V(x) = x^\top P x \tag{10}$$

This choice is motivated by the fact that for a linear (certain) system $\dot{x} = Ax$ the existence of a Lyapunov function of this form is a sufficient and necessary condition for asymptotic stability.

In the polytopic case we say that the system is quadratically stable if there exists a positive definite P such that

$$(\sum_{i=1}^{N} w_i A_i)^\top P + P(\sum_{i=1}^{N} w_i A_i) = A(w)^\top P + PA(w) < 0, \quad \text{for all} \quad w \in \mathscr{W}$$

Note that this condition implies that the Lyapunov derivative of $V(x) = x^\top P x$ is negative, i.e.

$$\dot{V}(x) = x^\top [A(w)^\top P + PA(w)]x < 0 \quad x \neq 0$$

The machinery to find the matrix $P > 0$ is given by the following linear matrix inequalities (LMIs) for P [29]:

$$A_i^\top P + PA_i < 0, \quad \forall i. \tag{11}$$

We have the following result.

Theorem 2 *System (8) is quadratically stable if and only if there exists $P > 0$ which satisfies the LMIs (11).*

Proof If there exists $P > 0$ which satisfies the LMIs (11), then

$$(\sum_{i=1}^{N} w_i A_i)^\top P + P(\sum_{i=1}^{N} w_i A_i) = A(w)^\top P + PA(w) < 0$$

for all $w \in \mathscr{W}$. Conversely, if the system is quadratically stable, the condition is true for all $w \in \mathscr{W}$ and then also on the vertices. ∎

Then a quadratically stable system admits a Lyapunov function of the form (10) where P is any positive definite matrix which satisfies (11). These inequalities have an interesting property. They are *convex* constraints. This means that if $P_1 > 0$ and $P_2 > 0$ are solutions, then also $P = \alpha P_1 + (1-\alpha)P_2 > 0$ is a solution. Then the solution P (if any) can be found by solving a convex optimization problem. Convex optimization can rely on nice properties [29] and efficient algorithms can be adopted.

The quadratic Lyapunov function does not depend on the parameters; hence, the criterion is conservative if the uncertainties are constant. The criterion is conservative *also* in the case of time-varying parameters. Indeed, it is possible that the system is stable for all $w(t) \in \mathscr{W}$ but no quadratic Lyapunov function exists.

Let us now consider the problem of determining a stabilizing state feedback of the form

$$u(t) = Kx(t)$$

for the system

$$\dot{x} = \sum_{i=1}^{N} w_i A_i x + \sum_{i=1}^{N} w_i B_i u$$

where $w_i \geq 0$ and $\sum_{i=1}^{N} w_i = 1$. By replacing $u = Kx$, we arrive at the conditions

$$[A_i + B_i K]^\top P + P[A_i + B_i K] < 0, \quad \forall i$$

which are inequalities but nonlinear. This implies that convexity is lost in this expression. However, we can multiply both sides by $S = P^{-1}$, and define $R = KS$ to get the equivalent set of inequalities

$$A_i S + B_i R + S A_i^\top + R^\top B_i^\top < 0, \quad \forall i \tag{12}$$

which are linear in S and R. This means that convexity holds again.

We have the following theorem.

Theorem 3 *The polytopic system is quadratically stabilizable via linear feedback $u = Kx$ if and only if (12) is solvable. In this case, the control $u = Kx$ is achieved by taking $K = RS^{-1}$.*

As mentioned before, the existence of a common quadratic function is a sufficient condition for robust stability and stabilizability, but it is not necessary. Indeed, there are linear uncertain systems with time-varying parameters which are stable (or stabilizable) but not quadratically stable (or stabilizable). In the next subsection we will show that there are classes of Lyapunov functions (for instance, the piecewise linear or polyhedral norm) whose existence is necessary and sufficient for stability or stabilization.

3.4 Non-quadratic Stability

Many classes of non-quadratic functions have been presented in the literature. These include the piecewise linear, polynomial, piecewise quadratic functions. The piecewise linear or polyhedral functions are the most classical ones and we briefly present some of their properties here.

By polyhedral function, we mean a convex function which is positive definite, positively homogeneous of order one (a norm in the symmetric case). Such functions are non-differentiable. Then the Lyapunov derivative has to be considered in a generalized sense.

Consider a family of smooth functions $v_i(x)$, $i = 1, 2, \ldots, q$, and consider the max function

$$V = \max_i v_i(x)$$

For any state x consider the maximizer set (the indices where the maximum is achieved)

$$\mathscr{A}(x) = \{i : V(x) = v_i(x)\} \subset \{1, 2, \ldots, q\}$$

which can have one or more elements. Then, given the system

$$\dot{x} = f(x, w)$$

the Lyapunov derivative is

$$\dot{V}(x, w) = \max_{i \in \mathscr{A}(x)} \nabla v_i(x)\dot{x} = \max_{i \in \mathscr{A}(x)} \nabla v_i(x) f(x, w)$$

Special cases of candidate Lyapunov functions are the polyhedral norms which can be expressed in the form

$$V(x) = \|Fx\|_\infty$$

(plane representation) with F full column rank.

Now take the unit ball $\mathscr{P} = \{x : \|Fx\|_\infty \leq 1\}$ and let X be the matrix such that the vertices of $\mathscr{P}$ are the columns of X or their opposite; then we have the following dual representation (vertex representation):

$$V(x) = \min\{\|p\|_1 : x = Xp\}$$

For instance if

$$F = \begin{bmatrix} 1 & 0 \\ 0 & 1 \\ 1 & 1 \end{bmatrix}, \quad \text{then} \quad X = \begin{bmatrix} 1 & 0 & 1 \\ 0 & 1 & -1 \end{bmatrix}$$

Given an induced norm $\| \cdot \|_*$, we call measure of the square matrix M with respect to such a norm the quantity defined as

$$\mu_* = \lim_{h^+ \to 0} \frac{\|I + hM\|_* - 1}{h}$$

Then we have the following results.

Theorem 4 *Given the polytopic system (8)–(9), and a polyhedral norm represented by F and X, the following statements are equivalent:*

- *$V(x)$ is a Lyapunov function;*
- *there exist matrices H_i with $\mu_\infty(P_i) < 0$ such that*

$$FA_i = H_i F, \quad i = 1, 2 \ldots N$$

- *there exist matrices P_i with $\mu_1(P_i) < 0$ such that*

$$A_i X = X P_i, \quad i = 1, 2 \ldots N$$

Theorem 5 *The polytopic system (8)–(9) is asymptotically stable iff there exists a polyhedral Lyapunov function.*

These results are due to [85–87]. The theorem can be extended to stabilization [15, 18].

Example 7 Consider the chemical reaction network of Fig. 11. For brevity, assume that the species are subject to an infinitesimal degradation, i.e. $\dot{x} = -\varepsilon x + Sg(x) + g_0$, with $\varepsilon > 0$ arbitrarily small (otherwise some additional non-singularity assumptions are necessary [16, 17]). Then adopting the procedure in [16, 17] it is possible to derive the polyhedral function $V(x)$ with

$$X = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & -1 & 0 & -1 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 & -1 & 0 & -1 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 1 & 1 & 0 & 0 & -1 & -1 & -1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 & 2 \end{bmatrix}$$

If we now consider the chemical reaction network in Fig. 12, the procedure provides

$$X = \begin{bmatrix} 0 & -2 & 0 & 2 \\ 0 & 0 & 2 & 2 \\ 2 & 0 & -2 & 0 \end{bmatrix}$$

These networks are hence stable; however, they fail to admit quadratic Lyapunov functions [17].

Theorem 6 *Given the polytopic system with control u, the following statements are equivalent.*

- *the polytopic system is stabilizable;*

- *there exists a polyhedral control Lyapunov function* $V(x)$*;*
- *there exist matrices* P_i *with* $\mu_1(P_i) < 0$ *and a matrix* U *such that*

$$A_i X + B_i U = X P_i, \quad i = 1, 2 \ldots N$$

The advantage of having universality is offset by the fact that it is not easy to compute the matrices X and U or F. Moreover, the number of rows of F or columns of X can be huge. If these are given, checking that they work is a simple linear programming problem. Conversely, if they have to be found, a recursive numerical procedure is available, but its complexity can be very high [18].

There are other classes of non-quadratic functions such as piecewise quadratic [72] and smoothed polyhedral. Piecewise quadratic functions have a strongly reduced complexity with respect to polyhedra. The smoothed polyhedral functions are useful because in general the control function associated with these functions is not linear. A piecewise linear function can be found of the form $u = K(x)x$ where $K(x)$ is constant over suitable simplicial cones. Since the Lyapunov function is non-differentiable, the gradient-based control does not work. A possible remedy is to smoothen the polyhedral function by replacing $\|Fx\|_\infty$ by $\|Fx\|_{2p}$ with p integer and large enough. This control works if B is certain or the system has a matched input disturbance.

4 Parametric Approach

In this section, we propose some criteria for the stability analysis of uncertain systems with *constant uncertainties*. Most of the existing results are based on the analysis of the characteristic polynomial. The interested reader can find further details in [7, 112]. We will conclude the section by presenting fundamental results on time-invariant uncertain linear systems.

We start from the basic consideration that there is a big difference between time-varying and constant uncertainties, even if they belong to the same bounding set. A very popular example is the equation

$$\ddot{y}(t) + 2\xi \dot{y}(t) + \sigma y(t) = 0$$

with $0 < \sigma^- \leq \sigma \leq \sigma^+$ and $\xi > 0$. If the uncertainty σ is constant, then the system is robustly Hurwitz. Conversely, for $\xi > 0$ small and a sufficiently large interval $[\sigma^-, \sigma^+]$ a time-varying σ can destabilize the system (see, for instance, [18] for details).

4.1 Value Set and Zero Exclusion Theorem

We now consider the stability analysis for a linear time-invariant uncertain system having uncertain constant parameter. We assume that the uncertain characteristic polynomial has the form

$$p(s,q) = p_n(q)s^n + p_{n-1}(q)s^{n-1} + \cdots + p_1(q)s + p_0(q), \quad q \in \mathscr{Q}$$

where $\mathscr{Q}$ is a generic region in the parameter space. The coefficients p_i are assumed to be continuous functions of a parameter vector q. A typical representation is

$$p_i = p_i(q) \quad q \in \mathscr{Q} = \{q_k^- \leq q_k \leq q_k^+\}$$

where the hyper-rectangle $\mathscr{Q}$ is called *box*. For brevity, we assume the following.

Assumption $p_n(q) \neq 0$ for all $q \in \mathscr{Q}$.

This assumption is always true, for instance, if we consider the characteristic polynomial of a matrix $A(q)$: $\det(sI - A(q))$, for which $p_n(q) \equiv 1$.

Example 8 In the case of the levitator described in Sect. 2.1, the parameters are a, b and c. Considering the compensator $G(s) = \kappa\frac{s+\beta}{s+\alpha}$, the closed-loop characteristic polynomial is

$$p(s,a,b,c) = s^4 + (b+\alpha)s^3 + (\alpha b - a)s^2 + c\kappa - a\alpha - ab)s + (k\beta - ab\alpha)$$

so its coefficients $p_k(a,b,c)$ are function of the three parameters for which we assume to know proper intervals in which they lie.

In general the coefficient vector

$$p(q) = [p_0 \; p_1 \ldots p_n]$$

can be a function of a vector parameter $q \in \mathscr{Q}$, and, in general $\mathscr{Q}$ must not be necessarily a hyper-rectangle.

Now the basic question is how to establish robust stability of the system, namely:

- is the polynomial $p(s,q)$ Hurwitz for all $q \in \mathscr{Q}$?

The robustness analysis test is based on the notion of *value set*.

Definition 3 Given the frequency $\omega \geq 0$, the value set is

$$\mathscr{V}_\omega = \{p(j\omega, q), \quad q \in \mathscr{Q}\} \subset \mathbb{C}$$

The following is a fundamental result [70, 71].

Theorem 7 (Zero exclusion theorem) *The polynomial $p(s,q)$ is Hurwitz for all $q \in \mathscr{Q}$ if and only if*

- *$p(s,q^*)$ is Hurwitz for some $q^* \in \mathscr{Q}$;*
- $0 \notin \mathscr{V}_\omega$, *for all* $\omega \geq 0$.

Now, to apply the criterion we have to consider the following property. Consider a nominal polynomial $p(s,q^*)$. This is typically derived from the model adopted for the synthesis. Of course, $p(s,q^*)$ must be Hurwitz and a preliminary test should be made.

If the nominal polynomial is Hurwitz, then it verifies the following phase increasing property. Without any restriction, we assume that the coefficients of $p(s, q^*)$ are positive.

Property 1 (Michailov theorem) *The polynomial $p(s, q^*)$ is Hurwitz if and only if its phase is well defined, namely, $p(j\omega, q^*) \neq 0$ for all $\omega \geq 0$, and $p(j\omega, q^*)$ encircles the origin counterclockwise by an angle $n\frac{\pi}{2}$.*

The value set for a single polynomial at some frequency is a singleton: a complex number. The value set $\mathcal{V}_\omega$ of an uncertain polynomial at some frequency is a cloud. As the frequency varies, the cloud navigates in the complex plane. Therefore, *if we are able to draw the value set in the complex plane* we can use a graphical test, implementable on a computer, to ensure that $0 \notin \mathcal{V}_\omega$, for all $\omega \geq 0$.

Under some assumptions on the functional form of the coefficients $p_k(q)$, the value set can be easily depicted and the stability test simply performed. In other cases, it is not easy to draw the value set, and typically one must resort to considering a region which includes it at each frequency and apply the test to that region. Unfortunately, this test is *conservative*, namely, it provides sufficient but not necessary conditions.

4.2 Vertex and Edge Theorems

Consider the case in which the coefficients $p_k(q)$ of the polynomial are affine functions of the parameters q.

$$p_k(q) = \sum_{h=1}^{r} f^{kh} q_h, \qquad q_h^- \leq q_h \leq q_h^+$$

Then we have that

$$p(s, q) = p_n(q)s^n + p_{n-1}(q)s^{n-1} + \cdots + p_1(q)s + p_0(q) = \sum_{h=1}^{r} q_h \phi_h(s)$$

namely, the overall polynomial is linear combinations of polynomials $\phi_h(s)$.

$$\phi_h(s) = f^{n,h}s^n + f^{n-1,h}s^{n-1} + \cdots + f^{1h}s + f^{0h}$$

This family is called *a polytope of polynomials*. It is not difficult to see that, for fixed frequency ω, the value set is

$$\mathcal{V}_\omega = \left\{ \sum_{h=1}^{r} q_h \phi_h(j\omega), \;\; q_h^- \leq q_h \leq q_h^+ \right\} = \left\{ \Phi(j\omega)q, \;\; q_h^- \leq q_h \leq q_h^+ \right\}$$

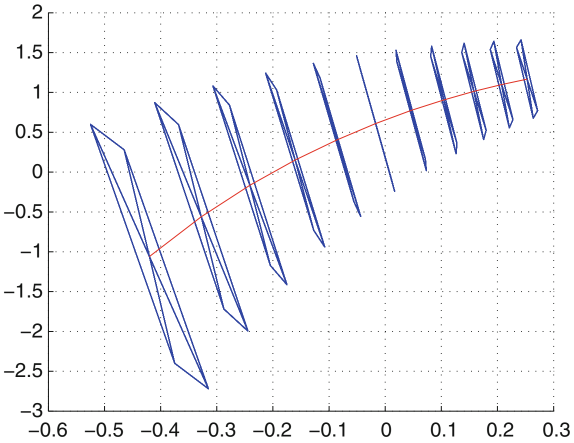


Fig. 15 The value set for the magnetic levitator system at frequencies in the range 100–150 equi-spaced with sampling interval $\Delta\omega = 5$. The adopted gain is $\kappa = 3000$. The value set hits the origin at $\omega \approx 125$. The value set for the magnetic levitator system at frequencies in the range 100–150 equi-spaced with sampling interval $\Delta\omega = 5$. The adopted gain is $\kappa = 3000$. The value set hits the origin at $\omega \approx 125$

where we have adopted the compact notation $\Phi(j\omega) \doteq [\phi_1(j\omega)\ \phi_2(j\omega)\ldots\phi_r(j\omega)]$ which is a vector of complex elements. The resulting set is a convex polygon in the complex plane for each ω.

Example 9 As an example consider the levitator system with the control introduced in Sect. 2.1. If we assume both the resistance and the inductance are precisely known (which is a reasonable assumption since they can be measured accurately), we have that only a and c are uncertain: $q_1 = a = 1400 \pm 400, q_2 = c = 1500 \pm 500$. Hence, the closed-loop polynomial is

$$p(s, q_1, q_2) = s^4 + (b+\alpha)s^3 + (\alpha b - q_1)s^2 + (q_2\kappa - q_1(\alpha + b))s + (k\beta - q_1 b\alpha)$$

with

$$1000 \leq q_1 \leq 1800, \qquad 1000 \leq q_2 \leq 2000$$

The value set is a polygon depicted in Fig. 15 at some selected frequencies. Note that the system fails the robustness test, because the value set hits the origin at the frequency $\omega \approx 125$: the robustness test fails. If we reduce the gain to $\kappa = 2500$ the system passes the robustness test. Indeed, the sequence of value sets passes above the origin, without touching it, as shown in Fig. 16.

Remark 3 For a proper graphical test, it is advisable to normalize the value set. Indeed, the set $\mathscr{V}_\omega$ tends to become large (indeed explode) for ω getting large. Clearly, one can equivalently replace $\mathscr{V}_\omega$ by $\psi(\omega)\mathscr{V}_\omega$ where $\psi(\omega)$ is any strictly positive function, to keep the value set bounded. It is not difficult to see that this does not alter the test results because $0 \in \psi(\omega)\mathscr{V}_\omega$ is equivalent to $0 \in \mathscr{V}_\omega$. In the plot we used

$$\psi(\omega) = \frac{1}{1+\omega^4}$$

Fig. 16 The value set for the magnetic levitator system at frequencies in the range 100–150 equi-spaced with sampling interval 5. The gain is $\kappa = 2500$. The value set sequence does not hit the origin: the robustness test has passed

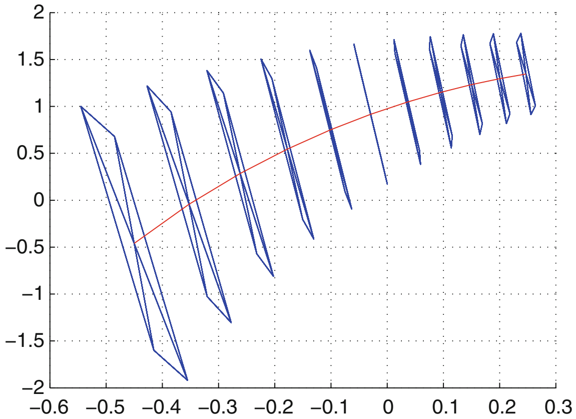
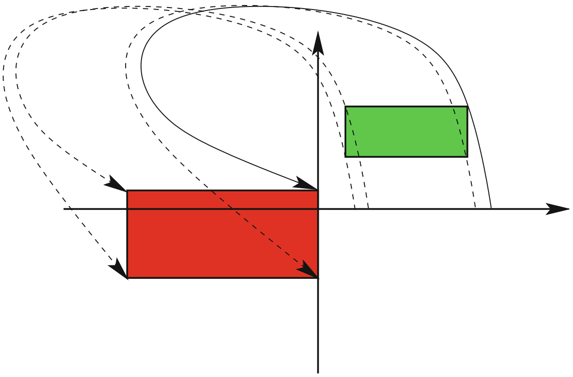


Fig. 17 The value set of an interval polynomial



A particular case of uncertainty class is given by the class of interval polynomials. This is the case where the function representing the coefficients is identities

$$p(s,q) = q_n s^n + q_{n-1} s^{n-1} + \cdots + q_1 s + q_0, \qquad q_k^- \leq q_k \leq q_k^+$$

In this case the polygon is a rectangle with edges parallel to the axes. Indeed, consider the real and imaginary parts of the polynomial $p(j\omega, q)$ as follows:

$$p(j\omega, q) = \left[q_0 - q_2\omega^2 + q_4\omega^4 - q_6\omega^6 \ldots\right] + j\left[q_1\omega - q_3\omega^3 + q_5\omega^5 - q_7\omega^7 \ldots\right]$$

Real and imaginary parts are independent, because they depend, respectively, on the even and odd coefficients only.

It is not difficult to see that the vertices of the value set are the four polynomials where the coefficients are taken as follows:

$$\begin{array}{cccccccccc}
q_0^+ & q_1^+ & q_2^- & q_3^- & q_4^+ & q_5^+ & q_6^- & q_7^- & \cdots \\
q_0^- & q_1^+ & q_2^+ & q_3^- & q_4^- & q_5^+ & q_6^+ & q_7^- & \cdots \\
q_0^- & q_1^- & q_2^+ & q_3^+ & q_4^- & q_5^- & q_6^+ & q_7^+ & \cdots \\
q_0^+ & q_1^- & q_2^- & q_3^+ & q_4^+ & q_5^- & q_6^- & q_7^+ & \cdots
\end{array}$$

These are called the *Kharitonov* polynomials. Adopting a phase argument one can prove the following famous theorem [76].

Theorem 8 (Kharitonov theorem)
The interval polynomial

$$p(s,q) = \sum_{k=0}^{n} q_k s^k, \qquad q_k^- \le q_k \le q_k^+$$

is robustly Hurwitz if and only if the 4 *Kharitonov polynomials are Hurwitz.*

The phase argument is the following: for the value set hitting the origin, necessarily one of the Kharitonov polynomials (the one corresponding to right top vertex in Fig. 17) must violate the phase increasing condition (Property 1).

Kharitonov's theorem is a vertex type result. For interval polynomials, it is sufficient to check a finite number of polynomials to test the entire family. Similar vertex results are available for certain classes of polynomials coming from specific applications. These include the application of a lead-lag compensator to an interval plant [10]. Unfortunately, for general polytopes of polynomials, it is not sufficient (just necessary) to check the vertex polynomials and see if they are Hurwitz. This is true for special classes of polynomials.

What it is known is the following fundamental result [4]. An edge of the box $\mathcal{Q}$ is a one-dimensional set of points of the form

$$[q_1^{\pm}\ q_2^{\pm}\ \ldots\ (\alpha q_k^- + (1-\alpha)q_k^+)\ \ldots\ q_r^{\pm}] \qquad 0 \le \alpha \le 1$$

where $q_h^{\pm}$ are values take at the extrema, either q_h^+ or q_h^-, and α is a parameter.

Theorem 9 (Edge theorem) *(Bartlett, Hollot and Huang 1988) A polytope of polynomials is robustly Hurwitz if and only if all the one-dimensional families of polynomials associated with the one-dimensional edges of the box (see Fig. 18) are Hurwitz.*

The theorem leads to the following nice graphical test. The value set of a polytope of polynomials is a polygon whose exposed edges come from the edges of the box (see again Fig. 18). Hence, to draw the value set, we need just to plot all vertex polynomials and consider the convex hull (there are many available routines which efficiently do this task) at any frequency.

The theorem also implies that one should perform a robust stability test only on the edges of $\mathcal{Q}$ to make sure that all the polynomials of the box are Hurwitz. So as a first step one has to test all vertex polynomials. If they are Hurwitz (otherwise the

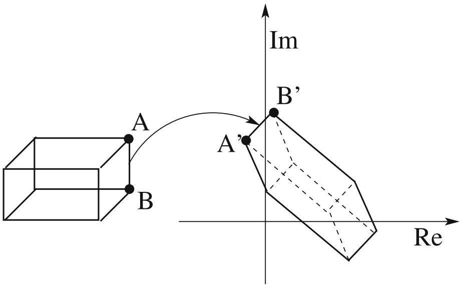


Fig. 18 Image of the box $\mathcal{Q}$: the delimiting edges of the polygon come from edges of the box

stability test fails) as a second step all the edges must be checked. Then the problem reduces to the following one.

- Given two Hurwitz polynomials $p_0(s)$ and $p_1(s)$, is their convex combination

$$p(s,\alpha) = (1-\alpha)p_0(s) + \alpha p_1(s)$$

Hurwitz?

In general the answer is: not necessarily. There are several possible numerical tests, see [7] for a summary of the existing results.

One result is the following. Write the convex combination as

$$p(s,\alpha) = p_0(s) + \alpha q(s)$$

with $q(s) = p_1(s) - p_0(s)$. Then the polynomial $q(s)$ is called a convex direction in the polynomial parameter space if the stability of $p_0(s)$ and $p_0(s) + q(s)$ implies the stability of the whole edge. A phase characterization of the convex stability has been proposed in [102].

4.3 Multilinear Uncertainties

Now we consider the case in which the uncertain polynomial does not have a linear structure but a multilinear one.

A function $\psi(q_1, q_2, \dots, q_r)$ is multilinear or multi-affine if it is affine in any variable, namely, if we fix all variable but one, q_k, we have an affine function in q_k. For instance,

$$\psi = 4 - q_1 + 2q_1q_2 + q_2q_3 + 3q_1q_2q_3$$

is multilinear. The function $-q_1 + 2q_2^2$ is not multilinear because of the square.

The interest for multilinear structures is motivated by the interval matrices. An interval uncertain matrix has entries a_{ij} with

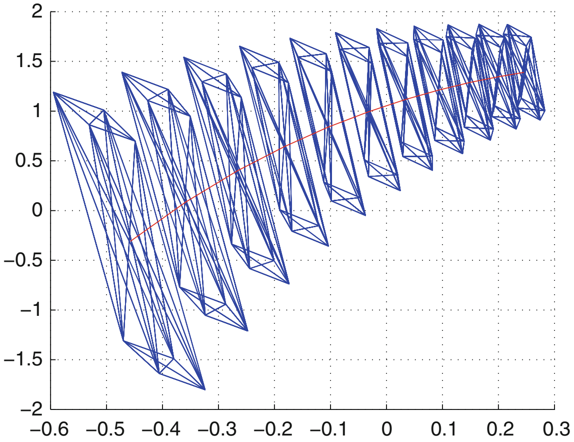


Fig. 19 The value set of the magnetic levitator is inside the convex hull of the vertex images represented in the figure. The frequencies are in the range 100–150 equi-spaced with sampling interval 5. The adopted gain is $\kappa = 2500$

$$a_{ij}^{-} \leq a_{ij} \leq a_{ij}^{+}$$

for given bounds a_{ij}^{-} and a_{ij}^{+}. The coefficients of the characteristic polynomial are multi-affine functions of the parameters if the matrix is expressed in terms of the BDC decomposition [16, 17, 59], namely, $A = BDC$ with D diagonal matrix with positive coefficients d_k. In the case of the magnetic levitator, if we consider $a = q_1$, $c = q_2$ and $b = q_3$ all uncertain, then the polynomial has a multi-affine structure

$$p(s, q_1, q_2, q_3) = s^4 + (q_3 + \alpha)s^3 + (\alpha q_3 - q_1)s^2 + (q_2\kappa - q_1\alpha - q_1 q_3)s + (k\beta - q_1 q_3 \alpha)$$

The value set is not so easy to be depicted for multilinear (or polynomial) uncertainties. The following theorem is of a fundamental help since it shows that we can easily compute a polygon that includes the true value set.

Theorem 10 (Mapping theorem). *If the coefficients of $p(s, q)$ are multi-affine functions and*

$$\mathscr{Q} = \{q : q^{-} \leq q \leq q^{+}\},$$

then

$$V_\omega \subseteq conv\left\{p(j\omega, q^k),\ \ q^k \in vert\mathscr{Q}\right\}$$

In the case of the levitator with bounds

$$1000 \leq q_1 \leq 1800, \quad 1000 \leq q_2 \leq 2000, \quad 340 \leq q_3 \leq 360$$

the polygon including the value set is represented in Fig. 19. The frequencies are the same as considered before. The adopted gain is $\kappa = 2500$ and the system passes the robust stability test. The ‘true’ value set is inside.

As a further example consider a chemical reaction network

$$\dot{x}(t) = Sg(x) + g_0$$

and assume that an equilibrium point $\bar{x}$ exists such that $0 = Sg(\bar{x}) + g_0$. Consider the linearization of this system around the equilibrium. According to Proposition 1, the linearized system can be described by means of the BDC decomposition as

$$\dot{z}(t) = [BDC]z(t) = \left[\sum_k b_k c_k^\top d_k\right] z$$

If we assume bounds of the form $d_k^- \leq d_k(\cdot) \leq d_k^+$, it is quite interesting to notice that this is a case in which the Mapping theorem applies. To prove this it is sufficient to notice that the characteristic polynomial is

$$p(s) = \det(sI - \sum_k b_k c_k^\top d_k)$$

so that its coefficients are multilinear functions of the parameters d_k [59].

4.4 Other Classes of Uncertainties and Stability Domains

The value set technique is a formidable analysis tool as long as we have an efficient way to draw it. In general, this is a difficult task, yet there are several cases in which this is possible. For instance, consider the so-called spherical families [7]

$$p(s) = \sum_{k=1}^{n} p_k s^k, \quad \|p\|_2 \leq \mu$$

where $\|p\|_2$ is the Euclidean norm of the coefficient vector. In this case, the value set is an ellipse. It is not difficult to see that we could generalize the norm to be any quadratic norm $\|p\| = \sqrt{p^\top Sp}$, with S positive definite.

The drawing issue restricts the class of uncertainty we can consider in practice. Conversely, there is no essential restriction on the stability domain.

Definition 4 Given a convex and closed domain $\mathscr{D}$ in the convex plane, we say that $p(s, q)$ is robustly $\mathscr{D}$-stable if its roots are inside the interior of $\mathscr{D}$.

Interesting cases of such domains are the translated left half-plane

$$\{s \in \mathscr{C} : \ Re(s) \leq -\alpha\}$$

the damping sector

$$\{s \in \mathscr{C} : \ Re(s) + \beta|Im(s)| \leq 0\}$$

and the unit circle

$$\{z \in \mathscr{C} : \ |z| \leq 1\}$$

The zero exclusion theorem can be extended without difficulties by requiring that the family has one $\mathscr{D}$-stable element and that the value set referred to as the Nyquist plot of the $\mathscr{D}$ domain does not hit the origin. For instance, exploring stability with respect to the unit circle would require considering the value set

$$\mathscr{V}_\theta = \left\{p(e^{j\theta}, q), \ \ q \in \mathscr{Q}\right\} \subset \mathbb{C}$$

parameterized in θ, with $0 \leq \theta \leq 2\pi$. Again the reader is referred to [7, 112] for further details. For a geometric view of the problem via optimization over semialgebraic sets, see [68].

4.5 Unstructured Uncertainties

In this subsection, we consider a robustness analysis for linear systems with constant uncertainties, where perturbations are present on the state matrix A_{un}. Let the state matrix be expressed as

$$A_{un} = A + L\Delta N$$

where A is a given Hurwitz matrix and Δ is a complex norm bounded matrix, subject to

$$\|\Delta\| \leq \alpha$$

while L and N are known matrices describing the effect of the uncertainty. This class of uncertainties is often referred to as *norm bounded*.

To assess robustness of the systems, one can consider the *complex stability radius* defined as the smallest perturbation for which the system has an eigenvalue on the stability boundary

$$r_{comp}(A, L, N){=}\{\text{inf } \|\Delta\|, \ \Delta \text{ complex} : \ A{+}L\Delta N \text{ has an eigenvalue in} : \ s{=}j\omega\},$$

which we can also write as follows:

$$r_{comp}(A, L, N) = \inf_{\omega}\{\text{inf } \|\Delta\|, \Delta \text{ complex} : \ det(I - \Delta G(j\omega)) = 0\}$$

where

$$G(s) = N(sI - A)^{-1}L$$

To prove the equivalence of the two previous expressions, note that $A_{un} = A + L\Delta N$ can be seen as the state matrix achieved by closing the loop with the transfer function

$G(s)$ with the perturbation block Δ, i.e.

$$\begin{aligned}\dot{x}(t) &= Ax(t) + Lw(t)\\ z(t) &= Nx(t)\\ w(t) &= \Delta z(t)\end{aligned}$$

The robust stability of this system is deeply related to the ∞ norm of the system (A, L, N), as can be easily understood by recalling the celebrated *small-gain theorem*, see Sect. 5.1, Theorem 13, saying that the feedback loop (and then A_{un}) is asymptotically stable if

$$\|G(s)\|_\infty < \alpha^{-1}$$

where $\|G(s)\|_\infty = \sup_{\omega \geq 0} \|G(j\omega)\|$. Conversely, if $\|G(s)\|_\infty \geq \alpha^{-1}$ then there exists an element Δ^* with norm not greater than α which renders A_{un} non-Hurwitz. Indeed, let $\bar{\omega}$ be the frequency associated with the maximum norm of $N(j\omega I - A)^{-1}L$ (such maximum is the H_∞ norm) and let $x \neq 0$ be such that

$$L^\top(-j\bar{\omega}I - A^\top)^{-1}N^\top N(j\bar{\omega}I - A)^{-1}Lx = \lambda^2 x$$

with $\lambda \geq \alpha^{-1}$. Define $y = (j\bar{\omega}I - A)^{-1}Lx$ to get

$$Lx = (j\bar{\omega}I - A)y, \quad \text{and} \quad L^\top(-j\bar{\omega}I - A^\top)^{-1}N^\top Ny = \lambda^2 x$$

Now let

$$\Delta^* = \frac{1}{\lambda^2}L^\top(-j\bar{\omega}I - A^\top)^{-1}N^\top, \quad \|\Delta^*\| = \frac{1}{\lambda} \leq \alpha$$

Then, with a few algebra,

$$(j\bar{\omega}I - A - L\Delta^* N)y = 0$$

The above reasoning proved that A_{un} is stable for all complex Δ of norm not exceeding α if and only if the norm of $N(sI - A)^{-1}L$ is less than α^{-1}. Therefore

$$r_{comp}(A, L, N) = \frac{1}{\|G(s)\|_\infty}$$

is the *complex stability radius*, which is the smallest norm of Δ capable of destabilizing A_{un}.

The next problem we consider is how to address the more difficult case in which Δ is assumed to be real, see [100]. The *real stability radius* for the perturbed matrix

$$A_{un} = A + L\Delta N$$

is defined as the largest norm of Δ *real* for which stability is ensured. The formal definition is as follows:

$$r_{real}(A, L, N) = \{\inf \|\Delta\|,\ \Delta \text{ real}:\ A + L\Delta N \text{ has an eigenvalue in}:\ s = j\omega\}$$

Reconsidering the previously introduced transfer function matrix $G(s)$ we have

$$r_{real}(A, L, N) = \inf_{\omega}\{\inf \|\Delta\|, \Delta \text{ real}:\ \det(I - \Delta G(j\omega)) = 0\}$$

For a complex matrix M, consider the quantity μ_{real} defined as

$$\mu_{real}(M) = \{\inf \|\Delta\|,\ \Delta \text{ real} : \det(I - \Delta M) = 0\}^{-1}$$

Then it holds true that

$$\mu_{real}(M) = \inf_{\gamma\in(0,1]} \sigma_2\left(\begin{bmatrix} Re(M) & -\gamma Im(M) \\ \gamma^{-1}Im(M) & Re(M) \end{bmatrix}\right)$$

where σ_2 is the second largest real singular value. Based on this result it can be seen that

$$r_{real}^{-1}(A, L, N) = \sup_{\omega} \inf_{\gamma\in(0,1]} \sigma_2\left(\begin{bmatrix} Re(G(j\omega)) & -\gamma Im(G(j\omega)) \\ \gamma^{-1}Im(G(j\omega)) & Re(G(j\omega)) \end{bmatrix}\right)$$

Note that the class of complex Δ's includes the real ones, hence, $r_{real} \geq r_{comp}$. Then the complex stability radius provides a more conservative value than the real one.

Example 10 Consider the system

$$\dot{x}(t) = (A + L\Delta N)x(t)$$

with

$$A = \begin{bmatrix} -1 & 2 \\ -2 & -3 \end{bmatrix},\quad L = \begin{bmatrix} 1 \\ 0 \end{bmatrix},\quad \Delta = \begin{bmatrix} \delta_1 & \delta_2 \end{bmatrix},\quad N = I$$

We first consider the real case, namely, the largest possible norm of a *real* perturbation Δ such that $A + L\Delta N$ remains Hurwitz (real stability radius), then we compare it with the complex case. To this aim note that

$$A + L\Delta N = \begin{bmatrix} -1 + \delta_1 & 2 + \delta_2 \\ -2 & -3 \end{bmatrix},$$

and that

$$\det[\lambda I - A - L\Delta N] = \lambda^2 + (4 - \delta_1)\lambda + 7 - 3\delta_1 + 2\delta_2$$

Therefore, stability is ensured if

$$\delta_1 < 4, \quad \delta_2 > 1.5\delta_1 - 3.5$$

By a graphical test, in the δ_1–δ_2 space, it can be shown that the circle centred in the origin of greatest radius is tangent to the line $\delta_2 = 1.5\delta_1 - 3.5$. The tangent point is the intersection of this line and the orthogonal one passing through the origin

$$\delta_2 = 1.5\delta_1 - 3.5, \quad \delta_2 = -2/3\delta_1$$

Then

$$\delta_1 = 21/13, \quad \delta_2 = -14/13$$

which means that the real stability radius is

$$r_{real}(A, L, N) = \sqrt{\delta_1^2 + \delta_2^2} = \frac{\sqrt{637}}{13} = 1.9415$$

which can be computed with the formula proposed before.

The complex stability radius is $1/\|G(s)\|_\infty$, where $G(s) = N(sI - A)^{-1}L$. We get

$$G(s) = \begin{bmatrix} s+3 \\ -2 \end{bmatrix} \frac{1}{s^2 + 4s + 7}.$$

Hence, denoting by $G^{\sim}(s) = G^{\top}(-s)$,

$$G^{\sim}(s)G(s) = \frac{13 - s^2}{(s^2 + 4s + 7)(s^2 - 4s + 7)}$$

Then

$$\|G(s)\|_\infty = \|\frac{\sqrt{13} + s}{s^2 + 4s + 7}\|_\infty = 0.519$$

As expected, the complex radius is not larger than the real one

$$r_{comp}(A, L, N) = \frac{1}{1.1232} = 1.9269 < 1.9415 = r_{real}(A, L, N)$$

The complex stability radius is related to *quadratic stability* of an uncertain system. We say that the set of matrices A_{un}, achieved by choosing a complex Δ with $\|\Delta\| < \alpha$ is quadratically stable if there exists a positive definite matrix P which satisfies

$$A_{un}^{\sim} P + P A_{un} < 0, \quad \forall \Delta, \quad \|\Delta\| \leq \alpha$$

where $A_{un}^{\sim}$ is the complex conjugate of A_{un} ($A_{un}^{\sim} = (A_{un}^{*})^{\top}$). The next theorem is a fundamental result because it links a frequency-domain condition with quadratic stability, which will be further developed later.

Theorem 11 *The set of matrices A_{un} is quadratically stable iff*

$$\|N(sI - A)^{-1}L\|_\infty < \alpha^{-1}$$

Proof Quadratic stability implies that any element in the family is Hurwitz. On the other hand, any A_{un} is Hurwitz for any Δ with norm smaller than α if and only if the norm of $N(sI - A)^{-1}L$ is less than α^{-1} (see small gain Theorem 13).

Conversely assume the H_∞ norm of system (A, L, N) is less than α^{-1} is equivalent (see again Sect. 5.1) to the existence of $P > 0$ satisfying the matrix inequality

$$A^\top P + PA + \alpha^2 PLL^\top P + N^\top N < 0$$

Then

$$\begin{aligned} 0 &> A^\top P + PA + \alpha^2 PLL^\top P + N^\top N \\ &\geq A^\top P + PA + \alpha^2 PLL^\top P + N^\top \alpha^{-2}\Delta^\sim \Delta N \\ &= A_{un}^\sim P + PA_{un} + \alpha^{-2}\left[(PL\alpha^2 - N^\top \Delta^\sim)(L^\top P\alpha^2 - \Delta N) +\right] \\ &\geq A_{un}^\sim P + PA_{un} \end{aligned}$$

The proof is then concluded.

4.6 Parameter-Dependent Lyapunov Function

In this subsection, we consider again the case of polytopic systems

$$\dot{x} = \sum_{i=1}^{N} \sigma_i A_i \, x = A(\sigma)x, \quad \sigma \in \Lambda_N$$

where Λ_N is the simplex

$$\Lambda_N := \left\{\lambda \in R^N \ : \ \sum_{i=1}^{N} \lambda_i = 1, \ \lambda_i \geq 0\right\}$$

We wish to find conditions ensuring Hurwitz stability of all the elements of the set.

As a preliminary observation, according to the results in Sect. 3.3, if there exists a symmetric matrix $P > 0$ that simultaneously solves the inequalities

$$A_i^\top P + PA_i < 0, \quad \forall i \tag{13}$$

then the system is quadratically stable, and hence, any element in the simplex is Hurwitz.

Requiring the existence of a single quadratic Lyapunov function is very conservative and it would ensure stability even under arbitrary variations of $A(\sigma(t))$ (and even under time-varying $\sigma(t)$ it would be conservative).

A technique tailored for constant uncertainties is achieved as follows. Consider a set of positive definite matrices $\{P_1, \cdots, P_N\}$ and consider the function

$$V(x) := x^\top (\sum_{i=1}^{N} \sigma_i P_i) x = x^\top P(\sigma) x$$

associated with the system (8). Note that $P(\sigma) > 0$. These types of functions are known as *parameter-dependent Lyapunov functions*. The next theorem holds.

Theorem 12 *Assume that there exist positive definite matrices* $\{P_1, \cdots, P_N\}$*, and a pair of matrices* V *and* G *of compatible dimensions which satisfy the inequalities*

$$\begin{bmatrix} A_i' G + G' A_i & P_i + A_i' V - G' \\ P_i + V' A_i - G & -V - V' \end{bmatrix} < 0, \quad \forall i \tag{14}$$

for all $i = 1, \cdots, N$*. Then the system (8) is robustly Hurwitz for any (constant)* $\sigma \in \Lambda_N$*.*

Proof Take $\sigma \in \Lambda_N$, and multiply the inequality (14) by the non-negative scalar σ_i and then sum up. We have

$$\begin{bmatrix} A(\sigma)^\top G + G^\top A(\sigma) & P(\sigma) + A(\sigma)^\top V - G^\top \\ P(\sigma) + V^\top A(\sigma) - G & -V - V' \end{bmatrix} < 0$$

Multiply this inequality on the left and on the right by $[I \ \ A(\sigma)']$ and $[I \ \ A(\sigma)']^\top$, respectively,[3] to get

$$A(\sigma)^\top P(\sigma) + P(\sigma) A(\sigma) < 0$$

The theorem is then proven. ■

Note that (14) implies

$$A_i^\top G + G^\top A_i < 0,$$

where G is not necessarily symmetric positive definite, in general. We thus see that the condition of Theorem 12 is less conservative than quadratic stability (13).

5 Small-Gain Theorems

In this section, we briefly recall important results concerning robust stability and robust performances of linear systems and special classes of nonlinear ones. The

[3] Note that if $Q > 0$ then $T^\top Q T > 0$ for any full column rank matrix T.

main theoretical tools rely on the concept of H_∞ norm of a linear system, which we now recall.

5.1 H_∞ Analysis

The H_∞ norm of a linear system with transfer function $G(s)$ is defined, in the frequency domain, as the worst (largest) value of the norm of $G(j\omega)$, i.e.

$$\|G(s)\|_\infty = \sup_\omega \|G(j\omega)\|$$

To be feasible, it is only required that $G(s)$ be well defined for all $s = j\omega$ (no poles on the imaginary axis). For $G(s)$ stable (analytic in $\mathrm{Re}(s) \geq 0$) it can be written also as

$$\|G(s)\|_\infty = \sup_{\mathrm{Re}(s)>0} \|G(s)\|$$

The space of proper rational matrix functions $G(s)$ that are analytic in the closed right half-plane is normally indicated by the symbol H_∞ and the above norm is the so-called H_∞ norm of $G(s) \in H_\infty$. The meaning of such a norm is important from an input–output perspective, since it represents the maximum amplification in the output of a (stable) system fed by a sinusoid at the 'worst' frequency, say $\bar{\omega}$. This characterization of $G(j\bar{\omega})$, with different denomination, is well known since the early days of what we can call the 'classical' automatic control theory, just after the Second World War. State-space characterisations of the H_∞ norm of $G(s) = C(sI - A)^{-1}B + D$ came much later, in terms of the associated Hamiltonian matrix and hence of the Riccati equation

$$A^\top P + PA + (PB + C^\top D)(\gamma^2 I - D^\top D)^{-1}(B^\top P + D^\top C) + C^\top C = 0 \quad (15)$$

Note that if we take $D = 0$ and we relax the equality to an inequality we get

$$A^\top P + PA + \gamma^{-2} PBB^\top P + C^\top C < 0$$

a condition we had anticipated in Theorem 11.

In particular, the H_∞ norm of $G(s)$ is less than $\gamma > 0$ if and only if A is Hurwitz, $\|D\| < \gamma$ and the Riccati equation admits a positive semidefinite solution P such that $A + B(B^\top P + C^\top D)(\gamma^2 I - D^\top D)^{-1}$ is Hurwitz stable, see [96]. This result is also known as *bounded real lemma* and has an interpretation in terms of Linear Matrix Inequalities. To be precise $\|G(s)\|_\infty < \gamma$ if and only if there exists $P > 0$ satisfying

$$\begin{bmatrix} A^\top P + PA & PB & C^\top \\ B^\top P & -\gamma^2 I & D^\top \\ C & D & -I \end{bmatrix} < 0 \quad (16)$$

The mathematical step from the Riccati equation (15) (relaxed to an inequality) to the Riccati inequality in the form of the LMI (16) hinges on the celebrated *Schur Lemma* for a block symmetric matrix inequality

$$V = \begin{bmatrix} Q & R \\ R^\top & S \end{bmatrix} > 0$$

saying that $V > 0$ is equivalent to $Q > 0$ and the Schur complement $S - R^\top Q^{-1} R > 0$ or $S > 0$ and the Schur complement $Q - RS^{-1}R^\top > 0$.

Assume now that the system (A, B, C, D) is affected by polytopic uncertainties with vertices (A_i, B_i, C_i, D_i), i.e.

$$(A, B, C, D) = \sum_{i=1^N} (A_i, B_i, C_i, D_i)\alpha_i(t), \quad \alpha_i(t) \geq 0, \quad \sum_{i=1}^{N} \alpha_i(t) = 1$$

and that $P > 0$ satisfies the LMIs (16) in the vertex matrices (A_i, B_i, C_i, D_i). Then, the cost $V(x) = x^\top Px$ is a common quadratic cost function for the uncertain system (A, B, C, D). Indeed, feasibility of the associated LMIs with a common P for each vertex of the system (with input w and output z) ensures that

$$\dot{V}(x(t)) + z(t)^\top z(t) < \gamma^2 w(t)^\top w(t)$$

so that, if $x(0) = 0$,

$$\frac{\int_0^\infty z^\top(t)z(t)dt}{\int_0^\infty w(t)^\top w(t)dt} < \gamma^2, \quad \forall 0 \neq w \in \mathscr{L}_2$$

As for norm bounded uncertain systems, consider

$$\begin{bmatrix} A & B \\ C & D \end{bmatrix} = \begin{bmatrix} A_n & B_n \\ C_n & D_n \end{bmatrix} + \begin{bmatrix} L_1 \\ L_2 \end{bmatrix} \Delta \begin{bmatrix} N_1 & N_2 \end{bmatrix}$$

with $\|\Delta\| < 1$, and assume that there exists $P > 0$ satisfying

$$\begin{bmatrix} A_n^\top P + PA_n & PB_n & C_n^\top & PL_1 & N_1^\top \\ B_n^\top P & -I & D_n^\top & 0 & N_2^\top \\ C_n & D_n & -I & L_2 & 0 \\ L_1^\top P & 0 & L_2^\top & -I & 0 \\ N_1 & N_2 & 0 & 0 & -I \end{bmatrix} < 0$$

Then $x^\top Px$ is a common quadratic Lyapunov function for the uncertain system. This is actually the condition (recall Theorem 11) for the H_∞ norm of the system $(\hat{A}, \hat{L}, \hat{N})$ to be less than 1, where

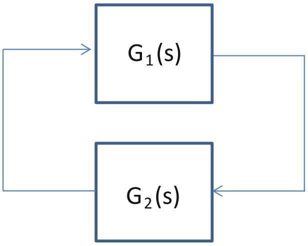


Fig. 20 Feedback scheme

$$\hat{A} = \begin{bmatrix} A_n & B_n \\ C_n & D_n \end{bmatrix}, \quad \hat{L} = \begin{bmatrix} L_1 \\ L_2 \end{bmatrix}, \quad \hat{N} = \begin{bmatrix} N_1 & N_2 \end{bmatrix}$$

The extension of quadratic stability results to synthesis problems is quite straightforward. For instance, dealing with robust state-feedback stabilization of the norm bounded uncertain system

$$\dot{x} = (A + L\Delta N_1)x + (B + L\Delta N_2)u$$

with $\|\Delta\| < 1$, we can consider the Lyapunov inequality associated with the closed-loop system

$$(A_n + B_n K)S + S(A_n + B_n K)^\top + (N_1 + N_2 K)^\top (N_1 + N_2 K) + L^\top L < 0$$

and give this inequality an LMI formulation letting $KS = W$ and exploiting the Schur lemma, yielding

$$\begin{bmatrix} A_n S + B_n W + W^\top B_n^\top + S A_n^\top & L^\top & S N_1^\top + W^\top N_2^\top \\ L & -I & 0 \\ N_1 S + N_2 W & 0 & -I \end{bmatrix} < 0 \qquad (17)$$

Therefore, if there exist $S > 0$ and W satisfying the LMI (17), then $u = Kx$ with $K = WS^{-1}$ is a quadratically stabilizing gain. The associated quadratic Lyapunov function is $x^\top S^{-1} x$. Notice that such a gain K is such that $A_n + B_n K + L\Delta(N_1 + N_2 K)$ is Hurwitz for any Δ (also complex) with $\|\Delta\| < 1$, and also that the transfer function

$$G_K(s) = (N_1 + N_2 K)(sI - A - BK)^{-1} L$$

has H_∞ norm less than 1. On the other hand, the closed-loop system coincides with the feedback connection between $G_K(s)$ and Δ.

The synthesis problems under uncertainties fall under the legacy of a very important result, called *small-gain theorem*, that provides a strong link between robust stabilization and H_∞. Considering Fig. 20, the following result can be proven.

Theorem 13 *Let $G_1(s) \in H_\infty$ be an assigned $p \times m$ transfer function and $G_2(s) \in H_\infty$ an arbitrary $m \times p$ transfer function with $\|G_2\|_\infty < \alpha \neq 0$. Then*

(i) The feedback connected system shown in Fig. 20 is stable for any $G_2(s)$ if $\|G_1(s)\|_\infty \leq \alpha^{-1}$.
(ii) If $\|G_1(s)\|_\infty > \alpha^{-1}$, there exists a transfer function $G_2(s)$ which destabilizes the feedback connected system shown in Fig. 20.

The H_∞ norm is an induced norm in the sense of the worst input–output ratio in the L_2 norm. Other norms can be associated with a system with different input–output characterizations. For instance, consider again the Riccati equation (15), but associated with a stable strictly proper system with transfer function $G(s) = C(sI - A)^{-1}B$. It is interesting to look at what happens if γ becomes arbitrarily large. It can be seen that for $\gamma \to \infty$ the following Lyapunov equation is obtained:

$$A^\top P + PA + C^\top C = 0$$

It is important here to remember that the trace of $B^\top PB$ corresponds to the square of the so-called H_2 norm of $G(s)$, say $\|G(s)\|$. It is easy to see that

$$\begin{aligned}\|G(s)\|^2 &= \frac{1}{2\pi}\int_{-\infty}^{\infty} \text{trace}(G(-j\omega)^\top G(j\omega))d\omega \\ &= \text{trace}(B^\top PB) = \sum_{i=1}^{m}\int_{0^-}^{\infty} y^{[i]}(t)^\top y^{[i]}(t)dt\end{aligned}$$

where $y^{[i]}(t)$ is the impulse response of the system when the initial state is 0 and the impulse is an impulse at the i-th input channel Three (the i-th column of $Ce^{At}B$). Here, we have used the fact that the solution of the Lyapunov equation is

$$P = \int_0^\infty e^{A^\top t}C^\top Ce^{At}dt$$

and the Parseval identity

$$\frac{1}{2\pi}\int_{-\infty}^{\infty} \text{trace}(G(-j\omega)^\top G(j\omega))d\omega = \int_0^\infty \text{trace}[B^\top e^{A^\top t}C^\top Ce^{At}B]dt$$

As a compromise between the two defined norms, very important in the mixed H_2–H_∞ control problem, it is possible to define the so-called γ-entropy of $G(s)$. To be precise, consider $G(s)$, strictly proper and stable, whose H_∞ norm is less than γ, and define the γ-entropy as follows:

$$I_\gamma(G) = -\frac{\gamma^2}{2\pi}\int_{-\infty}^{\infty} \ln\det\left[I - \frac{G(-j\omega)^\top G(j\omega)}{\gamma^2}\right]d\omega$$

This quantity is well defined thanks to the assumption that $\|G(s)\|_\infty < \gamma$. We can write

$$I_\gamma(G) = -\frac{\gamma^2}{2\pi}\int_{-\infty}^{\infty}\sum_{i=1}^{m}\ln\left[1-\frac{1}{\gamma^2}\sigma_i^2\left(G(j\omega)\right)\right]d\omega$$

where $\sigma_i(\cdot)$ denotes the i-th singular value.

Theorem 14 *Consider $G(s) = C(sI - A)^{-1}B$ and assume that A is Hurwitz and $\|G(s)\|_\infty < \gamma$, for some $\gamma > 0$. Moreover, let $\beta = \frac{\gamma^2}{\|G(s)\|_\infty^2}$. Then*

$$\|G(s)\|_2^2 \le I_\gamma(G) \le -\beta\log(1-\beta^{-1})\|G(s)\|_2^2$$

$$I_\gamma(G) = \text{trace}[B^\top PB]$$

where P is the stabilizing solution of the Riccati equation

$$A^\top P + PA + \gamma^{-2}PBB^\top P + C^\top C = 0$$

i.e. such that $A + \gamma^{-2}BB^\top P$ is Hurwitz.

Proof The squared H_2 norm can be written as follows:

$$\|G(s)\|_2^2 = \frac{1}{2\pi}\int_{-\infty}^{\infty}\sum_{i=1}^{m}\sigma_i^2\left(G(j\omega)\right)d\omega$$

Then, defining the function

$$f(x^2) = -\gamma^2\log\left(1-\frac{x^2}{\gamma^2}\right)$$

we have that

$$I_\gamma(G) = \frac{1}{2\pi}\int_{-\infty}^{\infty}\sum_{i=1}^{m}f\left(\sigma_i^2\left(G(j\omega)\right)\right)d\omega$$

Notice that $f(x^2) \ge x^2$, $\forall x$, and hence, the first conclusion is that

$$\|G(s)\|_2^2 \le I_\gamma(G)$$

Now let $r_i = \frac{\gamma^2}{\sigma_i^2(G(j\omega))}$ and notice that $r_i \ge \beta > 1$. Function $x\log(1 - x^{-1})$ is negative and monotonically increasing for $x > 1$. Therefore,

$$\log\left[1-\frac{1}{r_i}\right] \ge \frac{\beta}{r_i}\log\left[1-\frac{1}{\beta}\right]$$

and hence

$$
\begin{aligned}
I_\gamma(G) &= \frac{1}{2\pi}\int_{-\infty}^{\infty}\sum_{i=1}^{m} -\gamma^2 \log\left[1-\frac{1}{r_i}\right] d\omega \\
&\leq -\beta \log(1-\beta^{-1})\frac{1}{2\pi}\int_{-\infty}^{\infty}\sum_{i=1}^{m}\sigma_i^2\left(G(j\omega)\right) d\omega \\
&= -\beta \log(1-\beta^{-1})\|G(s)\|_2^2
\end{aligned}
$$

Finally, the condition $\|G(s)\|_\infty < \gamma$ is equivalent to the existence of the stabilizing solution $P \geq 0$ of the Riccati equation

$$A^\top P + PA + \gamma^{-2}PBB^\top P + C^\top C = 0$$

Letting $Y(s) = I - \gamma^{-2}B^\top P(sI-A)^{-1}B$, it turns out that

$$I - \gamma^{-2}G^{\sim}(s)G(s) = Y^{\sim}(s)Y(s)$$

and then, using the Cauchy integral formula

$$
\begin{aligned}
I_\gamma(G) &= -\frac{\gamma^2}{2\pi}\int_{-\infty}^{\infty} \log\det\left[Y(-j\omega)^\top Y(j\omega)\right] d\omega \\
&= \lim_{z\to\infty} -\frac{\gamma^2}{\pi}\int_{-\infty}^{\infty} \log\ \left|\det\left[Y(j\omega)\right]\right| \frac{z^2}{z^2+\omega^2} d\omega \\
&= \lim_{z\to\infty} -\gamma^2 z \log\ \left|\det\left[Y(z)\right]\right| \\
&= \lim_{z\to\infty} -\gamma^2 z \log\ \left|\det\left[I-\gamma^{-2}B^\top P(zI-A)^{-1}B\right]\right| \\
&= \lim_{z\to\infty} -\gamma^2 z \log\ \left|\det\left[I-\gamma^{-2}z^{-1}B^\top PB - \gamma^{-2}z^{-2}B^\top PA\sum_{k=0}^{\infty} z^{-k}A^k B\right]\right| \\
&= \lim_{z\to\infty} -\gamma^2 z \log\ \left|1-\gamma^{-2}z^{-1}\mathrm{trace}\left[B^\top PB\right] + \mathcal{O}(z^{-2})\right| \\
&= \lim_{z\to\infty} -\gamma^2 z\left[-\gamma^{-2}z^{-1}\mathrm{trace}\left[B^\top PB\right] + \mathcal{O}(z^{-2})\right] \\
&= \mathrm{trace}\left[B^\top PB\right]
\end{aligned}
$$

The proof is concluded. ■

For single-input single-output systems, it is possible to characterize $I_\gamma(G)$ in a probabilistic way, making reference to the transfer function of a feedback system

$$G_{cl}(s) = \frac{G(s)}{1-\Delta(s)G(s)}$$

where $G(s)$ is stable with$\|G(s)\|_\infty < \gamma$ and $\Delta(j\omega)$, for each ω, a random variable with $\Delta(j\omega_1)$ and $\Delta(j\omega_2)$ independent of each other if $\omega_1 \neq \omega_2$ and uniformly distributed in the disc of radius γ^{-1}. It can be proven that

$$I_\gamma(G) = E_\Delta(\|\frac{G(s)}{1-\Delta(s)G(s)}\|_2^2)$$

where E_Δ indicates the expectation with respect to distribution of Δ. Indeed, let $\Delta(j\omega) = \rho e^{j\theta}$ and $G(j\omega) = \lambda e^{j\phi}$, and note that the distribution of $\Delta(j\omega)$ is γ^2/π. Therefore

$$E_\Delta(\|\frac{G(j\omega)}{1-\Delta(j\omega)G(j\omega)}\|^2) = \frac{\gamma^2\lambda^2}{\pi}\int_0^{\gamma^{-1}} \rho \left[\int_0^{2\pi} \frac{1}{1+\lambda^2\rho^2-2\lambda\rho cos(\theta+\phi)} d\theta\right] d\rho$$

$$= 2\gamma^2\lambda^2 \int_0^{\gamma^{-1}} \frac{\rho}{1-\rho^2\lambda^2} d\rho = -\gamma^2 ln(1-\lambda^2\gamma^{-2})$$

Hence

$$E_\Delta(\|\frac{G(s)}{1-\Delta(s)G(s)}\|_2^2) = \frac{-\gamma^2}{2\pi}\int_{-\infty}^{\infty} ln(1-|G(j\omega)|^2\gamma^{-2}) d\omega = I_\gamma(G)$$

5.2 H_∞ *Design*

The control problem for linear time-invariant systems has been classically tackled in the frequency domain; see the classical scheme in Fig. 21. The aim is, loosely speaking, guaranteeing the *stability* of the control system and achieving *satisfactory performances*. Usually, such performances are evaluated in terms of the behaviour of suitable variables of interest to be specified according to the problem at hand and must be attained in spite of the disturbances acting on the system and of inaccurate knowledge of the process model. In general, the philosophy underlying the adopted synthesis procedure strongly affects the result: for instance, the choice of either ignoring or taking into account the *inaccurate* knowledge of the process model makes a great difference to the controller. Moreover, the design procedure significantly depends on the adopted description of the uncertainty. A design problem can be solved under *nominal conditions*, meaning that the process is supposed to be perfectly

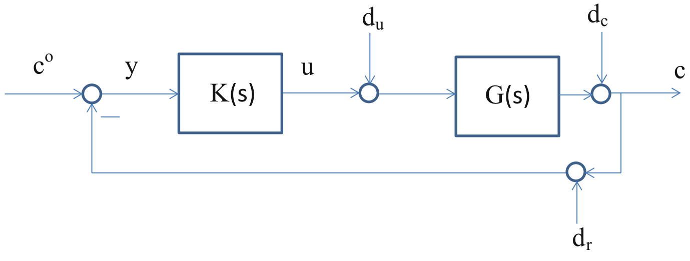


Fig. 21 Standard control problem

known, or can be tackled in a *robust way*, namely, that it is stated within various uncertainty scenarios for the plant.

A good solution to the design problem spontaneously calls for making small, in some suitable sense to be specified, the effects of the disturbances on the variables of interest. Thus, the desire is of *making small* one sensitivity function, say $\varphi(s)$, shaped in frequency by a *shaping function*, say $W(s)$, i.e.

$$\|W(s)\varphi(s)\|_\infty < 1$$

It is often realistic to assume that the process model, say $G(s)$, belongs to some specified set $\mathcal{G}$ rather than being perfectly known. Moreover, the so-called nominal model $G_n(s)$ is usually taken as an element of the set $\mathcal{G}$ and therefore viewed as a *first-order approximation* of the *true model* $G(s)$. Consistently, a description of the set $\mathcal{G}$ can be performed by *parametrizing* it by means of a transfer function $\varDelta(s)$ belonging to a suitable set $\mathcal{D}_\alpha$: the *perturbations* which $G_n(s)$ may undergo are then defined by the adopted parametrization and the structure of the set $\mathcal{D}_\alpha$.

Classical perturbations are those qualified only in terms of their *amplitude* as specified by the set

$$\mathcal{D}_\alpha := \{\bar{\varDelta}(s) \mid \bar{\varDelta}(s) \in H_\infty\ ,\ \ \|\bar{\varDelta}(s)\|_\infty < \alpha\} \tag{18}$$

Some particularly meaningful examples of parametrization of $\mathcal{G}$ are presented in the following equations:

$$\mathcal{G} := \{G(s) \mid G(s) = G_n(s) + \bar{\varDelta}(s)\} \tag{19}$$

$$\mathcal{G} := \{G(s) \mid G(s) = G_n(s)[I + \bar{\varDelta}(s)]\} \tag{20}$$

$$\mathcal{G} := \{G(s) \mid G(s) = [I + \bar{\varDelta}(s)]G_n(s)\} \tag{21}$$

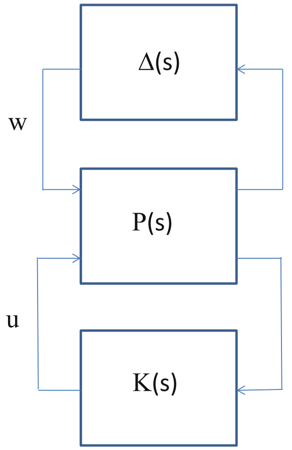
$$\mathcal{G} := \{G(s) \mid G(s) = [I - \bar{\varDelta}(s)]^{-1}G_n(s)\} \tag{22}$$

$$\mathcal{G} := \{G(s) \mid G(s) = [I - G_n(s)\bar{\varDelta}(s)]^{-1}G_n(s)\} \tag{23}$$

It is easy to verify that each of the sets (19)–(23) is suited to describe meaningful types of uncertainties in a fairly natural way. For instance, the set (22) can easily absorb right half-plane poles (as an example: $\bar{\varDelta}(s) = 10/(1+s)$, so that $[1 - \bar{\varDelta}(s)]^{-1} = (1+s)/(s-9)$.

The design problem in an uncertain environment consists of selecting a controller, say $K(s)$, which ensures stability as well as satisfactory performances not only in nominal conditions (e.g. $G(s) = G_n(s)$) but also when the plant undergoes *finite perturbations*. As for the basic stability requirement, a controller $K(s)$ is said to guarantee *robust stability* if, given a set $\mathcal{D}_\alpha$, the control system is stable for each $G(s) \in \mathcal{G}$. In a similar way, a controller $K(s)$ is said to guarantee *robust performances* if, given a set $\mathcal{D}_\alpha$, the control system satisfies some specified performance requirements for each $G(s) \in \mathcal{G}$, for a sensitivity function $\varphi(s)$, which includes the functions

Fig. 22 The standard three-block configuration



$$
\begin{aligned}
S(s) &= [I + G(s)K(s)]^{-1} \\
T(s) &= G(s)K(s)[I + G(s)K(s)]^{-1} \\
V(s) &= K(s)[I + G(s)K(s)]^{-1} \\
M(s) &= [I + G(s)K(s)]^{-1}G(s)
\end{aligned}
$$

Accordingly, the *robust sensitivity performance*, the *robust complementary sensitivity performance*, the *robust control sensitivity performance* and the *robust output sensitivity performance* are guaranteed if, given the sets $\mathscr{G}$ and $\mathscr{D}_\alpha$, the control system is stable for all $G(s) \in \mathscr{G}$ and

$$\|W_1(s)S(s)\|_\infty < 1\,, \quad \forall G(s) \in \mathscr{G} \tag{24}$$

$$\|W_2(s)T(s)\|_\infty < 1\,, \quad \forall G(s) \in \mathscr{G} \tag{25}$$

$$\|W_3(s)V(s)\|_\infty < 1\,, \quad \forall G(s) \in \mathscr{G} \tag{26}$$

$$\|W_4(s)M(s)\|_\infty < 1\,, \quad \forall G(s) \in \mathscr{G} \tag{27}$$

respectively.

It is possible to call for the simultaneous matching of two or even all the inequalities (24)–(27). On the other hand, under some circumstances, the controller might be required to guarantee robust stability together with satisfactory performances in nominal conditions only (therefore with $G_n(s)$ replacing $G(s)$ in the inequalities (24)–(27)).

The introduced design problems can all be reduced to a unique *standard problem* in the H_∞ context, described in the block configuration in Fig. 22 where $P(s)$ is the so-called *augmented system*, $K(s)$ is the *controller* to be designed, $\Delta(s)$ accounts for the uncertainties, z contains the performance variables, u the control inputs, y the measurement outputs and w the disturbances. The controller $K(s)$ in Fig. 21 which solves one of the design problems is the same controller that in Fig. 22 guarantees stability and the boundedness of the H_∞ norm of the transfer function T_{zw} from w to z. The interest in reformulating the original design problem in terms of the block structure of Fig. 22 lies in the fact that the augmented plant $P(s)$ depends only on the nominal plant $G_n(s)$, on the particular set $\mathscr{G}$ of the given perturbations and on the required performances of the control system.

The procedure underlying the passage from the standard control scheme in Fig. 21 to that of Fig. 22, i.e. the definition of the system $P(s)$ and signals w and z, turns out to be very simple when dealing with a design problem in nominal conditions ($\bar{\Delta}(s) = 0$), henceforth referred to as *nominal design*.

For instance, consider the problem of *nominal design with joint sensitivity and complementary sensitivity performance*. To this end, with reference to Fig. 21 (with $G(s) = G_n(s)$) and Fig. 22 (with $\Delta(s) = 0$), define $z := [y^\top W_1(s)^\top \;\; c^\top W_2(s)^\top]^\top$ and $w := c^o$. Then,

$$P(s) = \begin{bmatrix} W_1(s) & -W_1(s)G_n(s) \\ 0 & W_2(s)G_n(s) \\ I & -G_n(s) \end{bmatrix}$$

The solutions of control problems under norm bounded uncertainties hinge on the small gain Theorem 13. The first problem to be fulfilled is of course *robust stability*. Assume that one of the parameterizations of $\mathscr{G}$ in Eqs. (19)–(23) has been adopted to account for the uncertain knowledge of the plant. It is then obvious how important would be the choice of a controller $K(s)$ which guarantees closed-loop stability for an assigned set of perturbations, i.e. for an assigned value of the scalar variable α (recall the definition of $\mathscr{D}_\alpha$, Eq. (18)). Along these lines the control problem that spontaneously arises is a *robust stability* design problem. Such a problem is easily reformulated in that of determining (if any) a controller $K(s)$ that, with reference to the scheme of Fig. 21, guarantees stability and is such that the H_∞ norm of the transfer function T_{zw} is less than a suitable scalar β. The augmented plant $P(s)$ depends on the choice of the particular set $\mathscr{G}$, as now shown for the sets in Eqs. (19) and (21). For instance, considering the set $\mathscr{G}$ given by Eq. (19) (*additive perturbations*), define z as the input of $\bar{\Delta}(s)$, w as the output of $\bar{\Delta}(s)$, and observe that the control system of Fig. 21 is completely equivalent to that of Fig. 22 (with $\Delta(s) = \bar{\Delta}(s)$) if

$$P(s) = \begin{bmatrix} 0 & I \\ -I & -G_n(s) \end{bmatrix}$$

In view of Theorem 13, $K(s)$ guarantees stability for any $\bar{\Delta}(s) \in \mathscr{D}_\alpha$ if and only if $\|T_{zw}(s))\|_\infty < \beta := \alpha^{-1}$.

It is also easy to cope with the problem of *robust stability with nominal performances*. For instance, consider the control system depicted in Fig. 21 and assume that a description of the uncertainty, based on the set $\mathscr{G}$ in (21), is given, with the uncertain block $\Delta(s)$ shaped as follows:

$$\bar{\Delta}(s) = W_5(s)\Delta^*(s)W_4(s), \quad \|W_5(s)\|_\infty = 1, \quad \|\Delta^*(s)\|_\infty < 1 \qquad (28)$$

Moreover, assume that the design problem is that of determining a controller $K(s)$ which guarantees robust stability and the fulfilment, under nominal conditions, of a preassigned performance requirements, for instance, the sensitivity performance (24) under nominal conditions. Therefore, the problem is stated as one of determining $K(s)$ such that $\|W_1(s)S_n(s)\|_\infty < 1$ and $\|W_4(s)T_n(s)W_5(s)\|_\infty < 1$. Since $\|W_5(s)\|_\infty = 1$, it is apparent that the design specifications are met if

$$\left\| \begin{bmatrix} W_1(s)S_n(s) \\ W_4(s)T_n(s) \end{bmatrix} \right\|_\infty < 1,$$

i.e. if $\|T_{zw}(s))\|_\infty < 1$, with

$$z := \begin{bmatrix} z_1 \\ z_2 \end{bmatrix}, \quad P(s) := \begin{bmatrix} W_1(s) & -W_1(s)G_n(s) \\ 0 & W_4(s)G_n(s) \\ I & -G_n(s) \end{bmatrix}$$

The problem is then cast in the context illustrated in Fig. 22 with

$$\Delta(s) = \begin{bmatrix} 0 \\ \Delta^*(s) \end{bmatrix}$$

Things get more involved when the design problem is aimed at simultaneously achieving *robust stability and robust sensitivity performances*.

Again considering the sets (21) and (28),

$$S(s) = [I + G(s)K(s)]^{-1} = S_n(s)[I + W_5(s)\Delta^*(s)W_4(s)T_n(s)]^{-1},$$

the robust sensitivity performance can be expressed as

$$\|W_1(s)S_n(s)[I + W_5(s)\Delta^*(s)W_4(s)T_n(s)]^{-1}\|_\infty < 1$$

$$\forall \Delta^*(s) \in H_\infty, \quad \|\Delta^*(s)\|_\infty < 1 \tag{29}$$

whereas the robust stability requirement is stated as

$$\|W_4(s)T_n(s)W_5(s)\|_\infty < 1. \tag{30}$$

It is easy to verify that Eqs. (29) and (30) are satisfied if

$$\left\| \begin{bmatrix} W_1(s)S_n(s) \\ W_4(s)T_n(s) \end{bmatrix} \right\|_\infty < \frac{\sqrt{2}}{2},$$

i.e. $\|T_{zw}(s)\|_\infty < \frac{\sqrt{2}}{2}$ for the augmented plant $P(s)$ as above. Notice that the simultaneous request of both robust stability and robust performance has lowered, not really surprisingly, the bound on the value of the norm.

The foregoing discussion has shown that a number of meaningful control problems can be treated in a unified fashion. As a matter of fact, it has been shown that they are all amenable to the problem of synthesizing a controller $K(s)$ which stabilizes the control system in Fig. 21 and is such that the H_∞ norm of the transfer function $T_{zw}(s)$ from the input w to the output z is less than a prescribed attenuation level $\gamma > 0$. To this aim, first of all it is convenient to introduce the following time-domain description of the process under control (augmented plant), which will be frequently quoted in the development of the present chapter.

$$\dot{x} = Ax + B_1 w + B_2 u \tag{31}$$
$$z = C_1 x + D_{11} w + D_{12} u \tag{32}$$
$$y = C_2 x + D_{21} w + D_{22} u \tag{33}$$

The controller is constrained to be a finite-dimensional, time-invariant, linear system, described by

$$\dot{\xi} = F\xi + Gy \tag{34}$$
$$u = H\xi + Ey \tag{35}$$

Hence,

$$P(s) = \left[\begin{array}{c|cc} A & B_1 & B_2 \\ \hline C_1 & D_{11} & D_{12} \\ C_2 & D_{21} & D_{22} \end{array}\right] \;, \quad K(s) = \left[\begin{array}{c|c} F & G \\ \hline H & E \end{array}\right]$$

Of course, the feedback connection of system (31)–(33) with system (34)–(35) must be well defined. For such a condition to be verified, it is necessary that

$$\det[I - ED_{22}] \neq 0 \tag{36}$$

so that the algebraic loop which is created by the insertion of the controller is automatically solvable.

Notice that $T_{zw}(s) \in H_\infty$ entails that such a function is stable: this objective is obviously satisfied if the internal stability of the closed-loop system is ensured, i.e. if $K(s)$ in (34), (35) internally stabilizes system (31)–(33). This is equivalent to requiring the stability of the dynamic matrix of the closed-loop system, i.e.

$$Re(\lambda_i(A_F)) < 0 \;, \quad \forall i \tag{37}$$

$$A_F := \begin{bmatrix} A + B_2(I - ED_{22})^{-1}EC_2 & B_2(I - ED_{22})^{-1}H \\ G[I + D_{22}(I - ED_{22})^{-1}E]C_2 & F + GD_{22}(I - ED_{22})^{-1}H \end{bmatrix} \tag{38}$$

The controller $K(s)$ is *admissible* in H_∞ for $P(s)$ if conditions (36)−(38) are verified.

The main result concerns the resolution of two precise points, following the scheme formally presented in Problem 1 below: the existence of a controller such that $\|T_{zw}(s)\|_\infty < \gamma$ and the parametrization of all such controllers. Problem 1 refers to the feedback configuration of Fig. 21 and to the set $\mathscr{F}_{\infty\gamma}$ which represents the family of all admissible controllers in H_∞ for $P(s)$ such that $\|T_{zw})\|_\infty < \gamma$.

Problem 1 (Standard problem in H_∞) Let a positive scalar γ be fixed.

(a) Find a necessary and sufficient condition for the existence of a controller $K(s)$ which is admissible in H_∞ for $P(s)$ and such that $\|T_{zw}(s)\|_\infty < \gamma$.
(b) Find a family of controllers $\mathscr{F}_{\infty\gamma r} \subseteq \mathscr{F}_{\infty\gamma}$ whose elements generate the whole set of functions $T_{zw}(s)$ which are generated by the elements of $\mathscr{F}_{\infty\gamma}$.

Three main problems have been associated with a particular structure of the system $P(s)$: they are referred to as the *full information* problem, the *output estimation* problem and the *partial information* problem. In more detail, the last problem will be tackled by exploiting the solutions of the former ones, which, in turn, are strictly related to each other by structural relations: the complete picture puts into sharp relief important duality and separation properties. The main reference providing a complete state-space solution via Riccati equations for the nominal case, i.e. $\Delta = 0$ in Fig. 22, is [45]. To be precise, under the assumptions that $D_{11} = 0$, $D_{22} = 0$, $D_{12}^\top C_1 = 0$, $D_{21}^\top B_1 = 0$, $D_{12}^\top D_{12} = I$, $D_{21} D_{21}^\top = I$, a controller $R(s)$ such that the closed-loop system is stable and $\|T_{zw}(s)\|_\infty < \gamma$ exists if and only if there exist stabilizing solutions $P \geq 0$ and $S \geq 0$ of Riccati equations

$$0 = A^\top P + PA + \frac{1}{\gamma^2} P B_1 B_1^\top P - P B_2 B_2^\top P + C_1^\top C_1$$
$$0 = AS + SA^\top + \frac{1}{\gamma^2} S C_1 C_1^\top S - S C_2 C_2^\top S + B_1 B_1^\top$$

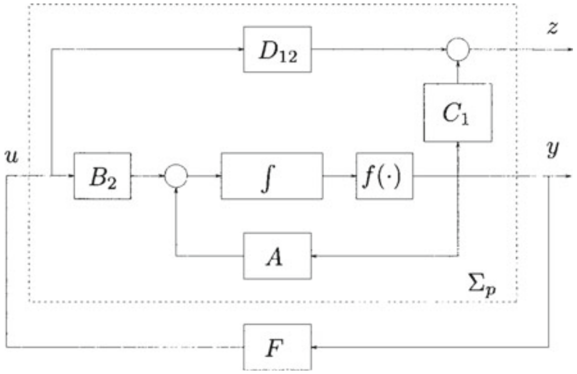
with the constraint $\|PS\| < \gamma$. One controller $K(s)$ is then given by

$$\dot{\hat{x}} = \left(A - B_2 B_2^\top P + \frac{1}{\gamma^2} B_1 B_1^\top P - (I - \frac{1}{\gamma^2} SP)^{-1} S C_2^\top C_2 \right) \hat{x} + (I - \frac{1}{\gamma^2} SP)^{-1} S C_2^\top y$$
$$u = -B_2^\top P \hat{x}$$

The well-known separation principle (valid for H_2 problems) is eventually lost, and recovered for $\gamma \to \infty$. Interestingly, it is also possible to parametrize the class of performant controllers with a free stable parameter.

As apparent from the simple examples above, in most problems the solution is conservative, in the sense that it encompasses uncertainties, $\Delta(s)$ in Fig. 22 that are unstructured, i.e. $\Delta(s)$ is a unique block with bounded norm. The so-called μ-synthesis, [106], generalizes the theory to cope with structured uncertainties (for instance, block-diagonal $\Delta(s)$). Even more, in the paper [107] the classical μ-synthesis tools are generalized to the integral quadratic constraint (IQC) framework.

Fig. 23 Persidiskii's system



This provides a systematic design of robust controllers based on an iteration of standard nominal controller synthesis with general dynamic multipliers, thus enabling to perform robust controller synthesis for a significantly large class of uncertainties, like sector-bounded and slope restricted nonlinearities, time-varying parametric uncertainties and uncertain time-varying time delays, both with bounds on the rate-of-variation.

Many directions were pursued trying to enrich the corpus of available theoretical results on robust control design and numerical analysis of the solutions. One interesting problem is the so-called *mixed* H_2–H_∞ design problem, aiming at finding a compensator capable to minimize the H_2 norm between a pair of input–output variables while keeping the H_∞ norm between (another or the same) pair of input–output variables less than a given attenuation level. The two problems, separately taken, are amenable to be solved via convex programming. Unfortunately, convexity is generally lost when taking into account jointly the two objectives. The *minimum entropy* design approach provides a suboptimal solution to the mixed problem [88], whereas, under suitable assumptions, the *blending control* approach [21] results in the optimal solution of the mixed problem at the expense of increased dimension of the controller.

5.3 Nonlinear Perturbations

In this subsection nonlinear robust stability and performance are considered. In this topic, advantage is taken of the a priori knowledge of the class of nonlinear perturbations acting on the open-loop model. Two main classes are of importance, namely, multiplicative and additive nonlinear perturbations, leading to what we call Persidiskii and Lur'e robust design procedures. In both cases, the control structure is assumed to be linear and the whole state vector is available for feedback.

5.3.1 Persidiskii Design

Consider the robust control design of a class of nonlinear systems subject to state-dependent nonlinear perturbations called *multiplicative* perturbations, see Fig. 23. The proposed design procedure will be expressed in terms of convex programming problems only. The open-loop system is subject to a class of perturbations such that, when they occur, the whole state vector x changes to $f(x)$. Then, the perturbation occurrence also changes the measured output y accordingly.

Assuming the state vector has dimension n and the nonlinear function $f(x)$ is not exactly known, the only available information is that it belongs to the uncertain domain $\mathscr{D}_f$ composed of all vector valued functions with the following properties:

(1) Each component of $f(x)$, namely, $f_j(x), j = 1, 2, \ldots, n$ is a real valued function such that

$$f_j(x) = f_j(x_j)$$

where $x_j \in R$ denotes the j-th component of the vector $x \in R^n$.

(2) Each component $f_j(x_j)$, $j = 1, 2, \ldots, n$ is such that

$$\begin{aligned} f_j(0) &= 0 \\ f_j(\xi)\xi &> 0 \ \forall \xi \neq 0 \in R \\ \int_0^\infty f_j(\xi) d\xi &= \infty \end{aligned}$$

The second condition says that the graph of $f(\cdot)$ must be contained in the first and third quadrants of the (f, ξ) plane. Since $f(x) = x \in \mathscr{D}_f$ the corresponding linear system is called the *nominal* system (Σ_n) and it has the following state-space representation:

$$\begin{aligned} \dot{x} &= Ax + B_2 u \ , \ \ x(0) = x_0 \\ z &= C_1 x + D_{12} u \end{aligned}$$

Adapting the previous design goals to cope with nonlinear systems stability and performance, we proceed trying to determine (if any exists) a linear state-feedback control law $u = Fx$ for Σ_n such that the origin $x = 0$ of the perturbed system Σ_p

$$\begin{aligned} \dot{x} &= (A + B_2 F) f(x) \ , \ \ x(0) = x_0 \\ z &= (C_1 + D_{12} F) f(x) \end{aligned}$$

is globally stable for all $f \in \mathscr{D}_f$. Furthermore, among all state-feedback gains with this property, find the one, namely, F_f, which solves the associated guaranteed cost control problem

$$\bar{\rho}_f(x_0) := \min \bar{\rho}(F, x_0) \tag{39}$$

where

$$\int_0^\infty z(t)^\top z(t)dt \leq \bar{\rho}(F, x_0) \ , \ \ \forall \ f \in \mathscr{D}_f$$

Similarly the minimum value of $\bar{\rho}(F, x_0)$ with respect to all F preserving stability is called the minimum guaranteed cost associated with the optimal feedback gain $F = F_f$. To accomplish the first goal concerning the robust stability of Σ_n we need to introduce the following important result.

To ease notation, for any square matrix P, the subscript ‘d’ indicates that $P = P_d$ is constrained to be a diagonal matrix.

Theorem 15 (Persidiskii theorem) *For any given state-feedback matrix F, suppose that there exists a positive definite diagonal matrix P_d such that*

$$0 \geq (A + B_2 F)^\top P_d + P_d(A + B_2 F) + Q \tag{40}$$

for some matrix $Q = Q^\top > 0$. Then, the origin $x = 0$ of the perturbed system Σ_p is globally asymptotically stable for all $f \in \mathscr{D}_f$.

In the above result, it is clear that matrix Q must be positive definite but does not need to have any particular structure. This degree of freedom is used in the next lemma to get the upper bound defined in (39).

Lemma 1 *Assume that, for all $f \in \mathscr{D}_f$, there exist n positive and finite parameters r_j, $j = 1, 2, \cdots, n$, such that*

$$\int_0^{x_j(0)} f_j(\xi)d\xi \leq \frac{r_j}{2} \ , \ \ j = 1, 2, \ldots, n$$

For any state-feedback control gain F such that there exists $P = P_d$ satisfying the matrix inequality (40), it is possible to choose $Q = Q^\top > 0$ such that the upper bound $\bar{\rho}(F, x_0)$ is given by

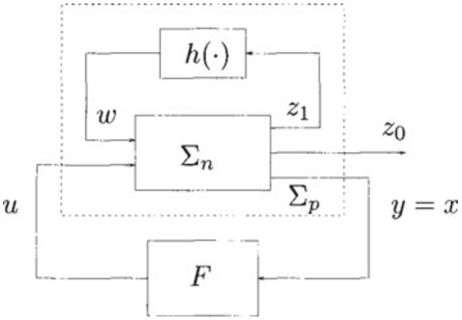
$$\bar{\rho}(F, x_0) := \sum_{j=1}^{n} P_{jj} r_j .$$

The guaranteed cost control problem (39) can be converted into a convex one by means of Schur complements. To this end, consider the affine matrix function which is defined by all pairs of matrices (X, Y) of appropriate dimensions with the first one being symmetric

$$\mathscr{A}_f(X, Y) := \begin{bmatrix} AX + B_2Y + XA^\top + Y^\top B_2^\top & XC_1^\top + Y^\top D_{12}^\top \\ C_1X + D_{12}Y & -I \end{bmatrix}$$

The following preliminary result is of particular importance towards the complete solution of the guaranteed cost control problem stated before.

Fig. 24 Lur'e system



Theorem 16 *Define the convex set*

$$\mathscr{C}_f := \left\{(X, Y) \ : \ X = X_d > 0 \ , \ \ \mathscr{A}_f(X, Y) < 0\right\}$$

The set of all state-feedback matrices F such that (40) holds for some $Q > 0$, denoted as $\mathscr{K}_f$, is alternatively given by

$$\mathscr{K}_f := \left\{YX^{-1} \ : \ (X, Y) \in \mathscr{C}_f\right\}$$

From this result, we are able to generate by means of a feasibility convex problem all gains belonging to the non-convex set $\mathscr{K}_f$. The elements of this set assure robust stability of the nominal closed-loop system against all nonlinear perturbations $f \in \mathscr{D}_f$. Besides, using Lemma 1 and defining the matrix

$$D := \text{diag}\left[\ \sqrt{r_1}, \sqrt{r_2}, \ldots, \sqrt{r_n}\ \right]$$

the elements of the set $\mathscr{C}_f$ allow for determination of the upper bound $\bar{\rho}(F, x_0)$ for all $F \in \mathscr{K}_f$ as

$$\bar{\rho}(F, x_0) = \sum_{j=1}^{n} P_{jj} r_j = \text{trace}\left[D^\top X^{-1} D\right]$$

valid for all $(X, Y) \in \mathscr{C}_f$ and $F = YX^{-1}$. From this fact the minimum guaranteed cost is readily calculated from

$$\bar{\rho}_f(x_0) = \inf\left\{\text{trace}\left[D^\top X^{-1} D\right] \ : \ (X, Y) \in \mathscr{C}_f\right\} \tag{41}$$

which is a convex programming problem. Once the global solution of the right-hand side of (41) is calculated, the corresponding state-feedback gain, optimal solution of the left-hand side of the same equation, is provided simply by $F_f = YX^{-1}$.

5.3.2 Lur'e Design

Let us consider now another important robust control design for the class of output-dependent nonlinear *additive* perturbations, see Fig. 24. The *nominal* system is denoted by Σ_n. The perturbed dynamic system, namely, Σ_p, is subject to the nonlinear perturbation $h(\cdot)$ which is a vector valued function not known a priori. The available information is that it belongs to the uncertain domain $\mathscr{D}_h$ composed of all functions having the following properties:

(1) The vector valued function $h(\cdot)$ is defined for all $\xi \in R^r$ and $h(\cdot) \in R^r$ where r is a positive integer less than or equal to the dimension n of the state vector $x \in R^n$.

(2) It is such that

$$\begin{aligned} h(0) &= 0 \\ h(\xi)^\top \xi &\leq 0 \;\; \forall\, \xi \in R^r \end{aligned}$$

The first condition imposes that, in Fig. 24, the vectors w and z_1 have the same dimension. The second one implies that the nonlinear function $-h(\cdot)$ belongs to the sector $[0, \infty)$. In fact, in the one-dimensional case, the graph of $-h(\xi)$ in the plane (h, ξ) is in the first and third quadrants. The state-space equations of the nominal open-loop system Σ_n, corresponding to $h(\cdot) = 0 \in \mathscr{D}_h$, are the standard ones

$$\begin{aligned} \dot{x} &= Ax + B_1 w + B_2 u \ , \quad x(0) = x_0 \\ z_0 &= C_0 x + D_0 u \\ z_1 &= C_1 x + D_{12} u \\ y &= x \end{aligned}$$

As before, the goal is to design a state-feedback control law, namely, $u = Fx$, such that the closed-loop perturbed system Σ_p obtained from Σ_n together with $w = h(z_1)$ possesses the following properties associated with its state-space representation:

$$\begin{aligned} \dot{x} &= (A + B_2 F)x + B_1 h(z_1) \ , \quad x(0) = x_0 \\ z_0 &= (C_0 + D_0 F)x \\ z_1 &= (C_1 + D_{12} F)x \end{aligned}$$

First, the origin $x = 0$ must be globally stable for all $h \in \mathscr{D}_h$. From all state-feedback gains with this property, select (if possible) one, namely, F_h, which solves the following guaranteed cost control problem:

$$\bar{\rho}_h(x_0) := \min \bar{\rho}(F, x_0) \tag{42}$$

where

$$\int_0^\infty z_0(t)^\top z_0(t)dt \leq \bar{\rho}(F, x_0)\ , \quad \forall\, h \in \mathscr{D}_h$$

The guaranteed cost control problem (42) is similar to the mixed H_2/H_∞ control problem. The existence of the nonlinear function $h \in \mathscr{D}_h$ does not allow us to express it in the frequency domain. Instead, the guaranteed cost is given in terms of an upper bound to the above integral of the controlled output. Accordingly, the exogenous signal is replaced by an arbitrary initial condition $x(0) \neq 0$.

Theorem 17 (Passivity theorem) *For any given state-feedback matrix F suppose that there exists a symmetric and positive definite matrix P such that*

$$0 \geq (A + B_2F)^\top P + P(A + B_2F) + Q \tag{43}$$

$$B_1^\top P = (C_1 + D_{12}F) \tag{44}$$

for some matrix $Q = Q^\top > 0$. Then, the origin $x = 0$ of the perturbed system Σ_p is globally asymptotically stable for all $h \in \mathscr{D}_h$.

Contrary to Theorem 15, the simultaneous feasibility of constraints (43) and (44) depends strongly on the particular choice of matrix $Q > 0$.

Even so, to be able to express the upper bound $\bar{\rho}(F, x_0)$ conveniently we need to impose

$$Q = (C_0 + D_0F)^\top (C_0 + D_0F) + \varepsilon I \tag{45}$$

with $\varepsilon > 0$ being an arbitrarily small parameter. Indeed, with this particular choice, introducing $V(x) = x(t)^\top Px(t)$, we have that

$$\begin{aligned}\dot{V}(x(t)) &\leq -x(t)^\top Qx(t)\\ &\leq -z_0(t)^\top z_0(t)\ , \quad \forall\, t \geq 0\end{aligned}$$

which after integration from $t = 0$ to $t = \infty$ provides

$$\int_0^\infty z_0(t)^\top z_0(t)dt \leq v(x(0)) = x_0^\top Px_0$$

Based on this, it is natural to define

$$\bar{\rho}(F, x_0) := x_0^\top Px_0 \tag{46}$$

as a valid upper bound for all $h \in \mathscr{D}_h$. Furthermore, let us denote by $\mathscr{K}_h$ the set of all state-feedback gains F such that with $Q > 0$ given in (45) both constraints (43) and (44) are simultaneously satisfied for some $P > 0$, and introduce the affine matrix functions defined for all pairs of matrices (X, Y) of appropriate dimension with the first one being symmetric

$$\mathscr{A}_h(X,Y) := \begin{bmatrix} AX + B_2Y + XA^\top + Y^\top B_2^\top & XC_0^\top + Y^\top D_0^\top \\ C_0X + D_0Y & -I \end{bmatrix}$$

and

$$\mathscr{B}_h(X,Y) := C_1X + D_{12}Y - B_1^\top$$

The following theorem provides a complete parametrization of the set $\mathscr{K}_h$ in terms of a convex set. It is the basis for the solution of the associated optimal guaranteed cost control problem (42).

Theorem 18 *Define the convex set*

$$\mathscr{C}_h := \{(X,Y) \ : \ X > 0 \ , \ \ \mathscr{A}_h(X,Y) < 0 \ , \ \mathscr{B}_h(X,Y) = 0\}$$

The set $\mathscr{K}_h$ is alternatively given by

$$\mathscr{K}_h := \left\{YX^{-1} \ : \ (X,Y) \in \mathscr{C}_h\right\}$$

We have now all elements to address the optimal guaranteed cost control problem (42). From Theorem 18 and (46), it reduces to the problem

$$\bar{\rho}_h(x_0) = \inf\left\{x_0^\top X^{-1} x_0 \ : \ (X,Y) \in \mathscr{C}_h\right\}$$

which is a convex programming problem. Its global optimal solution provides both the minimum guaranteed cost $\bar{\rho}_h(x_0)$ and the associated optimal state-feedback gain $F_h = YX^{-1}$.

5.4 L_1 Conditions

We briefly sketch some known robustness conditions of the small-gain type [40, 41] which are expressed in the time domain. Consider again the usual set-up

$$\begin{aligned} \dot{x}(t) &= Ax(t) + E\sigma(t) \\ \eta(t) &= Hx(t) \\ \sigma &= \Delta \circ \eta \end{aligned} \tag{47}$$

where A is an Hurwitz matrix and now Δ is an operator defined in the time domain. Let us introduce the following norm for signals with positive support (i.e. null for $t < 0$):

$$\|\eta\| \doteq \sup_{t \geq 0} \|\eta(t)\|_\infty$$

Then given a linear operator Δ, the ∞–to–∞ induced norm

$$\|\Delta\|_1 \doteq \sup_{\|\eta\|\leq 1} \|\Delta \circ \eta\|_\infty$$

is called the L_1 norm. The question now is whether the loop remains stable for a bounded Δ.

As a first step we consider the norm of the operator (A, E, H). Let $W(t)$ be the impulse response

$$W(t) = He^{At}E$$

Then

$$\|(A, E, H)\|_1 = \max_i \sum_j \int_0^\infty |W_{ij}(t)|dt$$

where the integral converges, thanks to the assumed asymptotic stability of A. This norm can be computed via numerical integration.

Define the quantity

$$\rho_{max} = \sup \rho > 0 : \quad \text{the loop (47) is robustly stable, with } \|Delta\|_1 < \rho$$

The following theorem holds, which can be considered the L_1 version of the small-gain theorem (Theorem 13).

Theorem 19 $\rho_{max} = \|(A, E, H)\|_1^{-1}$.

Let us briefly comment on the case of an SISO system. We have that

$$\|W\|_\infty = \sup_{\omega\geq 0} |H(j\omega I - A)^{-1}E| \leq \|(A, E, H)\|_1^{-1}$$

The property is easily proved by noticing that

$$|W(j\omega)| = \left|\int_0^\infty e^{j\omega t} W_{ij}(t)dt\right| \leq \int_0^\infty |e^{j\omega t} W_{ij}(t)|dt = \int_0^\infty |W_{ij}(t)|dt$$

where, with a slight abuse of notation, $W(s)$ is the transfer function of the open loop system. This basically implies that the small-gain robust stability conditions based on L_1 norm are typically more conservative than those based on $\mathscr{H}_\infty$ norms.

To explain why this criterion can be very conservative, we consider the case of the perturbation Δ to which we add a delay operator

$$D_\tau(\Delta \circ \eta) = [\Delta \circ \eta](t - \tau)$$

It is apparent that the delay does not change the norm. Hence if the criterion above holds, then the loop remains stable with arbitrary delay. This is quite a strong condition, in many cases unrealistic.

6 Related Topics

Parameter modulation in active control scenarios is a very up-to-date research area with the aim of improving the overall plant performance. In this section, we provide a very brief overview of the analysis of linear systems with either switching or periodic parameters.

6.1 Switching Systems

There is an enormous interest in dynamic systems whose behaviour can be described mathematically using a mixture of logic based switching and difference/differential equations. This interest has been primarily motivated by the realization that many man-made systems, and some physical systems, may be modelled using such a framework. Such systems are often referred to as switching systems, and arise frequently in communication networks, control theory, biology and many stochastic systems that can be described as an iterated function system. The link to systems with time-varying polytopic uncertainties has already been discussed in previous sections. Here, we briefly address stability and stabilization of switched systems. Important references are the monographs [81, 111], and the recent ones for positive switched systems [22, 25].

A switching system is a system of the form

$$\dot{x}(t) = f_\sigma(x(t)) \tag{48}$$

where the signal $\sigma(t)$ belongs to

$$\sigma(t) \in \Sigma = \{1, 2, \ldots, q\}$$

a finite set. By their nature they are discontinuous, so the existence of a solution is an issue, unless we admit that $\sigma(t)$ cannot have two commutations which differ in time less than a value $\tau > 0$, that is called dwell time. In this case a solution obviously exists if the functions f_i are regular.

A special interesting case is the linear one

$$\dot{x}(t) = A_\sigma x(t) \tag{49}$$

with A_i assigned matrices.

As far as $\sigma(t)$ is concerned, we have two possibilities.

- $\sigma(t)$ is an arbitrary signal (a disturbance);
- $\sigma(t)$ is a governable signal (a control).

The two cases are deeply different, and the latter case is much more difficult to study.

Let us consider the arbitrary signal case. We can associate with the switching system the following polytopic system

$$\dot{x}(t) = \sum_{i-1}^{q} w_i(t) f_i(x(t)) \quad w_i > 0, \quad \sum_{i=1}^{q} w_i = 1 \tag{50}$$

If we assume that $f(0) = 0$, it turns out that the stability of the 0 states of (48) and (50) are equivalent. In particular in the linear case we have the following result, which coincides with Theorem 4, already provided in Sect. 3.4 for polytopic systems.

Theorem 20 *The following statements are equivalent:*

- *The switching system (49) is asymptotically stable under arbitrary switching.*
- *The polytopic system (8)–(9) is asymptotically stable.*

These results are due to Molchanov and Pyatnitskii [85–87].[4]

The above necessary and sufficient conditions are very hard to be assessed, in general. Therefore, many sufficient conditions have been carried out to find a common Lyapunov function, that we know exists if and only if the system is stable under arbitrary switching, see also [108]. Interestingly, it was shown that a homogeneous polynomial Lyapunov function (of sufficiently high degree) exists if and only if the switching system is stable, see [35]. LMI techniques are also amenable to be used in the twin problem of assessing stability under a 'hard' dwell time τ, meaning that the switching signal is arbitrary but the interval between two successive jumps is lower bounded by $\tau > 0$. For such a problem, given τ it is clear that stability is always guaranteed with τ sufficiently high, being the single dynamical systems stable, and therefore, a sensible problem is that of finding the infimum τ guaranteeing stability. Piecewise quadratic Lyapunov functions can be used, see [36, 52]. A necessary and sufficient condition, along the lines of piecewise polyhedral Lyapunov functions is provided in [19], whereas the extension to piecewise homogeneous polynomial Lyapunov functions (manageable with LMIs) is in [35]. For the dwell time problem of nonlinear switching systems, see [38].

The theory can be extended to state-feedback stabilization [15, 18], namely, to systems of the form

$$\dot{x}(t) = A_\sigma x(t) + B_\sigma u$$

Clearly the equivalence of the stability of (49) and (8) allows us to use the same efficient LMI techniques for stability and stabilization.

The problem of stabilization is much more difficult. An explanation is that, even in the linear case, stabilizability does not imply the existence of a convex Lyapunov function [20]. There are examples of systems which can be stabilized adopting a switching rule $\sigma(x)$ but which do not admit convex Lyapunov functions.

A powerful sufficient condition is given by the following theorem.

[4] Notwithstanding the fact that some literature unfairly attributes the result to much more recent contributions.

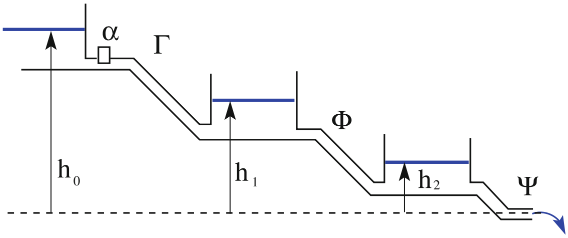


Fig. 25 Tank system

Theorem 21 *Assume that there exists a function $\bar{f}(x)$ such that*

$$\bar{f}(x) \in conv\{f_i(x)\}$$

the convex hull of the point $f_i(x)$, and that the system

$$\dot{x}(t) = \bar{f}(x(t))$$

has an equilibrium in 0, $\bar{f}(0) = 0$, *that is globally asymptotically stable and admits a Lyapunov function $V(x)$ such that*

$$\dot{V}(t) = \nabla V(x)\bar{f}(x(t)) \leq -\phi(x)$$

with $\phi(x)$ regular and negative definite. Then the switching system is stabilizable with the switching rule

$$\hat{\sigma}(x) = \arg\min_{i \in \Sigma} \nabla V(x) f_i(x)$$

Proof Since $\bar{f}(x) = \sum_{i=1}^{q} w_i f_i(x)$

$$-\phi(x) \geq \dot{V}(t) = \nabla V(x)\bar{f}(x(t)) = \nabla V(x) \sum_{i-1}^{q} w_i f_\sigma(x) = \sum_{i-1}^{q} w_i[\nabla V(x) f_i(x)]$$

Hence, being w_i non-negative and summing up to one, there exists at least one element i which satisfies $\nabla V(x) f_i(x) \leq -\phi(x)$. So if we take the minimum $\hat{\sigma}(x)$

$$\nabla V(x) f_{\hat{\sigma}(x)}(x) \leq -\phi(x)$$

■

As a corollary, we have that, in the linear case, the existence of a Hurwitz convex combination in the set of matrices A_i implies stabilizability. This condition is not necessary [81].

Note that all the individual systems associated with the functions f_i might be unstable. Moreover, f_i might not even have 0 as equilibrium state $f_i(0) \neq 0$, only $\bar{f}(0) = 0$ is required.

Example 11 Consider the tank system represented in Fig. 25. The equations are

$$\begin{aligned}\dot{h}_1 &= -\Phi(h_1 - h_2) + \alpha\Gamma(h_0 - h_1)\\ \dot{h}_2 &= +\Phi(h_1 - h_2) - \Psi(h_2)\end{aligned}$$

where h_1 and h_2 are the levels of the two tanks, h_0 is a constant level of the reservoir, $\alpha \in \{0, 1\}$ is a switching signal (an on–off valve). Functions Γ, Φ and Ψ are assumed to be unknown, monotonically increasing smooth functions, which represent the flows as functions of the levels. We assume that $\Gamma(0) = 0$, $\Phi(0) = 0$ and $\Psi(0) = 0$.

Let $\bar{\alpha}$ such that $0 < \bar{\alpha} < 1$ be an intermediate level of the switching parameter and let $\bar{h}_1$ and $\bar{h}_2$ be the equilibrium levels which would correspond to $\bar{\alpha}$ (which is clearly unfeasible). In the new variables $x_1 = h_1 - \bar{h}_1$ and $x_2 = h_2 - \bar{h}_2$, the equations become

$$\begin{aligned}\dot{x}_1 &= -\Phi(\bar{h}_1 + x_1 - \bar{h}_2 - x_2) + \bar{\alpha}\Gamma(h_0 - \bar{h}_1 - x_1) - (\alpha - \bar{\alpha})\Gamma(h_0 - \bar{h}_1 - x_1)\\ \dot{x}_2 &= +\Phi(\bar{h}_1 + x_1 - \bar{h}_2 - x_2) - \Psi(\bar{h}_2 + x_2)\end{aligned}$$

If we could apply the signal $\alpha = \bar{\alpha}$ by construction, the equilibrium would be $(0, 0)$. Note that at the equilibrium we have

$$\Phi(\bar{h}_1 - \bar{h}_2) = \bar{\alpha}\Gamma(h_0 - \bar{h}_1) \quad \text{and} \quad \Phi(\bar{h}_1 - \bar{h}_2) = \Psi(\bar{h}_2)$$

Consider the fake system with $\alpha = \bar{\alpha}$ and the candidate Lyapunov function $V(x_1, x_2) = (x_1^2 + x_2^2)/2$. The Lyapunov derivative is, after adding a term which is zero in view of the relations above,

$$\begin{aligned}\dot{V}(x_1, x_2) &= -\Phi(\bar{h}_1 - \bar{h}_2 + x_1 - x_2)(x_1 - x_2) + \bar{\alpha}\Gamma(h_0 - \bar{h}_1 - x_1)x_1 - \Psi(\bar{h}_2 - x_2)x_2\\ &\quad + \underbrace{\Phi(\bar{h}_1 - \bar{h}_2)(x_1 - x_2) - \bar{\alpha}\Gamma(h_0 - \bar{h}_1)x_1 + \Psi(\bar{h}_2)x_2}_{=0} =\\ &= -\left[\Phi(\bar{h}_1 - \bar{h}_2 + x_1 - x_2) - \Phi(\bar{h}_1 - \bar{h}_2)\right](x_1 - x_2)\\ &\quad -\bar{\alpha}\left[\Gamma(h_0 - \bar{h}_1) - \Gamma(h_0 - \bar{h}_1 - x_1)\right]x_1 - \left[\Psi(\bar{h}_2 + x_2)x_2 - \Psi(\bar{h}_2)\right]x_2\\ &= -\bar{\phi}(x_1, x_2)\end{aligned}$$

It is not difficult to see that $\bar{\phi}(x_1, x_2)$ is positive definite. Unfortunately, $\alpha = \bar{\alpha}$ cannot be actuated. However, the derivative with the true signal $\alpha \in \{0, 1\}$ differs from the original expression just by the term $x_1(\alpha - \bar{\alpha})\Gamma(h_0 - \bar{h}_1 - x_1)$

$$\dot{V}(x_1, x_2) = -\bar{\phi}(x_1, x_2) + x_1(\alpha - \bar{\alpha})\Gamma(h_0 - \bar{h}_1 - x_1)$$

So assuming that the term $\Gamma(h_0 - \bar{h}_1 - x_1) > 0$ (this physically means that the level of the first tank cannot be greater than the level of the reservoir) the switching law, minimizing the derivative yields

$$\alpha = \begin{cases} 0 & \text{if } x_1 > 0 \\ 1 & \text{if } x_1 < 0 \end{cases}$$

This law would introduce chattering, which would be not suitable in the application. The problem is solved by introducing a dwell time. This control does not require the knowledge of the functions and works quite well, in practice. Experimental results are shown in [18], section 9.4. ■

Remark 4 The reservoir system is asymptotically stable in both configurations $\alpha = 0$ and $\alpha = 1$, so in principle the problem seems a trivial one (stabilizing a stable system). This is not quite true, because we can achieve some performance improving the convergence speed. Assume that

$$\nabla V(x)\bar{f}(x) \leq -\beta V(x)$$

with $\beta > 0$. This ensures β contractivity of the system $\dot{x} = \bar{f}(x)$ in the convex hull, i.e.

$$V(x(t)) \leq -e^{-\beta t} V(x(0))$$

Theorem 21 can be equivalently stated in terms of β contractiveness: if this condition is ensured for the system $\dot{x} = \bar{f}(x)$, then the switching law ensures the same speed of convergence. This can be seen also for the suspension system, although we do not develop this case here.

A powerful method for stabilization of switched systems is provided by the so-called Lyapunov–Metzler inequalities. The theory was first presented in [52] and then extended to nonlinear systems in [38]. The idea, for switching linear systems, is to use an 'argmin' switching strategy to drive the system state in the steepest negative direction of the directional derivative of the Lyapunov function. To be precise, consider the switched system (49) and assume that there exist $P_i > 0$ and a Metzler matrix $\Lambda = \Lambda_{ij}$ with $\mathbf{1}^\top \Lambda^\top$ such that

$$A_i^\top P_i + P_i A_i + \sum_{j=1}^{q} \lambda_{ij} P_j < 0, \quad \forall i.$$

Then

$$\sigma(t) = \arg\min_i x(t)^\top P_i x(t) \tag{51}$$

is stabilizing. A nice characteristic of such inequalities is that they are identical to the inequalities associated with mean square stability of a Markov jump system with $\sigma(t)$ being a Markov process with infinitesimal transition matrix equal to Λ. This parallelism has nice consequences in the interpretation of the argmin switching rule with respect to the stochastic jumps induced by the equivalent Markovian system. The introduction of a cost to be minimized

$$J = \int_0^\infty x(t)^\top Q_\sigma(t) x(t) dt, \quad Q \geq 0$$

leads to Lyapunov–Metzler inequalities

$$A_i^\top P_i + P_1 A_i + \sum_{j=1}^{N} \lambda_{ij} P_j + Q_i < 0, \quad \forall i \tag{52}$$

that are amenable to provide a 'surrogate' of the Hamilton–Jacobi–Bellman equations, providing, with the above switching rule (51), a suboptimal cost that can be very close to the optimal one; see [22] for applications in the context of positive switched systems. For the system

$$\dot{x}(t) = A_{\sigma(t)} x(t) + Bw(t) \tag{53}$$
$$z(t) = E_{\sigma(t)} x(t) \tag{54}$$

with initial state $x(0)$ one can define the H_2 norm

$$J = \sum_{k=1}^{m} \int_0^\infty z^{[k]}(t)^\top z^{[k]}(t) dt \tag{55}$$

where $z^{[k]}(t)$ is the output response to an impulse at the k-th channel of $w(t)$. The minimization of J via a switching control law is a formidable complicated problem. The Lyapunov–Metzler approach provides a suboptimal control. Indeed, it can be proven that the switching law (51) based on P_i satisfying (52) with $Q_i = E_i^\top E_i$ is such that

$$J < \min_i \operatorname{trace}\left(B^\top P_i B\right)$$

The Lyapunov Metzler approach can be extended to the partial measurement case (only a linear combination of the state is available as measurement), as proved in [55]. Consider system (53), (54) with

$$y(t) = C_{\sigma(t)} x(t) + Dw(t) \tag{56}$$

with $x(0)$. The stabilizing switching rule is a functional of y via a suitable filter

$$\dot{\hat{x}}(t) = \hat{A}_{\sigma(t)} \hat{x}(t) + \hat{B}_{\sigma(t)} y(t) \tag{57}$$

where

$$\hat{B}_i = V^{-1} L_i \tag{58}$$
$$\hat{A}_i = V^{-1} M_i (X - Z_i)^{-1} V \tag{59}$$
$$M_i = A_i^\top Z_i + X A_i + L_i C_i + E_i^\top E_i \tag{60}$$

Table 1 Performance of closed-loop strategies

	OF_2	SF	SH	ADD	PS_1	PS_2
$\int_0^\infty \ddot{\xi}(t)^2 dt$ for $\ddot{\xi}_r(t) = \delta(t)$	7.835	7.721	8.288	8.150	26.548	8.307
$\frac{\int_0^T \ddot{\xi}(t)^2 dt}{\int_0^T \ddot{\xi}_r(t)^2 dt}$ for $T = 20$	0.697	0.643	0.787	0.823	3.558	0.719

and the matrices V, L_i, X, Z_i are specified in the following theorem.

Theorem 22 *Assume that there exist a Metzler matrix Λ with $\Lambda \mathbf{1} = 0$, a positive definite matrix X, a set of positive definite matrices (Z_i, R_{ij}) and a set of matrices L_i for all i, j, such that the following matrix inequalities:*

$$A_i^\top Z_i + Z_i A_i + \sum_{j=1}^{N} \lambda_{ij} R_{ij} + Q_i < 0$$

$$A_i^\top X + X A_i + C_i^\top L_i^\top + L_i C_i + Q_i < 0$$

$$R_{ii} < Z_i, \quad \begin{bmatrix} R_{ij} - Z_j & Z_j - Z_i \\ \bullet & X - Z_j \end{bmatrix} > 0, \quad i \neq j$$

hold. Then, the filter (57)–(60), along with the switching rule

$$\sigma(t) = \arg \min_i \hat{x}(t)^\top V^\top (X - Z_i)^{-1} V \hat{x}(t)$$

where V is an arbitrary non-singular matrix, makes the equilibrium solution $x = 0$ of (53)–(56) globally asymptotically stable and the associated cost (55) satisfies $J < \min_i \mathrm{Tr}(W_i)$ whenever matrices W_i are such that the linear matrix inequality

$$\begin{bmatrix} W_i & B' Z_i & B' X + D' L'_\ell \\ \bullet & Z_i & Z_i \\ \bullet & \bullet & X \end{bmatrix} > 0$$

holds for all i.

Example 12 Consider the motivating example presented in Sect. 2.5 The output feedback stabilization problem has been solved by taking the following set of parameters: $M = 400\,\mathrm{kg}$, $m = 50\,\mathrm{kg}$, $k = 2.0 \times 10^4\,\mathrm{N/m}$, $k_t = 2.5 \times 10^5\,\mathrm{N/m}$, $c_{min} = 3.0 \times 10^2\,\mathrm{N\,s/m}$ and $c_{max} = 3.9 \times 10^3\,\mathrm{N\,s/m}$. For these parameters the two matrices A_1 and A_2 are both stable (although with poorly damped oscillating modes); hence, our main goal here is to improve the transient dynamical behaviour of the system by minimizing the vertical acceleration of the chassis.

Two sets of simulations have been carried out. The first set refers to the response of $\ddot{\xi}(t)$ to a unit impulse on the road acceleration $\ddot{\xi}_r(t)$. The first row of Table 1 reports the integral of the squared chassis acceleration obtained with different control strategies. The symbols in the table have the following meaning:

- OF: Output feedback switching control of Theorem 22 designed with the output matrices of equations (1).
- SF: State-feedback switching control, designed via the switching rule (51) through the Lyapunov–Metzler inequalities (52).
- SH: Two-state sky-hook strategy (based on switching according to the sign of $\dot{\xi}(t)(\dot{\xi}(t) - \dot{\xi}_t(t))$).
- ADD: Acceleration-driven damper strategy with sampling period $\delta_T = 10^{-3} sec$ (based on a switching law depending on the sign of $\ddot{\xi}(t)(\dot{\xi}(t) - \dot{\xi}_t(t))$).
- PS$_1$: Passive suspension with fixed damping coefficient equal to c_{min}.
- PS$_2$: Passive suspension with fixed damping coefficient equal to c_{max}.

The design of OF depends on the tuning parameters r_1, r_2 and Π, that have been optimized after a limited number of trials. The resulting tuning parameters in OF are

$$r_1 = 2.0, \; r_2 = 0.5, \; \Lambda = \begin{bmatrix} -100 & 100 \\ 10 & -10 \end{bmatrix}$$

As apparent from Table 1, the difference between the outcomes of OF and SF is relatively small. By the way, the state-feedback performance is quite close to that obtained by applying the theoretical optimal switching strategy corresponding to $k_t \to \infty$, see [26].

In the second set of simulations the road profile $\xi_r(t)$ has been generated as the double integral of a sample realization of a white noise process with power $\chi^2 = 0.1$. The performance of the seven algorithms above, with the same values of the tuning parameters, has been measured as the power attenuation on the chassis acceleration, namely, the ratio

$$\Theta_T = \frac{\int_0^T \ddot{\xi}(t)^2 dt}{\int_0^T \ddot{\xi}_r(t)^2 dt}$$

This value, for $T = 20$ s, is reported in the second row of Table 1.

Figure 26 shows the behaviour of the acceleration for the three methods OF, SH and ADD. The plot has been restricted to an interval of 2 s, in order to better represent the effects of the commutations in the three methods. The OF strategy outperforms the other two algorithms at the expense of faster switching commutation and shorter dwell intervals.

Finally, the power attenuation Θ_T as a function of T is plotted in Fig. 27 to show the effectiveness of the feedback strategies based on the Lyapunov–Metzler switching rule.

It should be noted that the Lyapunov–Metzler inequalities only provide a sufficient condition for stabilization. An extension to cope with systems for which they are unfeasible has been first provided in [2]. Interesting is also the use of differential LMIs, see [24], for switching, impulsive systems and systems with reset, for which

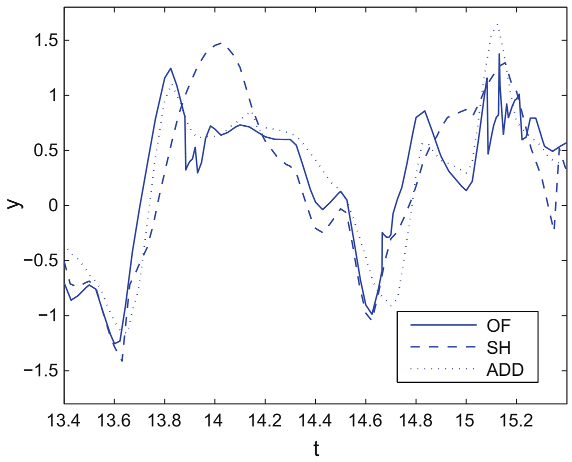


Fig. 26 Chassis acceleration during a short interval under a random road acceleration

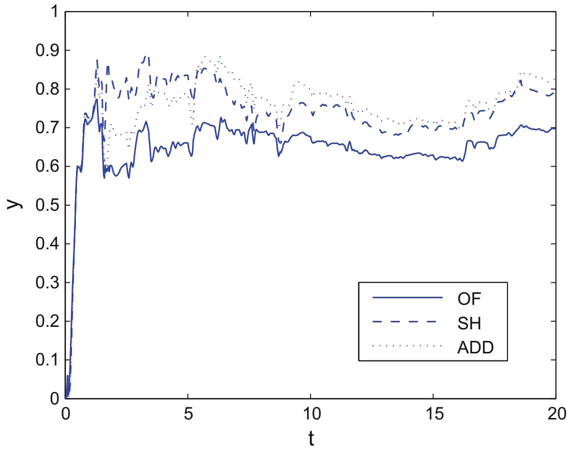


Fig. 27 Power attenuation under a random road acceleration

the theory encompasses problems related to optimization of continuous-time systems with intermittent measurements and digital input update control actions.

The class of 'dual switching systems', proposed in [28] for discrete-time systems and [25, 56] for continuous-time systems, is characterized by system parameters affected by two switching signals, one coming from a Markov chain, the other being either a deterministic disturbance or a control parameter. The presence of two signals allows to cope with interesting problems in contemporary technological society, where the interplay between performances, robustness and possibility of faults or malfunctions is very important. The associated new control strategies generalize switching and linear parameter varying control strategies determined so as to preserve stochastic stability and guaranteed performances.

6.2 Periodic Systems

Ordinary differential equations with periodic coefficients have a long history in physics and mathematics going back to the contributions of the nineteenth century by Faraday, Mathieu, Floquet, Rayleigh, Hill and many others. This has been emphasized by specific application demands, in particular in industrial process control, communication systems, natural sciences and economics. In control problems, the fact that a periodic operation may be advantageous is well known to mankind since time immemorial, but this observation germinated in industrial applications quite recently, in particular in the field of chemical engineering where it was seen that the performance of a number of catalytic reactors was improved by cycling. Suitable frequency-domain tests (such as the celebrated π-test) have been developed for this purpose in the early 1970s. In our days, the new possibilities offered by control technology, together with the theoretical developments of the field, opened the way for a wide use of periodic operations. For example, periodic control is useful in a variety of problems concerning under-actuated systems, namely, systems with a limited number of control inputs with respect to the degrees of freedom. Another example comes from non-holonomic mechanical systems, where in some cases stabilization cannot be achieved by means of a time-invariant differentiable feedback control law, but it is achievable with a periodic control law. Periodic control finds many applications in all fields of engineering and social sciences. For a monograph (including historical facts) see [14].

Consider the system

$$\dot{x}(t) = A(t)x(t)$$

where $A(t)$ is a matrix with bounded periodic coefficients (of period T). The system is asymptotically stable if and only if the so-called *characteristic multipliers* have modulus less than one. They are the eigenvalues of the *monodromy matrix* $\Phi_A(T, 0)$, where $\Phi_A(t, \tau)$ is the transition matrix, i.e. $x(t) = \Phi_A(t, \tau)x(\tau)$. It is clear that a switching system $\dot{x}(t) = A_{\sigma(t)}x(t)$ with $\sigma(t)$ periodic can be seen as a periodic system with discontinuous $A(t) = A_{\sigma(t)}$. For example, if $T = 2$ and

$$\sigma(t) = \begin{cases} 1 & t \in [0, 1) \\ 2 & t \in [1, 2) \end{cases}$$

then

$$\Phi_A(2, 0) = e^{A_2}e^{A_1}$$

and the system is stable if and only if the eigenvalues of $e^{A_2}e^{A_1}$ are in the open unit disc. Equivalently, one can associate with the periodic system a reset system

$$\dot{\xi}(t) = \begin{bmatrix} A_1 & 0 \\ 0 & A_2 \end{bmatrix}\xi(t), \quad \xi(t^+) = \begin{bmatrix} 0 & I \\ I & 0 \end{bmatrix}\xi(t)$$

Using a Lyapunov approach it can be seen that stability is equivalent to the existence of $P(t) > 0$ such that

$$\dot{P}(t) + \begin{bmatrix} A_1 & 0 \\ 0 & A_2 \end{bmatrix}^\top P(t) + P(t) \begin{bmatrix} A_1 & 0 \\ 0 & A_2 \end{bmatrix} < 0, \quad P(1) > \begin{bmatrix} 0 & I \\ I & 0 \end{bmatrix}^\top P(0) \begin{bmatrix} 0 & I \\ I & 0 \end{bmatrix}$$

This problem can be cast as a convex problem (by discretizing the differential linear matrix inequalities) so that it is amenable to deal with polytopic uncertainties on the parameters of the system, avoiding the computation of exponentials of matrices. On the other hand, a little thought reveals that the existence of $P(t) > 0$ solving the above inequalities is equivalent to the existence of $\bar{P} > 0$ solving

$$\bar{P} > \begin{bmatrix} 0 & e^{A_2} \\ e^{A_1} & 0 \end{bmatrix}^\top \bar{P} \begin{bmatrix} 0 & e^{A_2} \\ e^{A_1} & 0 \end{bmatrix}$$

that is equivalent to the Schur stability of $e^{A_2}e^{A_1}$.

From Floquet theory, it is possible to see that stability is equivalent to Hurwitz stability of the matrix $\bar{A}$ that solves $P(t)A(t) + \dot{P}(t) = \bar{A}P(t)$ for some Lyapunov–Floquet transformation (diffeomorphism) $P(t)$. The eigenvalues $\bar{\lambda}$ of $\bar{A}$ are called *characteristic exponents* and solve

$$\dot{\bar{x}}(t) + \bar{\lambda}\bar{x}(t) = A(t)\bar{x}(t)$$

for some periodic vector $\bar{x}(t)$, or, equivalently

$$((\sigma + \lambda)I - A(t))\bar{x}(t) = 0$$

where σ is the derivative operator. Notice that one can formally write a *characteristic equation* by solving the above differential equations, taking into account the skew commutative rule of periodic operators, i.e. given a periodic coefficient α. Then, the following operator identity holds:

$$\sigma\alpha = \dot{\alpha} + \alpha\sigma$$

With these simple considerations in mind we end this subsection by illustrating a toy example of stability and stabilization via periodic control taken from the literature [23, 37].

Example 13 Consider the linear time-invariant (LTI) system described by the transfer function

$$G(s) = \frac{s^2 - 1}{s^2 - 2\delta s + 1} \tag{61}$$

where δ is a real parameter in the range [0, 1]. The particular form of this transfer function comes from the Ph.D. thesis of Vincent Blondel [23], who offered a kilogram of Belgian chocolate for the solution of each of the following control problems:

Problem 2 Find the range of values for δ for which there exists an LTI, stable and minimum-phase stabilizing controller for plant (61).

Problem 3 Find an LTI, stable and minimum-phase stabilizing controller for plant (61) when $\delta = 0.9$.

Remark 5 Problem 3 was solved: a controller of 11th order was found using a randomized search method. Problem 2 is still unsolved. Note however that stabilization is impossible for $\delta = 1$ since then an unstable pole-zero cancellation occurs in plant $G(s)$. Results of complex analysis can be used to prove that there exists a value $\delta^* < 1$ such that stabilization is possible for all $\delta < \delta^*$, but impossible for $\delta \geq \delta^*$. A third-order controller was obtained for $\delta = 0.9$ but also for the more difficult case $\delta \approx 0.924$, and a fourth-order controller was obtained for $\delta \approx 0.951$, using non-convex non-smooth optimization with gradient sampling. It was also pointed out then that a third-order controller solving Problem 3 can be found by a suitable perturbation of necessary stabilizability conditions. For details, the interested reader is referred to [37].

One may argue that Blondel's chocolate problems are mainly of academic and mathematical interest. However, a better understanding of the mathematics of such difficult problems, even though academic ones, can help understand more applied and practical problems. For example, one should recall that control problems with near cancellation of unstable poles and zeros (just as in the Belgian chocolate problem) arise in physically relevant engineering problems, see, e.g. the X-29 prototype aircraft design problem or Klein's bicycle design problem mentioned in [3].

Here, it is shown, see [37], that simple control laws can be designed for plant (61) and any given value of $\delta < 1$, as soon as the assumption that the stable minimum-phase controller is LTI is relaxed.

The simple control action we are looking for is the one defined by

$$u(t) = F(t)y(t)$$

where $F(\cdot)$ is a suitable periodic function of period T. It is possible to show that the closed-loop system is asymptotically stable only if

$$T + \int_0^T \frac{F(t)}{1 - F(t)} dt < 0.$$

However, it is impossible to achieve stability with a bounded and smooth periodic gain if the function does not assume the forbidden value $F(\bar{t}) = 1$ at some time point $\bar{t}$.

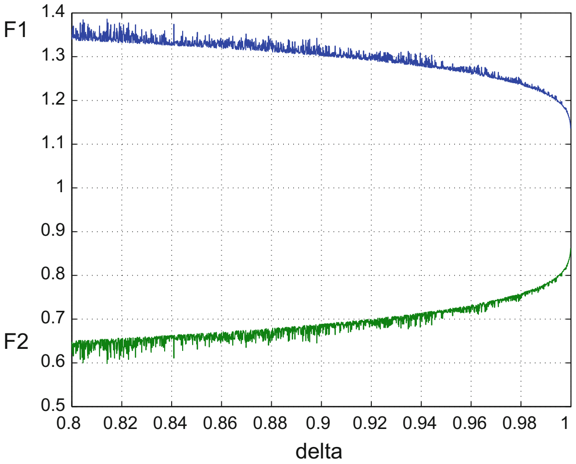


Fig. 28 F_1 (up) and F_2 (down) as functions of δ

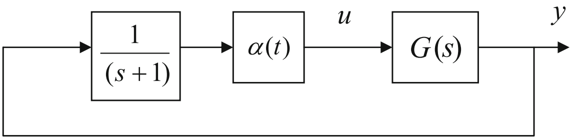


Fig. 29 First-order compensator

On the other hand, stabilization via piecewise constant $F(\cdot)$ is possible, for instance, with a periodic switching between two extreme values F_1 and F_2. The analytic dependence of F_1 and F_2 with respect to δ is hard to be found. However (taking a period $T = 1$), stabilization is possible for every δ in the fixed range, see Fig. 28.

The curves of F_1 and F_2 can be approximated by data fitting. Easy numerical computations show that the first-order polynomial ensures stability up to $\delta = 0.8897$, the third-order polynomial up to $\delta = 0.9716$ and the tenth-order polynomial up to $\delta = 0.994$.

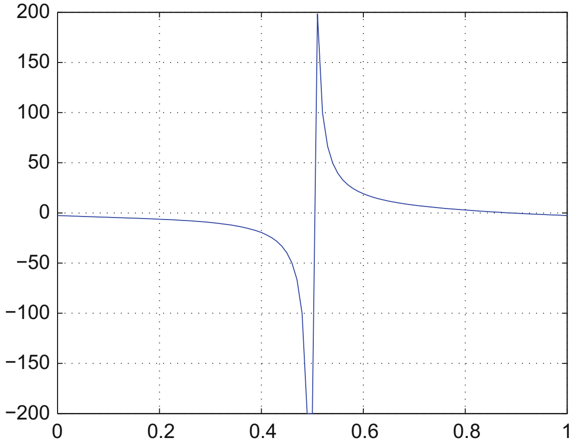
Finally, notice that the system has a stable zero that can be cancelled by the first-order controller that can be chosen with a smooth periodic gain. Hence, moving towards a first-order system, we consider the scheme depicted in Fig. 29.

The equations of the regulator are as follows:

$$\begin{aligned}\dot{\xi} &= -\xi + y \\ u &= \alpha(t)\xi\end{aligned}$$

Indicating by $\mathrm{av}(\cdot)$ the mean value of a periodic function, the set of all stabilizing periodic functions $\alpha(t)$ can be written as

Fig. 30 The gain α for $\delta = 0.999$



$$\alpha = \frac{-p_2^2 + 2\delta p_2 - \dot{p}_2}{1 - p_2}$$
$$0 > \mathrm{av}(p_2)$$
$$0 > \mathrm{av}\left(\frac{-1 + 2\delta - p_2 - \dot{p}_2}{1 - p_2}\right).$$

For instance, one can take $T = 1$ and the parameter p_2 as follows:

$$p_2(t) = -0.3 - \varepsilon(\delta) - 1.3\cos(2\pi t)$$

where $\varepsilon(\delta)$ is a small number depending on δ. We have chosen δ ranging from $\delta = 0.8$ to $\delta = 0.999$. It can be shown by simulation that $\varepsilon(\delta) = 0.00001$ can be chosen independent of δ in all the given range. In Fig. 30, the function $\alpha(t)$ associated with this constant choice and $\delta = 0.999$ is plotted.

Notice that the gain $\alpha(t)$ has an almost piecewise constant form.

7 Conclusions and Acknowledgements

This survey has provided an overview of a few decades of achievements in stability and control of uncertain systems. The amount of results is so massive that an exhaustive treatment was impossible. Thus, we have limited ourselves to embracing the topics we were personally involved in, starting from the mid-'80s, when the first (triennial) workshop organized before the IFAC World Congress by people in robust control took place. From the very beginning, we were accompanied and guided by a great colleague and friend, Roberto Tempo, prematurely passed away in 2017. We have been so fortunate to share more than 30 years of professional and human life with him. This survey is dedicated to him and has been written following his extraordinary fairness and teaching.

References

1. V. M. Admajan, D. Z. Arov, M. G. Krein, “Infinite block Hankel matrices and related extension problem”, *Americal Mathematical Society Translation*, Vol. 111, pp. 133–156, 1978.
2. L. I. Allerhand, U. Shaked, “Robust control of linear systems via switching”, *IEEE Transactions on Automatic Control*, Vol. 58, pp. 506–512, 2012.
3. K. J. Åström, “Limitations on control system performance”, *European Journal of Control*, Vol. 6, N. 1, pp. 2–20, 2000.
4. A. C. Bartlett, C. V. Hollot, L. Huang, “Root locations of an entire polytope of polynomials: it suffices to check the edges”, *Mathematics of Control, Signals and Systems*, Vol. 1, N. 1, pp. 61–71, 1988.
5. B. R. Barmish, “Stabilization of uncertain systems via linear control”, *IEEE Transactions on Automatic Control* Vol. 28, pp. 848–850, 1983.
6. B. R. Barmish, “Necessary and sufficient conditions for quadratic stabilizability of an uncertain system”, *J. Opt. Th. Appl.*, 46, pp. 399–408, 1985.
7. B. R. Barmish, *New Tools for Robustness of Linear Systems*, Macmillan, 1993.
8. B. R. Barmish, R. Tempo, “The robust root locus”, *Automatica*, Vol. 26, N. 2, pp. 283–292, 1990.
9. B. R. Barmish, M. Corless, and G. Leitmann, “A new class of stabilizing controllers for uncertain dynamical systems”, *SIAM Journal on Control and Optimization*, Vol. 21, N. 2, pp. 246–255, 1983.
10. B. R. Barmish, C. V. Hollot, F. J. Kraus, R. Tempo, “Extreme point results for robust stabilization of interval plants with first-order compensators”, *IEEE Transactions on Automatic Control*, Vol. 37, N. 6, pp. 707–714, 1992.
11. T. Basar, P. Bernhard, H_∞ *Optimal Control and Related Minimax Design Problems*, Birkhäuser, 1991.
12. D. S. Bernstein, W. M. Haddad, “LQG control with an H_∞ performance bound: a Riccati equation approach”, *IEEE Transactions on Automatic Control*, Vol. 34, pp. 293–305, 1989.
13. J. Bernussou, P. L. D. Peres, J. C. Geromel, “A linear oriented procedure for quadratic stabilization of uncertain systems”, *Systems and Control Letters*, Vol. 13, pp. 65–72, 1989.
14. S. Bittanti, P. Colaneri, *Periodic Systems: Filtering and Control*, Springer 2008.
15. F. Blanchini, “Non-quadratic Lyapunov function for robust control”, *Automatica*, Vol. 31, N. 3, pp. 451–461, 1995.
16. F. Blanchini, G. Giordano, “Piecewise-linear Lyapunov functions for structural stability of biochemical networks”, *Automatica*, vol. 50, N. 10, pp. 2482–2493, 2014.
17. F. Blanchini, G. Giordano, “Polyhedral Lyapunov functions structurally ensure global asymptotic stability of dynamical networks iff the Jacobian is non-singular”, *Automatica* no. 12, pp. 183–191, 2017.
18. F. Blanchini, S. Miani, *Set–Theoretic Methods in Control*, Birkhäuser, Boston, MA, 2015.
19. F. Blanchini, P. Colaneri, “Vertex/plane characterization of the dwelltime property for switching linear systems, *Proc. Conference on Decision and Control*, San Diego, 2010.
20. F. Blanchini and C. Savorgnan, “Stabilizability of switched linear systems does not imply the existence of convex Lyapunov functions”, *Automatica*, Vol. 44, N. 4, pp. 1166–1170, 2009.
21. F. Blanchini, P. Colaneri, F. A. Pellegrino, “Simultaneous performance achievement via compensator blending”, *Automatica*, 44, 1, pp. 1–14, 2008.
22. F. Blanchini, P. Colaneri, M. E. Valcher, *Positive Switched Linear Systems*, Foundation and Trends in Control Science, Now Publishing, 2015.

23. V. Blondel, *Simultaneous stabilization of linear systems*, Lecture Notes in Control and Information Sciences, 191, Springer-Verlag, Berlin, 1994.
24. C. Briat, “Dwell-time stability and stabilization conditions for linear positive impulsive and switched systems”, *Nonlinear Analysis: Hybrid Systems*, Vol. 24, pp. 198–226, 2017.
25. P. Bolzern, P. Colaneri, *Positive Markov Jump Systems*, Foundation and Trends in Control Science, Now Publishing, 2015.
26. P. Bolzern, P. Colaneri, J. C. Geromel, “Optimal switching of 1-dof oscillating systems”, in Hybrid Systems: Computation and control”, Alberto Bemporad, Antonio Bicchi and Giorgio Buttazzo Eds., Lecture Notes in Computer Science, Springer, pp. 118–130, 2007.
27. P. Bolzern, P. Colaneri, G. De Nicolao, “On the computation of upper covariance bounds for perturbed linear systems”, *IEEE Transactions on Automatic Control*, Vol. 39, pp. 623–626, 1994.
28. P. Bolzern, P. Colaneri, G. De Nicolao, “Design of stabilizing strategies for discrete time linear systems dual switching”, *Automatica*, Vol. 69, pp. 93–100, 2016.
29. S. Boyd, L. El Ghaoui, E. Feron, V. Balakrishnan, *Linear matrix inequalities in system and control theory*, SIAM Studies in Applied Mathematics, 1994.
30. R. K. Brayton and C. H. Tong, “Constructive stability and asymptotic stability of dynamical systems”, *IEEE Transactions on Circuits and Systems*, Vol. 27, N. 11, pp. 1121–1130, 1980.
31. B. C. Chang, J. B. Peaeson, “Optimal disturbance rejection in linear multivariable systems”, *IEEE Transactions on Automatic Control*, Vol. 29, pp. 880–887, 1984.
32. S. S. L. Chang, T. K. C. Peng, “Adaptive guaranteed cost control of systems with uncertain parameters”, *IEEE Transactions on Automatic Control*, Vol. 17, pp. 474–483, 1972.
33. B. M. Chen, *Robust and* H_∞ *control*, Springer, London, 2000.
34. G. Chesi, *Domain of Attraction: Analysis and Control via SOS Programming*, Lecture Notes in Control and Information Sciences, Springer-Verlag, 2011.
35. G. Chesi, P. Colaneri, J. C. Geromel, R. Middleton, R. Shorten, “A non-conservative LMI condition for stability of switched systems with guaranteed dwell time”, *IEEE Transaction on Automatic Control*, Vol. 57, N. 5, pp. 1297–1302, 2012.
36. P. Colaneri, “Dwell time analysis of deterministic and stochastic switched systems”, *European Journal of Control*, Vol. 15, pp. 228–248, 2009.
37. P. Colaneri, D. Henrion, *Switching and periodic control of the Belgian chocolate system*, IFAC Symposium Robust Control Design (Rocond), Toulouse, 2006.
38. P. Colaneri, J. C. Geromel, A. Astolfi, “Stabilization of continuous-time switched nonlinear Systems”, *Systems and Control Letters*, Vol. 57, pp. 95–103, 2008.
39. P. Colaneri, J. C. Geromel, A. Locatelli, *Control Theory and Design: An* RH_2 *and* RH_∞ *Viewpoint*, Academic Press, 1997.
40. M. A. Dahleh, I. D. Diaz-Bobillo, *Control of Uncertain Systems: A Linear Programming Approach*, Prentice-Hall, Englewood Cliffs, New Jersey, USA, 1995.
41. M. A. Dahleh, M. H. Khammash, “Controller design for plants with structured uncertainty”, *Automatica*, Vol. 29, N.1, pp. 37–56, 1993.
42. C. Davis, W. M. Kahan, H. F. Weinberger, “Norm preserving dilations and their applications to optimal error bound”, *SIAM Journal of Control and Optimization*, Vol. 20, pp. 445–469, 1982.
43. C. E. de Souza, U. Shaked, N. Fu, “Robust H_∞ filtering with parametric uncertainty and deterministic input signal”, *Proc. 31st IEEE Conf. Dec. Contr.* Tucson, USA, pp. 2305–2310,1992.
44. J. C. Doyle, *Lecture Notes in Advanced Multivariable Control*, ONR-Honeywell Workshop, Minneapolis, 1984.
45. J. C. Doyle, K. Glover, P. P. Khargonekar, B. Francis, “State space approach to standard H_2-H_∞ control problems”, *IEEE Transactions on Automatic Control*, Vol. 34, pp. 831–847, 1989.
46. B. A. Francis, *A Course in* H_∞ *Control Theory*, Lectures Notes on Control and Information Sciences. Springer Verlag, 1987.

47. B. A. Francis, G. Zames, “On L_∞-optimal sensitivity theory for SISO feedback systems”, *IEEE Transactions on Automatic Control*, Vol. 29, pp. 9–16, 1984.
48. R. A. Freeman and P. V. Kokotović, *Robust Nonlinear Control Design. State-Space and Lyapunov Techniques*, Systems and Control: Foundations and Applications. Birkhäuser Inc., Boston, MA, xii+256 pp. edition, 1996, ISBN: 0-8176-3930-6.
49. N. Fu, C. E. de Souza, L. Xie, “H_∞ estimation for uncertain systems”, *Int. J. Rob. and Nonlin. Contr.*, Vol. 2, pp. 82–105, 1992.
50. T. T. Georgiou, M. C. Smith, “Optimal robustness in the gap metric: controller design for distributed plants”, *IEEE Transactions on Automatic Control* Vol. 37, pp. 1133–1143, 1992.
51. J. C. Geromel, “On the determination of a diagonal solution of the Lyapunov equation”, *IEEE Transactions on Automatic Control*, Vol. 30, pp. 404–406, 1985.
52. J. C. Geromel, P. Colaneri, “Stability and stabilization of continuous-time switched systems”, *SIAM Journal on Control and Optimization*, Vol. 45, N. 5, pp. 1915–1930, 2006.
53. J. C. Geromel, A. O. E. Santo, “On the robustness of linear continuous-time dynamic systems”, *IEEE Transactions on Automatic Control*, Vol. 31, pp. 1136–1138, 1986.
54. J. C. Geromel, J. Bernussou, P. L. D. Peres,“ Decentralized control through parameter space optimization”. *Automatica*, Vol. 30, pp. 1565–1578, 1994.
55. J. C. Geromel, P. Colaneri, P. Bolzern, “Dynamic output feedback control of switched linear systems”, *IEEE Transaction on Automatic Control*, Vol. 53, N. 3, pp. 720–733, 2008.
56. J. C. Geromel, G. S. Daecto, P. Colaneri, “Minimax control of Markov jump linear systems”, *International Journal of Adaptive Control and Signal Processing*, Vol. 30, pp. 115–1162[, 2016.
57. J. C. Geromel, P. L. D. Peres, J. Bernussou, “On a convex parameter space method for linear control design of uncertain systems”, *SIAM Journal on Control and Optimization*, Vol. 29, pp. 381–402, 1991.
58. J. C. Geromel, P. L. D. Peres, S. R. Souza, “$\mathscr{H}_2$ Guaranteed cost control for uncertain continuous-time linear systems”, *Syst. Contr. Lett.*, Vol. 19, pp. 23–27, 1992.
59. G. Giordano, *Structural Analysis and Control of Dynamical Networks*, Ph.D. Thesis:, University of Udine, April, 2016.
60. K. Glover, “All optimal Hankel-norm approximations of linear multivariable systems and their L_∞ wrror bounds”, *International Journal of Control*, Vol. 39, pp. 1115–1193, 1984.
61. K. Glover, “Robust stabilization of linear multivariable systems: relaxation to approximations”, *International Journal of Control*, Vol. 43, pp. 74–766, 1986.
62. M. Green, D. J. Limebeer, *Linear Robust Control*, Dover Publications, 1995.
63. M. Green, K. Glover, D. J. N. Limebeer, J. C. Doyle, “J-spectral factorization approach to H_∞ control”, *SIAM Journal Control and Optimization*, Vol. 28, pp. 1130–1371, 1990.
64. E. Gruenbacher, P. Colaneri, L. Del Re, “Guaranteed robustness bounds for matched-disturbance nonlinear systems”, *Automatica*, Vol. 44, N. 9, pp. 2230–2240, 2008.
65. S. Gutman, “Uncertain dynamical systems—a Lyapunov min-max approach”, *IEEE Transactions on Automatic Control*, Vol. 24, N. 3, pp. 437–443, 1979.
66. W. M. Haddad, D. S. Bernstein, “Robust stabilization with positive real uncertainty: beyond the small gain theorem”, *Systems and Control Letters*, Vol. 17, pp. 191–208, 1991.
67. D. Hazony, “Zero cancellation synthesis using impedence operator”, *IRE Transactions on Circuit Theory*, Vol. 8, pp. 114–120, 1961.
68. D. Henrion, J. B. Lasserre, “Inner approximations for polynomial matrix inequalities and robust stability regions”, *IEEE Transactions on Automatic Control*, Vol. 57, N. 6, pp. 1456–1467, 2012.
69. H. P. Horisberger, P. R. Belanger, “Regulators for linear, time invariant plants with uncertain parameters”, *IEEE Transactions on Automatic Control*, Vol. 21, pp. 705–708, 1976.
70. I. Horowitz, A Synthesis theory for linear time-varying feedback systems with plant uncertainty, IEEE Transactions on Automatic Control, Vol. 20, N. 4, pp. 454–464, 1975.
71. I. Horowitz, *Quantitative Feedback Design Theory*, QFT Publishers, 1470 Grinnel Ave., Boulder, CO 80305, USA, 1993.

72. T. Hu, F. Blanchini, “Non-conservative matrix inequality conditions for stability/stabilizability of linear differential inclusions”, *Automatica*, vol. 46, N.1, pp. 190–196, 2010.
73. E. Kaszkurewicz, A. Bhaya, “Robust stability and diagonal Lyapunov functions”,*SIAM J. Matrix Analysis and Applications*, Vol. 14, pp. 508–520, 1993.
74. P.P. Khargonekar and M.A. Rotea. “Mixed H_2/H_∞ control: a convex optimization approach”, *IEEE Transactions on Automatic Control* AC-36, pp. 824–837, 1991.
75. P. P. Khargonekar, I. R. Petersen, K. Zhou, “Robust stabilization of uncertain linear systems: quadratic stabilizability and H_∞ Control Theory”, *IEEE Transactions on Automatic Control* Vol. 35, pp. 356–361, 1990.
76. V. L. Kharitonov, “Asymptotic stability of an equilibrium position of a family of systems of differential equations”, Differentsialnye uravneniya (in Russian), Vol. 14, pp. 2086–2088, 1978.
77. H. Kimura, “Robust stabilization of a class of trasfer functions”, *IEEE Transactions on Automatic Control*, Vol. 27, pp. 788–793, 1984.
78. H. Kimura, *Chain Scattering approach to H_∞ Control*, Boston, Birkhauser, 1997.
79. H. Kwakernaak, “The Polynomial Approach to the H_∞ Optimal Regulation”, H_∞ *Control*, E. Mosca and L. Pandolfi Eds. Lecture Notes in Mathematics, 1496, Springer-Verlag, 1990.
80. G. Leitmann, “ Guaranteed asymptotic stability for some linear systems with bounded uncertainties”, *Transactions of the ASME*, Vol. 101, pp. 212–216, 1979.
81. D. Liberzon, *Switching in Systems and Control*, Birkhäuser, 2003.
82. D. J. N. Limebeer, B. D. O. Anderson, B. Hendel. “A Nash game approach to mixed H_2/H_∞ Control”, *IEEE Transactions on Automatic Control*, Vol. 39, pp. 69–82, 1994.
83. K. Liu, B. M. Chen, Z. Lin, “On the problem of robust and perfect tracking for linear systems with external disturbances”, *International Journal of Control*, Vol. 74, N. 2, pp. 158–174, 2001.
84. D. McFarlane, K. Glover, “A loop shaping design procedure using H_∞ synthesis”, *IEEE Transactions on Automatic Control*, Vol. 37, N. 6, pp. 759–769, 1992.
85. A. P. Molchanov and E. S. Pyatnitskii, “Lyapunov functions that define necessary and sufficient conditions for absolute stability of nonlinear nonstationary control systems”, I *Autom. Remote Control*, Vol. 47, N. 3, pp. 344–354, 1986.
86. A. P. Molchanov and E. S. Pyatnitskii, “Lyapunov functions that define necessary and sufficient conditions for absolute stability of nonlinear nonstationary control systems”, II *Autom. Remote Control*, Vol. 47 N. 4, pp. 443–451, 1986.
87. A. P. Molchanov and E. S. Pyatnitskii, “Lyapunov functions that define necessary and sufficient conditions for absolute stability of nonlinear nonstationary control systems”, III *Autom. Remote Control*, Vol. 47, N. 5, pp. 620–630, 1986.
88. D. Mustafa, K. Glover, *Minimum Entropy H_∞ Control*, Lecture Notes in Control and Information Sciences, Vol. 146, Springer, Heidelberg, FRG, 1990.
89. R. Nevanlinna, “Uber beschrankte Funktionen, die in gegeben Punkten vorgeschriebenne Werte annehmen”, *Annales Academiae Scientiarum Fennicaeo*, Vol. 13, pp. 27–43, 1919.
90. P. Parrillo, *Structured Semidefinite Programs and Semialgebraic Geometry Methods in Robustness and Optimization*, Ph.D. Thesis, California Institute of Technology, Pasadena, CA, May 2000.
91. P. L. D. Peres, J. C. Geromel, “An alternate numerical solution to the linear quadratic problem”, *IEEE Transactions on Automatic Control*, Vol. 39, pp. 198–202, 1994.
92. P. L. D. Peres, J. C. Geromel, J. Bernussou, “Quadratic stabilizability of linear uncertain systems in convex-bounded domains”, *Automatica*, Vol. 29, pp. 491–493, 1993.
93. P. L. D. Peres, J. C. Geromel, S. R. Souza, “$\mathscr{H}_\infty$ Guaranteed cost control for uncertain continuous-time linear systems”, *Syst. Contr. Lett.*, Vol. 20, pp. 413–418, 1993.
94. P. L. D. Peres, J. C. Geromel, S. R. Souza, “Optimal $\mathscr{H}_\infty$ - State feedback control for continuous-time linear systems”. *J. of Opt. Theory and App.*, Vol. 82, pp. 343–359, 1994.
95. I. R. Petersen and B. R. Barmish, “Control effort considerations in the stabilization of uncertain dynamical systems”, *Proc. American Control Conference*, pp. 490–495, 1984.

96. I. R. Petersen, C. V. Hollot, “A Riccati equation approach to the stabilization of uncertain linear systems”, *Automatica*, Vol. 22, N. 4, pp. 397–412, 1986.
97. I. R. Petersen, D. C. McFarlane, “Robust State Estimation for Uncertain Systems”, *Proc. 30th Conf. Dec. Contr.* Brighton, England, pp. 2630–2631, 1991.
98. I. Petersen, R. Tempo, “Robust control of uncertain systems: Classical results and recent developments”, *Automatica*, Vol. 50, N. 5, pp. 1315–1538, 2014.
99. G. Pick, “Uber eschr a nkuungen analytischer Funktionen, welche durch vorgegebene Funktions wert bewirkt sind”, *Mathematischen Annalen*, Vol. 77, pp. 7–23, 1916.
100. L. Qiu, B. Bernarnhardsson, A. Rantzer, S. E. J. Davison, P. M. Young, J. C. Doyle, “A formula for computation of the real stability radius”, *Automatica*, Vol. 31, N. 6, pp. 879–890, 1995.
101. Z. Qu, *Robust Control of Nonlinear Uncertain Systems*, Wiley, New York, 1998.
102. A. Rantzer. “Stability conditions for polytopes of polynomials”, *IEEE Trans. Autom. Contr.*, Vol. 37, pp. 79–89, 1992.
103. R. T. Rockafellar, *Convex Analysis*, Princeton Univ. Press, 1970.
104. M. G. Safonov, D. J. N. Limebeer, “Simplifying the H_∞ theory via loop shifting”, *Proc. 27th Conf. on Dec. and Contr.*, Austin, USA, pp. 1399–1404, 1988.
105. R. Sanchez Pena, M. Sznaier, *Robust systems, Theory and Applications*, Wiley, New York, 1998.
106. M. G. Safonov, R. Y. Chiang, *Real–Complex Km-Synthesis without Curve-Fitting*, Academic Press, New York, 1993.
107. J. Veenman, C. W. Scherer, “IQC-synthesis with general dynamic multipliers”, *International journal of Robust and Nonlinear Control*, Vol. 0, pp. 1–38, 2013.
108. R. Shorten, F. Wirth, O. Mason, K. Wulff, C. King, “Stability criteria for switched and hybrid systems”, *Siam Review*, Vol. 49, No. 4, pp. 545–592, 2007.
109. A. A. Stoorvogel, *The H_∞ Control Problem*, Prentice Hall, 1992.
110. A. A. Stoorvogel, “The singular H_∞ Control Problem with Dynamic Measurement Feedback”, *SIAM Journal on Control and Optimization* 29, pp. 160–184, 1991.
111. Z. Sun, S. S. Ge, *Stability Theory of Switched Dynamical Systems*, Springer Verlag, 2011.
112. R. Tempo, F. Blanchini, “Robustness analysis with real parametric uncertaintiey”, in The Control Handbook, Edited by W. Levine, CRC Press, 2011 (2nd edition).
113. R. Tempo, G. Calafiore, and F. Dabbene, *Randomized Algorithms for Analysis and Control of Uncertain Systems*, Communications and Control Engineering Series2004. Springer-Verlag, London, 2004.
114. M. Vidyasagar, *Control Systems Analysis*, Prentice Hall, 1993.
115. S. Wieland, J. C. Willems, “Almost disturbance decoupling with internal stability”, *IEEE Transactions on Automatic Control*, Vol. 34, pp. 277–286, 1989.
116. D. C. Youla, “A new theory of cascade synthesis”, *IRE Transactions on Circuit Theory*, Vol. 9, pp. 244–260, 1961.
117. G. Zames, “Feedback and optimal sensitivity: model reference transformations, multiplicative seminorms, and approximate inverses”, *IEEE Transactions on Automatic Control*, Vol. 26, N. 2, pp. 301–320, 1981.
118. G. Zames, “Input-output feedback stability and robustness, 1959–1985”, *Control Systems Magazine*, IEEE, Vol. 16, N. 3, pp. 6–66, 1996.
119. G. Zames, B. A. Francis, “Feedback, minimax sensitivity, and optimal robustness”, *IEEE Transactions on Automatic Control*, Vol. 28, N. 5, pp. 585–601, 1983.
120. K. Zhou, P. P. Khargonekar, “Robust stabilization of linear systems with norm-bounded time-varying uncertainty”, *Systems and Control Letters*, Vol. 10, pp. 17–20, 1988.
121. K. Zhou, J. C. Doyle, K. Glower, *Robust and Optimal Control*, Prentice-Hall, Englewood Cliff, NJ, 1996.

Cooperative Resilient Estimation of Uncertain Systems Subjected to a Biasing Interference

Valery Ugrinovskii

Abstract The chapter revisits the recent methodology of distributed robust filtering using the H_∞ filtering approach. It summarizes some recent results on the analysis and design of networks of robust filters, which cooperate to produce high-fidelity estimates for uncertain plants. These results are applied to the problem of detecting and neutralizing biasing attacks on distributed observer networks, to obtain algorithms for cooperative detection of malicious biasing behaviour of compromised network nodes.

1 Introduction

Adoption of large-scale networks of smart devices is an emerging trend in the design of industrial control systems. Such systems operate by processing information obtained from multiple sensors to monitor and control physical plants. Cooperative filtering and estimation is one of the key methodologies to accomplish these tasks in a networked environment. In a cooperative filter, each network node receives information from several neighbouring nodes in the form of preprocessed data or raw measurements. It also shares similar information with its neighbours. The nodes then filter this information to extract the quantity of interest (e.g. the state of the observed plant) [1, 3, 4, 8, 11, 15, 16, 19, 20, 22, 34]. Because the nodes are usually spatially separated, such filtering algorithms are often referred to as distributed filters or distributed observers.

Distributing information processing among network nodes is known to have considerable advantages. Distributed sensor networks are more flexible in responding to failures, and the distributed organization helps to eliminate communication bottlenecks. At the same time, the information feedback which underpins these advantages is vulnerable to malicious attacks seeking to undermine the observer accuracy. For

V. Ugrinovskii (✉)
School of Engineering and IT, University of New South Wales Canberra, Canberra, ACT 2600, Australia
e-mail: v.ugrinovskii@gmail.com

© Springer Nature Switzerland AG 2018

T. Başar (ed.), *Uncertainty in Complex Networked Systems*,
Systems & Control: Foundations & Applications,
https://doi.org/10.1007/978-3-030-04630-9_2

instance, an adversary can exploit the dependency of a node on the information supplied by its neighbours and falsify this information to bias the node's decisions. Because in a cooperative environment the network components are tightly interwoven, this may lead to a catastrophic failure of the filtering algorithm [24].

The literature on the topic of distributed estimation and filtering is quite rich. Its main attention is focused on the so-called Distributed Kalman Filter problem. The latter term refers to a large family of estimation and filtering algorithms. It includes algorithms for calculation of optimal multisensor Kalman filter estimates in a distributed manner [25], multisensor data fusion algorithms [32], decentralized Kalman filters for vehicle positioning and formation control [29, 31], and consensus-based Kalman filters for maximum a posteriori (MAP) and linear minimum mean-square error (LMMSE) tracking of fast varying processes in wireless sensor networks [26]. A detailed account of publications on the topic of Distributed Kalman Filtering can be found in [12]. More recent results can be found, for example in [6, 13] and references therein.

The above references represent the mainstream trend in the literature on distributed estimation and filtering. In this mainstream setting the plant is typically assumed to be linear, and noise and disturbances are described as random Gaussian noise processes. A less common yet powerful approach to the design of distributed observers employs deterministic models of noise and uncertainty. It is based on the H_∞ theory and L_2 gain optimization [18, 21, 27, 28, 34, 36, 37, 39, 42]. We show in this chapter that this approach provides a convenient framework for analysing the impact of an adversarial biasing interference on an observer network. Such attacks aim to distort observer networks by biasing sensors and/or network communications, or by misappropriating the network nodes and can be modelled using deterministic models [33]. This motivates us to adopt the H_∞ framework for designing distributed observer networks resilient to biasing interference.

Resilience is a recent concept in the control system design [14, 43]. Resilient control schemes aim to ensure that the system is capable of maintaining acceptable (albeit possibly degraded) performance when it is subjected to a malicious interference and recovers quickly from the attack [23, 43]. The attention to the resilience problem has grown substantially after situations were discovered, where an adversary was able to interfere with the control task by injecting false information into the measurement data [5, 17]. These discoveries generated considerable interest in the development of techniques for detecting and neutralizing rogue behaviours within the system. The references [9, 10] are examples of the recent research on this topic specific to resilient estimation. This chapter also considers resilience of distributed observers to biasing attacks and develops a distributed estimation methodology for detecting such attacks and neutralizing their biasing impact.

Information sharing within an observer network presents abundant opportunities for monitoring its integrity. The information routinely collected and shared within the network can be utilized for both tracking the observed plant and detecting a biasing anomaly in the observer behaviour as well as correcting this anomaly.

We use this observation to analyse resilience of networked observers against conventional biasing attacks targeting the system sensors as well as the attacks compromising the integrity of the estimation algorithms rather than the data or communications. The latter attacks are known as misappropriation attacks [7]; cf. [10]. As remarked in [7], this type of attack is quite intricate since it allows the attacker to interfere with the system operation without any knowledge of the system model or the plant observed.

The organization of the chapter is as follows. In Sect. 2, we present a background on distributed robust estimation. The aim is to review an approach to the design of distributed H_∞ observers which will be used in the subsequent sections. This approach was originally introduced in [42]. Here, we show that this approach allows to reduce the attack detector design to a collection of decoupled H_∞ filtering problems, which can be solved independently from each other. This computational autonomy compares favourably with many existing algorithms, such as, for example the algorithms in [20, 34] where the computation of the observer gains relies on additional communication between the nodes; the latter can potentially be compromised. In contrast, the computation of the observer gains in [42] is essentially decentralized, although it does require a certain centralized initialization. However, this initialization involves only the information about the communication network, and does not require knowledge of the plant observed or the filters themselves.

The application of these ideas to the design of distributed biasing attack detectors for observer networks is presented in Sects. 3 and 4. In Sect. 3, we consider the detection of biasing misappropriation attacks, and Sect. 4 adapts this methodology to detecting biasing attacks on system sensors. One can trivially extend the results of Sect. 4 to include similar biasing attacks on communication channels. Remarks on using the information from the attack detectors for correcting the effects of biasing attacks are presented in Sect. 5. In Sect. 6, we present an illustrating example. The concluding remarks are given in Sect. 7.

Notation: $\mathbf{R}^n$ denotes the real Euclidean n-dimensional vector space, with the norm $\|x\| = (x'x)^{1/2}$; here the symbol $'$ denotes the transpose of a matrix or a vector. $\mathbf{R}^{n\times m}$ is the space of real $n \times m$ matrices. The symbol I denotes the identity matrix. For real symmetric $n \times n$ matrices X and Y, $Y > X$ (respectively, $Y \geq X$) means the matrix $Y - X$ is positive definite (respectively, positive semidefinite). $\mathrm{diag}[X_1, \dots, X_N]$ denotes the block diagonal matrix whose diagonal blocks are $X_1, \dots, X_N$. The notation $\mathscr{L}_2[0,\infty)$ refers to the Lebesgue space of $\mathbf{R}^n$-valued vector-functions $z(.)$, defined on the time interval $[0,\infty)$, with the norm $\|z\|_2 \triangleq \left(\int_0^\infty \|z(t)\|^2 dt\right)^{1/2}$ and the inner product $\int_0^\infty z_1'(t)z_2(t)dt$. When $z \in \mathscr{L}_2[0,\infty)$ we say that the signal z is $\mathscr{L}_2$-integrable and has a finite energy. Other notations include the vector of ones, $\mathbf{1} = [1 \ \dots \ 1]' \in \mathbf{R}^N$ and the symbol for the Kronecker product $\otimes$.

2 Preliminaries

2.1 Distributed Robust Estimation with H_∞ Performance

This section provides a background on the distributed H_∞ estimation problem. The presentation mainly follows [42], where an approach to this problem was developed which will be utilized in the subsequent sections.

Consider a continuous-time uncertain linear plant

$$\dot{x}(t) = A(t)x(t) + B(t)w(t), \quad x(0) = x_0, \tag{1}$$

evolving in the n-dimensional real Euclidean space, $x(t) \in \mathbf{R}^n$, governed by an m-dimensional unknown disturbance input w. The system coefficients $A(t) \in \mathbf{R}^{n\times n}$ and $B(t) \in \mathbf{R}^{n\times m}$ are assumed to be known. That is, we assume that the plant model is known, however the plant state $x(t)$ is not known because its initial state x_0 and the disturbance process are not available for direct measurement. The disturbance will be assumed to have a finite energy, $\|w\|_2^2 = \int_0^\infty \|w(t)\|^2 dt < \infty$. However, the plant is not assumed to be stable, therefore the plant trajectory is not guaranteed to be bounded.

As alluded in Introduction, we consider sensor networks where each node collects measurements about the plant and also receives a preprocessed data from its neighbours. The local measurements collected at node i are described by the equation

$$y_i(t) = C_i(t)x(t) + D_i(t)v_i(t). \tag{2}$$

That is, each node i measures a linear function of the plant state $x(t)$, and these measurements are contaminated by a measurement disturbance v_i. This disturbance is also assumed to have finite energy $\|v_i\|_2^2 = \int_0^\infty \|v_i(t)\|^2 dt < \infty$. The matrices C_i, D_i characterize the sensor at node i and can be time varying, although it is assumed that $D_i(t)D_i(t) > 0$ for all $t \geq 0$.

In addition, each node communicates with its neighbours j from the neighbourhood $\mathbf{V}_i$; the latter set is a subset of the node set $\mathbf{V} = \{1, \ldots, N\}$. The information that node j sends to node i is generally represented by a linear function of the state estimate obtained at node j at time t. However this information is sent over a communication channel, therefore the model for the signal c_{ij} received by node i takes into consideration disturbances in the communication channel,

$$c_{ij}(t) = W_{ij}\hat{x}_j(t) + H_{ij}v_{ij}(t). \tag{3}$$

Here, $\hat{x}_j(t)$ represents the estimate at node j, and v_{ij} denotes the channel disturbance. The latter will also be assumed to have finite energy. The matrices W_{ij} and H_{ij} are considered to be given constant matrices, which indicates that the channel is assumed to be time invariant. Also, it is assumed that $H_{ij}H_{ij}' > 0$ for all $j \in \mathbf{V}_i$, $i = 1, \ldots, N$. According to (3), our model assumes that the nodes communicate continuously and information is transmitted instantaneously. This is an idealization which can be overcome in some cases [38].

We now define the class of distributed observers for estimating the plant (1):

$$\dot{\hat{x}}_i(t) = A(t)\hat{x}_i(t) + L_i(t)(y_i(t) - C_i(t)\hat{x}_i(t)) + \sum_{j\in\mathbf{V}_i} K_{ij}(t)(c_{ij}(t) - W_{ij}\hat{x}_i(t)), \quad (4)$$
$$\hat{x}_i(0) = 0.$$

According to (4), each filter obtains its estimate $\hat{x}_i(t)$ using the measurement $y_i(t)$ and communication $c_{ij}(t)$, $j \in \mathbf{V}_i$, in the form of the innovations

$$\zeta_i = y_i - C_i(t)\hat{x}_i = C_i(t)(x - \hat{x}_i) + D_i(t)v_i, \quad (5)$$
$$\zeta_{ij} = c_{ij} - W_{ij}\hat{x}_i, \quad j \in \mathbf{V}_i. \quad (6)$$

The innovation ζ_i symbolizes the new information contained in the measurement acquired by node i, compared with its own prediction of that information. Likewise, the innovation ζ_{ij} symbolizes the new information contained in the message that node i receives from its neighbour $j \in \mathbf{V}_i$. These innovations will be used later for detecting a rogue biasing behaviour of misappropriated nodes. Both signals are readily available at node i; computing them only requires the local measurement y_i and the neighbour messages c_{ij}. $j \in \mathbf{V}_i$, available at node i, along with $\hat{x}_i$. The gains on these innovations, $L_i(t)$, $K_{ij}(t)$, are the parameters of the observer, and the distributed estimation problem boils down to determining these gains.

From the filter (4), we see that the neighbours' signals $c_{ij}(t)$ play a complementary role. They provide information about the plant that may not be present in the local measurements $y_i(t)$ but may be present in the measurements collected by the neighbouring nodes. This structure of the observer has proved to be useful in the situations where the plant was not detectable from the local measurements, and had to rely on the information sharing to accomplish the estimation task [34, 36, 37, 41]. More specifically, through the signals c_{ij} the observers (4) utilize a 'consensus' feedback from the network. Indeed, we can rewrite (4) in the form which makes the interconnections between the nodes explicit:

$$\begin{aligned}\dot{\hat{x}}_i(t) &= A(t)\hat{x}_i(t) + L_i(t)(y_i(t) - C_i\hat{x}_i(t)) \\ &\quad + \sum_{j\in\mathbf{V}_i} K_{ij}(t)W_{ij}(\hat{x}_j(t) - \hat{x}_i(t)) + \sum_{j\in\mathbf{V}_i} K_{ij}(t)H_{ij}v_{ij}(t).\end{aligned}$$

The second to last term is often called the diffusion term in the literature on multiagent consensus.

The filter (4) is admittedly not the only way to utilize the feedback from the neighbours. A number of references concerned with discrete-time distributed estimation problems employ a fusion of estimates, which is carried out before updating the state of the observer [4, 19, 20]. However, in continuous-time problems such as the problem considered here, the information fusion is difficult to carry out separately. Hence, we follow [34, 36–39, 42] and embed the processing of the innovations in the filter dynamics. Later, in the problem of detecting biasing attacks on distributed observer networks, we extend the principle of consensus feedback to enable the detectors to cooperatively monitor integrity of the network.

The distributed estimation problem is concerned with obtaining high-fidelity estimates of the plant state, while suppressing degrading effects of disturbances and/or noise on the estimation errors. A number of references, such as, for example [16, 22], do not consider disturbances or noise, and the estimation task is concerned with tracking the plant. Other references, such as [11, 13, 19], do consider disturbances/noise but focus on matching the performance of the centralized Kalman filter estimates. This often involves infinite [19] or finite [1] consensus loops at every step of the algorithm to enable the nodes to compute certain common quantities required to compute the optimal centralized estimates. Typically this requires carrying out a substantial number of consensus iterations during every update step, making the estimation algorithm computationally demanding. In other algorithms, the consensus step is augmented with blending the raw data [6, 13]. Effects of channel disturbances on this type of filters are often neglected—e.g., the references [1, 6, 11, 13, 16, 19, 22] do not consider the channel disturbances and their impact on the convergence of the filter. The approach which we follow in this chapter overcomes some of these drawbacks. It was originated in [34, 36–39, 42]. Within this approach, the task of distributed estimation is formalized as follows.

Distributed robust estimation problem with H_∞ performance: Determine the matrices $L_i(t)$ and $K_{ij}(t)$, $i = 1, \ldots, N$, $j \in \mathbf{V}_i$, such that the collection of interconnected filters (4) satisfies the following requirements:

(DE-i) *Internal stability of the distributed observer.* In the absence of disturbances, each node of the distributed estimator must exponentially converge to the true state of the plant. That is, there must exist some positive constants c_0, α such that:

$$\|x_i(t) - \hat{x}_i(t)\|^2 \leq c_0 e^{-\alpha t}, \quad \forall t \geq 0. \tag{7}$$

(DE-ii) *Global $\mathscr{L}_2$ disturbance attenuation.* In the presence of disturbances, the distributed estimator must deliver a certain $\mathscr{L}_2$ disturbance attenuation performance while its nodes track the plant. Following [42], here the performance of the filter is understood in the sense of a general $\mathscr{L}_2$ disturbance attenuation metric. Let P be an $Nn \times Nn$ symmetric positive-definite matrix, $P = P' > 0$, and define the vector of estimation errors $e = \mathbf{1} \otimes x - [\hat{x}_1' \ \ldots \ \hat{x}_N']'$. Clearly, $e = [e_1' \ \ldots \ e_N']'$, where $e_i = x - \hat{x}_i$ is the error of the observer at node i. The $\mathscr{L}_2$ disturbance attenuation performance requirement is formally defined as

$$\sup_{x_0,w,v_i,v_{ij},i,j=1,\ldots,N} \frac{\int_0^\infty e'(t)Pe(t)dt}{\sum_{i=1}^N \left(\|x_0\|^2_{X_i} + \int_0^\infty \left(\|w(t)\|^2 + \|v_i(t)\|^2 + \sum_{j\in \mathbf{V}_i}\|v_{ij}\|^2\right)dt\right)} \leq \gamma^2. \tag{8}$$

Here the matrices X_i weigh the uncertainty due to initiating the filter at node i at $\hat{x}_i(0) = 0$ against the plant, measurement and channel disturbances. These matrices are assumed to be selected in advance, similarly to how one selects an initial value for the a priori error covariance matrix in the Kalman filter algorithm.

Condition (8) captures several performance scenarios; see [42]. For example, letting P be a block diagonal matrix, $P = \text{diag}[P_1, \ldots, P_N]$, consisting of N diagonal $n \times n$ blocks P_i leads to a performance condition replicating the standard H_∞ filtering performance metric [2],

$$\sum_{i=1}^N \int_0^\infty e_i'(t)P_ie_i(t)dt \\ \leq \gamma^2 \sum_{i=1}^N \left(\|x_0\|^2_{X_i} + \int_0^\infty \left(\|w(t)\|^2 + \|v_i(t)\|^2 + \sum_{j\in\mathbf{V}_i}\|v_{ij}\|^2\right)dt\right). \tag{9}$$

Another interesting special case of the performance criterion (8) arises when $P = \begin{bmatrix} (\mathscr{L} + \mathscr{L}_T)\otimes I_n & 0 \\ 0 & (\mathscr{L} + \mathscr{L}_T)\otimes I_n \end{bmatrix}$, where $\mathscr{L}$, $\mathscr{L}_T$ are the Laplace matrices of the communication graph and its transpose graph (i.e. the graph whose edges are reversed), respectively. This special case corresponds to the consensus performance cost introduced in [34, 36, 37, 39]:

$$\int_0^\infty \sum_{i=1}^N \sum_{j\in\mathbf{V}_i} (\|\hat{x}_i(t) - \hat{x}_j(t)\|^2)dt \\ \leq \gamma^2 \sum_{i=1}^N \left(\|x_0\|^2_{X_i} + \int_0^\infty \left(\|w(t)\|^2 + \|v_i(t)\|^2 + \sum_{j\in\mathbf{V}_i}\|v_{ij}\|^2\right)dt\right). \tag{10}$$

In the literature, the performance objective (9) is encountered more frequently than (10). On the other hand, (10) explicitly puts an emphasis on the quality of the consensus feedback—since the filter (4) uses the disagreement between the neighbouring nodes, $\hat{x}_j(t) - \hat{x}_i(t)$, $j \in \mathbf{V}_i$ for feedback it makes sense to keep the detrimental effect of uncertainties and noise on this feedback signal to a minimum.

2.2 Design of the Distributed H_∞ Observer

To present an algorithm for solving the above-distributed estimation problem introduced in [42], let us consider the dynamics of the estimation errors of the observers (4) which evolve according to the equations

$$\begin{aligned}
\dot{e}_i &= (A(t) - L_i(t)C_i(t) - \sum_{j\in\mathbf{V}_i} K_{ij}(t)W_{ij})e_i + \sum_{j\in\mathbf{V}_i} K_{ij}(t)W_{ij}e_j \\
&\quad + B(t)w(t) - L_i(t)D_i(t)v_i - \sum_{j\in\mathbf{V}_i} K_{ij}(t)H_{ij}v_{ij}, \qquad (11)\\
e_i(0) &= x_0.
\end{aligned}$$

The system (11) at node i depends on the estimation errors at the neighbouring nodes $j \in \mathbf{V}_i$. This coupling is a major obstacle in the design of distributed observers. Standard decentralized approaches cannot be applied directly since they tend to treat interconnections between subsystems as undesirable disturbances. However, interconnections are often a crucial source of information about the plant especially at those nodes where the plant is not detectable locally [3, 35]. Because of this dependency on the interconnections, the design of distributed H_∞ observers has often to be done off-line in a centralized manner [27], or requires intensive calculations over the network [34, 40, 41]. The time-invariant nature of the plant and the observers exacerbates the issues arising from coupling between the filters.

The approach proposed in [42] partially circumvents these difficulties by breaking the original distributed filtering problem into a collection of auxiliary decentralized filtering subproblems, whose parameters are determined judiciously based on the global filtering objective. This endows the observer nodes with a computational autonomy in that the observer gains (but not the estimates themselves!) can be computed without interactions with other nodes, although the computations require an initial centralized setup. However when the observers are interconnected into the network, the estimates are generated in a distributed cooperative fashion.

The foundation of the decoupling procedure in [42] is a setup matrix inequality condition, which provides each node i with some positive-definite symmetric matrices R_i, Z_{ij}, $j \in \mathbf{V}_i$ that define the auxiliary local filtering problem to be solved at that node. We begin by introducing the matrices involved in that inequality:

$$\begin{aligned}
\Delta_i &\triangleq \sum_{j\in\mathbf{V}_i} W_{ij}'(W_{ij}Z_{ij}^{-1}W_{ij}' + H_{ij}H_{ij}')^{-1}Z_{ij}(W_{ij}Z_{ij}^{-1}W_{ij}' + H_{ij}H_{ij}')^{-1}W_{ij}, \\
\Phi_{ij} &\triangleq \begin{cases} \Delta_i & i = j, \\ -W_{ij}'(W_{ij}Z_{ij}^{-1}W_{ij}' + H_{ij}H_{ij}')^{-1}W_{ij}, & i \neq j,\ j \in \mathbf{V}_i, \\ 0, & i \neq j,\ j \notin \mathbf{V}_i. \end{cases} \qquad (12)
\end{aligned}$$

Also, define $R \triangleq \mathrm{diag}[R_1, \ldots, R_N]$, $\varDelta \triangleq \mathrm{diag}[\varDelta_1, \ldots, \varDelta_N]$ and $\varPhi \triangleq [\varPhi_{ij}]_{i,j=1,\ldots,N}$. Next, consider the differential Riccati equation:

$$\begin{aligned}\dot{\Sigma}_i &= A\Sigma_i + \Sigma_i A' - \Sigma_i\Bigg(C_i(t)'(D_i(t)D_i(t)')^{-1}C_i(t)\\ &\quad + \sum_{j\in\mathbf{V}_i} W'_{ij}(W_{ij}Z_{ij}W'_{ij} + H_{ij}H'_{ij})^{-1}W_{ij} - \frac{1}{\gamma^2}R_i\Bigg)^{-1}\Sigma_i + BB', \qquad (13)\\ \Sigma_i(0) &= X_i^{-1}.\end{aligned}$$

Equation (13) only depends on the parameters associated with node i. Once the matrices R_i, Z_{ij} are selected at this node, Eq. (13) can be solved autonomously by node i without interacting with its neighbours. The selection of these parameters is explained in the following theorem, which establishes the network of observers (4) as a solution to the distributed robust estimation problem under consideration. It is essentially the same as Theorem 1 from [42], except here the distributed observer is comprised of the standard H_∞ filters [2].

Theorem 1 (see [42]) *Given a positive semidefinite weighting matrix $P = P' \in \mathbf{R}^{nN\times nN}$, and positive-definite matrices $X_i \in \mathbf{R}^{n\times n}$, $i \in \mathbf{V}$, suppose a block diagonal matrix $R = R' > 0$, a collection of matrices $Z_{ij} = Z'_{ij} > 0$, $j \in \mathbf{V}_i$, $i \in \mathbf{V}$, and a constant $\gamma^2 > 0$ are such that*

$$R > P - \gamma^2(\varPhi + \varPhi' - \varDelta) \qquad (14)$$

and each Riccati equation (13) has a positive-definite-bounded[1] *solution on* $[0, \infty)$. *Then, the distributed observer obtained by interconnecting the node observers (4) equipped with the coefficients*

$$L_i(t) = \Sigma_i(t)C_i(t)'(D_i(t)D_i(t)')^{-1}, \qquad (15)$$
$$K_{ij}(t) = \Sigma_i(t)W'_{ij}(W_{ij}Z_{ij}W'_{ij} + H_{ij}H'_{ij})^{-1}, \qquad (16)$$

solves the problem of distributed estimation under consideration, in the sense that the conditions (7) and (8) are satisfied.

It is worth noting that despite the observers (4) with the gains (15), (16) are time varying, the inequality (14) only involves constant matrices. Therefore, it can be solved off-line in advance, and its solutions R_i, Z_{ij} can then be used in (13). These matrices do not need to be computed again, for as long as the network topologyand

[1] I.e., $\alpha_{1i}I \leq \Sigma_i(t) \leq \alpha_{2i}I$ $(\exists \alpha_{1i}, \alpha_{2i} > 0)$.

the communication channels remain the same. However, if the network topology or some of its channels change, the solutions of the inequality (14) must be updated. Also, once the matrices Z_{ij} are selected, the inequality (14) becomes a linear matrix inequality with respect to R_i, $i = 1, \dots, N$ and γ^2, it can be solved numerically using the existing software.

Finally, we note that each observer (4) is an H_∞ disturbance attenuating observer, which attenuates the impact of the process disturbance $w(t)$, the measurement v_i and channel disturbances v_{ij}, *as well as the estimation errors at neighbouring nodes* j*,* $j \in \mathbf{V}_i$:

$$\int_0^\infty e_i'(t) R_i e_i(t) dt \le \gamma^2 \left(\|x_0\|^2_{X_i} + \int_0^\infty \left(\|w(t)\|^2 + \|v_i(t)\|^2 \right.\right.$$
$$\left.\left. + \sum_{j \in \mathbf{V}_i} \|v_{ij}(t)\|^2 + \|e_j(t)\|^2_{Z_{ij}^{-1}} \right) dt \right). \tag{17}$$

Here, R_i, Z_{ij}, $j \in \mathbf{V}_i$, are symmetric positive-definite matrices which are selected from (14). From (17), one can see that R_i, Z_{ij} defines the local performance cost of the filter (4). Specifically, the matrix Z_{ij} imposes a weighting on the contribution of the neighbour j's error e_j into the i's performance. However, these parameters are not independent as they are linked via condition (14).

3 Robust Detection of Biasing Misappropriation Attacks

In this section, we turn to the problem of detecting malicious biasing attacks [33] on distributed observers. The problem was posed originally in [7], it is concerned with a situation where some observers in the network are misappropriated and are used to supply a biased information to its neighbours; cf. [30]. Specifically, we consider the situation where the adversary substitutes one or several observers (4) with a tempered version,

$$\dot{\hat{x}}_i = A(t)\hat{x}_i + L_i(t)(y_i(t) - C_i(t)\hat{x}_i) + \sum_{j \in \mathbf{V}_i} K_{ij}(t)(c_{ij} - W_{ij}\hat{x}_i) + F_i f_i, \tag{18}$$
$$\hat{x}_i(0) = 0.$$

Here, $F_i \in \mathbf{R}^{n \times n_{f_i}}$ is a constant matrix and $f_i \in \mathbf{R}^{n_{f_i}}$ is the unknown signal representing an attack input. In this section, we will present an algorithm for detecting and tracking these unknown inputs.

3.1 Biasing Attack Inputs

Following [7], we will consider a class of attacks consisting of biasing inputs $f_i(t)$ of the form

$$f_i(t) = f_{i1} + f_{i2}(t), \tag{19}$$

where f_{i1} is an unknown constant,[2] and $f_{i2}(t)$ is an unknown $\mathscr{L}_2$-integrable 'masking' signal, which the adversary may add to conceal the biasing component of its attack input [33]. It was shown in [7] that for any proper $n_{f_i} \times n_{f_i}$ transfer function $G_i(s)$ for which the transfer function $\frac{1}{s}(I + \frac{1}{s}G_i(s))^{-1}G_i(s)$ is stable, the signal

$$\hat{f}_i = \frac{1}{s}(I + \frac{1}{s}G_i(s))^{-1}G_i(s) f_i \tag{20}$$

approximates f_i asymptotically, and the approximation error

$$\eta_i = \hat{f}_i - f_i \tag{21}$$

is $\mathscr{L}_2$-integrable.

From (20), (21), the input–output relation between η_i and $\hat{f}_i$ is $\hat{f}_i = -\frac{1}{s}G_i(s)\eta_i$. Let

$$\begin{aligned} \dot{\varepsilon}_i &= \Omega_i \varepsilon_i + \Gamma_i \eta_i, \qquad \varepsilon_i(0) = 0, \\ \hat{f}_i &= \Upsilon_i \varepsilon_i. \end{aligned} \tag{22}$$

be the minimal realization of $-\frac{1}{s}G_i(s)$. It was shown in [7] that using the model (22), f_i can be observed from the data available to the observer network (18) up to an $\mathscr{L}_2$-integrable error. In this chapter, we follow this idea, however, the time-varying nature of the problem under consideration requires us to revisit the attack detection methodology developed in [7]. The method proposed in [7] involves solving certain coupled linear matrix inequalities, which in the time-varying case will have to be replaced by *differential* matrix inequalities with time-varying coefficients. Such inequalities are difficult to solve in general. On the contrary, the observer design methodology described in the previous section does not suffer from such difficulties. For this reason, the methodology of cooperative detection of biasing attacks presented here builds on this alternative technique. Its aim is to obtain an algorithm for computing the characteristics of the attack detector that can be used in real time and preferably in a decentralized manner, to reduce communication overheads.

[2] Formally, the attack model introduced in [7] is somewhat more general, it allows f_{i1} to be time varying, although it must satisfy certain additional constraints. This more general model can be used here as well, and this will not cause any technical issues. For that reason we restrict ourselves to the case where f_{i1} is a constant, to simplify the presentation.

Despite the apparent freedom in selecting the transfer function $G_i(s)$, its choice may be influenced by practical considerations. For example, [7] suggests using first-order low-pass filters, i.e. $G_i(s) = \frac{g_i}{s+2\beta_i}I$. In this case, β_i, g_i must be selected to ensure a sufficiently fast transient performance of the attack detector.

3.2 Design of Attack Detectors

To detect a biasing attack, we rely on the information already available at the observer nodes; that is, we aim to utilize the same two innovation signals (5), (6). We pose the problem of detecting a biasing attack of the form (19) on the observer network comprised of observers (18) as the problem of designing a network of innovation-based attack detectors

$$\begin{aligned}\dot{\mu}_i &= \mathscr{A}_d(t)\mu_i + L_{d,i}(t)(\zeta_i - W_{d,i}\mu_i) + \sum_{j\in\mathbf{V}_i} K_{d,ij}(t)(\zeta_{ij} - W_{d,ij}(\mu_j - \mu_i)),\\ \varphi_i &= C_{d,i}\mu_i, \qquad \mu_i(0) = 0.\end{aligned} \tag{23}$$

In the fault detection and isolation theory, the output $\varphi_i(t)$ is termed the *residual output*, it has the purpose of tracking the attack input f_i. Accordingly, the problem of attack detection is to determine the matrix-valued coefficients $\mathscr{A}_d(t)$, $L_{d,i}(t)$, $K_{d,ij}(t)$, $W_{d,i}$, $W_{d,ij}$, $C_{d,i}$ to ensure that $\varphi_i(t)$ converges to f_i when $f_i \neq 0$, and $\varphi_i(t)$ converge to 0 otherwise. These objectives are now stated formally.

Distributed H_∞ attack detection problem: Given the observer network consisting of the plant (1) and the filters (4), construct a network of filters (23) which, when interconnected with the state observers (18) achieve the following properties:

(AD-i) In the absence of disturbances and when the system is not under attack, at every node i, the detector outputs φ_i converge to 0 exponentially.

(AD-ii) In the presence of uncertainties and/or attack, each output φ_i tracks the corresponding attack input f_i in the $\mathscr{L}_2$ sense; that is,

$$\int_0^{+\infty} \|\varphi_i - f_i\|^2 dt < +\infty \quad \forall i. \tag{24}$$

In particular, when node i is not under attack, the corresponding residual φ_i must have a bounded energy.

Note that the proposed detectors (23) rely on the information received from their neighbouring nodes contained in the innovation signals ζ_{ij}. Essentially, the detectors (23) have the same structure as the observers (4), therefore our approach to detector design will be based on the results in Sect. 2.

3.3 Design of the Distributed Detector for Biasing Misappropriation Attacks

Our attack scenario allows to treat both the healthy and misappropriated nodes of the network in the same manner—the observers at the healthy nodes can be regarded as a special case of the compromised observers corresponding to the attack input $f_i = 0$. Using again the notation $e_i = x - \hat{x}_i$ for the estimation error at node i, it follows from (1), (18) that the errors of the observer (18) evolve according to

$$\begin{aligned}\dot{e}_i &= (A(t) - L_i(t)C_i(t) - \sum_{j\in\mathbf{V}_i} K_{ij}(t)W_{ij})e_i + \sum_{j\in\mathbf{V}_i} K_{ij}(t)W_{ij}e_j \\ &\quad + B(t)w - L_i(t)D_i(t)v_i - \sum_{j\in\mathbf{V}_i} K_{ij}(t)H_{ij}v_{ij} - F_i f_i, \qquad (25)\\ e_i(0) &= x_0.\end{aligned}$$

Using (21), we can eliminate the input f_i from (25). The resulting equation will, however, include $\hat{f}_i$, which can be further eliminated by substituting its expression from (22). Combining (25) with (22) leads to the following extended system:

$$\begin{aligned}\dot{e}_i &= (A(t) - L_i(t)C_i(t) - \sum_{j\in\mathbf{V}_i} K_{ij}(t)W_{ij})e_i + \sum_{j\in\mathbf{V}_i} K_{ij}(t)W_{ij}e_j - F_i\Upsilon_i\varepsilon_i \\ &\quad +B(t)w - L_i(t)D_i(t)v_i - \sum_{j\in\mathbf{V}_i} K_{ij}(t)H_{ij}v_{ij} + F_i\eta_i, \\ \dot{\varepsilon}_i &= \Omega_i\varepsilon_i + \Gamma_i\eta_i, \qquad (26)\\ e_i(0) &= x_0, \quad \varepsilon_i(0) = 0.\end{aligned}$$

The system comprised of subsystems (26) can be regarded as an interconnected large-scale uncertain system governed by the finite energy disturbance inputs $w(t)$, $v_i(t)$, $v_{ij}(t)$ and the fictitious tracking error η_i which is also $\mathscr{L}_2$-integrable, according to our standing assumption. Therefore, we can attempt to estimate $\hat{f}_i = \Upsilon_i\varepsilon_i$ from the innovations (5), (6) which can be expressed as functions of e_i, e_j, $j \in \mathbf{V}_i$,

$$\zeta_i = C_i(t)e_i + D_iv_i, \qquad (27)$$

$$\zeta_{ij} = -W_{ij}(e_j - e_i) + H_{ij}v_{ij}, \quad j \in \mathbf{V}_i. \qquad (28)$$

These innovation signals are available for measurement at the corresponding nodes of the observer network.

The observer which we propose below for estimating the combined state $[e_i', \varepsilon_i']'$, $i = 1, \dots, N$, of the resulting large-scale uncertain system has the structure of the attack detector (23):

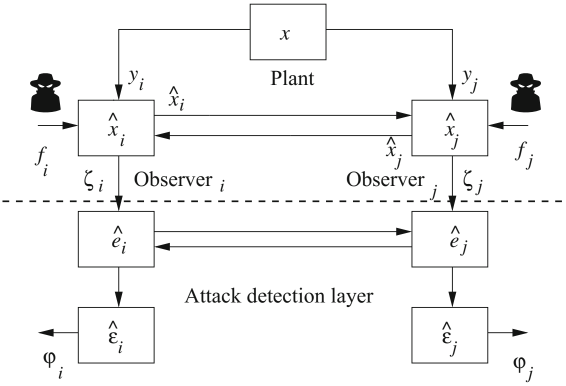


Fig. 1 A large-scale interconnected system including the plant, the state observers and the attack detectors

$$\begin{aligned}
\dot{\hat{e}}_i &= (A(t) - L_i(t)C_i(t) - \sum_{j\in\mathbf{V}_i} K_{ij}(t)W_{ij})\hat{e}_i + \sum_{j\in\mathbf{V}_i} K_{ij}(t)W_{ij}\hat{e}_j - F_i\Upsilon_i\hat{\varepsilon}_i \\
&\quad + \bar{L}_i(t)(\zeta_i - C_i(t)\hat{e}_i) + \sum_{j\in\mathbf{V}_i} \bar{K}_{ij}(t)(\zeta_{ij} - W_{ij}(\hat{e}_i - \hat{e}_j)), \\
\dot{\hat{\varepsilon}}_i &= \Omega_i\hat{\varepsilon}_i + \check{L}_i(\zeta_i - C_i\hat{e}_i) + \sum_{j\in\mathbf{V}_i} \check{K}_{ij}(t)(\zeta_{ij} - W_{ij}(\hat{e}_i - \hat{e}_j)), \\
\varphi_i &= \Upsilon_i\hat{\varepsilon}_i, \\
\hat{e}_i(0) &= 0, \quad \hat{\varepsilon}_i(0) = 0.
\end{aligned} \tag{29}$$

Its state μ_i is $\mu_i = [\hat{e}_i', \ \hat{\varepsilon}_i']'$, and the gains $L_{d,i}$, $K_{d,ij}$ are comprised of the gains $\bar{L}_i(t)$, $\bar{K}_{ij}(t)$, $\check{L}_i(t)$, $\check{K}_{ij}(t)$:

$$L_{d,i}(t) = \begin{bmatrix} \bar{L}_i(t) \\ \check{L}_i(t) \end{bmatrix}, \quad K_{d,ij}(t) = \begin{bmatrix} \bar{K}_{ij}(t) \\ \check{K}_{ij}(t) \end{bmatrix}.$$

The structure of the network including the proposed attack detectors is shown in Fig. 1. It shows that the original network of observers (18) is supplemented with an attack detection layer. The topology of this layer replicates the topology of the original network, and the detectors can utilize the same communication channels to exchange information.

Henceforth, our effort is directed towards finding a constructive method for computing the coefficients $\bar{L}_i(t)$, $\bar{K}_{ij}(t)$, $\check{L}_i(t)$, $\check{K}_{ij}(t)$ which ascertain that properties (AD-i) and (AD-ii) hold. To explain the construction of the proposed attack detector, let us define

$$\hat{L}_i = L_i + \bar{L}_i, \qquad \hat{K}_{ij} = K_{ij} + \bar{K}_i. \tag{30}$$

Also, let us introduce the notation

$$\mathbf{A}_i(t) = \begin{bmatrix} A(t) & -F_i\Upsilon_i \\ 0 & \Omega_i \end{bmatrix}, \quad \mathbf{B}_i = \begin{bmatrix} B(t) & F_i \\ 0 & \Gamma_i \end{bmatrix}, \quad \mathbf{C}_i(t) = \begin{bmatrix} C_i(t) & 0 \end{bmatrix},$$
$$\mathbf{W}_{ij} = \begin{bmatrix} W_{ij} & 0 \end{bmatrix}, \quad \mathbf{L}_i = \begin{bmatrix} \hat{L}_i \\ \check{L}_i \end{bmatrix}, \quad \mathbf{K}_{ij} = \begin{bmatrix} \hat{K}_{ij} \\ \check{K}_{ij} \end{bmatrix}. \tag{31}$$

Let Z_{ij} be a collection of positive definite $n \times n$ matrices, $i = 1, \dots, N$, $j \in \mathbf{V}_i$. Also, consider a collection of positive- definite matrices $\hat{P}_i$, $\check{P}_i$; each matrix $\hat{P}_i$ must be an $n \times n$ matrix, while matrices $\check{P}_i$ must have dimensions matching the corresponding vectors δ_i. These matrices will assign the weighting to the accuracy of estimating the components e_i and ε_i of the state of the system (26). Also, we introduce the matrices $\hat{X}_i = \hat{X}_i' > 0$, $\check{X}_i = \check{X}_i' > 0$; the former one reflects our knowledge of the accuracy of approximating $x(0) = x_0$ with 0.

Now, consider two sets of symmetric matrices $\{\hat{R}_i, i = 1, \dots, N\}$ and $\{\check{R}_i, i = 1, \dots, N\}$, which satisfy the matrix inequalities

$$\begin{aligned} &\hat{R} > \hat{P} - \gamma_d^2(\Phi + \Phi' - \Delta), \quad \hat{R}_i > 0, \\ &\check{R}_i > \check{P}_i, \end{aligned} \tag{32}$$

where $\hat{R} = \mathrm{diag}[\hat{R}_1, \dots, \hat{R}_N]$, $\hat{P} = \mathrm{diag}[\hat{P}_1, \dots, \hat{P}_N]$. The matrices Φ and Δ have been defined in (12). Also, for every $i = 1, \dots, N$, consider the differential Riccati equation of the form (13),

$$\begin{aligned} \dot{\mathbf{Y}}_i &= \mathbf{A}_i\mathbf{Y}_i + \mathbf{Y}_i\mathbf{A}_i' + \mathbf{B}_i\mathbf{B}_i' - \mathbf{Y}_i\Bigg(\mathbf{C}_i'(D_iD_i')^{-1}\mathbf{C}_i \\ &\quad + \sum_{j\in\mathbf{V}_i} \mathbf{W}_{ij}'(W_{ij}Z_{ij}W_{ij}' + H_{ij}H_{ij}')^{-1}\mathbf{W}_{ij} - \frac{1}{\gamma_d^2}\mathbf{R}_i\Bigg)\mathbf{Y}_i, \\ \mathbf{Y}_i(0) &= \mathbf{X}_i^{-1}, \end{aligned} \tag{33}$$

where $\mathbf{X}_i \triangleq \mathrm{diag}[\hat{X}_i, \check{X}_i]$, $\mathbf{R}_i \triangleq \mathrm{diag}[\hat{R}_i, \check{R}_i]$.

Theorem 2 *Suppose there exists a constant $\gamma_d > 0$ and positive-definite symmetric matrices $\hat{R}_i$, $\check{R}_i$, Z_{ij}, $j \in \mathbf{V}_i$, $i = 1, \dots N$, which satisfy the inequalities (32) and such that each differential Riccati equation (33) has a positive-definite symmetric bounded solution $\mathbf{Y}_i(t)$ on the interval $[0, \infty)$, i.e., for all $t \geq 0$, $\alpha_1 I < \mathbf{Y}_i(t) = \mathbf{Y}_i'(t) < \alpha_2 I$, for some $\alpha_{1,2} > 0$. Then the network of attack detectors (29) with the coefficients $\bar{L}_i$, $\bar{K}_{ij}$, $\check{L}_i$, $\check{K}_{ij}$, obtained by partitioning the matrices*

$$\begin{aligned} \mathbf{L}_i(t) &= \mathbf{Y}_i(t)\mathbf{C}_i(t)'(D_iD_i')^{-1}(t), \\ \mathbf{K}_{ij}(t) &= \mathbf{Y}_i(t)\mathbf{W}_{ij}'(W_{ij}Z_{ij}W_{ij}' + H_{ij}H_{ij}')^{-1} \end{aligned} \tag{34}$$

according to (31) and letting $\bar{L}_i = \hat{L}_i - L_i$, $\bar{K}_{ij} = \hat{K}_{ij} - K_{ij}$, guarantees the satisfaction of the following properties:

(AD-i') *In the absence of disturbances and attacks, the detection errors vanish exponentially as $t \to \infty$, and $\varphi \to 0$ exponentially as $t \to \infty$.*

(AD-ii') *In the presence of disturbances or attacks, the attack detectors provide a guarantee of H_∞-type attack detecting performance,*

$$\sum_{i=1}^{N} \int_0^\infty \|\varphi_i(t) - f_i(t)\|^2 dt \leq \frac{\gamma_d^2 \|\Upsilon_i\|^2}{\sigma_{\min}(\check{P}_i)} \sum_{i=1}^{N} \Bigg(\|x_0\|^2_{\hat{X}_i} + \int_0^\infty \Bigg(\|w(t)\|^2 + \|v_i(t)\|^2 + \|\eta_i(t)\|^2 + \sum_{j \in \mathbf{V}_i} \|v_{ij}\|^2 \Bigg) dt \Bigg) + \int_0^\infty \sum_{i=1}^{N} \|\eta_i\|^2 dt. \quad (35)$$

Proof Introduce the detector error variables $z_i = e_i - \hat{e}_i$, $\delta_i = \varepsilon_i - \hat{\varepsilon}_i$. The variable δ_i determines the accuracy of approximating $\hat{f}_i$ with the detector output φ_i, $\hat{f}_i - \varphi_i = \Upsilon_i \delta_i$. On the other hand, z_i represents the accuracy of estimating the error e_i of the biased observer (18) at node i. This variable will be used later for correcting the plant estimates produced by (18). The evolution of these variables is described by the equations

$$\begin{aligned}
\dot{z}_i &= (A(t) - \hat{L}_i(t)C_i(t) - \sum_{j \in \mathbf{V}_i} \hat{K}_{ij}(t)W_{ij})z_i - F_i \Upsilon_i \delta_i - \sum_{j \in \mathbf{V}_i} \hat{K}_{ij}(t)W_{ij}z_j \\
&\quad + Bw - \hat{L}_i(t)D_i(t)v_i - \sum_{j \in \mathbf{V}_i} \hat{K}_{ij}(t)H_{ij}v_{ij} + F_i \eta_i, \\
\dot{\delta}_i &= \Omega_i \delta_i - \check{L}_i(t)C_i(t)z_i - \sum_{j \in \mathbf{V}_i} \check{K}_{ij}(t)W_{ij}z_i - \sum_{j \in \mathbf{V}_i} \check{K}_{ij}(t)W_{ij}z_j \\
&\quad - \check{L}_i(t)D_i(t)v_i - \sum_{j \in \mathbf{V}_i} \check{K}_{ij}(t)H_{ij}v_{ij} + \Gamma_i \eta_i, \\
&\quad z_i(0) = x_0, \quad \delta_i(0) = 0.
\end{aligned} \quad (36)$$

Using the notation (31), the dynamics of the detector errors can be written in the form

$$\begin{aligned}
\dot{\lambda}_i &= (\mathbf{A}_i(t) - \mathbf{L}_i(t)\mathbf{C}_i(t) - \sum_{j \in \mathbf{V}_i} \mathbf{K}_{ij}\mathbf{W}_{ij})\lambda_i + \sum_{j \in \mathbf{V}_i} \mathbf{K}_{ij} W_{ij} z_j \\
&\quad + \mathbf{B}_i(t) \begin{bmatrix} w \\ \eta_i \end{bmatrix} - \mathbf{L}_i(t) D_i(t) v_i - \sum_{j \in \mathbf{V}_i} \mathbf{K}_{ij} H_{ij} v_{ij}, \qquad \lambda_i \triangleq \begin{bmatrix} z_i \\ \delta_i \end{bmatrix}.
\end{aligned} \quad (37)$$

Once again, we observe that detector error dynamics are interconnected, and therefore represent a large-scale interconnected uncertain system. In fact, this system is of the same type as the system describing the evolution of local errors of the unbiased observer (4) and comprised of subsystems (11). Therefore, the proof of Theorem 1 can be adapted to establish the validity of the theorem. Indeed, define the Lyapunov function candidate for the system (37), $\mathscr{V}_i = \sum_{i=1}^{N} \lambda_i' \mathbf{Y}_i^{-1} \lambda_i$. Using (34), (33), it is easy to show by completing the squares that

$$
\begin{aligned}
\dot{\mathscr{V}}_i = \sum_{i=1}^{N} \Bigg[& - \left\| \begin{bmatrix} w \\ \eta_i \end{bmatrix} - \mathbf{B}_i' \mathbf{Y}_i^{-1} \lambda_i \right\|^2 - \left\| v_i + D_i'(D_i D_i)^{-1} C_i z_i \right\|^2 \\
& - \sum_{j \in \mathbf{V}_i} \|v_{ij} + H_{ij}(W_{ij} Z_{ij} W_{ij}' + H_{ij} H_{ij}')^{-1} W_{ij} z_i\|^2 - 2 z_i' \sum_{j \in \mathbf{V}_i} \Phi_{ij} z_j \\
& - z_i'(\Delta_i + \frac{1}{\gamma^2} R_i) z_i - \frac{1}{\gamma^2} \delta_i' \check{R}_i \delta_i + \|w\|^2 + \|\eta_i\|^2 + \|v_i\|^2 + \sum_{j \in \mathbf{V}_i} \|v_{ij}\|^2 \Bigg] \\
< & -\frac{1}{\gamma^2} \sum_{i=1}^{N} (z_i' \hat{P}_i z_i + \delta_i' \check{P}_i \delta_i) + \sum_{i=1}^{N} \left(\|w\|^2 + \|\eta_i\|^2 + \|v_i\|^2 + \sum_{j \in \mathbf{V}_i} \|v_{ij}\|^2 \right). \qquad (38)
\end{aligned}
$$

The properties (AD-i'), (AD-ii') of the attack detector now readily follow from this inequality. Indeed, in the absence of disturbances and an attack, we have $w(t) = 0$, $v_i(t) = 0$, $v_{ij}(t) = 0$ and also $\eta_i(t) = 0$ since $f_i(t) = 0$ implies $\hat{f}_i(t) = 0$. Then the standard Lyapunov argument leads to the conclusion that $z_i(t)$, $\delta_i(t)$ converge to 0 exponentially. Also, in this case, $\varphi_i(t) = \Upsilon_i \hat{\varepsilon}_i(t) - \hat{f}_i(t) = -\Upsilon_i \delta_i(t)$ vanishes exponentially as $t \to \infty$. Thus, (AD-i') holds. On the other hand, when the system under consideration is subject to disturbances or when $f_i \neq 0$, then at least one of the signals $w(t)$, $v_i(t)$, $v_{ij}(t)$ or the attack approximation error $\eta_i(t)$ are non-zero. In this case, integrating both parts of the inequality (38) over $[0, T]$ and letting $T \to \infty$ leads to an inequality analogous to the condition (9):

$$
\begin{aligned}
& \sum_{i=1}^{N} \int_0^{\infty} ((e_i - \hat{e}_i)' \hat{P}_i (e_i - \hat{e}_i) + (\varepsilon_i - \hat{\varepsilon}_i)' \check{P}_i (\varepsilon_i - \hat{\varepsilon}_i)) dt \\
& \leq \gamma_d^2 \sum_{i=1}^{N} \left(\|x_0\|^2_{\hat{X}_i} + \int_0^{\infty} \left(\|w(t)\|^2 + \|v_i(t)\|^2 + \|\eta_i(t)\|^2 + \sum_{j \in \mathbf{V}_i} \|v_{ij}\|^2 \right) dt \right). \qquad (39)
\end{aligned}
$$

The bound (35) on the detection performance immediately follows from (39). □

We conclude this section by presenting a sufficient condition for the existence of a bounded solution to Eq. (33). It is based on the condition for the existence of a solution to Eq. (13) established in [42]. The condition applies only in the case when the plant and sensing patterns are time invariant, i.e. when the matrices A, B, C_i and D_i in (1) are constant.

Theorem 3 *Suppose (A, B) is stabilizable, and let P, Z_{ij}, $i = 1, \ldots, N$, $j \in \mathbf{V}_i$, be the matrices from Theorem 1. The following linear matrix inequality (LMI) conditions in the variables Π_i, γ_d^2 and $\hat{R}_i$, $\check{R}_i$, $i = 1, \ldots, N$,*

$$\begin{bmatrix} \begin{array}{c} \mathbf{A}_i'\Pi_i + \Pi_i\mathbf{A}_i + \mathbf{R}_i \\ -\gamma_d^2\left(\mathbf{C}_i'(D_iD_i')^{-1}\mathbf{C}_i + \sum_{j\in\mathbf{V}_i}\mathbf{W}_{ij}'(W_{ij}Z_{ij}W_{ij}' + H_{ij}H_{ij}')^{-1}\mathbf{W}_{ij}\right) \end{array} & \Pi_i\mathbf{B}_i \\ \mathbf{B}_i'\Pi_i & -\gamma_d^2 I \end{bmatrix} < 0,$$

$$\Pi_i = \Pi_i' > 0, \quad \gamma_d^2(\Phi + \Phi' - \Delta) + \hat{R} > \hat{P}, \quad \check{R}_i > \check{P}_i, \tag{40}$$

guarantee the satisfaction of the conditions of Theorem 1 for a sufficiently large X_i.

Proof Observe that every matrix pair $(\mathbf{A}_i, \mathbf{B}_i)$ is stabilizable. This conclusion is readily verified using the Hautus lemma. Indeed, $(\mathbf{A}_i, \mathbf{B}_i)$ is stabilizable if and only if for every s, Re $s > 0$, the equations

$$\begin{aligned} &(sI - A')z = 0, \\ &-\Upsilon_i'F_i'z + (sI - \Omega_i')\delta = 0, \\ &B'z = 0, \\ &F_i'z + \Gamma_i'\delta = 0 \end{aligned} \tag{41}$$

imply that $z = 0$, $\delta = 0$. Since (A, B) is stabilizable by assumption, then the first and the third equations in (41) yield $z = 0$. Therefore, the second and the fourth equations in (41) simplify as $(sI - \Omega_i')\delta = 0$, $\Gamma_i'\delta = 0$. Recall that (22) is a minimal realization, therefore we obtain that $\delta = 0$. This confirms that $(\mathbf{A}_i, \mathbf{B}_i)$ is stabilizable. The statement of the theorem now follows from Theorem 2 in [42]. □

The inequalities (40) are linear in the decision variables Π_i, $\hat{R}_i$, $\check{R}_i$ and γ_d^2, therefore the problem of finding a solution to the conditions of Theorem 2 can be reformulated as a convex problem,

$$\inf \gamma_d^2 \quad \text{subject to (40)}. \tag{42}$$

The problem (42) can be solved efficiently using the existing software, although this has to be done centrally. Its solutions can then be utilized in Theorem 2 to construct an attack detector.

3.4 Local Attack Detector Performance

The condition (35) reflects the overall global attack detection capability of the network. In practice, it may be more useful to monitor performance of individual detectors. By design, our method equips each observer with a local bound on the energy in the attack detection error, of the form (17):

$$\int_0^\infty (z_i'(t)\hat{R}_i z_i(t) + \delta_i(t)'\check{R}_i\delta_i(t))dt \leq \gamma_d^2\Bigg(\|x_0\|^2_{\hat{X}_i}$$
$$+\int_0^\infty\left(\|w(t)\|^2 + \|\eta_i(t)\|^2 + \|v_i(t)\|^2 + \sum_{j\in\mathbf{V}_i}\|v_{ij}(t)\|^2 + \|z_j(t)\|^2_{Z_{ij}^{-1}}\right)dt\Bigg). \quad (43)$$

Here, the matrices $\hat{R}_i$, $\check{R}_i$ are obtained from (32).

The bound (43) shows that by adjusting the matrices Z_{ij}, one can predict and adjust the disturbance attenuation level γ_d^2 as well as the weighted norms of the attack detector errors. The error δ_i is indicative of the attack input tracking performance at node i. It is of primary interest from the attack detection viewpoint. On the other hand, z_i indicates the accuracy of estimating the error of the plant observer at node i under the biasing attack. In Sect. 5, it will be used for correcting the bias introduced by the attack. The weights $\hat{R}_i$, $\check{R}_i$ on these errors are obtained through solving the inequality (32). Although this must be done centrally, this step must be carried out only once. In addition to equipping each node with the parameters needed for computing the gains of the attack detector, it also provides an assessment of the expected local performance of the detector.

4 Detection of Biasing False Data Injection Attacks

In this section, we extend the proposed H_∞ attack detection methodology to allow the detection of biasing attacks on local sensors and communication links. Instead of biased observers (4), we consider a more conventional situation where the adversary substitutes the measurements y_i collected by the sensor at node i with the tempered data,

$$y_i(t) = C_i(t)x(t) + D_i(t)v_i(t) + F_i f_i. \quad (44)$$

As was the case previously, $F_i \in \mathbf{R}^{m_i\times n_{f_i}}$ is a constant matrix, and $f_i \in \mathbf{R}^{n_{f_i}}$ is an unknown signal representing the attack input. We adopt the same model for the biasing signal f_i which was introduced in Sect. 3. That is, we assume that the signals f_i can be approximated, up to an error η_i using a signal $\hat{f}_i$ generated by a system of the form (22), and that the error $\eta_i = \hat{f}_i - f_i$ is $\mathscr{L}_2$-integrable. Biasing attacks on the communication links can be considered in the same manner, and the treatment of these attacks is no different from how we will treat the presence of biasing in the sensor data. Therefore, for the sake of keeping the presentation simple, we won't pursue this more general case here.

In contrast to the problem considered in Sect. 3, the biasing attacks (44) modify the innovation signals. Let us rewrite them in the form explicitly reflecting their dependency on the estimation errors and the state of the attack generator model ε_i,

$$\zeta_i = C_i(t)e_i + F_i\Upsilon_i\varepsilon_i + D_iv_i - F_i\eta_i, \tag{45}$$

$$\zeta_{ij} = -W_{ij}(e_j - e_i) + H_{ij}v_{ij}, \quad j \in \mathbf{V}_i. \tag{46}$$

The extended system combining the error dynamics of the observer (4) and the attack model (22) becomes

$$\begin{aligned}
\dot{e}_i &= (A(t) - L_i(t)C_i(t) - \sum_{j\in\mathbf{V}_i} K_{ij}(t)W_{ij})e_i + \sum_{j\in\mathbf{V}_i} K_{ij}(t)W_{ij}e_j - L_i(t)F_i\Upsilon_i\varepsilon_i \\
&\quad +B(t)w - L_i(t)D_i(t)v_i - \sum_{j\in\mathbf{V}_i} K_{ij}(t)H_{ij}v_{ij} + L_i(t)F_i\eta_i, \\
\dot{\varepsilon}_i &= \Omega_i\varepsilon_i + \Gamma_i\eta_i, \\
e_i(0) &= x_0, \quad \varepsilon_i(0) = 0.
\end{aligned} \tag{47}$$

Again, we observe that the effect of the attack is somewhat different in this case in that the gains at the variables associated with the attack input are scaled by the observer gain $L_i(t)$. This indicates that unless they are countered, biasing attacks may have a significant adverse impact on the performance of high-gain observers.

To obtain a network of attack detectors, we follow the idea presented in the previous section and employ a filter of the form (29) to estimate the state of the interconnected uncertain system comprised of the extended error dynamics (47) and track the attack signal f_i in the H_∞ sense. This will be done by processing the innovation signals (45), (46). This filter has the familiar form of the state observer

$$\begin{aligned}
\dot{\hat{e}}_i &= (A(t) - L_i(t)C_i(t) - \sum_{j\in\mathbf{V}_i} K_{ij}(t)W_{ij})\hat{e}_i + \sum_{j\in\mathbf{V}_i} K_{ij}(t)W_{ij}\hat{e}_j - L_i(t)F_i\Upsilon_i\hat{\varepsilon}_i \\
&\quad + \bar{L}_i(\zeta_i - C_i(t)\hat{e}_i - F_i\Upsilon_i\hat{\varepsilon}_i) + \sum_{j\in\mathbf{V}_i} \bar{K}_{ij}(t)(\zeta_{ij} - W_{ij}(\hat{e}_i - \hat{e}_j)), \\
\dot{\hat{\varepsilon}}_i &= \Omega_i\hat{\varepsilon}_i + \check{L}_i(t)(\zeta_i - C_i\hat{e}_i - F_i\Upsilon_i\hat{\varepsilon}_i) + \sum_{j\in\mathbf{V}_i} \check{K}_{ij}(t)(\zeta_{ij} - W_{ij}(\hat{e}_i - \hat{e}_j)), \\
\varphi_i &= \Upsilon_i\hat{\varepsilon}_i, \\
\hat{e}_i(0) &= 0, \quad \hat{\varepsilon}_i(0) = 0.
\end{aligned} \tag{48}$$

To obtain the gains of this observer $\bar{L}_i$, $\bar{K}_{ij}$, $\check{L}_i$, $\check{K}_{ij}$, we again resort to the analysis of disturbance attenuation properties of the corresponding large-scale system describing dynamics of the detector errors.

To present the formal statement of the detector design algorithm, we need to modify some of the notation in (31) as well as introduce an additional notation:

$$\begin{aligned}
\mathbf{A}_i(t) &= \begin{bmatrix} A(t) & 0 \\ 0 & \Omega_i \end{bmatrix}, \quad \mathbf{B}_i = \begin{bmatrix} B(t) & 0 & 0 \\ 0 & \Gamma_i & 0 \end{bmatrix}, \quad \mathbf{C}_i(t) = \begin{bmatrix} C_i(t) & F_i \Upsilon_i \end{bmatrix}, \\
\mathbf{W}_{ij} &= \begin{bmatrix} W_{ij} & 0 \end{bmatrix}, \quad \mathbf{D}_i = \begin{bmatrix} 0 & -F_i & D_i \end{bmatrix}, \\
\tilde{\mathbf{A}}_i(t) &= \mathbf{A}_i(t) - \mathbf{B}_i \mathbf{D}_i' (\mathbf{D}_i \mathbf{D}_i')^{-1} \mathbf{C}_i \\
&= \begin{bmatrix} A(t) & 0 \\ -\Gamma_i F_i'(F_i F_i' + D_i D_i')^{-1} C_i & \Omega_i - \Gamma_i F_i'(F_i F_i' + D_i D_i')^{-1} F_i \Upsilon_i \end{bmatrix}.
\end{aligned} \tag{49}$$

As previously, let Z_{ij} be a collection of positive definite $n \times n$ matrices, $i = 1, \dots, N$, $j \in \mathbf{V}_i$. Also, for every $i = 1, \dots, N$ consider the modification of the differential Riccati equation (33),

$$\begin{aligned}
\dot{\mathbf{Y}}_i &= \tilde{\mathbf{A}}_i \mathbf{Y}_i + \mathbf{Y}_i \tilde{\mathbf{A}}_i' \\
&\quad - \mathbf{Y}_i \Bigg(\mathbf{C}_i'(F_i F_i' + D_i D_i')^{-1} \mathbf{C}_i + \sum_{j \in \mathbf{V}_i} \mathbf{W}_{ij}'(W_{ij} Z_{ij} W_{ij}' + H_{ij} H_{ij}')^{-1} \mathbf{W}_{ij} \\
&\quad - \frac{1}{\gamma_d^2} \mathbf{R}_i \Bigg) \mathbf{Y}_i + \mathbf{B}_i (I - \mathbf{D}_i'(F_i F_i' + D_i D_i')^{-1} \mathbf{D}_i) \mathbf{B}_i', \qquad (50) \\
\mathbf{Y}_i(0) &= \mathbf{X}_i^{-1},
\end{aligned}$$

with a symmetric positive definite $\mathbf{X}_i = \mathrm{diag}[\hat{X}_i, \check{X}_i]$. As one can see, the differences in the attack model and the resulting structure of the attack detector lead to a somewhat more general differential Riccati equation involved in the calculation of the detector.

Theorem 4 *Suppose there exist a constant $\gamma_d > 0$ and positive-definite symmetric matrices $\hat{R}_i$, $\check{R}_i$, Z_{ij}, $j \in \mathbf{V}_i$, $i = 1, \dots N$, which satisfy the inequalities (32) and such that each differential Riccati equation (50) has a positive-definite symmetric-bounded solution $\mathbf{Y}_i(t)$ on the interval $[0, \infty)$, i.e. for all $t \geq 0$, $\alpha_1 I < \mathbf{Y}_i(t) = \mathbf{Y}_i'(t) < \alpha_2 I$, for some $\alpha_{1,2} > 0$. Then the network of observers (48) with the coefficients $\bar{L}_i$, $\bar{K}_{ij}$, $\check{L}_i$, $\check{K}_{ij}$, obtained by partitioning the matrices*

$$\begin{aligned}
\mathbf{L}_i(t) &= (\mathbf{Y}_i(t) \mathbf{C}_i(t)' + \mathbf{B}_i(t) \mathbf{D}_i(t)')(F_i F_i' + D_i D_i')^{-1}, \\
\mathbf{K}_{ij}(t) &= \mathbf{Y}_i(t) \mathbf{W}_{ij}'(W_{ij} Z_{ij} W_{ij}' + H_{ij} H_{ij}')^{-1}
\end{aligned} \tag{51}$$

according to (31) and letting $\bar{L}_i = \hat{L}_i - L_i$, $\bar{K}_{ij} = \hat{K}_{ij} - K_{ij}$, guarantees the satisfaction of the properties (AD-i'), (AD-ii') of Theorem 2. In particular, (35) holds.

Proof The proof of Theorem 4 is essentially the same as the proof of Theorem 2. It is based on the analysis of the errors of the attack detector (48),

$$
\begin{aligned}
\dot{z}_i &= (A(t) - \hat{L}_i(t)C_i(t) - \sum_{j\in\mathbf{V}_i} \hat{K}_{ij}(t)W_{ij})z_i - \hat{L}_i F_i \varUpsilon_i \delta_i - \sum_{j\in\mathbf{V}_i} \hat{K}_{ij}(t)W_{ij}z_j \\
&\quad + Bw - \hat{L}_i(t)D_i(t)v_i - \sum_{j\in\mathbf{V}_i} \hat{K}_{ij}(t)H_{ij}v_{ij} + \hat{L}_i F_i \eta_i, \\
\dot{\delta}_i &= \varOmega_i \delta_i - \check{L}_i(t)C_i(t)z_i - \sum_{j\in\mathbf{V}_i} \check{K}_{ij}(t)W_{ij}z_i - \sum_{j\in\mathbf{V}_i} \check{K}_{ij}(t)W_{ij}z_j \\
&\quad - \check{L}_i(t)D_i(t)v_i - \sum_{j\in\mathbf{V}_i} \check{K}_{ij}(t)H_{ij}v_{ij} + (\check{L}_i F_i + \varGamma_i)\eta_i, \qquad (52) \\
&\quad z_i(0) = x_0, \quad \delta_i(0) = 0.
\end{aligned}
$$

As in the proof of Theorem 2, the statement of the theorem follows using the Lyapunov function $\lambda_i' \mathbf{Y}_i^{-1} \lambda_i$ for this system, where as before, $\lambda_i = [z_i' \; \delta_i']'$. In particular, with this Lyapunov function we show that (39) holds in this case as well. □

5 Resilient Estimation Under Biasing Attacks

The analysis in the previous sections shows that the errors of the distributed observer can be estimated. Under normal circumstances when the system operates in a safe environment, these error estimates are not of any value. However, when the network is under attack, the estimates of the observer errors can be used for correcting the biasing effect of the attack, as demonstrated in the journal version of [7]. Indeed, let us express the true state of the plant as

$$
\begin{aligned}
x &= \hat{x}_i + e_i \\
&= \hat{x}_i + \hat{e}_i + z_i, \qquad (53)
\end{aligned}
$$

where $z_i = e_i - \hat{e}_i$ is the error of the attack detectors (29) and (48), respectively. That is, z_i can also be interpreted as the plant estimation error associated with choosing

$$
\hat{\hat{x}}_i(t) = \hat{x}_i(t) + \hat{e}_i(t) \qquad (54)
$$

to represent an estimate of the plant state. Furthermore, from (53), it follows that

$$
\int_0^T (x - \hat{\hat{x}}_i)' \hat{P}_i (x - \hat{\hat{x}}_i) dt = \int_0^T z_i' \hat{P}_i z_i dt. \qquad (55)
$$

Since the attack detectors (29) guarantee that the right- hand side of (55) remains bounded as $T \to \infty$ under conditions of Theorem 2, then we conclude that (29) can be utilized as a distributed plant observer in a situation when the network is subjected to a biasing interference. Similarly, Theorem 4 establishes that the attack detectors (48) provide an estimate of the plant state when an adversary injects biasing interference into some of the network sensors. This conclusion can be formalized as a theorem.

Theorem 5

(a) Consider the observer network (18) augmented with the distributed networked attack detector (29) whose coefficients $\bar{L}_i$, $\bar{K}_i$, $\check{L}_i$, $\check{K}_i$ are obtained using the procedure described in Theorem 2. Then, the following statements hold:

(i) In the absence of disturbances and attack, $\hat{\hat{x}}_i(t) \to x(t)$ exponentially for all $i = 1, \dots, N$;

(ii) In the presence of perturbations and biasing misappropriation attacks, the $\mathscr{L}_2$ performance cost of the combined observer (18), (29), $\sum_{i=1}^{T} \int_0^T (x - \hat{\hat{x}}_i)' \hat{P}_i (x - \hat{\hat{x}}_i) dt$ is bounded by the expression on the right-hand side of (39). Furthermore, the expression on the right-hand side of (43) provides a bound on the individual performance of each observer node, $\int_0^T (x - \hat{\hat{x}}_i)' \hat{R}_i (x - \hat{\hat{x}}_i) dt$.

(b) Similarly, consider the observer network (4) augmented with the distributed networked attack detectors (48) whose coefficients $\bar{L}_i$, $\bar{K}_i$, $\check{L}_i$, $\check{K}_i$ are obtained using the procedure described in Theorem 4. Then, in the absence of disturbances and biasing attacks on the sensors, the estimates $\hat{\hat{x}}_i(t)$ produced by the combined observer-detector network (4), (48) converges to $x(t)$ exponentially for all $i = 1, \dots, N$. Also, in the presence of disturbances and/or biasing data injection attacks, performance of these estimates is bounded, as explained in claim (ii) of part (a).

Note that both the extended observer (18), (29) and the extended observer (4), (48) produce two estimates of the plant state, $\hat{x}_i(t)$ and $\hat{\hat{x}}_i(t)$. In the absence of an attack, the estimates $\hat{x}_i(t)$ are produced by the filters (4) and are robust against the disturbances w, v_i and v_{ij}. In contrast with $\hat{x}_i(t)$, the 'corrected' estimates $\hat{\hat{x}}_i(t)$ are produced by the extended observer and are also robust against the fictitious disturbance η_i, representing an attack tracking error of the system (22). Even when the network is not under attack and $\eta_i = 0$, the signal η_i is still treated as a fictitious uncertainty input in the derivation of the observer for e_i. This makes the estimates $\hat{\hat{x}}_i(t)$ more conservative and less accurate than $\hat{x}_i(t)$. However, when the observer is under attack, $\hat{x}_i$ becomes biased whereas $\hat{\hat{x}}_i(t)$ remains unbiased. This shows that augmenting the observer network with the attack detectors (29) or (48) allows the network to remain functional under attack, albeit less accurate. Also, one can switch from $\hat{\hat{x}}_i(t)$ back to $\hat{x}_i$ once the attack has ceased. This allows the network to recover after the attack. That is, the observers augmented with the proposed attack detectors meet the requirements for resilience.

6 Illustrating Example

We use the example in [34] to demonstrate the efficacy of the proposed approach. We consider the misappropriation attack scenario discussed in Sect. 3. The plant is time invariant and evolves according to Eq. (1) with constant state and input matrices

$$A = \begin{bmatrix} 0.3775 & 0 & 0 & 0 & 0 & 0 \\ 0.2959 & 0.3510 & 0 & 0 & 0 & 0 \\ 1.4751 & 0.6232 & 1.0078 & 0 & 0 & 0 \\ 0.2340 & 0 & 0 & 0.5596 & 0 & 0 \\ 0 & 0 & 0 & 0.4437 & 1.1878 & -0.0215 \\ 0 & 0 & 0 & 0 & 2.2023 & 1.0039 \end{bmatrix}, \tag{56}$$

$$B = 0.1 I_{6\times 6}.$$

The plant is observed by six sensors. The first sensor measures the first and the second coordinates of the state vector, the second sensor measures the second and the third coordinates, etc, with the last sensor taking measurements of the sixth and the first coordinates. Therefore, for the fourth sensor, for example we have

$$C_{24} = \begin{bmatrix} 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \end{bmatrix}.$$

Also, $D_i = 0.01 I_2$, $\forall i$. With these matrices C_i, each matrix pair (A, C_i) is not detectable, however, it was demonstrated in [34] that a cyclic network of observers of the form (4) can be constructed which overcomes this deficiency. This network broadcasts the full vector $\hat{x}_i$, i.e. $W_{ij} = I_{6\times 6}$ for all $i, j \in \mathbf{V}_i$. However, we now allow for disturbances in the communication channels; cf. [36, 37]. Accordingly, we let $H_{ij} = 0.1 \times [1\ 1\ 1\ 1\ 1\ 1]'$.

We now wish to protect this network from biasing attacks. Let us assume that the attack inputs are scalar and therefore, we let $F_i = [1\ 1\ 1\ 1\ 1\ 1]'\ \forall i$. Accordingly we choose a scalar $G_i(s) = \frac{g_i}{s+2\beta_i}$, with $\beta_i = 20$, $g_i = 410$. These parameters were found to produce the attack detectors that were sufficiently fast compared with the duration of the attack; see Fig. 2. Next, we let $Z_{ij} = 0.01 \times I_{6\times 6}$ and solved the LMI optimization problem (42), to obtain the smallest $\gamma_d^2 = 5.7952$ for which the LMI conditions (41) were feasible. According to Theorem 3, this gave us a guarantee that the Riccati equations (33) with the found $\hat{R}_i$, $\check{R}_i$ and γ_d^2 have positive-definite bounded solutions as required in Theorem 2.

Next, the plant (1), the observers obtained in [34] and the attack detectors (29) were jointly simulated using MATLAB. The gains for the attack detector (29) were obtained from Theorem 2 using the found values of γ_d^2 and $\hat{R}_i$, $\check{R}_i$. To illustrate the impact of the disturbance, we applied sinusoidal signals of amplitude 50, with frequencies ranging from 100 to 330 rad/s. Also, a 3 s long attack input of amplitude 5 was applied at node 2 of the observer at time $t = 4$ s. That is, each component in the attack vector $F_2 f_2$ had the same amplitude as the components of the disturbance

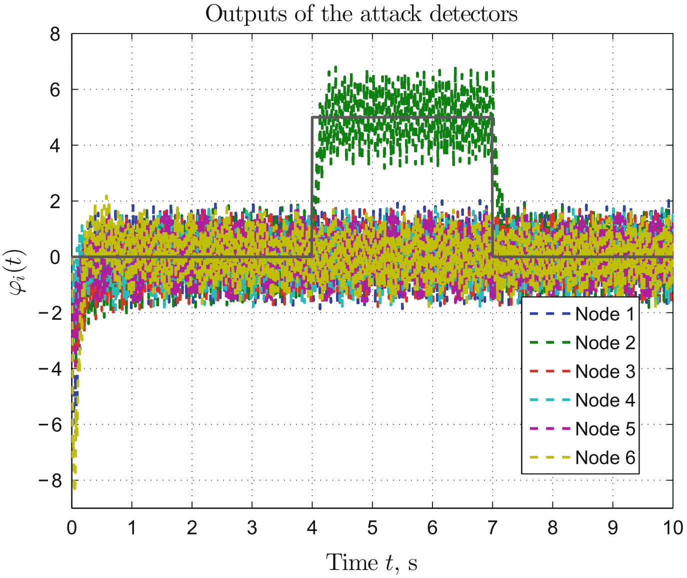


Fig. 2 Outputs of the attack detectors $\varphi_i(t)$. The solid line shows the attack input. The outputs of the detectors are plotted using the coloured dashed lines. The figure shows that the output of the detector at node 2 can be easily separated from the outputs of the other detectors

vectors $Bw(t)$ and $H_{ij}v_{ij}(t)$. Despite this, the attack detector was able to segregate the attack input from the disturbances and identify the node subjected to the attack, as shown in Fig. 2.

7 Concluding Remarks

The basis of the proposed approach to attack detection is modelling the biasing attack as an output of a fictitious tracking system (22) plus $\mathscr{L}_2$-integrable tracking error. This allowed us to recast the attack detector design problem as a problem of distributed stabilization of the system of detector error dynamics via output injection. The problem has been considered within an H_∞ framework, and we have proposed a decentralized H_∞ synthesis method for the design of distributed detectors of biasing attacks on distributed filter networks. The proposed detectors can pick a biasing attack from local sensory information complemented with information extracted from the routine information exchange within the network. This has been demonstrated for two types of biasing attacks, the false data injection attacks and the misappropriation attacks. The former bias the sensor measurements, and the latter target the network via directly biasing the dynamics of the compromised nodes. Extending our methodology to include similar biasing attacks on communication channels is a routine exercise which trivially follows the derivations in Sect. 4.

The derivation of the detectors employs the technique from [42], however it is applicable to a broad class of distributed observers (4) subjected to the biasing misappropriation and false data attacks described in the foregoing paragraph. This is because our algorithm computes the gains of the output injection stabilizing feedback, $\hat{L}_i$, $\hat{K}_{ij}$, $\check{L}_i$, $\check{K}_{ij}$, from which the gains of the attack detector are obtained by subtracting the gains of the original observer, as

$$\bar{L}_i = \hat{L}_i - L_i, \quad \bar{K}_{ij} = \hat{K}_{ij} - K_{ij}. \tag{57}$$

Thus, the attack detectors of the form (29) or (48) can be obtained regardless of how the gains L_i, K_{ij} of the original observer were obtained. For instance, the same procedure can be applied to design an attack detector network for a distributed Kalman filter of the form (4). However, the convergence and performance guarantees of our method rely on the assumption that the disturbances can be interpreted as finite energy perturbations.

Acknowledgements This chapter is dedicated to the memory of Roberto Tempo whose friendship and support will never go unappreciated. The work was supported in part by the Australian Research Council and the University of New South Wales.

References

1. B. Açıkmeşe, M. Mandić, and J.L. Speyer. Decentralized observers with consensus filters for distributed discrete-time linear systems. *Automatica*, 50(4):1037–1052, 2014.
2. T. Başar and P. Bernhard. *H^∞-optimal control and related minimax design problems: a dynamic game approach*. Birkhäuser, Boston, 2nd edition, 1995.
3. G. Battistelli and L. Chisci. Kullback-Leibler average, consensus on probability densities, and distributed state estimation with guaranteed stability. *Automatica*, 50(3):707–718, 2014.
4. R. Carli, A. Chiuso, L. Schenato, and S. Zampieri. Distributed Kalman filtering based on consensus strategies. *IEEE Journal on Selected Areas in Communications*, 26(4):622–633, 2008.
5. G. Dán and H. Sandberg. Stealth attacks and protection schemes for state estimators in power systems. In *First IEEE International Conference on Smart Grid Communications (SmartGridComm)*, pages 214–219, 2010.
6. C. E. de Souza, D. Coutinho, and M. Kinnaert. Mean square state estimation for sensor networks. *Automatica*, 72:108–114, 2016.
7. M. Deghat, V. Ugrinovskii, I. Shames, and C. Langbort. Detection of biasing attacks on distributed estimation networks. In *55th IEEE Conference on Decision and Control*, Las Vegas, NV, 2016.
8. F. Dorfler, F. Pasqualetti, and F. Bullo. Continuous-time distributed observers with discrete communication. *IEEE Journal of Selected Topics in Signal Processing*, 7(2):296–304, 2013.
9. Z. Guo, D. Shi, K.H. Johansson, and L. Shi. Optimal linear cyber-attack on remote state estimation. *IEEE Transactions on Control of Network Systems*, 4(1):4–13, March 2017.
10. D. Han, Y. Mo, and L. Xie. Robust state estimation against sparse integrity attacks. *arXiv preprint* arXiv:1601.04180, 2016.

11. M. Kamgarpour and C. Tomlin. Convergence properties of a decentralized Kalman filter. In *47th IEEE Conference on Decision and Control*, pages 3205–3210, 2008.
12. M.S. Mahmoud and H.M. Khalid. Distributed Kalman filtering: a bibliographic review. *IET Control Theory and Applications*, 7(4):483–501, 2013.
13. D. Marelli, M. Zamani, M. Fu, and B. Ninness. Distributed Kalman filter in a network of linear systems. *Systems and Control Letters*, 116:71–77, 2018.
14. N. Matni, Y.P. Leong, Y.S. Wang, S. You, M.B. Horowitz, and J.C. Doyle. Resilience in large scale distributed systems. *Procedia Computer Science*, 28:285–293, 2014.
15. P. Millán, L. Orihuela, C. Vivas, F.R. Rubio, D.V. Dimarogonas, and K.H. Johansson. Sensor-network-based robust distributed control and estimation. *Control Engineering Practice*, 21(9):1238–1249, 2013.
16. A. Mitra and S. Sundaram. Distributed observers for LTI systems. *IEEE Transactions on Automatic Control*, 63(12), 2018.
17. Y. Mo, T.H.-J. Kim, K. Brancik, D. Dickinson, H. Lee, A. Perrig, and B. Sinopoli. Cyber-physical security of a smart grid infrastructure. *Proceedings of the IEEE*, 100(1):195–209, 2012.
18. T.R. Nelson and R.A. Freeman. Decentralized H_∞ filtering in a multi-agent system. In *American Control Conference*, pages 5755–5760, St. Louis, MO, 2009.
19. R. Olfati-Saber. Distributed Kalman filter with embedded consensus filters. In *44th IEEE Conference on Decision and Control and 2005 European Control Conference*, pages 8179–8184, 2005.
20. R. Olfati-Saber. Distributed Kalman filtering for sensor networks. In *46th IEEE Conference on Decision and Control*, pages 5492–5498, 2007.
21. L. Orihuela, P. Millán, C. Vivas, and F.R. Rubio. Reduced-order H_2/H_∞ distributed observer for sensor networks. *International Journal of Control*, 86(10):1870–1879, 2013.
22. S. Park and N. C. Martins. Design of distributed LTI observers for state omniscience. *IEEE Transactions on Automatic Control*, 62(2):561–576, 2017.
23. F. Pasqualetti, F. Dorfler, and F. Bullo. Attack detection and identification in cyber-physical systems. *IEEE Transactions on Automatic Control*, 58(11):2715–2729, 2013.
24. F. Pasqualetti, A. Bicchi, and F. Bullo. Consensus computation in unreliable networks: A system theoretic approach. *IEEE Transactions on Automatic Control*, 57:90–104, 2012.
25. B.S. Rao and H.F. Durrant-Whyte. Fully decentralised algorithm for multisensor Kalman filtering. *Control Theory and Applications, IEE Proceedings D*, 138(5):413–420, Sep 1991.
26. I.D. Schizas, G.B. Giannakis, S.I. Roumeliotis, and A. Ribeiro. Consensus in ad hoc WSNs with noisy links—Part II: Distributed estimation and smoothing of random signals. *IEEE Transactions on Signal Processing*, 56(4):1650–1666, 2008.
27. B. Shen, Z. Wang, and Y.S. Hung. Distributed H_∞-consensus filtering in sensor networks with multiple missing measurements: The finite-horizon case. *Automatica*, 46(10):1682–1688, 2010.
28. B. Shen, Z. Wang, and X. Liu. A stochastic sampled-data approach to distributed H_∞ filtering in sensor networks. *IEEE Transactions on Circuits and Systems I: Regular Papers*, 58(9):2237–2246, 2011.
29. R. Smith and F. Hadaegh. Closed-loop dynamics of cooperative vehicle formations with parallel estimators and communication. *IEEE Transactions on Automatic Control*, 52(8):1404–1414, 2007.
30. R.S. Smith. Covert misappropriation of networked control systems: Presenting a feedback structure. *IEEE Control Systems*, 35(1):82–92, 2015.
31. M.V. Subbotin and R.S. Smith. Design of distributed decentralized estimators for formations with fixed and stochastic communication topologies. *Automatica*, 45(11):2491–2501, 2009.
32. S.L. Sun and Z.L. Deng. Multi-sensor optimal information fusion Kalman filter. *Automatica*, 40(6):1017–1023, 2004.
33. A. Teixeira, I. Shames, H. Sandberg, and K.H. Johansson. A secure control framework for resource-limited adversaries. *Automatica*, 51:135–148, 2015.

34. V. Ugrinovskii. Distributed robust filtering with H_∞ consensus of estimates. *Automatica*, 47(1):1–13, 2011.
35. V. Ugrinovskii. Conditions for detectability in distributed consensus-based observer networks. *IEEE Transactions on Automatic Control*, 58:2659–2664, 2013.
36. V. Ugrinovskii. Distributed robust estimation over randomly switching networks using H_∞ consensus. *Automatica*, 49(1):160–168, 2013.
37. V. Ugrinovskii. Gain-scheduled synchronization of parameter varying systems via relative H_∞ consensus with application to synchronization of uncertain bilinear systems. *Automatica*, 50(11):2880–2887, 2014.
38. V. Ugrinovskii and E. Fridman. A Round-Robin protocol for distributed estimation with H_∞ consensus. *Syst. Contr. Lett.* 69:103–110, 2014.
39. V. Ugrinovskii and C. Langbort. Distributed H_∞ consensus-based estimation of uncertain systems via dissipativity theory. *IET Control Theory and Applications*, 5(12):1458–1469, 2011.
40. J. Wu, L. Li, V. Ugrinovskii, and F. Allgöwer. Distributed filter design for cooperative H_∞-type estimation. In *IEEE Multi-Conference on Systems and Control*, Sydney, Australia, Sept. 2015.
41. J. Wu, V. Ugrinovskii, and F. Allgöwer. Cooperative estimation and robust synchronization of heterogeneous multi-agent systems with coupled measurements. *IEEE Transactions on Control of Network Systems*, 5(4), 2018.
42. M. Zamani and V. Ugrinovskii. Minimum-energy distributed filtering. In *53rd IEEE Conference on Decision and Control*, Los Angeles, CA, 2014.
43. Q. Zhu, L. Bushnell, and T. Basar. Resilient distributed control of multi-agent cyber-physical systems. In D.C. Tarraf, editor, *Control of Cyber-Physical Systems*, volume 449 of *Lecture Notes in Control and Information Sciences*, pp. 301–316. Springer, 2013.

Robust Static Output Feedback Design with Deterministic and Probabilistic Certificates

D. Arzelier, F. Dabbene, S. Formentin, D. Peaucelle and L. Zaccarian

Abstract Static output feedback design for linear plants is well known to be a challenging non-convex problem. The presence of plant uncertainty makes this challenge even harder. In this chapter, we propose a new BMI formulation with S-variables which includes an interesting link between state feedback, output injection, state injection, and static output feedback gains in a unified framework. Based on this formulation, the robust design problem is suitably addressed by iterative optimization procedures with either deterministic or probabilistic viewpoints exploiting the fact that Lyapunov certificates are separated from the control gain design variables. The deterministic approach is for affine polytopic systems. The probabilistic approach requires no assumption on the uncertain system, and is based on the Scenario with Certificates (SwC) method which was recently proposed to address certain static anti-windup design problems. Numerical results illustrate the effectiveness of the approach in both deterministic and stochastic cases.

D. Arzelier · D. Peaucelle · L. Zaccarian
LAAS-CNRS, Université de Toulouse, CNRS, Toulouse, France
e-mail: arzelier@laas.fr

D. Peaucelle
e-mail: peaucelle@laas.fr

F. Dabbene (✉)
CNR-IEIIT, Politecnico di Torino, Turin, Italy
e-mail: fabrizio.dabbene@ieiit.cnr.it

S. Formentin
Politecnico di Milano, Milan, Italy
e-mail: simone.formentin@polimi.it

L. Zaccarian
Industrial Engineering Department, University of Trento, Trento, Italy
e-mail: zaccarian@laas.fr

© Springer Nature Switzerland AG 2018

T. Başar (ed.), *Uncertainty in Complex Networked Systems*,
Systems & Control: Foundations & Applications,
https://doi.org/10.1007/978-3-030-04630-9_3

1 Introduction

Static output feedback (SOF) represents probably the simplest and most intuitive way to design a feedback control law: the plant's output is measured and fed back to the input, multiplied by a specifically designed static gain. Its straightforward implementation and the fact that the full state vector is usually not accessible, and only a partial information is available via the measured output, render it particularly attractive to control designers and practitioners. Moreover, it is known that different problems related to the design of dynamic controllers, such as e.g., fixed/low-order control, can be recast as a SOF design problem by introducing a suitable re-parameterization [10].

However, it has been known since many years that the SOF implementation simplicity is counteracted by an intrinsic complexity in obtaining strong theoretical results: the problem is extremely difficult, and no systematic constructive numerical solutions exist guaranteeing SOF design, or allowing to determine whether such a feedback does not exist. Even its exact theoretical complexity is not known. Indeed, it is easy to see that the problem is immediately rewritten in terms of a bilinear matrix inequality (BMI), whose solution is known to be NP-hard [16]. The interested reader can refer to the 1997 survey [19], or to the most recent overview [18]. In particular, in [18], the different possible solutions proposed in the literature for tackling the SOF problem are discussed and classified according to the specific approach adopted: (i) Methods based on the numerical solution of the ensuing BMI problems: these techniques directly tackle the bilinear problem by making recourse to specific optimization solvers such as e.g., the PENBMI and PENLAB toolboxes [11, 14]. (ii) Methods based on Lyapunov theory: most of these methods present an iterative algorithm in which a set of linear matrix inequalities (LMIs) is iteratively repeated, until some termination criteria are met. (iii) Non-Lyapunov-based static output feedback control strategies, such as those based on the direct solution of the non-smooth optimization problem of minimizing the spectral abscissa of the closed-loop system, (see for instance, the free package HIFOO [2, 13] or the MATLAB© macro `HINFSTRUCT`).

Among the previously introduced classes, the Lyapunov-based one presents some very interesting features, as pointed out in [18]. First, these techniques allow keeping a clear insight into the original problems, an insight that is usually lost when directly tackling the problem through an optimization-based approach. Second, the formulation in many cases immediately extends to *uncertain* problems, i.e., problems in which the plant to be controlled is not perfectly known but instead is affected by uncertainty. This is an important characteristic, which is becoming of fundamental importance in modern control design. Hence, iterative LMI methods allow extending the approach to the solution of *robust static output feedback* (R-SOF) problems. Clearly, the presence of plant uncertainty makes this challenging problem even harder. Hence, if on one side, one may expect that the solution of the R-SOF problem will enlarge its practical interest, on the other side, one should be aware that R-SOF solutions may not exist in many cases.

In this chapter, we follow the Lyapunov-based approach, and we introduce a novel BMI formulation, based on S-variables. An interesting feature of the proposed formulation is that the involved design variables provide a clear link to the state feedback, output injection, state injection, and static output feedback gains. In particular, the formulation captures all these subproblems in a unified framework. Moreover, the formulation immediately extends to the uncertain case, so that vertex results available for polytopic-type uncertainty can be directly applied. Furthermore, it is observed that a key feature of the derived framework is that it maintains an explicit distinction between *design variables* (i.e., those variables directly involved in the definition of the controller gain), and the *certificates*, (i.e., variables whose existence is necessary to prove the existence of a SOF, but that are not involved in the controller construction). This paves the way to the use of recent results on probabilistic robust design, based on the so-called Scenario with Certificates (SwC) approach [12]. This approach represents one of the more recent findings in the area of randomized methods for systems and control [4, 20], emerged in the last decade to synergize with the standard deterministic methods for control of systems with uncertainty. Results in this area are based on a combination of probability and random sampling, and their goal is to provide the research engineer with robustness guarantees which hold only with high probability. The payback is a reduction in the computational complexity of classical control algorithms, and in the conservativeness of standard robust control techniques.

The chapter represents the confluence and combination of two different viewpoints to handle uncertainty in systems: the deterministic/robust approach, in which one is interested in obtaining guaranteed results, that hold for every possible instance of the uncertainty, and the so-called probabilistic approach, which characterizes the uncertain parameters as random variables, and then evaluates the system performance in terms of probabilities. This confluence was made possible by the farsightedness and vision of our colleague and friend Roberto Tempo (1956–2017), recently suddenly passed away. Roberto was a strong believer in collaboration and cross-fertilization of research. He always insisted that the two approaches should not be viewed as an alternative but rather complementary to each other: one adds to the other.

To describe the philosophy underlying the present work, we use Roberto's words, taken from the proposal of one of the first formal collaborations between our two groups[1]: *"Robustness can be tackled by two means. One, probabilistic, consists in testing a finite number of configurations among the infinitely many admissible ones. This approach is said to be optimistic in the sense that if a level of performance is valid for all tested cases, it may not hold for the actual ones. The second approach, deterministic, provides, using mathematical tools, a guaranteed level of performances for all configurations. It is unfortunately conservative (or pessimistic) in the sense that the guaranteed performance is usually worse than the worst case. The project aims at comparing and hence enriching the optimistic and pessimistic approaches."*

[1] Bilateral Project "Convex optimization and randomized algorithms for robust control" (CORARC), between IEIIT CNR and LAAS CNRS (2012).

We organize the presentation in three main sections. Section 2 presents the unified S-variable formulation used throughout the chapter. Section 3 illustrates deterministic results and a corresponding heuristic design procedure. Section 4 presents parallel results providing probabilistic guarantees. Finally, we illustrate the effectiveness of the proposed constructions on numerical examples in Sect. 5.

Notation: I stands for the identity matrix. A^T is the transpose of the matrix A. $\{A\}^{\mathscr{S}}$ stands for the symmetric matrix $\{A\}^{\mathscr{S}} = A + A^T$. For a matrix $A \in \mathbb{R}^{n\times m}$ of rank r, $A^{\perp} \in \mathbb{R}^{m\times(m-r)}$ is the matrix of maximal rank such that $AA^{\perp} = 0$, and $A^{\circ} \in \mathbb{R}^{r\times n}$ stands for the full rank matrix such that $A^{\circ}A$ is full rank. $A \succ B$ is the matrix inequality stating that $A - B$ is symmetric positive definite. The terminology *"congruence operation of A on B"* is used to denote $A^T BA$. If A is full rank, and $B \succ 0$, the congruence operation of A on B gives a positive definite matrix: $A^T BA \succ 0$. A matrix inequality of the type $N(X) \succ 0$ is said to be a linear matrix inequality (LMI for short), if $N(X)$ is affine in the decision variables X. In the following, decision variables are highlighted[2] using the blue color. $\Phi_{\bar{v}} = \{\phi_{v=1\ldots\bar{v}} \geq 0,\ \sum_{v=1}^{\bar{v}} \phi_v = 1\}$ is the unitary simplex in $\mathbb{R}^{\bar{v}}$. The elements ϕ of unitary simplexes are used to describe polytopic-type uncertainties. In the following, uncertainties ϕ are highlighted using the red color.

2 S-Variables Formulation of Robust Stability

We consider an LTI uncertain system that depends on a vector of constant but uncertain parameters q (to our knowledge, the proposed results are the first to address the case where all matrices are parameter dependent):

$$\begin{pmatrix} \dot{x} \\ y \end{pmatrix} = \begin{bmatrix} A(q) & B(q) \\ C(q) & 0 \end{bmatrix} \begin{pmatrix} x \\ u \end{pmatrix} \tag{1}$$

where $x \in \mathbb{R}^n$ is the state vector, $u \in \mathbb{R}^m$ is the vector of control inputs, and $y \in \mathbb{R}^p$ is the vector of measured outputs. We assume that the uncertain parameters q take values in a set $\mathbb{Q}$ whose structure will be specified next, depending on whether a deterministic or a probabilistic approach is adopted for the static output feedback stabilizer design.

For the uncertain system (1), our primary goal is to design a robustly stabilizing static output feedback gain. Such a design problem can be rigorously defined as follows:

(OF) Goal. Design a static output feedback gain F, such that the closed loop between plant (1) and $u = Fy$, corresponding to $\dot{x} = (A(q) + B(q)FC(q))x$ is robustly stable, namely the matrix $A(q) + B(q)FC(q)$ is Hurwitz for all $q \in \mathbb{Q}$.

[2] Not available in the html-only version of the book.

A suggestive aspect of the approach presented in the sequel is that a few auxiliary (and arguably simpler) problems will turn out to be instrumental for the solution of the (OF) Goal, and correspond to:

- **(SF) Goal**. Design a state feedback gain K such that the closed loop between plant (1) and $u = Kx$, corresponding to $\dot{x} = (A(q) + B(q)K)x$ is robustly stable.
- **(OI) Goal**. Design an output injection gain L such that the closed loop $\dot{x} = (A(q) + LC(q))x$ is robustly stable.
- **(SI) Goal**. Design a state injection gain J such that the closed loop $\dot{x} = (A(q) + J)x$ is robustly stable.

For the problems above, a necessary condition for the existence of a solution to the (SF) goal is that the pair $(A(q), B(q))$ is stabilizable for all $q \in \mathbb{Q}$. Moreover, a necessary condition for the existence of a solution to the (OI) goal is that the pair $(C(q), A(q))$ is detectable for all $q \in \mathbb{Q}$ (indeed, L is well understood as the gain of a full-order Luenberger observer). Finally, both conditions above are necessary for the existence of a solution to our main goal (OF), whereas goal (SI) is trivial and always feasible as long as $\mathbb{Q}$ is bounded and matrix $A(\cdot)$ is a locally bounded function.

The heuristic approach proposed in this chapter for the solution of the (OF) goal is based on a main result presented here, wherein we manage to represent all the design goals (OF), (SF), (OI) and (SI) listed above within a single matrix inequality depending bilinearly on a set of variables to be (optimally) selected. This matrix inequality arises from the dual calculations associated with [9, Thm 6.8], and involves a Lyapunov certificate $X(q) \succ 0$ and a number of S-variables. It corresponds to:

$$\begin{bmatrix} 0 & 0 & X(q) \\ 0 & 0 & 0 \\ X(q) & 0 & 0 \end{bmatrix} \prec \left\{ \begin{bmatrix} I \\ -\left(\lambda \begin{bmatrix} C(q) \\ 0_{p-n,n} \end{bmatrix} + M(q)\right) \\ -A(q) \end{bmatrix} S_1(q) + \begin{bmatrix} 0 \\ S_2 \\ B(q)Z \end{bmatrix} \begin{bmatrix} 0 & I & -H^T \end{bmatrix} \right\}^{\mathscr{S}} . \tag{2}$$

In particular, the relation between feasibility of (2) for certain selections of the blue variables, and the four design problems (OF), (OI), (SF), and (SI) is clarified in the next main result.

Theorem 1 *Consider system* (1) *and any selection of variables* $X > 0$, λ, M, S_1, S_2, Z, H *satisfying* (2) *for all* $q \in \mathbb{Q}$. *The following holds:*

- **(OF)** *if* $\lambda = 1$, $M(q) = 0$ *for all* $q \in \mathbb{Q}$ *and* S_2 *is nonsingular, then selection* $F = -ZS_2^{-1} \begin{bmatrix} I_p \\ 0_{n-p,p} \end{bmatrix}$ *solves the (OF) goal;*
- **(OI)** *if* $\lambda = 1$ *and* $M(q) = 0$ *for all* $q \in \mathbb{Q}$, *then selection* $L = H \begin{bmatrix} I_p \\ 0_{n-p,p} \end{bmatrix}$ *solves the (OI) goal;*

- **(SF)** *if* $\lambda = 0$, $M(q) = M$ *is common to all* $q \in \mathbb{Q}$, *and* S_2 *is nonsingular, then* $K = -ZS_2^{-1}M$ *solves the (SF) goal;*
- **(SI)** *if* $\lambda = 0$ *and* $M(q) = M$ *is common to all* $q \in \mathbb{Q}$, *then* $J = HM$ *solves the (SI) goal.*

We shall prove the four items of the theorem one by one. In particular, given any $q \in \mathbb{Q}$, for each one of the four items, we show below that the corresponding closed-loop matrix is Hurwitz. But before going into each individual proof, let us state the following facts. Assuming invertibility of S_2, the congruence operation of $\begin{bmatrix} I & 0 & 0 \\ 0 & B(q)(-ZS_2^{-1}) & I \end{bmatrix}$ on (2) implies

$$\begin{bmatrix} 0 & X(q) \\ X(q) & 0 \end{bmatrix} \prec \left\{ \begin{bmatrix} I \\ -A(q) - B(q)(-Z{S_2}^{-1}) \left(\lambda \begin{bmatrix} C(q) \\ 0_{p-n,n} \end{bmatrix} + M(q) \right) \end{bmatrix} \hat{S}_1(q) \right\}^{\mathscr{S}} \tag{3}$$

where $\hat{S}_1(q) = S_1 \begin{bmatrix} I & 0 & 0 \\ 0 & B(q)(-ZS_2^{-1}) & I \end{bmatrix}^T$, and the congruence operation of $\begin{bmatrix} I & 0 & 0 \\ 0 & H & I \end{bmatrix}$ on (2) implies

$$\begin{bmatrix} 0 & X(q) \\ X(q) & 0 \end{bmatrix} \prec \left\{ \begin{bmatrix} I \\ -A(q) - H \left(\lambda \begin{bmatrix} C(q) \\ 0_{p-n,n} \end{bmatrix} + M(q) \right) \end{bmatrix} \check{S}_1 \right\}^{\mathscr{S}} \tag{4}$$

where $\check{S}_1 = S_1 \begin{bmatrix} I & 0 & 0 \\ 0 & H & I \end{bmatrix}^T$. The uncertainty q will be omitted in most steps of the following proofs to simplify the notations.

Proof of (OF). We need to show that matrix $A(q) + B(q)FC(q) = A + BFC$ is Hurwitz. Using the assumption that $\lambda = 1$, $M(q) = 0$, invertibility of S_2, and the selection $F = -ZS_2^{-1} \begin{bmatrix} I_p \\ 0_{n-p,p} \end{bmatrix}$, inequality (3) implies

$$\begin{bmatrix} 0 & X \\ X & 0 \end{bmatrix} \prec \left\{ \begin{bmatrix} I \\ -A - BFC \end{bmatrix} \hat{S}_1 \right\}^{\mathscr{S}} . \tag{5}$$

This S-variable inequality, together with $X \succ 0$, implies that $A + BFC$ is Hurwitz (see [9]). This is also corroborated by performing a congruence operation of $\begin{bmatrix} A + BFC & I \end{bmatrix}$ on (5) which gives the classical Lyapunov inequality: $(A + BFC)X + X(A + BFC)^T \prec 0$.

Proof of (OI). We need to show that matrix $A(q) + LC(q) = A + LC$ is Hurwitz. Using the assumption that $\lambda = 1$, $M(q) = 0$, and the selection $L = H \begin{bmatrix} I_p \\ 0_{n-p,p} \end{bmatrix}$, inequality (4) implies

$$\begin{bmatrix} 0 & X \\ X & 0 \end{bmatrix} \prec \left\{ \begin{bmatrix} I \\ -A - LC \end{bmatrix} \check{S}_1 \right\}^{\mathscr{S}}.$$

This S-variable inequality, together with $X \succ 0$, implies that $A + LC$ is Hurwitz.
Proof of (SF). We need to show that matrix $A(q) + B(q)K = A + BK$ is Hurwitz. Using the assumption that $\lambda = 0$, invertibility of S_2, and the selection $K = -ZS_2^{-1}M$, inequality (3) implies

$$\begin{bmatrix} 0 & X \\ X & 0 \end{bmatrix} \prec \left\{ \begin{bmatrix} I \\ -A - BK \end{bmatrix} \hat{S}_1 \right\}^{\mathscr{S}}.$$

This S-variable inequality, together with $X \succ 0$, implies that $A + BK$ is Hurwitz.
Proof of (SI). We need to show that matrix $A(q) + J = A + J$ is Hurwitz. Using the assumption that $\lambda = 0$ and the selection $J = HM$, inequality (4) implies

$$\begin{bmatrix} 0 & X \\ X & 0 \end{bmatrix} \prec \left\{ \begin{bmatrix} I \\ -A - J \end{bmatrix} \check{S}_1 \right\}^{\mathscr{S}}.$$

This S-variable inequality, together with $X \succ 0$, implies that $A + J$ is Hurwitz. ■

Remark 1 Theorem 1 establishes conditions for specific selections of variables λ and M. Alternative cases are also of interest. In particular, for the general case when $M \neq 0$ and $\lambda \neq 1$, S-variable conditions (3) and (4) show, respectively, that matrices $A(q) - B(q)ZS_2^{-1}\left(\lambda \begin{bmatrix} C(q) \\ 0_{n-p,p} \end{bmatrix} + M(q)\right)$ and $A(q) + H\left(\lambda \begin{bmatrix} C(q) \\ 0_{n-p,p} \end{bmatrix} + M(q)\right)$ are Hurwitz. These properties clarify the rationale behind the heuristic algorithm proposed in the next section, which stems from picking an initial "simple" selection such that $A(q) + HM$ be Hurwitz, and then performing iterations aiming at minimizing the norm of $M(q)$ while converging to $\lambda = 1$, so that the first one of the two matrices above corresponds to the closed loop with the static output feedback gain.

3 Robust Deterministic Static Output Feedback Design

3.1 Deterministic Robust Stability

In the deterministic approach addressed in this section, we shall assume a polytopic uncertainty structure where the matrices in (1) lie in the convex hull of vertices computed at extremal values $q^{[v]}$, $v = 1, \ldots, \bar{v}$, with $\bar{v}$ being the number of vertices of the polytopic representation:

$$\mathbb{Q} = \{q = \sum_{v=1}^{\bar{v}} \phi_v q^{[v]}, \quad \phi = (\phi_1, \ldots, \phi_{\bar{v}}) \in \Phi_{\bar{v}}\}. \tag{6}$$

A model from this uncertain polytopic set is parameterized by the barycentric coordinates $\phi \in \Phi_{\bar{v}}$ in the following form:

$$\begin{pmatrix} \dot{x} \\ y \end{pmatrix} = \begin{bmatrix} A_\phi & B_\phi \\ C_\phi & 0 \end{bmatrix} \begin{pmatrix} x \\ u \end{pmatrix} = \sum_{v=1}^{\bar{v}} \phi_v \begin{bmatrix} A(q^{[v]}) & B(q^{[v]}) \\ C(q^{[v]}) & 0 \end{bmatrix} \begin{pmatrix} x \\ u \end{pmatrix}. \tag{7}$$

Alternatively, in the probabilistic approach addressed in Sect. 4, we will assume a more general not necessarily convex dependence of the matrices in (1) on q, not requiring convexity of the uncertainty set.

The polytopic representation in (7), together with the peculiar structure of the S-variable characterization in (2), allows providing bilinear conditions imposed at the vertices of the polytope (6). Then, we may apply convex combinations to conclude robust stability in the whole polytope, as long as the dependence on the uncertain parameters is affine. In particular, a problematic term arises from the product between $S_1(q)$ and other uncertain variables in (2). Due to this fact, we propose the use of a more conservative condition, corresponding to

$$\begin{bmatrix} 0 & 0 & X^{[v]} \\ 0 & 0 & 0 \\ X^{[v]} & 0 & 0 \end{bmatrix} \prec \left\{ \begin{bmatrix} I \\ -\left(\lambda \begin{bmatrix} C(q^{[v]}) \\ 0_{p-n,n} \end{bmatrix} + M^{[v]} \right) \\ -A(q^{[v]}) \end{bmatrix} S_1 + \begin{bmatrix} 0 \\ S_2 \\ B(q^{[v]}) Z \end{bmatrix} \begin{bmatrix} 0 & I & -H^T \end{bmatrix} \right\}^{\mathscr{S}}, \tag{8}$$

for all $v = 1 \dots \bar{v}$, where we selected a common value S_1 for all the values of $q \in \mathbb{Q}$. We may then prove the following Corollary to Theorem 1.

Corollary 1 *Consider system* (7) *and any selection of variables* $X^{[v]} \succ 0$, λ, $M^{[v]}$, S_1, S_2, Z, H *satisfying* (8) *for all* $v = 1 \dots \bar{v}$. *The following holds:*

- **(OF)** *if* $\lambda = 1$, $M^{[v]} = 0$ *for all* $v = 1 \dots \bar{v}$, *and* S_2 *is nonsingular, then selection* $F = -Z S_2^{-1} \begin{bmatrix} I_p \\ 0_{n-p,p} \end{bmatrix}$ *solves the (OF) goal;*
- **(OI)** *if* $\lambda = 1$ *and* $M^{[v]} = 0$ *for all* $v = 1 \dots \bar{v}$, *then selection* $L = H \begin{bmatrix} I_p \\ 0_{n-p,p} \end{bmatrix}$ *solves the (OI) goal;*
- **(SF)** *if* $\lambda = 0$, $M^{[v]} = M$ *is common to all* $v = 1 \dots \bar{v}$, *and* S_2 *is nonsingular, then* $K = -Z S_2^{-1} M$ *solves the (SF) goal;*
- **(SI)** *if* $\lambda = 0$ *and* $M^{[v]} = M$ *is common to all* $v = 1 \dots \bar{v}$, *then* $J = HM$ *solves the (SI) goal.*

Proof The proof is based on the selection of the parameter-dependent matrix $X_\phi = \sum_{v=1}^{\bar{v}} \phi_v X^{[v]} \succ 0$, which emerges naturally when performing a convex combination, through ϕ of inequalities (8), providing

$$\begin{bmatrix} 0 & 0 & X_\phi \\ 0 & 0 & 0 \\ X_\phi & 0 & 0 \end{bmatrix} \prec \left\{ \begin{bmatrix} I \\ -\left(\lambda \begin{bmatrix} C_\phi \\ 0_{p-n,n} \end{bmatrix} + M_\phi\right) \\ -A_\phi \end{bmatrix} S_1 + \begin{bmatrix} 0 \\ S_2 \\ B_\phi Z \end{bmatrix} \begin{bmatrix} 0 & I & -H^T \end{bmatrix} \right\}^{\mathscr{S}} . \tag{9}$$

Since Eq. (9) involves the matrices of the uncertain system (7), the proof is completed following steps parallel to those of the proof of Theorem 1, using the polytopic stability certificate X_ϕ. ■

Remark 2 While Theorem 1 provides conditions that are hard to check in practice, Corollary 1 corresponds to a conservative way to obtain a viable practical approach to the problem. In particular, the use of a common value of S_1 for all vertices is key to ensuring the affine nature of the conditions with respect to the polytopic uncertainty in (6), so that the convex combination can be carried over to the uncertainty-dependent conditions in (8). Note that while the common value of S_1 may be a source of conservatism of the tractable conditions of Corollary 1 (as compared to those of Theorem 1), a byproduct is the polytopic nature of the selected Lyapunov function (see [9, Lemma 3.3] which proves that the search for polytopic Lyapunov certificates is lossless under the constraint that the S-variable is common to all uncertainties). More general parameter-dependent Lyapunov functions may be effective at reducing the conservatism, perhaps at the expense of a higher computational burden. This is one of the goals of the probabilistic approach adopted in Sect. 4.

3.2 Iterative Heuristic for Deterministic Robust Control

In this section, we propose a heuristic procedure to design a robust static output feedback exploiting the matrix inequalities (8) and Corollary 1. The proposed approach is an iterative procedure to address the bilinear nature of (8), while starting from a reasonable initial condition. It consists of three fundamental phases:

- an *initialization phase*, which solves the (SI) and (SF) goals, also providing an initial guess of a solution to (8) having promising features in terms of convergence to the condition $\lambda = 1$ and $M(\cdot) \equiv 0$ required in item (OF) of Corollary 1;
- an *iteration phase*, which iterates between two steps aiming at refining the candidate solution to the BMI in the direction of this necessary condition $\lambda = 1$ and $M(\cdot) \equiv 0$;
- a *validation phase*, comprising a semidefinite program solving the (OF) and (OI) goals, thus providing a static output feedback selection if the previous phase converged to a solution sufficiently close to the condition $\lambda = 1$ and $M(\cdot) \equiv 0$.

Let us present the three abovementioned phases one by one.

3.2.1 Initialization Phase

The initialization phase aims at finding an initial selection of $X^{[v]} > 0$, λ, $M^{[v]}$, S_1, S_2, Z, H satisfying (8). A possible strategy is to fix variables λ, $M^{[v]}$, and H, so that optimizing the remaining variables is a convex LMI problem. We then select these variables according to the following straightforward consequence of Corollary 1.

Proposition 1 *For a selection $X^{[v]} \succ 0$, $\lambda = 0$, $M^{[v]} = M$, S_1, S_2, Z, H to be a solution of* (8)*, it is necessary that $A(q) + J = A(q) + HM$ be Hurwitz for all $q \in \mathbb{Q}$.*

Motivated by the proposition above, we propose the following selection:

$$\lambda = \lambda_0 = 0, \quad M^{[v]} = M_0, \quad H = H_0 = J_0 M_0^{-1}, \tag{10}$$

where J_0 ensures that $A(q) + J_0$ be Hurwitz for all $q \in \mathbb{Q}$, and M_0 is some invertible common selection of $M^{[v]}$. More specifically, keep in mind that we aim for the following convergence

$$\lambda \begin{bmatrix} C(q) \\ 0_{n-p,p} \end{bmatrix} + M(q) \to \begin{bmatrix} C(q) \\ 0_{n-p,p} \end{bmatrix}.$$

A natural choice of initial M_0 is hence such that its first p rows mimic $C(q)$. Therefore define $C_m = \frac{1}{\bar{v}} \sum_{v=1}^{\bar{v}} C(q^{[v]})$ the average of all matrices computed at vertices, and choose

$$M_0 = \begin{bmatrix} C_m^{\circ} C_m \\ C_m^{\perp T} \end{bmatrix}. \tag{11}$$

In this way, M_0 is square and non-singular and its first rows span the same range as the average of the $C(q)$ matrices.

For the initial guess of the SI matrix J_0 in (10), to ensure that $A(q) + J_0$ be robustly stable, let μ denote the maximum real part of all matrices $A(q^{[v]})$. Then we may select

$$J_0 = (-\mu - h)I, \tag{12}$$

where $h > 0$ is a positive scalar. For a sufficiently large value of h, matrix $A(q) + J_0$ is then robustly stable. Clearly, increasing h provides a natural way to strengthen this robust stability condition, and is the baseline intuition for the initialization algorithm below.

Phase 1. INITIALIZATION (PROVIDES SOLUTIONS TO (SI) AND (SF))

```
1: Input: Select the initial values as per (10)–(12) with $h = 1$.
2: Iteration: Solve the following LMI problem for $v = 1 \dots \bar{v}$:
```

$$X^{[v]} \succ 0,$$
$$\begin{bmatrix} 0 & 0 & X^{[v]} \\ 0 & 0 & 0 \\ X^{[v]} & 0 & 0 \end{bmatrix} \prec \left\{ \begin{bmatrix} I \\ -M_0 \\ -A(q^{[v]}) \end{bmatrix} S_1 + \begin{bmatrix} 0 \\ S_2 \\ B(q^{[v]})Z \end{bmatrix} \begin{bmatrix} 0 \; I \; -H_0^T \end{bmatrix} \right\}^{\mathscr{S}} . \tag{13}$$

If (13) is feasible, go to the next step. Otherwise, increase h, redefine H_0 according to (10), (12) and repeat Step 2. If for larger values of h no solution exists, then stop: the iterative heuristic fails.

3: **Output** If a solution $X^{[v]}$, S_1, S_2, Z to (13) is found, then output $S_{1,0} = S_1$, $\hat{K}_0 = -ZS_2^{-1}$, $K = -ZS_2^{-1}M_0$ and $J = H_0M_0$. From Corollary 1, K and J are proved to be robustly stabilizing SF and SI gains, respectively.

It is emphasized that there is no guarantee that the algorithm provides a correct solution, and even in the case where there exists a gain K inducing a common quadratic Lyapunov certificate X for the corresponding matrices $A(q^{[v]}) + B(q^{[v]})K$, it is unclear how to get a proof of its successful termination. Nevertheless, practical experience revealed that the algorithm is quite effective in finding a feasible solution to (8). Moreover, there was not a need to iterate on the value of h. If the LMIs were unfeasible for $h = 1$, then they happened to be unfeasible for larger values as well.

3.2.2 Iteration Phase

If the initialization phase provides an initial feasible solution to (8), we may proceed with the iteration phase, whose goal is (starting from $\lambda_0 = 0$ and M_0) to iteratively reach a solution where $\lambda = 1$ and $M^{[v]} = 0$. This is done by maximizing $\lambda \in [\,0 \;\; 1\,]$ with a constraint on the norm of $M^{[v]}$ of the type $(1 - \lambda)I \succ M^TM$ and as formalized next.

Phase 2. ITERATION

1: **Input**: Start from the initial guess $S_{1,0}$ and $\hat{K}_0$ provided by Phase 1 (initialization phase). Initialize $k = 0$.

2: **Step k, 1**: Let $k := k + 1$. For a fixed $\hat{K}_{k-1}$, $S_{1,k-1}$ is coming from the previous step, maximize λ under the following LMI conditions for $v = 1 \ldots \bar{v}$:

$$X^{[v]} \succ 0, \qquad \begin{bmatrix} (1-\lambda)I & M^{[v]T} \\ M^{[v]} & I \end{bmatrix} \succeq 0, \qquad \lambda \geq 0,$$

$$\begin{bmatrix} 0 & 0 & X^{[v]} \\ 0 & 0 & 0 \\ X^{[v]} & 0 & 0 \end{bmatrix}$$
$$\prec \left\{ \begin{bmatrix} I \\ -\left(\lambda \begin{bmatrix} C(q^{[v]}) \\ 0_{p-n,n} \end{bmatrix} + M^{[v]}\right) \\ -A(q^{[v]}) \end{bmatrix} S_{1,k-1} + \begin{bmatrix} 0 \\ -I \\ B(q^{[v]})\hat{K}_{k-1} \end{bmatrix} \begin{bmatrix} 0 \; -S_2 \;\; Y^T \end{bmatrix} \right\}^{\mathscr{S}}$$

at the optimum set $\lambda_k = \lambda$, $M_k^{[v]} = M^{[v]}$ and $H_k^T = S_2^{-1}Y^T$.

If $1-\lambda_k$ is smaller than a (small) tolerance, then $F_k = \hat{K}_{k-1}\begin{bmatrix} I_p \\ 0_{n-p,p}\end{bmatrix}$ and $L_k = H_k\begin{bmatrix} I_p \\ 0_{n-p,p}\end{bmatrix}$ are reasonable candidates (OF) and (OI) robustly stabilizing gains, respectively. Therefore, transfer selection H_k to the validation phase. Otherwise, go to Step k, 2.

3: **Step k, 2**: For fixed λ_k, $M_k^{[v]}$ and H_k coming from the previous step, search by bisection the smallest $\alpha \in [0\ 1]$ such that the following LMIs hold for $v = 1 \ldots \bar{v}$:

$$X^{[v]} \succ 0,$$

$$\begin{bmatrix} 0 & 0 & X^{[v]} \\ 0 & 0 & 0 \\ X^{[v]} & 0 & 0 \end{bmatrix} \prec \left\{ \begin{bmatrix} I \\ -\hat{M}^{[v]}(\alpha) \\ -A(q^{[v]}) \end{bmatrix} S_1 + \begin{bmatrix} 0 \\ S_2 \\ B(q^{[v]})Z \end{bmatrix} \begin{bmatrix} 0 & I & -H_k^T \end{bmatrix} \right\}^{\mathscr{S}}$$

where $\hat{M}^{[v]}(\alpha) = \left((1+\alpha(\lambda_k - 1)) \begin{bmatrix} C(q^{[v]}) \\ 0_{p-n,n} \end{bmatrix} + \alpha M_k^{[v]} \right)$. At the optimum set $\alpha_k = \alpha$, $\hat{K}_k = -ZS_2^{-1}$ and $S_{1,k} = S_1$.

If α_k is smaller than a (small) tolerance, then $F_k = \hat{K}_k\begin{bmatrix} I_p \\ 0_{n-p,p}\end{bmatrix}$ and $L_k = H_k\begin{bmatrix} I_p \\ 0_{n-p,p}\end{bmatrix}$ are reasonable candidates (OF) and (OI) robustly stabilizing gains, respectively. Therefore, transfer selection H_k to the validation phase. Otherwise, go to Step $k+1$, 1.

The key feature enjoyed by the two steps of the procedure listed in Phase 2 above, is that whenever moving from one step to the next one, the quality of the optimized solution (in terms of size of $1-\lambda$ or α) cannot get worse. This is clarified in the next proposition, whose proof is straightforward.

Proposition 2 *For the iterations listed in Phase 2, the following holds:*

- *Given any initial solution provided by Phase 1, the conditions at Step 1, 1 are feasible for $\lambda = 0$;*
- *Given any solution from Step k, 1, Step k, 2 is feasible for $\alpha = 1$;*
- *Given any solution from Step k, 2, Step k+1,1 is feasible for $\lambda = \lambda_k + (1-\lambda_k)(1-\alpha_k)$.*

3.2.3 Validation Phase

This heuristic algorithm is completed by a validation phase, which comprises the solution of an LMI, parameterized by matrix H, selected according to the iteration phase of the previous section.

Phase 3. VALIDATION (PROVIDES SOLUTIONS TO (OI) AND (OF))

1: **Input**: Start from matrix H, produced as an output of Phase 2 (iteration).

2: **Validation Step**: Solve the following LMI feasibility problem for $v = 1 \ldots \bar{v}$:

$$X^{[v]} \succ 0,$$

$$\begin{bmatrix} 0 & 0 & X^{[v]} \\ 0 & 0 & 0 \\ X^{[v]} & 0 & 0 \end{bmatrix} \prec \left\{ \begin{bmatrix} I \\ -\begin{bmatrix} C(q^{[v]}) \\ 0_{p-n,n} \end{bmatrix} \\ -A(q^{[v]}) \end{bmatrix} S_1 + \begin{bmatrix} 0 \\ S_2 \\ B(q^{[v]})Z \end{bmatrix} \begin{bmatrix} 0 & I & -H^T \end{bmatrix} \right\}^{\mathscr{S}}. \qquad (14)$$

If a solution is found, then from Corollary 1 $F = -ZS_2^{-1}\begin{bmatrix} I_p \\ 0_{n-p,p} \end{bmatrix}$ and $L = H\begin{bmatrix} I_p \\ 0_{n-p,p} \end{bmatrix}$ are, respectively, proved to be robustly stabilizing (OF) and (OI) gains. Otherwise, the validation phase fails and the algorithm should go back to the iteration phase reducing the tolerance for α and λ.

Note that, the LMI conditions in the Validation Step are guaranteed to be feasible whenever matrix H coming from the iteration step was associated with $\lambda = 1$ (equivalently, from Proposition 2, $\alpha = 0$). However, in general, one may find it convenient to run the Validation Step even for cases where these conditions are not exactly met. Due to this fact, and possibly due to numerical errors, it makes sense to possibly come back to the iteration phase (from a failing validation phase) to further improve, via extra iterations, the previous candidate selection of H.

4 Probabilistic Static Output Feedback Design

As emphasized in the previous section, using a polytopic approach to address the design of suitable matrices guaranteeing the conditions in Theorem 1 may be too conservative for the problem at hand. For this reason, in this section we follow an alternative paradigm based on a probabilistic approach, which allows for uncertain dynamics more general than (7) (thereby not requiring convexity with respect to the uncertainty q), enables using multipliers that are not necessarily common among the samples, reaches beyond the use of polytopic Lyapunov certificates, but comes at the expense of providing a probabilistic guarantee of robust stability (rather than a deterministic one), in addition to typically being computationally more expensive.

In particular, throughout this section, we do not assume that the uncertainty lies in a convex polytope, but we consider a more general setup, in which the state matrices in (1) may be generic continuous (possibly nonlinear) functions of the uncertainty parameter q. On the other hand, following a classical probabilistic approach [4, 20], we require to have additional probabilistic information on the uncertainty. Formally, we assume that $A(q)$, $B(q)$, $C(q)$ are continuous measurable functions of q, and that q is a random variable with probability distribution Pr with support $\mathbb{Q}$. Such a probability distribution may describe the likelihood of each occurrence of the uncertainty, or a user-defined weight for all possible uncertain situations.

Then, randomized algorithms are applied to design a controller that guarantees performance with a prescribed level of probability. These algorithms are based on the extraction of random samples of the uncertainty

$$q^{(1)} \dots q^{(\bar{r})} \in \mathbb{Q},$$

and the construction of sampled convex programs. The focus of this approach is in the derivation of sample complexity bounds, i.e., bounds on the number of samples to be extracted so as to ensure that the desired probabilistic guarantees are met.

In the next section, we briefly recall the so-called scenario approach originally presented in [3] for dealing with convex optimization problems in the presence of uncertainty.

4.1 The Scenario Approach

Let us consider a generic class of robust convex optimization problems of the form:

$$\begin{aligned}\theta_{\text{RO}} = \arg \ & \min_{\theta \in \Theta} c^T \theta \\ & \text{s.t. } f(\theta, q) \leq 0, \ \forall q \in \mathbb{Q},\end{aligned} \tag{15}$$

where $\theta \in \Theta$ denotes the design variable, bounded in a domain Θ, which is a convex and compact set in $\mathbb{R}^{n_\theta}$, and q is the uncertainty, bounded in the uncertainty set $\mathbb{Q}$, not necessarily compact. For a given $q \in \mathbb{Q}$, $f(\theta, q)$ is a convex function of the optimization variable. Furthermore, we assume that $f(\theta, q)$ is a continuous (possibly nonlinear) function of q for any given θ.

To construct a sampled convex program, N independent identically distributed (iid) samples are extracted according to the probability distribution of q, and the following scenario optimization problem, based on $\bar{r}$ instances (scenarios) of the uncertain constraints:

$$\begin{aligned}\theta_{\text{SO}} = \arg & \min_{\theta \in \Theta} c^T \theta \\ & \text{s.t. } f(\theta, q^{(r)}) \leq 0, \ r = 1 \ldots \bar{r}.\end{aligned} \tag{16}$$

Problem (16) represents a sampled relaxation of Problem (15), since it deals only with a subset of the (infinite number of) constraints considered in (15), according to the probability distribution of the uncertainty. However, under rather mild assumptions on Problem (15), by suitably choosing $\bar{r}$, this approximation may in practice become negligible in some probabilistic sense. Specifically, $\bar{r}$ can be selected depending on the level of "risk" of constraint violation that the user is willing to accept. To this end, we define the *violation probability* of a design θ as follows:

$$\text{Viol}(\theta) \doteq \Pr\{q \in \mathbb{Q} \ : \ f(\theta, q) > 0\} \tag{17}$$

The following result has been proven in [6].

Proposition 3 ([6]) *Assume that, for any multisample extraction, Problem (16) is feasible and attains a unique optimal solution. Then, given an accuracy level* $\varepsilon \in (0, 1)$*, the solution* θ_{SO} *of Problem* (16) *satisfies*

$$\Pr\{\text{Viol}(\theta_{\text{SO}}) > \varepsilon\} \leq \text{B}(\bar{r}, \varepsilon, n_\theta), \tag{18}$$

where

$$\text{B}(\bar{r}, \varepsilon, n_\theta) \doteq \sum_{k=0}^{n_\theta - 1} \binom{\bar{r}}{k} \varepsilon^k (1 - \varepsilon)^{(\bar{r} - k)}. \tag{19}$$

We note that nonuniqueness of the optimal solution can be circumvented by imposing additional “tie-break” rules in the problem, see, e.g., Appendix A of [3]. Also, in [5] it is shown that the feasibility assumption can be removed at the expense of substituting $n_\theta - 1$ with n_θ in $\mathrm{B}(\bar{r}, \varepsilon, n_\theta)$.

From (18), explicit bounds on the number of samples necessary to guarantee the “goodness” of the solution have been derived. The bound provided in [1] shows that, if, for given $\varepsilon, \delta \in (0, 1)$, the sample complexity $\bar{r}$ is chosen to satisfy the sample complexity bound

$$\bar{r} \geq \frac{\mathrm{e}}{\varepsilon(\mathrm{e}-1)}\left(\ln\frac{1}{\delta} + n_\theta - 1\right) \tag{20}$$

(where e denotes the Euler number), then the solution θ_{SO} of Problem (16) satisfies $\mathrm{Viol}(\theta_{\mathrm{SO}}) \leq \varepsilon$ with probability $1 - \delta$. This bound improves by a constant factor upon previous bounds, see e.g., [5], and it shows that Problem (16) exhibits linear dependence in $1/\varepsilon$ and n_θ, and logarithmic dependence on $1/\delta$. Note however that, from a practical viewpoint, it is always preferable to numerically solve the one-dimensional problem of finding the smallest integer $\bar{r}$ such that $\mathrm{B}(\bar{r}, \varepsilon, n_\theta) \leq \delta$.

4.2 Scenario with Certificates

The classical scenario approach previously discussed deals with uncertain optimization problems where all variables θ are to be designed. On the other hand, in the design with certificates approach we distinguish between *design variables* θ and *certificates* ξ. The certificates are represented here in green color, and correspond to those variables which are not involved in the construction of the design, but whose existence is necessary for its derivation. A classical example of certificates are Lyapunov functions for proving stability.

Formally, we consider a function $f(\theta, \xi, q)$, *jointly convex* in $\theta \in \Theta$ and $\xi \in \Xi \subseteq \mathbb{R}^{n_\xi}$ for given $q \in \mathbb{Q}$, and study the following robust optimization problem with certificates:

$$\begin{aligned} \theta_{\mathrm{RwC}} = \arg\min_{\theta} c^T\theta \\ \text{s.t. } \theta \in \mathscr{S}(q),\ \forall q \in \mathbb{Q}, \end{aligned} \tag{21}$$

where the set $\mathscr{S}(q)$ is defined as

$$\mathscr{S}(q) \doteq \{\theta \in \Theta \,|\, \exists \xi = \xi(q) \in \Xi \text{ satisfying } f(\theta, \xi, q) \leq 0\}\,.$$

From the above formulation, the role of certificates is clear: for any value of the uncertainty, the existence of a certificate (possibly depending on the given value of q) is required.

A key observation is that the set $\mathscr{S}(q)$ is convex in θ for any given q, see [12, Theorem 1]. These observations lead to the introduction of the following *scenario with certificates* problem, based again on a sample extraction, to approximate Problem (21), inspired by a similar approach proposed in [17] for the iterative solution of parameter-dependent LMIs:

$$\begin{aligned} \theta_{\mathrm{SwC}} = \arg \min_{\theta, \xi^{(1)}, \ldots, \xi^{(\bar{r})}} & \; c^T \theta \\ \text{s.t. } & f(\theta, \xi^{(r)}, q^{(r)}) \leq 0, \;\; r = 1 \ldots \bar{r}. \end{aligned} \tag{22}$$

Note that, contrary to Problem (16), in this case, a new certificate variable $\xi^{(r)}$ is created for every sample $q^{(r)}$, $r = 1, \ldots, \bar{r}$. To analyze the properties of the solution θ_{SwC}, we note that, in the case of SwC, the violation probability of design θ are given by

$$\mathrm{Viol}(\theta) = \Pr\Big\{ q \in \mathbb{Q} \mid \nexists \xi \in \Xi \text{ satisfying } f(\theta, \xi, q) \leq 0 \Big\}.$$

Then, the following theorem can be stated, from [12, Thm 1].

Theorem 2 ([12]) *Assume that, for a multisample extraction, Problem* (22) *is feasible and attains a unique optimal solution. Then, given an accuracy level* $\varepsilon \in (0, 1)$, *the solution* θ_{SwC} *of Problem* (22) *satisfies*

$$\Pr\left\{ \mathrm{Viol}(\theta_{\mathrm{SwC}}) > \varepsilon \right\} \leq \mathrm{B}(\bar{r}, \varepsilon, n_\theta). \tag{23}$$

We remark that Problem (22) has $\bar{r}$ separate constraints, one for each $q^{(r)}$, and each constraint involves a different certificate. However, notice that the dimension n_ξ of the certificates ξ does not enter into the right-hand side of the probability bound (23) in Theorem 2. Hence, the sample complexity of Problem (22) is smaller than that of the scenario counterpart of the problem with common certificates, in which both θ and ξ play the role of design variables. On the other hand, the complexity of solving Problem (22) is higher, since the number of optimization variables significantly increases, because a different variable $\xi^{(r)}$ is introduced for every sample $q^{(r)}$. This increase in complexity is not surprising, being Problem (22) much more difficult than the robust problem involving common certificates.

4.3 Probabilistic Robust Stability

In this section, we exploit the SwC setting previously discussed to derive a sample-based heuristic for designing a SOF controller guaranteeing robust stability in probability.

To this end, we revisit the heuristic approach presented in Sect. 3.2, and observe that both the initialization and the validation phases involve the solution of uncertain

LMI problems, where the values of λ, M and H are fixed ($\lambda = 0$, $M = M_0$, $H = H_0$ in Phase 1 and $\lambda = 1$, $M = 0$, $H = H$ in Phase 3). In that case, the necessity of having a convex formulation with respect to q forced us to impose the S-variable $S_1(q)$ to be fixed and independent of q. This limitation can be lifted in the sample-based approach, due to the general result of Theorem 2.

In the following corollary, which is a direct application of Theorems 1 and 2, we show how a sample-based approximation of Problem (2) with fixed values of λ, M, and H, can be derived, together with a precise characterization of its probabilistic properties.

Corollary 2 *Given $\varepsilon, \delta \in (0, 1)$, extract $\bar{r}$ iid samples $q^{(1)} \dots q^{(\bar{r})}$ of the uncertainty $q \in \mathbb{Q}$, where $\bar{r}$ satisfies*

$$\bar{r} \geq \frac{\mathrm{e}}{\varepsilon(\mathrm{e}-1)} \left(\ln \frac{1}{\delta} + n(n+m) - 1 \right). \tag{24}$$

Consider a selection of $\lambda \in [0, 1]$, M and H. If there exist matrices S_2, Z, and certificates $X^{(r)} \succ 0$, $S_1^{(r)}$, satisfying for $r = 1 \dots \bar{r}$,

$$\begin{bmatrix} 0 & 0 & X^{(r)} \\ 0 & 0 & 0 \\ X^{(r)} & 0 & 0 \end{bmatrix} \prec \left\{ \begin{bmatrix} I \\ -\left(\lambda \begin{bmatrix} C(q^{(r)}) \\ 0_{p-n,n} \end{bmatrix} + M \right) \\ -A(q^{(r)}) \end{bmatrix} S_1^{(r)} + \begin{bmatrix} 0 \\ S_2 \\ B(q^{(r)}) Z \end{bmatrix} \begin{bmatrix} 0 & I & -H^T \end{bmatrix} \right\}^{\mathscr{S}}, \tag{25}$$

then, we guarantee with confidence at least $1-\delta$, a probability of at least $1-\varepsilon$ that

- **(OF)** *if $\lambda = 1$, $M = 0$ and S_2 is invertible, then selection $F = -Z S_2^{-1} \begin{bmatrix} I_p \\ 0_{n-p,p} \end{bmatrix}$ solves the (OF) goal;*
- **(OI)** *if $\lambda = 1$ and $M = 0$, then selection $L = H \begin{bmatrix} I_p \\ 0_{n-p,p} \end{bmatrix}$ solves the (OI) goal;*
- **(SF)** *if $\lambda = 0$ and S_2 is invertible, then $K = -Z S_2^{-1} M$ solves the (SF) goal;*
- **(SI)** *if $\lambda = 0$, then $J = HM$ solves the (SI) goal.*

4.4 Iterative Heuristic for Probabilistic Robust Control

In the sequel, we show how the application of Corollary 2 allows deriving a sample-based version of the heuristic introduced in Sect. 3.2 for the deterministic case. In particular, the proposed approach involves again three phases:

1. An initialization phase, in which a sample-based SwC problem is solved, leading to the construction of initial candidate variables to be passed to the iteration phase. In particular, this first phase returns, as a side result, the design of probabilistic solutions to the robust state feedback (SF) and state injection (SI) goals. More specifically, we are able to assess precise probabilistic properties of these solutions, in terms of the measure of the uncertainty set that it may fail to stabilize.
2. An iteration phase, in which a small subset $\bar{r}_\mathrm{d} \leq \bar{r}$ of the samples employed in the first phase is randomly selected (for instance, the first $\bar{r}_d$ ones), and is used in an iterative way to "push" the design obtained in the first phase (wherein $\lambda = 0$) toward a SOF design (wherein we need $\lambda = 1$).
3. A validation phase where, based on matrix H of the previous phase, a sample-based SwC problem is solved for the design of probabilistic solutions to the robust output feedback (OF) and output injection (OI) goals.

It should be noted that if Phase 3 fails, one may investigate more accurate selections of matrix H by repeating phase 2 with a larger number $\bar{r}_\mathrm{d}$ of samples from Phase 1.

4.4.1 Sample-Based Initialization Phase

The initialization phase represents, substantially, the sample-based equivalent of Phase 1 presented in Sect. 3.2.1.

Phase 1. SAMPLE- BASED INITIALIZATION (PROVIDES PROBABILISTIC SOLUTIONS TO (SI) AND (SF))

1: **Input**: Select the initial values as per (10)–(12) with $h = 1$.

2: **Sample generation**: Given probabilistic levels $\delta, \varepsilon \in [0, 1]$, set $\bar{r}$ as per (24), and generate $\bar{r}$ iid samples $q^{(1)} \dots q^{(\bar{r})}$ according to distribution Pr.

3: **Iteration**: Solve the following sampled feasibility problem for $r = 1 \dots \bar{r}$:

$$X^{(r)} \succ 0,$$

$$\begin{bmatrix} 0 & 0 & X^{(r)} \\ 0 & 0 & 0 \\ X^{(r)} & 0 & 0 \end{bmatrix} \prec \left\{ \begin{bmatrix} I \\ -M_0 \\ -A(q^{(r)}) \end{bmatrix} S_1^{(r)} + \begin{bmatrix} 0 \\ S_2 \\ B(q^{(r)})Z \end{bmatrix} \begin{bmatrix} 0 & I & -H_0^T \end{bmatrix} \right\}^{\mathscr{S}} \qquad (26)$$

If (26) is feasible, go to the next step. Otherwise, increase h, redefine H_0 according to (10), (24) and repeat step 2. If for larger values of h no solution exists, then stop: the iterative heuristic fails.

4: **Output**: If a feasible solution to (26) is found, then output $\hat{K}_0 = -ZS_2^{-1}$, $K = -ZS_2^{-1}M_0$, $J = H_0 M_0$, together with samples $q^{(1)} \dots q^{(\bar{r})}$ and the corresponding S-variables $S_1^{(1)} \dots S_1^{(\bar{r})}$, S_2, Z.

Also in this case, there is no guarantee, even in a probabilistic sense, that this step of the algorithm will return a feasible solution. However, if a solution is returned, then by Corollary 2 K and J defined in Step 4 are guaranteed to solve in a probabilistic way the robust state feedback (SF) and state injection (SI) goals, respectively. Moreover, its output constitutes the initialization step of the iteration phase presented next.

4.4.2 Sample-Based Iteration Phase

The objective of this phase is to iteratively “push” the initial solution to (25) provided by Phase 1 (with $\lambda_0 = 0$, $M = M_0$) toward a solution to (25) with $\lambda = 1$ and $M = 0$. This phase represents a completely heuristic procedure, which, if successful, returns a parameter H for the next validation phase, which is instead based on the rigorous results of Corollary 2.

This phase is the one that is computationally most expensive. To alleviate the computational load, a subset of $\bar{r}_\mathrm{d} \le \bar{r}$ *design samples* is selected among the samples returned by Phase 1.

Phase 2. Sample- based iteration

1: **Input**: Start from the initial guess $\hat{K}_0$ provided by Phase 1 (initialization phase). Initialize $k = 0$

2: **Design samples selection**: Select a (small) number $\bar{r}_\mathrm{d} \le \bar{r}$ of samples $q^{(r)}$ and the corresponding S-variables $S_{1,0}^{(r)} := S_1^{(r)}$, $r = 1 \dots \bar{r}_\mathrm{d}$ returned by Phase 1.

3: **Step k, 1**: Let $k := k + 1$. For fixed $\hat{K}_{k-1}$, $S_{1,k-1}^{(r)}$ coming from the previous step, maximize λ under the following conditions for $r = 1 \dots \bar{r}_\mathrm{d}$

$$X^{(r)} \succ 0, \qquad \begin{bmatrix} (1-\lambda)I & {M^{(r)}}^T \\ M^{(r)} & I \end{bmatrix} \succeq 0, \qquad \lambda \ge 0,$$

$$\begin{bmatrix} 0 & 0 & X^{(i)} \\ 0 & 0 & 0 \\ X^{(r)} & 0 & 0 \end{bmatrix} \prec \left\{ \begin{bmatrix} I \\ -\left(\lambda \begin{bmatrix} C(q^{(r)}) \\ 0_{p-n,n} \end{bmatrix} + M^{(r)} \right) \\ -A(q^{(r)}) \end{bmatrix} S_{1,k-1}^{(r)} + \begin{bmatrix} 0 \\ -I \\ B(q^{(r)})\hat{K}_{k-1} \end{bmatrix} \begin{bmatrix} 0 & -S_2 & Y^T \end{bmatrix} \right\}^{\mathscr{S}}$$

at the optimum set $\lambda_k = \lambda$, $M_k^{(r)} = M^{(r)}$ and $H_k^T = S_2^{-1} Y^T$.

If $1 - \lambda_k$ is smaller than a (small) tolerance level, then $F_k = \hat{K}_{k-1} \begin{bmatrix} I_p \\ 0_{n-p,p} \end{bmatrix}$ and $L_k = H_k \begin{bmatrix} I_p \\ 0_{n-p,p} \end{bmatrix}$ are reasonable candidates (OF) and (OI) robustly stabilizing gains, respectively. Therefore, transfer selection H_k to the validation phase. Otherwise, go to Step k, 2.

4: **Step k, 2**: For fixed λ_k, $M_k^{(r)}$ and H_k coming from the previous step, search by bisection the smallest $\alpha \in [0\ 1]$ such that the following inequalities hold for $r = 1 \dots \bar{r}_\mathrm{d}$

$$X^{(r)} \succ 0,$$

$$\begin{bmatrix} 0 & 0 & X^{(r)} \\ 0 & 0 & 0 \\ X^{(r)} & 0 & 0 \end{bmatrix} \prec \left\{ \begin{bmatrix} I \\ -\hat{M}(q^{(r)}, \alpha) \\ -A(q^{(r)}) \end{bmatrix} S_1^{(r)} + \begin{bmatrix} 0 \\ S_2 \\ B(q^{(r)})Z \end{bmatrix} \begin{bmatrix} 0 & I & -H_k^T \end{bmatrix} \right\}^{\mathscr{S}}$$

where $\hat{M}(q^{(r)}, \alpha) = \left((1 + \alpha(\lambda_k - 1)) \begin{bmatrix} C(q^{(r)}) \\ 0_{p-n,n} \end{bmatrix} + \alpha M_k^{(r)} \right)$. At the optimum set $\alpha_k = \alpha$, $\hat{K}_k = -Z S_2^{-1}$ and $S_{1,k}^{(r)} = S_1^{(r)}$.

If α_k is smaller than a (small) tolerance level, then $F_k = \hat{K}_k \begin{bmatrix} I_p \\ 0_{n-p,p} \end{bmatrix}$ and $L_k = H_k \begin{bmatrix} I_p \\ 0_{n-p,p} \end{bmatrix}$ are reasonable candidates OF and OI robustly stabilizing gains, respectively. Therefore, transfer selection H_k to the validation phase. Otherwise, go to Step $k+1$, 1.

Similar to its deterministic counterpart, the iterations in Phase 2 guarantee that the quality of the optimized solution does not get worse. In particular, the results of Proposition 2 still hold.

4.4.3 Sample-Based Validation Phase

The validation phase uses matrix H returned by the iterations in Phase 2 to construct a scenario with certificates problem.

Phase 3. SAMPLE- BASED VALIDATION (PROVIDES PROBABILISTIC SOLUTIONS TO (OI) AND (OF))

1: **Input**: Start from matrix H, produced as an output of Phase 2 (iteration).

2: **Sample generation**: Given probabilistic levels $\delta, \varepsilon \in [0, 1]$, set $\bar{r}$ as per (2), and generate $\bar{r}$ iid samples $q^{(1)} \dots q^{(\bar{r})}$ according to distribution Pr.

3: **Validation Step**: Solve the following sampled problem for $r = 1 \dots \bar{r}$:

$$X^{(r)} \succ 0,$$
$$\begin{bmatrix} 0 & 0 & X^{(r)} \\ 0 & 0 & 0 \\ X^{(r)} & 0 & 0 \end{bmatrix} \prec \left\{ \begin{bmatrix} I \\ -\begin{bmatrix} C(q^{(r)}) \\ 0_{p-n,n} \end{bmatrix} \\ -A(q^{(r)}) \end{bmatrix} S_1^{(r)} + \begin{bmatrix} 0 \\ S_2 \\ B(q^{(r)})Z \end{bmatrix} \begin{bmatrix} 0 & I & -H^T \end{bmatrix} \right\}^{\mathscr{S}} \tag{27}$$

If a solution is found, then from Corollary 2 selections, $F = -ZS_2^{-1} \begin{bmatrix} I_p \\ 0_{n-p,p} \end{bmatrix}$ and $L = H \begin{bmatrix} I_p \\ 0_{n-p,p} \end{bmatrix}$ are probabilistic solutions to the output feedback (OF) and the output injection (OI) goals, respectively. Otherwise, the validation phase fails and the algorithm should go back to the iteration phase, increasing the number of selected samples by choosing a larger number $\bar{r}_{\mathrm{d}}$.

Note that, the sample-based Validation Step is by nature less conservative than the corresponding deterministic one (14) for two main reasons: (i) it does not require the solution to be feasible for all possible values of the uncertainty, but it requires feasibility only for a suitably selected number of samples, (ii) it does not require a common S-variable S_1, but it allows for parameter-dependent certificates. This is done at the expense of giving up deterministic robustness, but instead allowing for a (typically small) probability of failure.

However, if one is indeed interested in robustly guaranteed results, it should be pointed out that nothing prevents us from testing the output of the probabilistic Phase 2 by means to the corresponding deterministic Validation Step (14).

5 Numerical Examples

5.1 OF Design Without Uncertainties

Although the results are intended for robust stability, the heuristic algorithm can also be applied to systems without uncertainties. In this case, there is only one sample and the deterministic and probabilistic algorithms coincide. The CompLeib library provides a collection of such systems. We have tested the heuristic on some of these examples (of low order). The results are as follows ($h = 1$ in all cases). The algorithm finds a stabilizing output feedback gain

- at iteration $k = 1$ for examples, AC1–AC5, AC12, AC15–AC17, HE2
- at iteration $k = 2$ for examples, AC6, AC7, AC9, HE4
- at iteration $k = 4$ for example, AC8
- at iteration $k = 7$ for example, HE3
- does not converge for AC11, AC18, HE1, and HE5.

These results are quite encouraging because some of these examples were proved to be hard when tested with similar tools in [9].

5.2 Deterministic OF Design with Uncertainties

The next example is borrowed from [8] with slight modifications to ensure that all system matrices A, B, and C are uncertain. These uncertain matrices belong to a polytope with two vertices:

$$\begin{aligned} A(q^{[1]}) &= \begin{bmatrix} -1 & 4 & 0 \\ 0 & 0 & 1 \\ \underline{a} & 6 & -1 \end{bmatrix}, \quad B(q^{[1]}) = \begin{bmatrix} 0 \\ 0 \\ 0.5 \end{bmatrix}, \; C(q^{[1]}) = \begin{bmatrix} 1 & 1 & 0 \\ 0 & 1 & 0 \end{bmatrix}, \\ A(q^{[2]}) &= \begin{bmatrix} -1 & 1 & 0 \\ 0 & -5 & 1 \\ \overline{a} & 1 & -1 \end{bmatrix}, \; B(q^{[2]}) = \begin{bmatrix} 0 \\ 0 \\ 2 \end{bmatrix}, \quad C(q^{[2]}) = \begin{bmatrix} 1 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix}. \end{aligned} \tag{28}$$

The peculiarity of this numerical example is that the uncertain input matrices $C(q)$ are of full row rank except at one of the vertex of the polytope. This rank deficiency corresponds to a failure of one of the sensors of the system. This uncertain system may be robustly stabilized via static output feedback for different ranges of the parameter $a \in [\underline{a}, \overline{a}]$ as indicated in Table 1. For each integer value of $\underline{a} \in [0, 10]$, we search for the maximal integer $\overline{a}$ such that the algorithm finds a solution to the (OF) goal. K_0 is the state feedback (SF) gain found at the initialization Phase 1. $\bar{k}$ is the number of iterations in Phase 2 of the algorithm. The maximal number of iterations was set to 10, therefore $\bar{k} = 10$ indicates that Phase 2 terminated because this maximal

Table 1 Robust stabilizing SOF gains for numerical example (28)

$[\underline{a}, \overline{a}]$	K_0	$\bar{k}$	$F_{\bar{k}}$	Time (s)
[0, 9]	$\begin{bmatrix} -1.5939 & -17.5869 & -7.1516 \end{bmatrix}$	2	$\begin{bmatrix} -1.3792 & -27.4388 \end{bmatrix}$	6.5
[1, 14]	$\begin{bmatrix} -3.0824 & -20.0953 & -7.4500 \end{bmatrix}$	2	$\begin{bmatrix} -2.6507 & -29.2585 \end{bmatrix}$	7.4
[2, 23]	$\begin{bmatrix} -5.1340 & -24.8380 & -8.4098 \end{bmatrix}$	2	$\begin{bmatrix} -4.9091 & -37.1852 \end{bmatrix}$	7.4
[3, 29]	$\begin{bmatrix} -7.1790 & -32.6351 & -10.1779 \end{bmatrix}$	7	$\begin{bmatrix} -6.8056 & -48.0987 \end{bmatrix}$	33.8
[4, 49]	$\begin{bmatrix} -16.1234 & -48.9351 & -13.5936 \end{bmatrix}$	10	$\begin{bmatrix} -13.0042 & -52.9144 \end{bmatrix}$	51.3
[5, 58]	$\begin{bmatrix} -21.4924 & -57.7095 & -14.8185 \end{bmatrix}$	5	$\begin{bmatrix} -14.4984 & -47.2955 \end{bmatrix}$	28.1
[6, 72]	$\begin{bmatrix} -27.1872 & -59.7972 & -14.1779 \end{bmatrix}$	7	$\begin{bmatrix} -18.9969 & -45.5168 \end{bmatrix}$	45.6
[7, 77]	$\begin{bmatrix} -31.3509 & -62.1565 & -14.7895 \end{bmatrix}$	10	$\begin{bmatrix} -21.6073 & -42.3976 \end{bmatrix}$	45.6
[8, 80]	$\begin{bmatrix} -34.4863 & -63.0824 & -14.8762 \end{bmatrix}$	2	$\begin{bmatrix} -23.2575 & -38.5662 \end{bmatrix}$	7.5
[9, 83]	$\begin{bmatrix} -37.6496 & -63.8179 & -14.9218 \end{bmatrix}$	2	$\begin{bmatrix} -25.2797 & -35.1410 \end{bmatrix}$	8.0
[10, 86]	$\begin{bmatrix} -40.7372 & -64.4806 & -14.9106 \end{bmatrix}$	2	$\begin{bmatrix} -26.9999 & -31.4823 \end{bmatrix}$	6.8
[0, 1000]	$\begin{bmatrix} -1305.7 & -695.5 & -73.5 \end{bmatrix}$	10	Fail	59.3

number is reached. Otherwise, the iterations stop when $1 - \lambda < 10^{-7}$. The column $F_{\bar{k}}$ gives the value of the output feedback gain when the algorithm succeeds in finding a robustly stabilizing one. Results are given in Table 1. They outperform significantly those of [8]. Note that in many cases the number of iterations is very low (typically 2) and, hence, the computation time is not prohibitive. The last row of the table is a test of the method's capability to find robustly stabilizing state feedback gains. For this last test we have set $h = 10$ in the initialization Phase 1. In all other tests $h = 1$.

5.3 A Comparison Between the Deterministic and Probabilistic Approaches

The following example is taken from [7]. The system is given by

$$A(q) = \begin{bmatrix} 0 & -0.5 + q_1 \\ 0.5 + q_2 & 0 \end{bmatrix}, \quad B(q) = \begin{bmatrix} 0.5 + q_1 \\ 0.5 - q_2 \end{bmatrix}, \quad C = \begin{bmatrix} 1 & 0 \end{bmatrix}.$$

To compare the deterministic and the probabilistic approaches, we let q_1 and q_2 be defined as

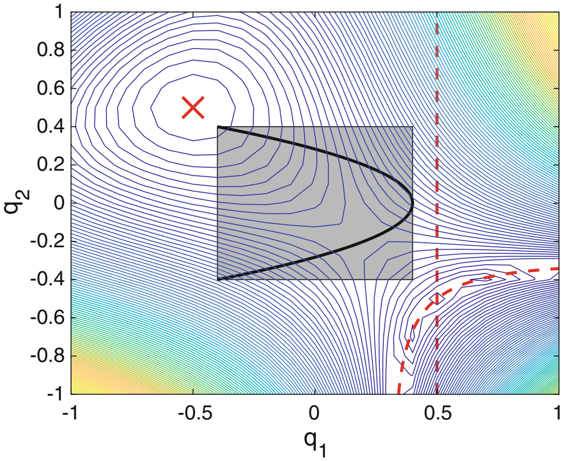


Fig. 1 Deterministic uncertainty set (gray-shaded box) and real uncertainty set (black solid line) for $\bar{q} = 0.4$, namely the limit value for a successful deterministic OF design. In the background, the contour plot of the absolute value of the determinant of the reachability matrix of the system in [7] is also illustrated. This is null corresponding to the red cross and along the red curve in the right bottom part of the figure. Also, a stabilizing SOF cannot be found if the system is unobservable, that is for all points on the dashed line $q_1 = 0.5$

$$q_1 = \bar{q}\cos(2\theta), \quad q_2 = \bar{q}\sin(\theta), \quad \theta \in \left[-\frac{\pi}{2}, \frac{\pi}{2}\right],$$

where $\bar{q}$ is a fixed parameter determining the upper bound on the absolute value of the uncertain parameters for any value of θ.

To perform a robust deterministic design, we need to assume two *independent* uncertainties $|q_1| \leq \bar{q}$ and $|q_2| \leq \bar{q}$, without explicitly considering their (nonlinear) dependence on the common parameter θ. With this assumption, the parameterization becomes convex and we can run the procedure in Sect. 3.2. Such an overparameterization of the uncertainty results in a larger uncertainty set: see the gray-shaded box in Fig. 1 as compared to the real set indicated by the black solid line. Clearly, the corresponding design is convex but more conservative.

For this simple example, one can compute by hand the robustly stabilizing OF gains and these are exactly such that $\frac{\bar{q}-0.5}{\bar{q}+0.5} < F < 0$. Moreover for $\bar{q} > 0.36$, there is no state feedback that may quadratically stabilize the system (stability may not be proved with a common to all uncertainties Lyapunov matrix). For $\bar{q} > 0.36$ only parameter-dependent Lyapunov certificates may be used to prove robust stability of the closed loop.

The heuristic algorithms are applied to the example for various values of $\bar{q}$. The results are given in Tables 2, 3 and 4. K_0 is the state feedback (SF) gain found at the initialization Phase 1. $\bar{k}$ is the number of iterations in Phase 2 of the algorithm. The maximal number of iterations is set to 10, therefore $\bar{k} = 10$ indicates that Phase 2 terminated because this maximal number is reached. Otherwise, the iterations

Table 2 Numerical results for the deterministic approach

$\bar{q}$	$\frac{\bar{q}-0.5}{\bar{q}+0.5}$	K_0	$\bar{k}$	$1-\lambda_{\bar{k}}$	$F_{\bar{k}}$	Validation	Time (s)
0.1	-0.6667	$\begin{bmatrix}-0.6589 & -0.9263\end{bmatrix}$	2	$7.5 \cdot 10^{-9}$	-0.5082	OK	5.3
0.2	-0.4286	$\begin{bmatrix}-0.4012 & -1.0939\end{bmatrix}$	3	$2.9 \cdot 10^{-9}$	-0.3166	OK	8.6
0.3	-0.2500	$\begin{bmatrix}-0.1776 & -1.0576\end{bmatrix}$	3	$1.4 \cdot 10^{-8}$	-0.1185	OK	8.8
0.4	-0.1111	$\begin{bmatrix}-0.0471 & -0.9240\end{bmatrix}$	7	$3.5 \cdot 10^{-8}$	-0.1104	OK	26.7
0.41	-0.0989	$\begin{bmatrix}-0.0427 & -0.9057\end{bmatrix}$	10	$8.4 \cdot 10^{-7}$	-0.1030	-	45.0
0.42	-0.0870	$\begin{bmatrix}-0.0387 & -0.8886\end{bmatrix}$	10	$2.7 \cdot 10^{-6}$	-0.0928	-	53.3
0.43	-0.0753	$\begin{bmatrix}-0.0345 & -0.8728\end{bmatrix}$	10	$7.7 \cdot 10^{-6}$	-0.0827	-	47.0
0.44	-0.0638	$\begin{bmatrix}-0.0319 & -0.8654\end{bmatrix}$	10	$1.5 \cdot 10^{-5}$	-0.0712	-	54.0
0.45	-0.0526	$\begin{bmatrix}-0.0337 & -0.8739\end{bmatrix}$	10	$3.8 \cdot 10^{-5}$	-0.0623	-	50.1
0.455	-0.0471	$\begin{bmatrix}-0.0353 & -0.8758\end{bmatrix}$	10	$8.3 \cdot 10^{-5}$	-0.0546	-	49.2
0.46	-0.0417	-					1.2

Table 3 Numerical results for the probabilistic method applied to the deterministic model

$\bar{q}$	$\frac{\bar{q}-0.5}{\bar{q}+0.5}$	K_0	$\bar{k}$	$1-\lambda_{\bar{k}}$	$F_{\bar{k}}$	Validation	Time (s)
0.1	-0.6667	$\begin{bmatrix}-0.6763 & -0.8962\end{bmatrix}$	2	$9.4 \cdot 10^{-9}$	-0.5409	OK	4.5
0.2	-0.4286	$\begin{bmatrix}-0.4403 & -0.9932\end{bmatrix}$	3	$3.4 \cdot 10^{-9}$	-0.3657	OK	8.4
0.3	-0.2500	$\begin{bmatrix}-0.2174 & -0.9742\end{bmatrix}$	4	$1.1 \cdot 10^{-8}$	-0.2371	OK	11.7
0.4	-0.1111	$\begin{bmatrix}-0.0569 & -0.8682\end{bmatrix}$	4	$1.8 \cdot 10^{-9}$	-0.1012	OK	12.9
0.41	-0.0989	$\begin{bmatrix}-0.0522 & -0.8492\end{bmatrix}$	4	$1.2 \cdot 10^{-8}$	-0.0927	OK	17.0
0.42	-0.0870	$\begin{bmatrix}-0.0480 & -0.8297\end{bmatrix}$	3	$1.6 \cdot 10^{-8}$	-0.0792	OK	9.4
0.43	-0.0753	$\begin{bmatrix}-0.0443 & -0.8095\end{bmatrix}$	2	$2.9 \cdot 10^{-8}$	-0.0638	OK	5.1
0.44	-0.0638	$\begin{bmatrix}-0.0406 & -0.7889\end{bmatrix}$	5	$1.9 \cdot 10^{-9}$	-0.0554	OK	22.2
0.45	-0.0526	$\begin{bmatrix}-0.0392 & -0.7867\end{bmatrix}$	10	$5.6 \cdot 10^{-6}$	0.0950	-	50.8
0.455	-0.0471	$\begin{bmatrix}-0.0413 & -0.7968\end{bmatrix}$	5	$1.5 \cdot 10^{-8}$	-0.0337	OK	22.4
0.46	-0.0417	-					1.2

stop when $1-\lambda < 10^{-7}$. The column $1-\lambda_{\bar{k}}$ shows how close λ is to the value 1 when the iterations stop. The column $F_{\bar{k}}$ gives the value of the OF gain when the algorithm stops. The column named 'Validation' indicates whether the validation

Phase 3 is successful (OK) or not (-). The computation time is the total time including initialization and termination phases. For each tested value of $\bar{q}$, the theoretical limit of the stabilizing output feedback gains is recalled.

The algorithms are applied for three cases. The first one (results of Table 2) corresponds to the purely deterministic case described in Sect. 3. The last one (results of Table 4) corresponds to the purely probabilistic case described in Sect. 4. The probabilistic designs are done considering $\bar{r}_{\rm d} = 10$ samples for the iterations in Phase 2. The number of samples in Phases 1 and 3 is set to $\bar{r} = 450$. This value is computed from (20) with $\varepsilon = 0.05$ and $\delta = 10^{-4}$. The results of Table 3 correspond to an intermediate case where the probabilistic approach is applied to the deterministic model, that is when considering only the four vertices as samples. This case does not allow to conclude robustness (no deterministic nor probabilistic robustness can be deduced) but corresponds to simultaneous stabilization of the four vertices. The goal is to illustrate the degrees of freedom obtained when relaxing S_1 from being common to all vertices/samples.

Some conclusions about these results:

- The deterministic (OF) design is successful up to $\bar{q} = 0.4$ but when looking at the results for larger bounds on the uncertainties, it seems that the algorithm is close to converging to valid values. One reason for this non convergence is due to the heuristic nature of the algorithm. The other possible reasons is that there might not be any solution to the BMIs for $\bar{q} > 0.4$. The true bound on $\bar{q}$ for the existence of a stabilizing output feedback gain is $\bar{q} < 0.5$, but we cannot say that this bound can be approached by the proposed conservative BMI conditions.
- When relaxing the constraints on having common variables S_1, the (OF) goal is almost always attained whatever $\bar{q} \leq 0.455$. The fact that it fails for $\bar{q} = 0.45$ can be due to numerical errors at some stage of the iteration, or because the heuristic fails by going in an inappropriate direction. The improvements when comparing Tables 2 and 3 illustrate the potential reduction of conservatism that can be achieved by probabilistic methods.
- The approach allows achieving the robustly (SF) goal up to $\bar{q} = 0.455$. We know that such a goal cannot be achieved when imposing common Lyapunov matrices to all uncertainties. This illustrates the fact that the new LMIs of the initialization Phase 1 have quite some potential for the robust state feedback design problem that remains open.
- The probabilistic approach allows going further in terms of the (SF) goal. This is not surprising since, compared to the deterministic approach, there is some (small) tolerance on stability violation. Typically, during Phase 1 of the probabilistic approach, the extremal values of the uncertainties (which happen in this example to be the worst-case values) have low probability to be drawn. Phase 1 is hence applied considering a large scenario of samples ($N = 450$ in our case) but there might be no value close the critical samples $|q_1| = |q_2| = \bar{q}$. The relaxed (SF) goal is hence feasible (in probability).
- Since the approach is dependent on the samples that have been drawn, there is no possible monotonicity in the results. This is illustrated for the case where

Table 4 Numerical results for the randomized approach

$\bar{q}$	$\frac{\bar{q}-0.5}{\bar{q}+0.5}$	K_0	$\bar{k}$	$1-\lambda_{\bar{k}}$	$F_{\bar{k}}$	Validation	Time (s)
0.1	-0.6667	$\begin{bmatrix} -0.7028 & -0.8958 \end{bmatrix}$	2	$5.2 \cdot 10^{-9}$	-0.6125	OK	31.2
0.2	-0.4286	$\begin{bmatrix} -0.4885 & -1.0711 \end{bmatrix}$	3	$4.0 \cdot 10^{-9}$	-0.4047	OK	43.1
0.3	-0.2500	$\begin{bmatrix} -0.2597 & -1.1284 \end{bmatrix}$	4	$1.1 \cdot 10^{-8}$	-0.2480	OK	53.4
0.4	-0.1111	$\begin{bmatrix} -0.0855 & -1.0275 \end{bmatrix}$	6	$2.8 \cdot 10^{-8}$	-0.1296	-	79.0
0.4	-0.1111	$\begin{bmatrix} -0.0750 & -1.0250 \end{bmatrix}$	3	$2.7 \cdot 10^{-8}$	-0.1501	-	43.0
0.41	-0.0989	$\begin{bmatrix} -0.0619 & -1.0026 \end{bmatrix}$	4	$2.8 \cdot 10^{-8}$	-0.1275	-	55.2
0.42	-0.0870	$\begin{bmatrix} -0.0464 & -1.0069 \end{bmatrix}$	3	$5.0 \cdot 10^{-9}$	-0.0547	-	37.5
0.43	-0.0753	$\begin{bmatrix} -0.0365 & -0.9873 \end{bmatrix}$	2	$1.5 \cdot 10^{-8}$	-0.0942	-	35.0
0.44	-0.0638	$\begin{bmatrix} -0.0241 & -0.9828 \end{bmatrix}$	2	$2.7 \cdot 10^{-8}$	-0.0619	-	52.0
0.45	-0.0526	$\begin{bmatrix} -0.0109 & -0.9624 \end{bmatrix}$	3	$4.7 \cdot 10^{-8}$	-0.0293	-	42.25
0.455	-0.0471	$\begin{bmatrix} -0.0052 & -0.9607 \end{bmatrix}$	2	$1.2 \cdot 10^{-8}$	-0.0173	-	38.2
0.46	-0.0417	$\begin{bmatrix} -0.0045 & -0.9433 \end{bmatrix}$	2	$3.6 \cdot 10^{-8}$	-0.0174	-	39.1
0.465	-0.0363	$\begin{bmatrix} -0.0019 & -0.9434 \end{bmatrix}$	3	$2.1 \cdot 10^{-8}$	-0.0027	-	54.8
0.47	-0.0309	$\begin{bmatrix} 0.0006 & -0.9293 \end{bmatrix}$	-	num pb		-	88.6
0.48	-0.0204	$\begin{bmatrix} 0.0053 & -0.9237 \end{bmatrix}$	5	$6.4 \cdot 10^{-8}$	$8.98 \cdot 10^{-5}$	-	84.9
0.49	-0.0101	$\begin{bmatrix} 0.0010 & -0.8937 \end{bmatrix}$	-	num pb		-	54.4
0.495	-0.0050	$\begin{bmatrix} 0.0024 & -0.8724 \end{bmatrix}$	4	$1.2 \cdot 10^{-8}$	-0.0127	-	80.0
0.50	0	-					12.4
0.505	No	-					15.2

we applied the method for two different scenarios, and the results are inevitably different. For $\bar{q}$ close to 0.5 the problem becomes very constrained and for some cases (depending on the samples) we noticed numerical errors in the algorithm (the LMIs become unfeasible during the iterations although the sequence λ_k is proved to be theoretically decreasing monotonously).

- The iterations of Phase 2 are done on a subset of the scenario. When it converges, and this is quite often the case as illustrated by the value of $1 - \lambda$, the conclusion is that we have good candidates for stabilization of the few systems ($\bar{r}_\mathrm{d} = 10$) used during this phase. There is no guarantee of robustness, not even probabilistic. This is the reason why the termination Phase 3 usually fails (except for $\bar{q} \leq 0.4$). It fails even though the computed value of the (OF) gain actually does solve the problem. This result illustrates the fact that, even for the scenario situation, the BMIs are conservative. Conservatism comes from the fact that the S-variable S_2 is imposed to be the same for all samples.

The computations were done on a MacBook Pro 2.9 GHz Intel Core i5 with Matlab2016b. The LMIs were coded using YALMIP (release R20141030) by [15] and solved using SDPT3 (version 4.0) by [21].

6 Conclusions

In this chapter, we proposed a new robust static output feedback design method stemming from an S-variable description of the set of feasible solutions. The proposed approach leads to both deterministic and probabilistic designs, the first one providing worst-case guarantees (pessimistic approach) requiring polytopic uncertainty sets and specific multipliers structures, and the second one removing these assumptions at the cost of extra computational burden and probabilistic guarantees (optimistic approach). The derived conditions are coded as bilinear matrix inequalities for both cases so that a heuristic procedure is proposed for their solution. Interestingly, the heuristic approach starts from solving a robustly stabilizing state injection gain and a robust state feedback stabilizer, which is then refined into a robust output feedback stabilizer and a robustly stabilizing output injection gain. Numerical tests on some examples taken from the literature have shown good performance of the heuristic, first in finding nominally stabilizing output feedback gains, and then addressing robust stabilization problems.

Future works involve better characterizing the properties of the suggested heuristic algorithms, possibly providing more sophisticated solution methods for the proposed BMI problems, in addition to further characterizations of the merits of the proposed approach on relevant case studies.

References

1. T. Alamo, R. Tempo, A. Luque, and D.R. Ramirez. Randomized methods for design of uncertain systems: Sample complexity and sequential algorithms. *Automatica*, 52:160–172, 2015.
2. D. Arzelier, G. Deaconu, S. Gumussoy, and D. Henrion. H_2 for HIFOO. In *International Conference on Control and Optimization with Industrial Applications*, Ankara, Turkey, 2011.
3. G. Calafiore and M.C. Campi. The scenario approach to robust control design. *IEEE Transactions on Automatic Control*, 51(5):742–753, 2006.
4. G. Calafiore, F. Dabbene, and R. Tempo. Research on probabilistic methods for control system design. *Automatica*, 47:1279–1293, 2011.
5. G.C. Calafiore. Random convex programs. *SIAM Journal on Optimization*, 20(6):3427–3464, 2010.
6. M.C. Campi and S. Garatti. The exact feasibility of randomized solutions of robust convex programs. *SIAM Journal on Optimization*, 19(3):1211–1230, 2008.
7. P Colaneri, J.C Geromel, and A. Locatelli. *Control Theory and Design: an RH_2* and *RH_∞ viewpoint.* Academic Press, 1997.

8. J. Dong and G-H. Yang. Robust static output feedback control synthesis for linear continuous systems with polytopic uncertainties. *Automatica*, 49:1821–1829, 2013.
9. Y. Ebihara, D. Peaucelle, and D. Arzelier. *S-Variable Approach to LMI-based Robust Control*. Communications and Control Engineering. Springer, 2015.
10. L. El Ghaoui, F. Oustry, and M. AitRami. A cone complementary linearization algorithm for static output-feedback and related problems. *IEEE Transactions on Automatic Control*, 42:1171–1176, 1997.
11. J. Fiala, M. Kocvara, and M. Stingl. PENLAB - a solver for nonlinear semidefinite programming. Technical report, 2013.
12. S. Formentin, F. Dabbene, R. Tempo, L. Zaccarian, and S.M. Savaresi. Robust linear static anti-windup with probabilistic certificates. *IEEE Transactions on Automatic Control*, 62(4):1575–1589, 2017.
13. S. Gumussoy, D. Henrion, M. Millstone, and M.L. Overton. Multiobjective robust control with HIFOO 2.0. In *IFAC Symposium on Robust Control Design*, Haifa, Israel, 2009.
14. D. Henrion, J. Lofberg, M. Kocvara, and M. Stingl. Solving polynomial static output feedback problems with PENBMI. In *44th IEEE Conference on Decision and Control (CDC) and the European Control Conference (ECC)*, pages 7581–7586, 2005.
15. J. Löfberg. YALMIP, 2014.
16. Toker O and H. Ozbay. On the NP-hardness of solving bilinear matrix inequalities and simultaneous stabilization with static output feedback. In *IEEE American Control Conference*, pages 2525–2526, Seattle, 1995.
17. Y. Oishi. Probabilistic design of a robust controller using a parameter-dependent Lyapunov function. In Giuseppe Calafiore and Fabrizio Dabbene, editors, *Probabilistic and Randomized Methods for Design under Uncertainty*, pages 303–316. Springer London, 2006.
18. M.S. Sadabadi and D. Peaucelle. From static output feedback to structured robust static output feedback: A survey. *Annual Reviews in Control*, 42:11–26, 2016.
19. V.L. Syrmos, C.T. Abdallah, P. Dorato, and K. Grigoriadis. Static output feedback–a survey. *Automatica*, 33(2):125–137, 1997.
20. R. Tempo, G.C. Calafiore, and F. Dabbene. *Randomized Algorithms for Analysis and Control of Uncertain Systems: With Applications*. Springer, 2nd edition, 2013.
21. T.C. Toh, M.J. Todd, and R.H. Tutuncu. SDPT3 - a MATLAB software package for semidefinite programming. *Optimization Methods and Software*, 11:545–581, 1999.

Robust Control Against Uncertainty Quartet: A Polynomial Approach

Di Zhao, Chao Chen, Sei Zhen Khong and Li Qiu

In memory of Robert Tempo.

Abstract One of the main components of a robust control theory is a quantifiable description of system uncertainty. A good uncertainty description should have three desirable properties. First, it is required to capture important unmodeled dynamics and perturbations. Second, it needs to be mathematically tractable, preferably by using elementary tools. Third, it should lead to a self-contained robust control theory, encompassing analysis and synthesis techniques that are accessible to both researchers and practitioners. While the additive uncertainty and multiplicative uncertainty are two of the most commonly employed uncertainty descriptions in systems modeling and control, they come up short of fulfilling the requirements above. In this chapter, we introduce the uncertainty quartet, a.k.a. the $+ - \times \div$ uncertainty (as is simpler to pronounce in oriental languages), which combines in a unifying framework the additive, multiplicative, subtractive and divisive uncertainties. An elementary robust control theory, involving mostly polynomial manipulations, is developed based on the uncertainty quartet. The proposed theory is demonstrated in a case study on controlling an under-sensed and under-actuated linear (USUAL) inverted pendulum system.

D. Zhao · C. Chen · L. Qiu (✉)
Department of Electronic and Computer Engineering, The Hong Kong
University of Science and Technology, Clear Water Bay, Kowloon, Hong Kong, China
e-mail: eeqiu@ust.hk

D. Zhao
e-mail: dzhaoaa@ust.hk

C. Chen
e-mail: cchenap@ust.hk

S. Z. Khong
Department of Electrical and Electronic Engineering, The University of Hong Kong,
Pokfulam, Hong Kong, China
e-mail: szkhong@hku.hk

© Springer Nature Switzerland AG 2018

T. Başar (ed.), *Uncertainty in Complex Networked Systems*,
Systems & Control: Foundations & Applications,
https://doi.org/10.1007/978-3-030-04630-9_4

1 Uncertainty in Dynamical Systems

Model-based control synthesis is ubiquitous in engineering. It involves designing a controller based on a mathematical model of the system to be controlled (a.k.a. the plant). Every model, irrespective of its complexity, can at best approximate the dynamics of a real system. In other words, uncertainty is inherent to any mathematical model of a system. An uncertainty description or model provides a useful means to characterizing certain unmodeled dynamics of and unmeasured perturbations on a system. In what follows, we review several uncertainty models that have been widely adopted in practice, and discuss their advantages and shortcomings with the aid of an illustrative example involving a double integrator. Moreover, we introduce a powerful uncertainty description, known as the uncertainty quartet, that can be used to model a large class of uncertainties. A robust control theory based on the uncertainty quartet is subsequently developed in the succeeding sections.

1.1 Common Uncertainty Descriptions

The additive and multiplicative uncertainty descriptions constitute two of the most well-studied models in robust control. As an illustration, consider a nominal double integrator system

$$P(s) = \frac{1}{s^2},$$

which may model an ideal rigid body undergoing a forced linear motion. The real system, however, would have an elastic body. The dynamics arising from the presence of elasticity are not captured in this model and may correspond to an additive damped oscillatory term

$$\tilde{P}(s) = P(s) + \frac{\delta\omega_n^2}{s^2 + 2\zeta\omega_n s + \omega_n^2},$$

where δ denotes a small gain, ζ the damping ratio and ω_n the natural frequency, which can all be uncertain. The value of δ provides a quantification of the difference between the nominal system $P(s)$ and its perturbed model $\tilde{P}(s)$. The additive uncertainty description of the form

$$\tilde{P}(s) = P(s) + \varDelta_+(s)$$

can be used to model the aforementioned uncertainty satisfactorily. Observe that in this example, $\varDelta_+(s)$ is stable and has a small magnitude response as determined by the small parameter δ.

The use of the additive uncertainty model alone can be restrictive, as we explain below. Suppose that the double integrator is subject to an uncertain gain instead and the real system takes the form

$$\tilde{P}_1(s) = \frac{1+\delta}{s^2} \quad \text{or} \quad \tilde{P}_2(s) = \frac{1}{(1+\delta)s^2},$$

where δ denotes a small parameter. To model $\tilde{P}_i(s)$ using the additive uncertainty model, one would have to let

$$\Delta_+(s) = \frac{\delta}{s^2} \quad \text{or} \quad \Delta_+(s) = \frac{-\delta}{(1+\delta)s^2}.$$

In this case, both the uncertainty terms above are unstable, which are in no sense small since they both have infinite induced gains. In order to model the aforementioned perturbations with reasonably small uncertainties, we appeal to alternative uncertainty models having the multiplicative form

$$\tilde{P}_1(s) = (1 + \Delta_\times(s))P(s)$$

or the divisive form (a.k.a. the relative form)

$$\tilde{P}_2(s) = \frac{P(s)}{1 + \Delta_\div(s)}.$$

With respect to these models, we have $\Delta_\times(s) = \delta$ and $\Delta_\div(s) = \delta$, both of which are stable and small in magnitudes, as desired.

The following further demonstrates that the uncertainty descriptions covered above are still inadequate from a practical point of view. Due to the existence of a small stiffness in the rigid body motion, suppose the real system takes the form

$$\tilde{P}(s) = \frac{1}{s^2 + \varepsilon^2},$$

where ε is a small parameter. In this case, it can be verified that applying the additive, multiplicative and divisive models would result in an uncertainty term that is unstable. On the contrary, the subtractive form (a.k.a. the feedback form)

$$\tilde{P}(s) = \frac{P(s)}{1 + \Delta_-(s)P(s)}$$

gives $\Delta_-(s) = \varepsilon^2$, which is stable and small in magnitude.

1.2 The Uncertainty Quartet

By integrating the additive, subtractive, multiplicative, and divisive uncertainty models within a unifying framework, we arrive at the following uncertainty description:

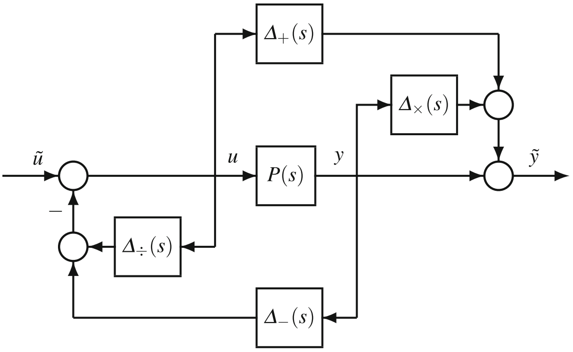


Fig. 1 An uncertain system with $+ - \times \div$ uncertainty

$$\tilde{P}(s) = \frac{(1+\Delta_\times(s))P(s) + \Delta_+(s)}{1 + \Delta_\div(s) + \Delta_-(s)P(s)}; \tag{1}$$

see the block diagram in Fig. 1 for a depiction of $\tilde{P}(s)$ as a mapping from $\tilde{u}$ to $\tilde{y}$. We call the band of the four uncertainties the **uncertainty quartet** (or the $+ - \times\div$ uncertainty), and the 2-by-2 transfer matrix

$$\boldsymbol{\Delta}(s) := \begin{bmatrix} \Delta_\div(s) & \Delta_-(s) \\ \Delta_+(s) & \Delta_\times(s) \end{bmatrix}$$

the uncertainty quartet matrix. It is straightforward to see that (1) gives rise to a versatile form that can be used to model a wide class of uncertainties.

To motivate the utility of the uncertainty quartet, let us revisit the example of the double integrator. Suppose the real system has dynamics of the form

$$\tilde{P}(s) = \frac{1+\delta_2}{s^2 + \delta_1 s + \varepsilon^2},$$

where ε^2 is a small stiffness term, δ_1 a small damping coefficient and δ_2 a small uncertain gain. It can be verified that using only the additive and multiplicative forms of uncertainty would result in unstable $\Delta_+(s)$ and $\Delta_\times(s)$ regardless of the values of ε^2, δ_1 and δ_2. Likewise, adopting only the relative and the feedback form of the uncertainty would give rise to unstable $\Delta_\div(s)$ and $\Delta_-(s)$. On the other hand, if we characterize $\tilde{P}(s)$ with the uncertainty quartet by applying equation (1), we obtain

$$\boldsymbol{\Delta}(s) = \begin{bmatrix} \Delta_\div(s) & \Delta_-(s) \\ \Delta_+(s) & \Delta_\times(s) \end{bmatrix} = \frac{1}{s+1}\begin{bmatrix} \delta_1 & (\delta_1 + \varepsilon^2)s + \varepsilon^2 \\ 0 & \delta_2 \end{bmatrix}.$$

Each member in this uncertainty quartet is stable and small in magnitude. This example demonstrates the fact that while each of the individual uncertainty models falls short of providing a satisfactory characterization of the uncertainty, their combination (1) introduces a powerful framework in which we can model various types of perturbations.

Mathematically, the map from $P(s)$ to $\tilde{P}(s)$ is a linear fractional transformation (LFT). In particular, let

$$\mathrm{LFT}\left(\begin{bmatrix} T_{11}(s) & T_{12}(s) \\ T_{21}(s) & T_{22}(s) \end{bmatrix}, P(s)\right) = \frac{T_{22}(s)P(s) + T_{12}(s)}{T_{11}(s) + T_{21}(s)P(s)},$$

then

$$\tilde{P}(s) = \mathrm{LFT}\left(\begin{bmatrix} 1 + \Delta_{\div}(s) & \Delta_{-}(s) \\ \Delta_{+}(s) & 1 + \Delta_{\times}(s) \end{bmatrix}, P(s)\right).$$

The study of various uncertainty models has a long history in the field of robust control. The additive, subtractive, multiplicative, and divisive models were covered in such classic books as [2, 29] and revisited more recently in, for example, [16]; see also the survey paper [19]. The uncertainty quartet unifies all four of the aforementioned uncertainties within one powerful framework for robustness analysis and control synthesis. It is worth noting that the notation of $+ - \times \div$ uncertainty was first used in [10]. It is shown in [8] that the uncertainty quartet is closely related to the gap metric and its variations [6, 7, 20, 21, 24, 27]. The uncertainty quartet has also been used to describe the interferences and distortions within a communication channel modeled by a two-port network [9, 28]. Moreover, one may relate the uncertainty quartet to the coprime-factor uncertainty [6, 17, 23], namely, a pair of dynamic uncertainties additive to the coprime factors of a nominal system. It is noteworthy that in the uncertainty quartet, each member acts directly on the input and output of the nominal system, whereas the coprime-factor uncertainty depends on a particular coprime factorization of the nominal system. In addition to the uncertainty quartet, many other types of dynamic uncertainties have been studied over the past decades; see, for instance, [13, 16, 19, 29].

1.3 Notation

We formalize the notation in this chapter. Let $\boldsymbol{I}$ denote the identity matrix of a proper dimension. Let $\mathscr{R}^{p\times m}$ denote the set of all $p \times m$ proper real-rational transfer function matrices. The set of elements in $\mathscr{R}^{p\times m}$ containing bounded singular values on the imaginary axis is denoted by $\mathscr{RL}_{\infty}^{p\times m}$ and the set of elements in $\mathscr{RL}_{\infty}^{p\times m}$ with bounded singular values on the right complex plane $\mathrm{Re}\, s > 0$ is denoted by $\mathscr{RH}_{\infty}^{p\times m}$. A transfer function $P(s) \in \mathscr{R}^{p\times m}$ is said to be *stable* if $P(s) \in \mathscr{RH}_{\infty}^{p\times m}$. Define the set of uncertain systems of size $r \in [0, 1)$ centered at $P(s)$ as

$$\mathscr{B}(P(s), r) = \left\{\mathrm{LFT}\left(\boldsymbol{I} + \boldsymbol{\Delta}(s), P(s)\right) : \boldsymbol{\Delta}(s) \in \mathscr{RH}_{\infty}^{2\times 2},\ \|\boldsymbol{\Delta}(s)\|_{\infty} \le r\right\}. \quad (2)$$

Throughout, the superscripts corresponding to the dimensions may be omitted for notational simplicity.

Recall the standard Lebesgue space $\mathscr{L}_2$ endowed with the norm $\|\cdot\|_2$ and Hardy space $\mathscr{H}_2 \subset \mathscr{L}_2$. The orthogonal complement of $\mathscr{H}_2$ in $\mathscr{L}_2$ is denoted by $\mathscr{H}_2^{\perp}$. In other words, $\mathscr{L}_2 = \mathscr{H}_2 \oplus \mathscr{H}_2^{\perp}$, where $\oplus$ denotes the orthogonal sum. For a $G(s) \in \mathscr{R}\mathscr{L}_\infty$, we have

$$\|G(s)\|_\infty = \sup_{U(s)\in\mathscr{L}_2} \frac{\|G(s)U(s)\|_2}{\|U(s)\|_2}.$$

Moreover, if $U_1(s) \in \mathscr{H}_2$ and $U_2(s) \in \mathscr{H}_2^{\perp}$, then

$$\|U_1(s) + U_2(s)\|_2^2 = \|U_1(s)\|_2^2 + \|U_2(s)\|_2^2.$$

2 Robust Closed-Loop Stability

As explained in the last section, uncertainty is intrinsic to every mathematical model of a system. This fact is particularly problematic to model-based control—if a model does not accurately describe the behavior of a system, how can we be certain that a controller designed based on the model will perform well when it is implemented on the system? Feedback, which underlies the field of systems and control, is most commonly adopted to resolve this issue. It is a powerful tool with which we desensitize a dynamical system to the effect of uncertainty. The theory of feedback control, which will be briefly reviewed in this section, has been well studied over recent decades and demonstrated to be effective in many application scenarios [22, 25, 29]. We begin with the notion of a standard feedback (or closed-loop) system, and define its closed-loop stability. Then we analyze robust closed-loop stability when the plant is subject to uncertainty quartet, based on which we derive a robust stability condition.

A closed-loop system composed of a plant $P(s) \in \mathscr{R}$ and a feedback controller $C(s) \in \mathscr{R}$ is illustrated in Fig. 2. We denote it by $P(s)\,\#\,C(s)$. We say that $P(s)\,\#\,C(s)$ is stable if for all exogenous signals $w_1, w_2 \in \mathscr{H}_2$, the endogenous signals u_1, u_2, y_1, y_2 exist and belong to $\mathscr{H}_2$. Intuitively, stability means that the energy within the feedback system stays bounded when it is injected with bounded-energy

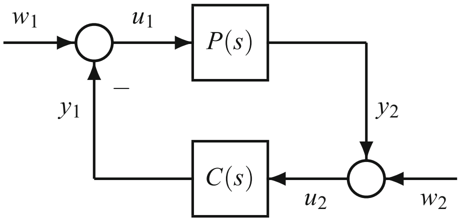


Fig. 2 A standard feedback system

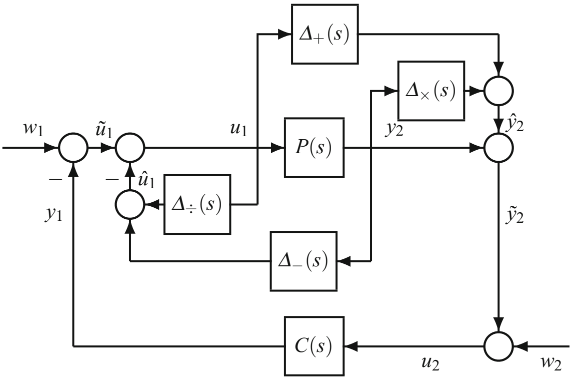


Fig. 3 A closed-loop system with $+ - \times \div$ uncertainty quartet at plant side

exogenous signals. It is known that $P(s) \# C(s)$ is stable if and only if the associated **Gang of Four** transfer matrix

$$P(s) \# C(s) := \begin{bmatrix} 1 \\ P(s) \end{bmatrix} (1 + P(s)C(s))^{-1} \begin{bmatrix} 1 & C(s) \end{bmatrix} = \begin{bmatrix} \dfrac{1}{1 + P(s)C(s)} & \dfrac{C(s)}{1 + P(s)C(s)} \\ \dfrac{P(s)}{1 + P(s)C(s)} & \dfrac{P(s)C(s)}{1 + P(s)C(s)} \end{bmatrix}$$

is stable [1]. Here, both the closed-loop system and its associated Gang of Four transfer matrix are denoted by $P(s) \# C(s)$ for notational simplicity.

Recall from the preceding section that an uncertainty quartet is useful for modeling a rich class of uncertainties. When model-based feedback control design is performed based on a mathematical model of a plant, it gives rise to a stable nominal closed-loop system. In the following, we analyze the closed-loop stability when the plant is subject to uncertainty quartet and provide a quantification of how much uncertainty is tolerable while the feedback system remains stable. Mathematically, let $P(s) \# C(s)$ be a nominal closed-loop system. We derive an upper bound on $r > 0$ such that $\tilde{P}(s) \# C(s)$ is stable for all $\tilde{P}(s) \in \mathscr{B}(P(s), r)$. First recall from (2) that every $\tilde{P}(s) \in \mathscr{B}(P(s), r)$ can be expressed as

$$\tilde{P}(s) = \text{LFT}\left(\begin{bmatrix} 1 + \Delta_\div(s) & \Delta_-(s) \\ \Delta_+(s) & 1 + \Delta_\times(s) \end{bmatrix}, P(s) \right) = \frac{(1 + \Delta_\times(s))P(s) + \Delta_+(s)}{1 + \Delta_\div(s) + \Delta_-(s)P(s)},$$

where

$$\|\boldsymbol{\Delta}(s)\|_\infty = \left\| \begin{bmatrix} \Delta_\div(s) & \Delta_-(s) \\ \Delta_+(s) & \Delta_\times(s) \end{bmatrix} \right\|_\infty \leq r.$$

See Fig. 3 for a depiction of the perturbed closed-loop system $\tilde{P}(s) \# C(s)$. It is shown in [9] that the signals within the perturbed closed-loop system satisfy the following relations

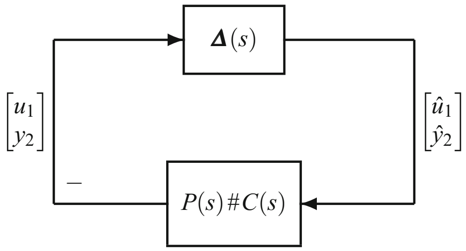

$\boldsymbol{\Delta}(s)$
$\begin{bmatrix} u_1 \\ y_2 \end{bmatrix}$
$\begin{bmatrix} \hat{u}_1 \\ \hat{y}_2 \end{bmatrix}$
$-$
$P(s)\#C(s)$

Fig. 4 An equivalent closed-loop system composed of an uncertainty quartet and the Gang of Four transfer matrix

$$\begin{bmatrix} \hat{u}_1 \\ \hat{y}_2 \end{bmatrix} = \boldsymbol{\Delta}(s) \begin{bmatrix} u_1 \\ y_2 \end{bmatrix} \quad \text{and} \quad \begin{bmatrix} u_1 \\ y_2 \end{bmatrix} = P(s)\,\#\,C(s) \begin{bmatrix} \hat{u}_1 \\ \hat{y}_2 \end{bmatrix}.$$

As a result, we can equivalently transform the perturbed closed-loop system in Fig. 3 into a standard feedback interconnection of an uncertainty quartet $\boldsymbol{\Delta}(s)$ and the Gang of Four transfer matrix $P(s)\,\#\,C(s)$ as shown in Fig. 4. Furthermore, it can be verified that the stability of $\tilde{P}(s)\,\#\,C(s)$ is equivalent to that of the closed-loop system in Fig. 4.

Since both open-loop systems $\boldsymbol{\Delta}(s)$ and $P(s)\,\#\,C(s)$ are stable, robust stability of the closed-loop system in Fig. 4 can be analyzed by means of the well-known small-gain theorem [29, Theorem 8.1]. In particular, the closed-loop system $\boldsymbol{\Delta}(s)\,\#\,[P(s)\,\#\,C(s)]$ is stable for all $\|\boldsymbol{\Delta}(s)\| \le r$ if, and only if, $r < \|P(s)\,\#\,C(s)\|_\infty^{-1}$. Consequently, we have the following robust stability condition.

Theorem 1 *Let $r \in [0, 1)$. The perturbed closed-loop system $\tilde{P}(s)\,\#\,C(s)$ in Fig. 3 is stable for all $\tilde{P}(s) \in \mathscr{B}(P(s), r)$ if and only if*

$$r < \|P(s)\,\#\,C(s)\|_\infty^{-1}.$$

By virtue of Theorem 1, it is natural to define $\|P(s)\,\#\,C(s)\|_\infty^{-1}$ as the robust stability margin of the nominal closed-loop system $P(s)\,\#\,C(s)$. The larger the margin is, the more robust the closed-loop system will be against model uncertainties characterized in the form of an uncertainty quartet. An optimal control problem naturally arises from this context. It involves designing a feedback controller $C(s)$ for a nominal plant $P(s)$ such that the stability margin $\|P(s)\,\#\,C(s)\|_\infty^{-1}$ is maximized, or equivalently, solving the following $\mathscr{H}_\infty$ control problem:

$$\min_{C(s)} \|P(s)\,\#\,C(s)\|_\infty. \tag{3}$$

The optimally robust stability margin is thus given by

$$\alpha(P(s)) := \left(\min_{C(s)} \|P(s)\,\#\,C(s)\|_\infty\right)^{-1}. \tag{4}$$

3 Optimally Robust Controller Design

In this section, our aim is to derive an optimally robust controller $C(s)$ that minimizes $\|P(s) \# C(s)\|_\infty$. This is an $\mathscr{H}_\infty$ optimal control problem. It has favorable properties and can be solved efficiently using state-space methods [3, 17] based on algebraic Riccati equations. Given the simplicity of our setup, formulated in terms of scalar transfer functions, we provide below a more straightforward and efficient alternative to solving the optimization problem via a polynomial approach.

It is worth noting that a certain polynomial method was proposed in [15], [22, Chapter 9], where the $\mathscr{H}_\infty$ optimal control problem is solved by calculating a Hankel matrix based on special bases. Another similar polynomial method was proposed in [12, Chapter 5] based on solving polynomial equations. By contrast, in this chapter, we propose an even simpler alternative polynomial method, involving only elementary matricial and polynomial manipulations.

3.1 Main Algorithm

First we introduce some notation. Consider an arbitrary polynomial with real coefficients

$$f(s) = f_0 s^n + f_1 s^{n-1} + \cdots + f_n,$$

whose degree, denoted by deg $f(s)$, is no larger than n. Correspondingly to the polynomial $f(s)$, define

$$\boldsymbol{f} := \begin{bmatrix} f_0 \\ \vdots \\ f_n \end{bmatrix}, \quad \boldsymbol{L}_f := \begin{bmatrix} f_0 & 0 & \cdots & 0 \\ f_1 & f_0 & \ddots & \vdots \\ \vdots & \ddots & \ddots & 0 \\ f_{n-1} & \cdots & f_1 & f_0 \end{bmatrix} \quad \text{and} \quad \boldsymbol{U}_f := \begin{bmatrix} f_n & f_{n-1} & \cdots & f_1 \\ 0 & f_n & \ddots & \vdots \\ \vdots & \ddots & \ddots & f_{n-1} \\ 0 & \cdots & 0 & f_n \end{bmatrix}.$$

Let $\boldsymbol{J}$ be a sign matrix, defined as

$$\boldsymbol{J} := \begin{bmatrix} (-1)^{n-1} & & & \\ & \ddots & & \\ & & -1 & \\ & & & 1 \end{bmatrix}.$$

The matrices $\boldsymbol{L}_f$, $\boldsymbol{U}_f$ and $\boldsymbol{J}$ are helpful in transforming a polynomial equation into a system of linear equations. As we shall see, such a transformation plays an important role in the proposed polynomial approach.

Suppose we are given an nth order plant

$$P(s) = \frac{b(s)}{a(s)} = \frac{b_0 s^n + b_1 s^{n-1} + \cdots + b_n}{a_0 s^n + a_1 s^{n-1} + \cdots + a_n}, \tag{5}$$

where $a_0 \neq 0$, and $a(s)$ and $b(s)$ are coprime. The following algorithm computes an optimally robust controller

$$C_{\text{opt}}(s) = \underset{C(s)}{\arg\min} \, \| P(s) \,\#\, C(s) \|_\infty .$$

Algorithm 1 Optimally Robust Controller Design

Step 1: (*Spectral factorization*) Find a stable polynomial

$$d(s) = d_0 s^n + d_1 s^{n-1} + \cdots + d_n$$

such that

$$a(-s)a(s) + b(-s)b(s) = d(-s)d(s).$$

Step 2: (*Matrix construction*) Construct

$$\boldsymbol{H} = \boldsymbol{J}\boldsymbol{L}_d^{-1}\boldsymbol{J}\begin{bmatrix}\boldsymbol{L}_b\boldsymbol{J} & -\boldsymbol{L}_a\boldsymbol{J}\end{bmatrix}\begin{bmatrix}\boldsymbol{L}_a & \boldsymbol{L}_b \\ \boldsymbol{U}_a & \boldsymbol{U}_b\end{bmatrix}^{-1}\begin{bmatrix}\boldsymbol{L}_d \\ \boldsymbol{U}_d\end{bmatrix}.$$

Step 3: (*Eigen-computation*) Find the eigenvalue of $\boldsymbol{H}$ whose magnitude equals to the spectral radius $\rho(\boldsymbol{H})$. Let $\boldsymbol{e}$ be an eigenvector corresponding to this eigenvalue[1].

Step 4: (*Pole placement*) Compute

$$\begin{bmatrix}\boldsymbol{p} \\ \boldsymbol{q}\end{bmatrix} = \begin{bmatrix}\boldsymbol{L}_a & \boldsymbol{L}_b \\ \boldsymbol{U}_a & \boldsymbol{U}_b\end{bmatrix}^{-1}\begin{bmatrix}\boldsymbol{L}_d \\ \boldsymbol{U}_d\end{bmatrix}\boldsymbol{e} \quad \text{and} \quad \begin{cases} p(s) = \begin{bmatrix} s^{n-1} & s^{n-2} & \cdots & 1\end{bmatrix}\boldsymbol{p} \\ q(s) = \begin{bmatrix} s^{n-1} & s^{n-2} & \cdots & 1\end{bmatrix}\boldsymbol{q} \end{cases}.$$

An optimal controller is given by $C_{\text{opt}}(s) = \dfrac{q(s)}{p(s)}$.

Step 5: (*Optimal robustness margin computation*)

$$\alpha(P(s)) = \frac{1}{\sqrt{1 + \rho^2(\boldsymbol{H})}}.$$

[1] It can be shown that all the eigenvalues of $\boldsymbol{H}$ are real. For clarity of presentation, it is implicitly assumed that there exists a unique eigenvalue of $\boldsymbol{H}$ whose magnitude is $\rho(\boldsymbol{H})$. This is generically the case. The more involved situation is discussed specifically in Section 3.3.

Notice that only basic matricial and polynomial manipulations, such as spectral factorization, eigenvalue decomposition and matrix inversion are required in the algorithm above. See Sect. 3.2 for an illustrative example of applying the algorithm.

Whereas **Steps 2, 3, and 5** in Algorithm 1 are concerned with the optimal control design, **Steps 1 and 4** are standard and well known, as we elaborate below. Denote by $\mathscr{P}_n$ the set of all the polynomials with real coefficients and of degree n. That is, for $d(s) \in \mathscr{P}_n$, it holds $d(s) = d_0 s^n + d_1 s^{n-1} + \cdots + d_n$ with $d_0 \neq 0$. This polynomial $d(s)$ is said to be stable if all its roots have negative real parts.

Let the plant $P(s)$ be given as in (5). Observe that the polynomial

$$a(-s)a(s) + b(-s)b(s) \tag{6}$$

is self-conjugate, i.e., its conjugate coincides with itself. Consequently, if z is a root of this polynomial, then so is $-z$. Together with the coprimeness of $a(s)$ and $b(s)$, it follows that this polynomial has no roots on the imaginary axis and all its roots are symmetric about the imaginary axis. **Step 1** in Algorithm 1 can be carried out by first solving for the roots of the polynomial in (6) and then obtaining a stable polynomial $d(s) \in \mathscr{P}_n$ such that

$$a(-s)a(s) + b(-s)b(s) = d(-s)d(s). \tag{7}$$

This process is known as the spectral factorization [11, Section 3.4], [22, Section 8.1].

Given two coprime polynomials with real coefficients $p(s)$ and $q(s)$, by defining

$$C(s) := \frac{q(s)}{p(s)}, \tag{8}$$

we know from the definition of the Gang of Four transfer matrix $P(s) \# C(s)$ that the closed-loop poles are the roots of the characteristic polynomial

$$a(s)p(s) + b(s)q(s).$$

One way to obtain $p(s)$ and $q(s)$ is via the pole placement method as follows. Let $e(s) \in \mathscr{P}_{n-1}$ be a stable polynomial. By solving the following polynomial Diophantine equation [11, Section 4.5] [22, Section 3.6]

$$a(s)p(s) + b(s)q(s) = d(s)e(s), \tag{9}$$

we obtain $p(s)$ and $q(s)$ with $\max\{\deg p(s), \deg q(s)\} \leq n - 1$. A controller $C(s)$ defined as in (8) then places the closed-loop poles at the roots of $d(s)e(s)$. Such a process is called the pole placement design, and the resulting $C(s)$ is called a pole placement controller. In particular, equating the coefficients in (9) yields the following system of linear equations:

$$\begin{bmatrix} \boldsymbol{L}_a & \boldsymbol{L}_b \\ \boldsymbol{U}_a & \boldsymbol{U}_b \end{bmatrix} \begin{bmatrix} \boldsymbol{p} \\ \boldsymbol{q} \end{bmatrix} = \begin{bmatrix} \boldsymbol{L}_d \\ \boldsymbol{U}_d \end{bmatrix} \boldsymbol{e}, \tag{10}$$

where the elements in $\boldsymbol{p}$ and $\boldsymbol{q}$ are the unknowns. The matrix

$$\begin{bmatrix} \boldsymbol{L}_a & \boldsymbol{L}_b \\ \boldsymbol{U}_a & \boldsymbol{U}_b \end{bmatrix}$$

is called a Sylvester's resultant matrix [22, Section 3.6], which is a $2n$-by-$2n$ nonsingular matrix as $a(s)$ and $b(s)$ are coprime. By inverting this matrix as in **Step 4** of Algorithm 1, we obtain the solution to equation (9), as well as the pole placement controller.

It is now obvious that **Steps 2 and 3** of Algorithm 1 serve the purpose of computing a partial set of the closed-loop poles, based on which the pole placement controller resulting from **Step 4** gives rise to an optimally robust controller. The proof of this fact is deferred to Sect. 4.

3.2 An Illustrative Example

Here we revisit the simple example of a double integrator and apply Algorithm 1 to obtain an optimally robust controller.

Example 1 Let

$$P(s) = \frac{1}{s^2}.$$

Objective: find an optimal controller $C(s)$ such that $\|P(s) \# C(s)\|_\infty$ is minimized with Algorithm 1 .

1. (*Spectral factorization*)
$$s^4 + 1 = d(-s)d(s).$$
This gives $d(s) = s^2 + \sqrt{2}s + 1$.
2. (*Matrix computation*) We can compute that
$$\boldsymbol{H} = \begin{bmatrix} 1 & \sqrt{2} \\ \sqrt{2} & 1 \end{bmatrix}.$$
3. (*Eigen-computation*) The eigenvalues of $\boldsymbol{H}$ are $1 \pm \sqrt{2}$. The eigenvalue with the largest magnitude is $1 + \sqrt{2}$ and the corresponding eigenvector satisfies
$$\begin{bmatrix} 1 & \sqrt{2} \\ \sqrt{2} & 1 \end{bmatrix} \boldsymbol{e} = (1 + \sqrt{2})\boldsymbol{e}.$$

This gives $\boldsymbol{e} = \begin{bmatrix} 1 \\ 1 \end{bmatrix}$. Thus,

$$e(s) = \begin{bmatrix} s & 1 \end{bmatrix} \boldsymbol{e} = s + 1.$$

4. (*Pole placement*) We obtain $p(s) = s + 1 + \sqrt{2}$ and $q(s) = (1 + \sqrt{2})s + 1$ from

$$\begin{bmatrix} \boldsymbol{p} \\ \boldsymbol{q} \end{bmatrix} = \begin{bmatrix} \boldsymbol{L}_a & \boldsymbol{L}_b \\ \boldsymbol{U}_a & \boldsymbol{U}_b \end{bmatrix}^{-1} \begin{bmatrix} \boldsymbol{L}_d \\ \boldsymbol{U}_d \end{bmatrix} \boldsymbol{e} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}^{-1} \begin{bmatrix} 1 & 0 \\ \sqrt{2} & 1 \\ 1 & \sqrt{2} \\ 0 & 1 \end{bmatrix} \begin{bmatrix} 1 \\ 1 \end{bmatrix} = \begin{bmatrix} 1 \\ 1 + \sqrt{2} \\ 1 + \sqrt{2} \\ 1 \end{bmatrix}.$$

An optimally robust controller is then given by

$$C_{\text{opt}}(s) = \frac{(1 + \sqrt{2})s + 1}{s + 1 + \sqrt{2}}.$$

5. (*Optimal robustness margin computation*)

$$\alpha(P(s)) = \frac{1}{\sqrt{4 + 2\sqrt{2}}}.$$

3.3 The Nongeneric Case

In **Step 3** of Algorithm 1, the generic case where $\boldsymbol{H}$ admits a unique eigenvalue of magnitude $\rho(\boldsymbol{H})$ is dealt with. Here we mention without proof a method to handle the singular case where $\boldsymbol{H}$ has multiple eigenvalues of magnitude $\rho(\boldsymbol{H})$. This will not be pursued further elsewhere in this chapter. First we introduce some notation. For a square matrix $\boldsymbol{A} \in \mathbb{R}^{n \times n}$, denote by $\lambda_k(\boldsymbol{A})$, $k = 1, 2, \ldots, n$ its k-th eigenvalue counting multiplicity, ordered according to

$$|\lambda_1(\boldsymbol{A})| = \cdots = |\lambda_v(\boldsymbol{A})| > |\lambda_{v+1}(\boldsymbol{A})| \geq \cdots \geq |\lambda_n(\boldsymbol{A})|.$$

The spectral radius of $\boldsymbol{H}$, $\rho(\boldsymbol{H})$, is hence $|\lambda_1(\boldsymbol{A})|$. Let the number of the eigenvalues of magnitude $\rho(\boldsymbol{H})$ be $m(\boldsymbol{A}) := v > 1$.

It can be shown using the spectral factorization relation in **Step 1** of Algorithm 1 that $\boldsymbol{H}$ is diagonalizable and all its eigenvalues are real, hence either or both of $\rho(\boldsymbol{H})$ and $-\rho(\boldsymbol{H})$ are eigenvalues of $\boldsymbol{H}$. If $\rho(\boldsymbol{H})$ is an eigenvalue, let $\mathscr{E}_1$ be the corresponding eigenspace; otherwise $\mathscr{E}_1 = \{\boldsymbol{0}\}$. Similarly, let $\mathscr{E}_2$ be the eigenspace corresponding to $-\rho(\boldsymbol{H})$. Then Algorithm 1 with **Step 3** replaced by **Step 3*** below yields an optimally robust controller whose order is no larger than $n - m(\boldsymbol{H})$:

Step 3*: (*Eigen-computation*) Find $\mathbf{0} \neq \boldsymbol{e} \in \mathscr{E}_1 \cup \mathscr{E}_2$ such that the degree of $e(s)$ is minimized, where

$$e(s) = \begin{bmatrix} s^{n-1} \; s^{n-2} \; \cdots \; 1 \end{bmatrix} \boldsymbol{e}.$$

The following example of a special all-pass system illustrates how we utilize the algorithm when $m(\boldsymbol{H}) > 1$.

Example 2 Consider the following all-pass plant

$$P(s) = \frac{(s-1)(s-2)(s-3)}{(s+1)(s+2)(s+3)} = \frac{s^3 - 6s^2 + 11s - 6}{s^3 + 6s^2 + 11s + 6}.$$

Objective: find an optimal controller $C(s)$ such that $\|P(s) \# C(s)\|_\infty$ is minimized using Algorithm 1 equipped with **Step 3*** above.

1. (*Spectral factorization*)

$$d(s) = \sqrt{2}(s+1)(s+2)(s+3) = \sqrt{2}(s^3 + 6s^2 + 11s + 6).$$

2. (*Matrix computation*) We can compute that

$$\boldsymbol{H} = \begin{bmatrix} -1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & -1 \end{bmatrix}.$$

3. (*Eigen-computation*) The eigenvalues of $\boldsymbol{H}$ are 1, -1 and -1, all of which have magnitude 1. Hence $m(\boldsymbol{H}) = n = 3$. The eigenspaces corresponding to eigenvalues 1 and -1 are, respectively,

$$\mathscr{E}_1 = \text{span} \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix} \quad \text{and} \quad \mathscr{E}_2 = \text{span} \begin{bmatrix} 1 & 0 \\ 0 & 0 \\ 0 & 1 \end{bmatrix}.$$

The vector $\mathbf{0} \neq \boldsymbol{e} \in \mathscr{E}_1 \cup \mathscr{E}_2$ such that the degree of $e(s)$ is minimized is given by

$$\boldsymbol{e} = \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}, \quad \text{whereby} \quad e(s) = 1, \quad \text{and} \quad \deg e(s) = 0 = n - m(\boldsymbol{H}).$$

4. (*Pole placement*) We obtain $p(s) = \sqrt{2}$ and $q(s) = 0$ from

$$\begin{bmatrix} \boldsymbol{p} \\ \boldsymbol{q} \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 1 & 0 & 0 \\ 6 & 1 & 0 & -6 & 1 & 0 \\ 11 & 6 & 1 & 11 & -6 & 1 \\ 6 & 11 & 6 & -6 & 11 & -6 \\ 0 & 6 & 11 & 0 & -6 & 11 \\ 0 & 0 & 6 & 0 & 0 & -6 \end{bmatrix}^{-1} \begin{bmatrix} \sqrt{2} & 0 & 0 \\ 6\sqrt{2} & \sqrt{2} & 0 \\ 11\sqrt{2} & 6\sqrt{2} & \sqrt{2} \\ 6\sqrt{2} & 11\sqrt{2} & 6\sqrt{2} \\ 0 & 6\sqrt{2} & 11\sqrt{2} \\ 0 & 0 & 6\sqrt{2} \end{bmatrix} \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ \sqrt{2} \\ 0 \\ 0 \\ 0 \end{bmatrix}.$$

Hence, an optimally robust controller is given by

$$C_{\text{opt}}(s) = 0.$$

5. (*Optimal robustness margin computation*)

$$\alpha(P(s)) = \frac{1}{\sqrt{2}}.$$

4 Proof of Optimality

The purpose of this section is to prove that the controller $C_{\text{opt}}(s)$ obtained from Algorithm 1 is optimal, in the sense that it satisfies

$$C_{\text{opt}}(s) = \arg\min_{C(s)} \| P(s) \,\#\, C(s) \|_{\infty}.$$

4.1 Preliminaries

Given a plant $P(s) = \dfrac{b(s)}{a(s)}$ and a stable polynomial $e(s)$, we can rewrite the spectral factorization in (7) and the pole placement in (9) as, respectively,

$$M(-s)M(s) + N(-s)N(s) = 1 \quad \text{and} \quad M(s)X(s) + N(s)Y(s) = 1, \tag{11}$$

where

$$M(s) := \frac{a(s)}{d(s)}, \quad N(s) := \frac{b(s)}{d(s)}, \quad X(s) := \frac{p(s)}{e(s)}, \quad \text{and} \quad Y(s) := \frac{q(s)}{e(s)}. \tag{12}$$

Based on these relations, the set of all controllers $C(s)$ for which $P(s) \,\#\, C(s)$ is stable is given by the Youla parametrization [26] as

$$\mathscr{S}(P(s)) = \left\{ C(s) = \frac{Y(s) + M(s)Q(s)}{X(s) - N(s)Q(s)} \,:\; Q(s) \in \mathscr{RH}_\infty \right\}. \tag{13}$$

Obviously, an optimally robust controller belongs to the set $\mathscr{S}(P(s))$.

It can be shown with some algebraic manipulations [22, Chapter 9] that

$$\|P(s) \# C(s)\|_\infty = \left(1 + \left\| \frac{P(-s) - C(s)}{1 + P(s)C(s)} \right\|_\infty^2 \right)^{\frac{1}{2}}.$$

As a consequence,

$$\min_{C(s) \in \mathscr{S}(P(s))} \|P(s) \# C(s)\|_\infty$$

is equivalent to

$$\min_{C(s) \in \mathscr{S}(P(s))} \left\| \frac{P(-s) - C(s)}{1 + P(s)C(s)} \right\|_\infty =: \gamma(P(s)). \tag{14}$$

Furthermore, the optimal robust stability margin is

$$\alpha(P(s)) = \frac{1}{\sqrt{1 + \gamma^2(P(s))}}.$$

Later in the section, we will show that $\gamma(P(s)) = \rho(\boldsymbol{H})$, whereby

$$\alpha(P(s)) = \frac{1}{\sqrt{1 + \rho^2(\boldsymbol{H})}}$$

as in **Step 5** of Algorithm 1.

In the sequel, we derive an alternative form, which is easier to work with, for the $\mathscr{H}_\infty$ optimal control problem in (14). In particular, given the set of all stabilizing controllers $\mathscr{S}(P(s))$ in (13) and $M(s)$, $N(s)$, $X(s)$, $Y(s)$ defined in (12), we have

$$\begin{aligned}
\gamma(P(s)) &= \inf_{C(s) \in \mathscr{S}(P(s))} \left\| \frac{P(-s) - C(s)}{1 + P(s)C(s)} \right\|_\infty \\
&= \inf_{Q(s) \in \mathscr{RH}_\infty} \left\| \frac{M(s)[N(-s)X(s) - M(-s)Y(s) - Q(s)]}{M(-s)} \right\|_\infty \\
&= \inf_{Q(s) \in \mathscr{RH}_\infty} \|N(-s)X(s) - M(-s)Y(s) - Q(s)\|_\infty \\
&= \inf_{Q(s) \in \mathscr{RH}_\infty} \|G(s) - Q(s)\|_\infty ,
\end{aligned} \tag{15}$$

where

$$G(s) := N(-s)X(s) - M(-s)Y(s) = \frac{b(-s)p(s) - a(-s)q(s)}{d(-s)e(s)} \in \mathscr{RL}_\infty, \quad (16)$$

and the third equality follows from the fact that

$$\frac{M(s)}{M(-s)}$$

is an all-pass transfer function. Consequently, solving for an optimally robust controller is equivalent to finding a $Q(s) \in \mathscr{RH}_\infty$ that lies the closest to $G(s) \in \mathscr{RL}_\infty$ in the $\mathscr{L}_\infty$ norm. This special $\mathscr{H}_\infty$ optimal control problem is a **Nehari's problem** [18], [5, Chapter 12], which is closely related to the partial pole placement problem introduced in what follows.

4.2 Partial Pole Placement

Definition 1 Given an nth order plant $P(s) = \dfrac{b(s)}{a(s)}$ and a stable polynomial $d(s) \in \mathscr{P}_n$ obtained from the spectral factorization in (7), we say that a triplet of polynomials $\{p(s), q(s), e(s)\}$ solves the **partial pole placement problem** for $\lambda \in \mathbb{R}$ if it satisfies

$$\begin{aligned} a(s)p(s) + b(s)q(s) &= d(s)e(s), \\ b(-s)p(s) - a(-s)q(s) &= \lambda d(s)e(-s), \\ \max\{\deg p(s), \deg q(s)\} &\leq \deg e(s) \leq n-1. \end{aligned} \quad (17)$$

A partial pole placement problem is distinguished from a pole placement problem in (9), since the closed-loop poles, namely the roots of $d(s)e(s)$, are not completely prescribed ahead of time and need to be determined from the equations in (17). Two questions arise naturally from this problem:

(i) What are the possible solutions to the partial pole placement problem?
(ii) How are these solutions related to the Nehari's problem in (15)?

We answer these questions below, and in doing so complete the main part of the derivation for the optimal controller from Algorithm 1. We begin with Question (ii).

Recall the expression of $G(s)$ in (16). In a similar manner, define a series of transfer functions in $\mathscr{RL}_\infty$ for $k = 1, 2, \ldots, n$ by

$$G_k(s) = \frac{b(-s)p_k(s) - a(-s)q_k(s)}{d(-s)e_k(s)}, \quad (18)$$

where $\{p_k(s), q_k(s), e_k(s)\}$ is a solution of (17) with respect to λ_k and $e_k(s)$ has exactly $k-1$ anti-stable roots. In particular, $e_1(s)$ is stable. Recall that the $e(s) \in \mathscr{P}_{n-1}$ in (16) is required to be a stable polynomial. Henceforth, let $e(s) = e_1(s)$, whereby $G(s) = G_1(s)$. Denote by $\mathscr{RL}_\infty^{[k]} \subset \mathscr{RL}_\infty$ the set of all the transfer functions that have at most $k-1$ anti-stable poles. Specifically, $\mathscr{RL}_\infty^{[1]} = \mathscr{RH}_\infty$. Similarly to (15), consider the series of optimization problems

$$\inf_{Q_k(s)\in\mathscr{RL}_\infty^{[k]}} \|G_k(s) - Q_k(s)\|_\infty, \; k = 1, 2, \ldots, n. \tag{19}$$

When $k = 1$, the optimization problem reduces to (15). As k increases, the enlargement of the feasible set of the kth optimization problem is more than enough to compensate for the additional anti-stable pole in $G_k(s)$. As a result, for $k = 1, 2, \ldots, n-1$, we have

$$\inf_{Q_k(s)\in\mathscr{RL}_\infty^{[k]}} \|G_k(s) - Q_k(s)\|_\infty \geq \inf_{Q_{k+1}(s)\in\mathscr{RL}_\infty^{[k+1]}} \|G_{k+1}(s) - Q_{k+1}(s)\|_\infty. \tag{20}$$

The following lemma shows that the series of optimization problems above admit analytic solutions.

Lemma 1 *Given $G_k(s)$ as defined in* (18), we have

$$\inf_{Q_k(s)\in\mathscr{RL}_\infty^{[k]}} \|G_k(s) - Q_k(s)\|_\infty = |\lambda_k|,$$

where the infimum is achieved when $Q_k(s) = 0$.

Proof Since $\{p_k(s), q_k(s), e_k(s)\}$ is a solution of (17) with respect to λ_k, it follows that

$$G_k(s) = \frac{b(-s)p_k(s) - a(-s)q_k(s)}{d(-s)e_k(s)} = \lambda_k \frac{d(s)e_k(-s)}{d(-s)e_k(s)}.$$

Consequently,

$$\inf_{Q_k(s)\in\mathscr{RL}_\infty^{[k]}} \|G_k(s) - Q_k(s)\|_\infty \leq \|G_k(s) - 0\|_\infty = \left\| \lambda_k \frac{d(s)e_k(-s)}{d(-s)e_k(s)} \right\|_\infty = |\lambda_k|, \tag{21}$$

where the fact that

$$\frac{d(s)e_k(-s)}{d(-s)e_k(s)}$$

is all-pass has been used. We show below

$$\inf_{Q_k(s)\in\mathscr{RL}_\infty^{[k]}} \|G_k(s) - Q_k(s)\|_\infty \geq |\lambda_k|,$$

from which it follows that equality is achieved when $Q_k(s) = 0$.

Let $e_k(s) = f_k(s)g_k(-s)$, where $f_k(s)$ and $g_k(s)$ are stable polynomials and $\deg g_k(s) = k-1$. For an arbitrary transfer function $Q_k(s) \in \mathscr{RL}_\infty^{[k]}$, we can write

$$Q_k(s) = \frac{h_k(s)}{h_k(-s)}\tilde{Q}_k(s),$$

where $h_k(s) \in \mathscr{P}_{k-1}$ is stable and $\tilde{Q}_k(s) \in \mathscr{RH}_\infty$. Define

$$U_k(s) := \frac{f_k(s)h_k(-s)}{d(s)},$$

which is an element in $\mathscr{H}_2$ since $d(s)$ is stable and

$$\deg f_k(s) + \deg h_k(s) \leq \deg e_k(s) < \deg d(s).$$

Observe that

$$G_k(s)U_k(s) = \lambda_k \frac{h_k(-s)f_k(-s)g_k(s)}{d(-s)g_k(-s)} \in \mathscr{RH}_2^\perp$$

and

$$\|G_k(s)U_k(s)\|_2 = \\ |\lambda_k| \left\| \frac{h_k(-s)f_k(-s)g_k(s)}{d(-s)g_k(-s)} \right\|_2 = |\lambda_k| \left\| \frac{h_k(-s)f_k(-s)}{d(-s)} \right\|_2 = |\lambda_k| \|U_k(s)\|_2.$$

On the other hand, we have

$$Q_k(s)U_k(s) = \frac{f_k(s)h_k(s)}{d(s)}\tilde{Q}_k(s) \in \mathscr{H}_2.$$

Therefore,

$$\begin{aligned} \|G_k(s) - Q_k(s)\|_\infty &\geq \frac{\|G_k(s)U_k(s) - Q_k(s)U_k(s)\|_2}{\|U_k(s)\|_2} \\ &= \sqrt{\frac{\|G_k(s)U_k(s)\|_2^2}{\|U_k(s)\|_2^2} + \frac{\|Q_k(s)U_k(s)\|_2^2}{\|U_k(s)\|_2^2}} \\ &= \sqrt{|\lambda_k|^2 + \frac{\|Q_k(s)U_k(s)\|_2^2}{\|U_k(s)\|_2^2}} \\ &\geq |\lambda_k|, \end{aligned}$$

as required. □

Below we provide an answer to Question (i). Lying at the core of the answer is the matrix $\boldsymbol{H}$ defined in **Step 2** of Algorithm 1. First recall the notation introduced at the start of Sect. 3.1.

Lemma 2 *A triplet of polynomials* $\{p(s), q(s), e(s)\}$ *is a solution to the partial pole placement problem in Definition 1 with respect to a* $\lambda \in \mathbb{R}$ *if, and only if,* $\{\boldsymbol{p}, \boldsymbol{q}, \boldsymbol{e}\}$

satisfies

$$\boldsymbol{H}\boldsymbol{e} = (-1)^n \lambda \boldsymbol{e}$$

and

$$\begin{bmatrix} \boldsymbol{p} \\ \boldsymbol{q} \end{bmatrix} = \begin{bmatrix} \boldsymbol{L}_a & \boldsymbol{L}_b \\ \boldsymbol{U}_a & \boldsymbol{U}_b \end{bmatrix}^{-1} \begin{bmatrix} \boldsymbol{L}_d \\ \boldsymbol{U}_d \end{bmatrix} \boldsymbol{e}, \tag{22}$$

where

$$\boldsymbol{H} := \boldsymbol{J}\boldsymbol{L}_d^{-1}\boldsymbol{J}\begin{bmatrix} \boldsymbol{L}_b\boldsymbol{J} & -\boldsymbol{L}_a\boldsymbol{J} \end{bmatrix} \begin{bmatrix} \boldsymbol{L}_a & \boldsymbol{L}_b \\ \boldsymbol{U}_a & \boldsymbol{U}_b \end{bmatrix}^{-1} \begin{bmatrix} \boldsymbol{L}_d \\ \boldsymbol{U}_d \end{bmatrix}.$$

Proof Equating the coefficients of the polynomials on both sides of the equations in (17) results in the following two systems of linear equations:

$$\begin{aligned} \begin{bmatrix} \boldsymbol{L}_a & \boldsymbol{L}_b \\ \boldsymbol{U}_a & \boldsymbol{U}_b \end{bmatrix} \begin{bmatrix} \boldsymbol{p} \\ \boldsymbol{q} \end{bmatrix} &= \begin{bmatrix} \boldsymbol{L}_d \\ \boldsymbol{U}_d \end{bmatrix} \boldsymbol{e}, \\ \begin{bmatrix} (-1)^n\boldsymbol{J} & \\ & \boldsymbol{J} \end{bmatrix} \begin{bmatrix} \boldsymbol{L}_b & -\boldsymbol{L}_a \\ \boldsymbol{U}_b & -\boldsymbol{U}_a \end{bmatrix} \begin{bmatrix} \boldsymbol{J} & \\ & \boldsymbol{J} \end{bmatrix} \begin{bmatrix} \boldsymbol{p} \\ \boldsymbol{q} \end{bmatrix} &= \lambda \begin{bmatrix} \boldsymbol{L}_d \\ \boldsymbol{U}_d \end{bmatrix} \boldsymbol{J}\boldsymbol{e}. \end{aligned} \tag{23}$$

Since $a(s)$ and $b(s)$ are coprime, the matrix

$$\begin{bmatrix} \boldsymbol{L}_a & \boldsymbol{L}_b \\ \boldsymbol{U}_a & \boldsymbol{U}_b \end{bmatrix}$$

is invertible. Thus, from (23), we obtain (22) and

$$\begin{bmatrix} \boldsymbol{J} & \\ & (-1)^n\boldsymbol{J} \end{bmatrix} \begin{bmatrix} \boldsymbol{L}_b & -\boldsymbol{L}_a \\ \boldsymbol{U}_b & -\boldsymbol{U}_a \end{bmatrix} \begin{bmatrix} \boldsymbol{J} & \\ & \boldsymbol{J} \end{bmatrix} \begin{bmatrix} \boldsymbol{L}_a & \boldsymbol{L}_b \\ \boldsymbol{U}_a & \boldsymbol{U}_b \end{bmatrix}^{-1} \begin{bmatrix} \boldsymbol{L}_d \\ \boldsymbol{U}_d \end{bmatrix} \boldsymbol{e} = (-1)^n\lambda \begin{bmatrix} \boldsymbol{L}_d \\ \boldsymbol{U}_d \end{bmatrix} \boldsymbol{J}\boldsymbol{e}. \tag{24}$$

There are a total of $2n$ linear equations involving the elements of $\boldsymbol{e}$ in (24). By equating the coefficients in the spectral factorization in (7), one may show that the first n equations are identical to the last n ones after some algebraic manipulations. Hence it suffices to consider the first n rows of (24):

$$\boldsymbol{J}\begin{bmatrix} \boldsymbol{L}_b\boldsymbol{J} & -\boldsymbol{L}_a\boldsymbol{J} \end{bmatrix} \begin{bmatrix} \boldsymbol{L}_a & \boldsymbol{L}_b \\ \boldsymbol{U}_a & \boldsymbol{U}_b \end{bmatrix}^{-1} \begin{bmatrix} \boldsymbol{L}_d \\ \boldsymbol{U}_d \end{bmatrix} \boldsymbol{e} = (-1)^n\lambda\boldsymbol{L}_d\boldsymbol{J}\boldsymbol{e}.$$

Since $\deg d(s) = n$, or equivalently, $d_0 \neq 0$, $\boldsymbol{L}_d$ is invertible. Left-multiplying both sides of the equation above by $\boldsymbol{J}\boldsymbol{L}_d^{-1}$ yields

$$\boldsymbol{H}\boldsymbol{e} = (-1)^n\lambda\boldsymbol{e}.$$

In other words, $\{(-1)^n\lambda, \boldsymbol{e}\}$ is an eigenpair of $\boldsymbol{H}$. □

4.3 Optimally Robust Controller

Based on the previous development, we state the main result as follows.

Theorem 2 *The controller* $C_{\rm opt}(s)$ *defined in* Algorithm 1 *is optimally robust, in the sense that*

$$C_{\rm opt}(s) = \mathop{\arg\min}_{C(s)\in\mathscr{S}(P(s))} \left\| \frac{P(-s)-C(s)}{1+P(s)C(s)} \right\|_\infty = \mathop{\arg\min}_{C(s)\in\mathscr{S}(P(s))} \|P(s)\,\#\,C(s)\|_\infty.$$

Moreover, the optimal robustness margin is

$$\alpha(P(s)) = \frac{1}{\sqrt{1+\rho^2(\boldsymbol{H})}}.$$

Proof By Lemma 1 and (20), we have for $k = 1, 2, \ldots, n-1$,

$$\begin{aligned} |\lambda_k| = \|G_k(s)\|_\infty &= \inf_{Q_k(s)\in\mathscr{RL}_\infty^{[k]}} \|G_k(s) - Q_k(s)\|_\infty \\ &\geq \inf_{Q_{k+1}(s)\in\mathscr{RL}_\infty^{[k+1]}} \|G_{k+1}(s) - Q_{k+1}(s)\|_\infty = \|G_{k+1}(s)\|_\infty = |\lambda_{k+1}|, \end{aligned}$$

where

$$G_k(s) := \frac{b(-s)p_k(s) - a(-s)q_k(s)}{d(-s)e_k(s)}$$

is as defined in (18), $\{p_k(s), q_k(s), e_k(s)\}$ is a solution to the partial pole placement problem in Definition 1 with respect to λ_k and $e_k(s)$ has exactly $k-1$ anti-stable roots. Furthermore, by Lemma 2, each $|\lambda_k|$ is the magnitude of an eigenvalue of $\boldsymbol{H}$ defined in **Step 2** of Algorithm 1. In particular, $|\lambda_1| = \rho(\boldsymbol{H})$. Since $e_1(s)$ is stable, it follows from (15) that

$$\begin{aligned} \gamma(P(s)) &= \inf_{C(s)\in\mathscr{S}(P(s))} \left\| \frac{P(-s)-C(s)}{1+P(s)C(s)} \right\|_\infty \\ &= \inf_{Q(s)\in\mathscr{RH}_\infty} \|G_1(s) - Q(s)\|_\infty \\ &= \inf_{Q_1(s)\in\mathscr{RL}_\infty^{[1]}} \|G_1(s) - Q_1(s)\|_\infty \\ &= \|G_1(s)\|_\infty \\ &= |\lambda_1| = \rho(\boldsymbol{H}). \end{aligned}$$

Therefore,

$$\alpha(P(s)) = \frac{1}{\sqrt{1+\gamma^2(P(s))}} = \frac{1}{\sqrt{1+\rho^2(\boldsymbol{H})}},$$

and again by Lemma 2, **Steps 3 and 4** of Algorithm 1 yield a triplet of polynomials $\{p(s), q(s), e(s)\}$ satisfying $p(s) = p_1(s), q(s) = q_1(s)$, and $e(s) = e_1(s)$, whereby

$$G_1(s) := \frac{b(-s)p_1(s) - a(-s)q_1(s)}{d(-s)e_1(s)} = \frac{b(-s)p(s) - a(-s)q(s)}{d(-s)e(s)}.$$

By defining $C_{\text{opt}}(s) := \dfrac{q(s)}{p(s)}$, we have

$$\begin{aligned}\|G_1(s)\|_\infty &= \left\| \frac{b(-s)p(s) - a(-s)q(s)}{d(-s)e(s)} \right\|_\infty \\ &= \left\| \frac{d(-s)a(s)}{a(-s)d(s)} \frac{b(-s)p(s) - a(-s)q(s)}{d(-s)e(s)} \right\|_\infty \\ &= \left\| \frac{P(-s) - C_{\text{opt}}(s)}{1 + P(s)C_{\text{opt}}(s)} \right\|_\infty,\end{aligned}$$

where the first equality in (17) has been used. In other words, the $C_{\text{opt}}(s)$ obtained in Algorithm 1 is optimally robust. □

5 Case Study: Control of a USUAL Inverted Pendulum

The inverted pendulum system has been one of the most popular control education apparatus since the 1950s. The system has been widely utilized for verifying the effectiveness of stabilizing algorithms due to its unstable and under-actuated properties. The system mimics the human stick balancing game: balancing a long stick upward on our finger tip. In the game, our fingers move in a horizontal plane and the stick can fall in all directions. In this scenario, the state of the stick is observed directly by the human vision. Unlike the game, the inverted pendulum in this case study, whose cart moves linearly along a straight rail and rod can only fail either to the front or to the back of the cart, is a simplified one-dimensional version of the game. See Fig. 5 for an illustration.

The human stick balancing game motivates us to reconsider certain issues of stabilizing the inverted pendulum. Conventionally, the inverted pendulum is equipped with two sensors, i.e., the cart position sensor and the rod angle sensor. The feedback stabilization of the inverted pendulum is usually done by using the measured two sensor outputs. If we recall the stick balancing game, it is highly unlikely that our eyes are focused on the finger position and the stick angle simultaneously. What do we really look at when we try to balance a stick using our hand? The researchers now tend to believe that the player in the game looks at the top end of the stick when

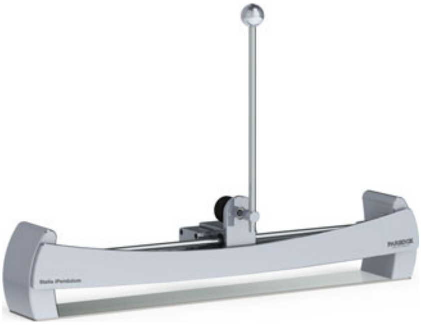


Fig. 5 A real inverted pendulum

the player tries to move the fingers [4, 14, 22]. To mimic this human behavior, we utilize a single position sensor, which measures the horizontal position of the upper tip of the rod, to achieve the stabilization of the inverted pendulum. Clearly, such an inverted pendulum system would be not only under-actuated, but also under-sensed. The control of such a system is much more challenging compared with controlling an inverted pendulum by using two measured outputs.

In this section, the output feedback stabilization of an under-sensed and under-actuated linear (USUAL) inverted pendulum, which has only one position sensor and one force actuator, is investigated. We successfully stabilize this USUAL inverted pendulum without sophisticated tuning. The optimally robust controller introduced earlier is demonstrated to be effective. To the best of our knowledge, this is the first successful experimental study on controlling a linear inverted pendulum by using a single position sensor measurement.

5.1 System Model

As shown in Fig. 6, a standard linear inverted pendulum consists of a cart and a rod. The cart, with a mass M_c, slides on a stainless shaft and is equipped with a linear motor. The rod, attached with a small ball, is mounted on the cart. The axis of rotation of the rod is perpendicular to the direction of the motion of the cart. The rod, of length L, has an evenly distributed mass M_p, and the small ball with a mass M_b can be regarded as a point mass. The system has two degrees of freedom. One is from the horizontal motion of the cart, and the other is from the rotational motion of the rod on the plane. Nevertheless, only the horizontal motion of the cart is actuated by the force $f(t)$ applied to the cart, and only the horizontal position of the tip of the rod $z(t)$ is measured by a single position sensor. Consequently, Fig. 6 indeed shows the schematic diagram of the USUAL inverted pendulum.

The differential equation model [22, Sections 2.10 and 3.9] of the USUAL inverted pendulum is given by

Fig. 6 A schematic diagram of the USUAL inverted pendulum with input $f(t)$ and output $z(t)$

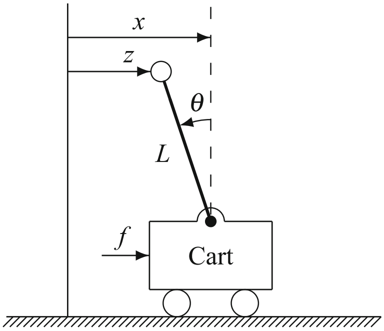


Table 1 Parameters of the USUAL inverted pendulum in our experimental setup

Parameter	Value
Mass of rod (M_p)	0.07 kg
Mass of the cart (M_c)	1.42 kg
Mass of the ball (M_b)	0.05 kg
Gravitational acceleration (g)	9.8 m/s^2
Length of the rod (L)	0.335 m

$$\begin{aligned} f(t) &= M_1\ddot{x}(t) - M_2L\ddot{\theta}(t)\cos\theta(t) + M_2L\dot{\theta}^2(t)\sin\theta(t), \\ 0 &= M_3L\ddot{\theta}(t) - M_2\ddot{x}(t)\cos\theta(t) - M_2g\sin\theta(t), \\ z(t) &= x(t) - L\sin\theta(t) \end{aligned} \tag{25}$$

where $f(t)$ is the system input, $z(t)$ is the system output, $x(t)$ is the cart position, $\theta(t)$ is the pendulum angle, g is the gravitational acceleration, $M_1 = M_p + M_c + M_b$, $M_2 = M_p/2 + M_b$, and $M_3 = M_p/3 + M_b$ are three constant coefficients of the practical system. The system given in (25) is highly nonlinear. Our control objective is to stabilize the rod around its upward direction, which is an unstable equilibrium point. Linearizing the system around the equilibrium point $x(t) = 0, \dot{x}(t) = 0, \theta(t) = 0, \dot{\theta}(t) = 0$ yields

$$\begin{aligned} f(t) &= M_1\ddot{x}(t) - M_2L\ddot{\theta}(t), \\ 0 &= M_3L\ddot{\theta}(t) - M_2\ddot{x}(t) - M_2g\theta(t), \\ z(t) &= x(t) - L\theta(t) \end{aligned}$$

together with the transfer function $P(s)$ from $F(s)$ to $Z(s)$ as

$$P(s) = \frac{(M_3/M_2 - 1)\,Ls^2 - g}{M_1s^2\left[(M_3/M_2 - M_2/M_1)\,Ls^2 - g\right]}. \tag{26}$$

Plugging the actual values of the parameters given in Table 1 into (26) results in

$$P(s) = \frac{-0.1104s^2 - 22.52}{s^2\left(s^2 - 36.23\right)}. \tag{27}$$

This is a highly unstable system with poles at 0, 0, ±6.019 and zeros at $\pm j14.28$. It is impossible to stabilize this system by using PD or PID control.

5.2 Optimally Robust Stabilization

In real applications, before we apply the design algorithm to $P(s)$, generally, loop-shaping for the plant $P(s)$ is carried out to help improve the control performance. The purpose of shaping the plant is to balance the system input and output possibly using a frequency-dependent weighting function. In our experimental setup, a simple weighting constant is demonstrated to be sufficient.

Specifically, for the USUAL inverted pendulum, first multiply the original plant $P(s)$ given in (27) by the simplest weighting function, i.e., a constant W, to form a new plant $\hat{P}(s) = WP(s)$, then carry out the optimally robust stabilization algorithm to obtain the resulting controller $\hat{C}(s)$. Note that the loop transfer function is given by

$$L(s) = \hat{P}(s)\hat{C}(s) = P(s)W\hat{C}(s) = P(s)C(s)$$

where $C(s) = W\hat{C}(s)$. In other words, to guarantee the same loop transfer function for the original $P(s)$, we need to absorb the weighting constant W into the controller $C(s)$. As a result, $C(s)$ is the optimally robust controller that we use in reality for the original plant $P(s)$.

The shaping constant W should be carefully tuned in actual applications. In our real USUAL inverted pendulum setup, we find that a large range of W is applicable, and we set $W = 400$ for our experiment.

In the following, we make use of the main algorithm to design an optimally robust stabilizing controller for the real USUAL inverted pendulum.

Example 3 [USUAL Inverted Pendulum] Consider the shaped plant

$$\hat{P}(s) = WP(s) = \frac{400(-0.1104s^2 - 23.52)}{s^4 - 36.23s^2}.$$

Following Algorithm 1, we try to find the optimal $\hat{C}(s)$ such that $\|\hat{P}(s)\,\#\,\hat{C}(s)\|_\infty$ is minimized.

1. (*Spectral factorization*)

$$(s^4 - 36.23s^2)^2 + 400^2(-0.1104s^2 - 23.52)^2 = d(-s)d(s).$$

This yields $d(s) = s^4 + 27.21s^3 + 334.0s^2 + 2335s + 9409$.

2. (*Matrix computation*) We can compute that

$$\boldsymbol{H} = \begin{bmatrix} -2.073 & -0.3017 & -2.962 \times 10^{-2} & -1.476 \times 10^{-3} \\ -45.48 & -6.136 & -0.5042 & -1.055 \times 10^{-2} \\ -425.9 & -55.28 & -3.755 & 1.115 \times 10^{-2} \\ -2839 & -278.7 & -13.89 & 0.3075 \end{bmatrix}.$$

3. (*Eigen-computation*) The four eigenvalues of $\boldsymbol{H}$ are -13.20, 1.977, -0.5058 and 7.574×10^{-2}. The one with the largest magnitude is $\rho(\boldsymbol{H}) = 13.20$ and its corresponding eigenvector is

$$\boldsymbol{e} = \begin{bmatrix} -1.315 \times 10^{-3} & -2.416 \times 10^{-2} & -0.1994 & -0.9796 \end{bmatrix}^T.$$

This gives

$$e(s) = \begin{bmatrix} s^3 & s^2 & s & 1 \end{bmatrix} \boldsymbol{e} = -1.315 \times 10^{-3} s^3 - 2.416 \times 10^{-2} s^2 - 0.1994 s - 0.9796.$$

4. (*Pole placement*) The pole placement equation follows that

$$(s^4 - 36.23 s^2) p(s) + 400(-0.1104 s^2 - 23.52) q(s) = d(s) e(s).$$

Solving the above equation gives

$$p(s) = s^3 + 45.57 s^2 + 438.6 s + 9834,$$

$$q(s) = -13.20 s^3 - 116.8 s^2 - 336.5 s - 744.8.$$

Therefore, the pole placement controller is given by

$$\hat{C}(s) = \frac{q(s)}{p(s)} = \frac{-13.20 s^3 - 116.8 s^2 - 336.5 s - 744.8}{s^3 + 45.57 s^2 + 438.6 s + 9834}.$$

Absorbing W yields the desired optimally robust controller in practical use

$$C(s) = W\hat{C}(s) = \frac{400(-13.20 s^3 - 116.8 s^2 - 336.5 s - 744.8)}{s^3 + 45.57 s^2 + 438.6 s + 9834}. \quad (28)$$

5. (*Optimal robustness margin computation*) We have

$$\alpha(\hat{P}(s)) = 7.552 \times 10^{-2}.$$

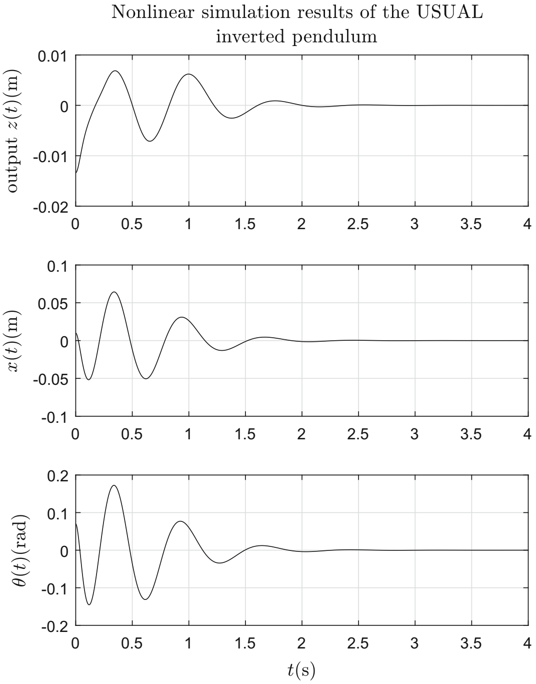


Fig. 7 The nonlinear simulation of the USUAL inverted pendulum: stabilization results of the output $z(t)$, cart position $x(t)$ and pendulum angle $\theta(t)$ with the initial conditions $x(t) = 0.01$ m, $\dot{x}(t) = 0.002$ m/s, $\theta(t) = 4\pi/180$ rad, $\dot{\theta}(t) = 0.5\pi/180$ rad/s

5.3 Simulation and Experimental Results

To verify the effectiveness of the design algorithm, we first show the simulation results for the nonlinear model given in (25) together with the optimally robust controller $C(s)$ designed in (28). Given the initial conditions $x(t) = 0.01$ m, $\dot{x}(t) = 0.002$ m/s, $\theta(t) = 4\pi/180$ rad, $\dot{\theta}(t) = 0.5\pi/180$ rad/s, the stabilization simulation results are shown in Fig. 7.

The closed-loop system starts with the initial conditions, and the simulation results show that the output $z(t)$, the cart position $x(t)$ and pendulum angle $\theta(t)$ converge to zero quickly when $t > 2$ s. This validates the effectiveness of the designed controller from a theoretical perspective.

In the following, we implement the optimally robust stabilizing controller $C(s)$ given in (28) to the real USUAL inverted pendulum. The experiment is carried out as follows. In the beginning, we show the stabilized behaviors of $z(t)$, $x(t)$, and $\theta(t)$ of the USUAL inverted pendulum. Then, we excite the system by knocking the pendulum gently on the top as performance testing. In the end, the behaviors of $z(t)$, $x(t)$ and $\theta(t)$ of the system against the knock are presented.

The real-time experimental data of three variables $z(t)$, $x(t)$, and $\theta(t)$ together with the performance testing are illustrated by Fig. 8. When $t < 9.7$ s, the output $z(t)$ is within a small range $[-0.04 \text{ m},\ 0.02 \text{ m}]$, and from $x(t)$ and $\theta(t)$, we know that the real USUAL inverted pendulum is indeed stabilized. Moreover, both $\theta(t)$

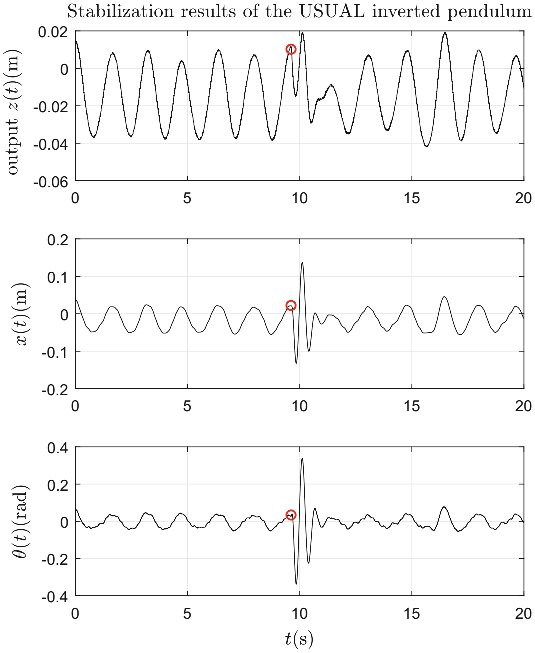


Fig. 8 The stabilization of the real USUAL inverted pendulum: results of the output $z(t)$, cart position $x(t)$ and pendulum angle $\theta(t)$. The circles represent the rough time $t = 9.7$ s when we excite the system by hitting the pendulum on the top

and $x(t)$ vary within a small range. The results indicate the closed-loop system is running with satisfactory stabilized behaviors.

In order to test the system performance, we excite the system by hitting the pendulum lightly on the top when the designed controller is in operation. As shown in Fig. 8, the circles represent the rough time $t = 9.7$ s when we excite the system. The results show that both $x(t)$ and $\theta(t)$ restore quickly to their stabilized behaviors. In the meantime, $z(t)$ is almost free of the impact of the hit since $z(t)$ is the controlled output. This validates the effectiveness of the designed controller from a practical point of view.

By simply shaping the USUAL inverted pendulum with a constant to balance the system input and output, the optimally robust controller can be implemented successfully without further complicated tuning. We conclude that the optimally robust control is demonstrated to be effective in the control of the USUAL inverted pendulum.

6 Conclusion

To characterize system uncertainties of different types and from multiple sources, we have proposed a special uncertainty model, namely, the uncertainty quartet. The uncertainty quartet combines and generalizes several commonly adopted uncertainty models, such as the additive, the multiplicative, the relative, and the feedback uncertainties. In correspondence with the uncertainty quartet, a robust stability condition was derived, resulting in a robust stability margin in terms of the Gang of Four transfer matrix. An optimally robust controller, maximizing the robust stability margin, was obtained through a proposed polynomial approach. This approach involves only basic matricial and polynomial manipulations. Moreover, the mathematical tools used in developing this polynomial approach are also rudimentary, e.g., the matrix analysis and basic $\mathscr{H}_\infty$ control theory. The clarity and simplicity of the polynomial approach may be beneficial to the popularization of the robust control theory for engineering applications.

The optimally robust controller was demonstrated to be effective by the case study on the USUAL inverted pendulum, a highly nonlinear and unstable single-input single-output system. This system is commonly seen in laboratories and familiar to most of people in the field of control. It is nontrivial to control such a system with simple methods, such as, PID control. As a result, the USUAL inverted pendulum may be regarded as a benchmark to validate the effectiveness of control methods in practice. For the purpose of education, the control of this system may serve as a qualifying test for control system designers and engineers.

Acknowledgements Useful discussions with Dr. Wei Chen on the writing of this chapter are gratefully acknowledged. This work was supported in part by the Research Grants Council of Hong Kong Special Administrative Region, China, under Project 16201115 and Theme-Based Research Scheme T23-701/14-N, and in part by the People's Government of Pengjiang District, Jiangmen, Guangdong, China.

References

1. Åström, K. J. and Murray, R. M. (2008). *Feedback Systems: An Introduction for Scientists and Engineers*. Princeton, NJ: Princeton University Press.
2. Doyle, J. C., Francis, B. A., and Tannenbaum, A. R. (1990). *Feedback Control Theory*. London: Macmillan Publishers Ltd.
3. Doyle, J. C., Glover, K., Khargonekar, P. P., and Francis, B. A. (1989). State-space solutions to standard $\mathscr{H}_2$ and $\mathscr{H}_\infty$ control problems. *IEEE Trans. Automat. Contr.*, 34(8):831–847.
4. Doyle, J. C., Nakahira, Y., Leong, Y. P., Jenson, E., Dai, A., Ho, D., and Matni, N. (2016). Teaching control theory in high school. In *Proc. 55th IEEE Conf. on Decision and Contr. (CDC)*, pages 5925–5949.
5. Fuhrmann, P. A. (2012). *A Polynomial Approach to Linear Algebra*. New York, NY: Springer Science & Business Media.
6. Georgiou, T. T. and Smith, M. C. (1990). Optimal robustness in the gap metric. *IEEE Trans. Automat. Contr.*, 35(6):673–686.
7. Georgiou, T. T. and Smith, M. C. (1997). Robustness analysis of nonlinear feedback systems: an input-output approach. *IEEE Trans. Automat. Contr.,*, 42(9):1200–1221.
8. Gu, G. and Qiu, L. (1998). Connection of multiplicative/relative perturbation in coprime factors and gap metric uncertainty. *Automatica*, 34(5):603–607.
9. Gu, G. and Qiu, L. (2011). A two-port approach to networked feedback stabilization. In *Proc. 50th IEEE Conf. on Decision and Contr. and European Contr. Conf. (CDC-ECC)*, pages 2387–2392.

10. Halsey, K. M. and Glover, K. (2005). Analysis and synthesis of nested feedback systems. *IEEE Trans. Automat. Contr.*, 50(7):984–996.
11. Kailath, T. (1980). *Linear Systems*. NJ: Prentice-Hall Englewood Cliffs.
12. Kanno, M. (2003). *Guaranteed Accuracy Computations in Systems and Control*. PhD thesis, University of Cambridge.
13. Lanzon, A. and Papageorgiou, G. (2009). Distance measures for uncertain linear systems: A general theory. *IEEE Trans. Automat. Contr.,* 54(7):1532–1547.
14. Leong, Y. P. and Doyle, J. C. (2016). Understanding robust control theory via stick balancing. In *Proc. 55th IEEE Conf. on Decision and Contr. (CDC)*, pages 1508–1514.
15. Liang, Y. and Qiu, L. (2009). A polynomial solution to an $\mathscr{H}_\infty$ robust stabilization problem. In *Proc. 7th IEEE Asian Contr. Conf. (ASCC)*, pages 642–647.
16. Liu, K.-Z. and Yao, Y. (2016). *Robust Control: Theory and Applications*. Singapore: John Wiley & Sons.
17. McFarlane, D. and Glover, K. (1990). *Robust Controller Design Using Normalized Coprime Factor Plant Descriptions*. New York: Springer-Verlag.
18. Nehari, Z. (1957). On bounded bilinear forms. *Annals of Mathematics*, pages 153–162.
19. Petersen, I. R. and Tempo, R. (2014). Robust control of uncertain systems: Classical results and recent developments. *Automatica*, 50(5):1315–1335.
20. Qiu, L. and Davison, E. J. (1992a). Feedback stability under simultaneous gap metric uncertainties in plant and controller. *Syst. Contr. Lett.*, 18(1):9–22.
21. Qiu, L. and Davison, E. J. (1992b). Pointwise gap metrics on transfer matrices. *IEEE Trans. Automat. Contr.*, 37(6):741–758.
22. Qiu, L. and Zhou, K. (2009). *Introduction to Feedback Control*. Upper Saddle River, NJ: Prentice Hall.
23. Vidyasagar, M. (1985). *Control System Synthesis: A Factorization Approach*. Cambridge, MA: M.I.T. Press.
24. Vinnicombe, G. (1993). Frequency domain uncertainty and the graph topology. *IEEE Trans. Automat. Contr.*, 38(9):1371–1383.
25. Vinnicombe, G. (2000). *Uncertainty and Feedback: $\mathscr{H}_\infty$ Loop-shaping and the ν-gap Metric*. Singapore: World Scientific.
26. Youla, D. C., Bongiorno, J. J., and Jabr, H. A. (1976). Modern Wiener-Hopf design of optimal controllers: part I. *IEEE Trans. Automat. Contr.*, 21(1):3–13.
27. Zames, G. and El-sakkary, A. K. (1980). Unstable systems and feedback: the gap metric. In *Proc. 16th Allerton Conf.*, pages 380–385.
28. Zhao, D. and Qiu, L. (2016). Networked robust stabilization with simultaneous uncertainties in plant, controller and communication channels. In *Proc. 55th IEEE Conf. on Decision and Contr. (CDC)*, pages 2376–2381.
29. Zhou, K. and Doyle, J. C. (1998). *Essentials of Robust Control*. Upper Saddle River, NJ: Prentice Hall.

Part II
Randomization and Probabilistic Methods

Randomization in Robustness, Estimation, and Optimization

B. Polyak and P. Shcherbakov

Abstract This is an attempt to discuss the following question: When is a random choice better than a deterministic one? That is, if we have an original deterministic setup, is it wise to exploit randomization methods for its solution? There exist numerous situations where the positive answer is obvious; e.g., stochastic strategies in games, randomization in experiment design, randomization of inputs in identification. Another type of problems where such approach works successfully relates to treating uncertainty, see Tempo R., Calafiore G., Dabbene F., "Randomized algorithms for analysis and control of uncertain systems," Springer, New York, 2013. We will try to focus on several research directions including optimization problems with no uncertainty and compare known deterministic methods with their stochastic counterparts such as random descent, various versions of Monte Carlo etc., for convex and global optimization. We survey some recent results in the field and ascertain that the situation can be very different.

1 Introduction

The use of a random mechanism to solve a problem in a deterministic setup is very common not only in mathematics but much beyond formal framework. One can remember that random decisions were performed in ancient times, and the procedure of drawing lots was very common. Moreover, political events such as election of governing officers in Athens were randomized. Nowadays, elements of randomization are often exploited in sport competitions to equalize the chances of the participants. Very important role of random mixing in medical and biological experiments is of no doubt.

B. Polyak (✉) · P. Shcherbakov
Institute of Control Science, Russian Academy of Sciences, Moscow, Russia
e-mail: polyak@ipu.ru

P. Shcherbakov
e-mail: cavour118@mail.ru

© Springer Nature Switzerland AG 2018

T. Başar (ed.), *Uncertainty in Complex Networked Systems*,
Systems & Control: Foundations & Applications,
https://doi.org/10.1007/978-3-030-04630-9_5

Probably one of the first application of stochastic approach in mathematics is the theory of mixed strategies in zero-sum games by John von Neumann. Here the role of randomization is to make secret a strategy of the player against the competitor. Approximately at the same time Fisher proposed to apply mixed strategies in experiment design; here their role was different. A real breakthrough was the invention of the Monte Carlo methods by Ulam, Metropolis, von Neumann and Teller [43], and the ideas of random sampling became very popular in modeling and numerical analysis. Thus randomization methods found numerous applications in various fields of research; to survey all of them does not seem to be realistic. In this chapter we restrict ourselves to some problems related to estimation, robustness, and continuous optimization. The typical question to be analysed is as follows. Given a deterministic problem (say, unconstrained smooth optimization), how randomization ideas can be exploited for its solution and are randomized methods true competitors with deterministic ones? We will see that situation differs in various domains of interest.

The role of Roberto Tempo in progress of this approach can not be overestimated. His research since 2000 was mostly dedicated to randomization methods in control, robustness, and optimization, see the monograph [77]. In the present chapter, we continue this line of research, but also we address the directions which have little intersections with [77] as well with other monographs and surveys on randomization [14, 28, 29].

Due to the wide spectrum of the problems under consideration, we are forced to provide really brief presentation of the problems; the references do not pretend to be complete. However we have tried to emphasize the pioneering works and surveying publications.

2 Uncertainty and Robustness

Mathematical models for systems and control are often unsatisfactory due to the incompleteness of the parameter data. For instance, the ideas of off-line optimal control can only be applied to real systems if all the parameters, exogenous perturbations, state equations, etc., are known precisely. Moreover, feedback control also requires a detailed information which is not available in most cases. For example, to drive a car with four-wheel control, the controller should be aware of the total weight, location of the center of gravity, weather conditions and highway properties as well as many other data which may not be known. In that respect, even such a relatively simple real-life system can be considered a *complex* one; in such circumstances, control under uncertainty is a highly important issue.

In this section we consider the *parametric uncertainty*; other types of uncertainty can be treated within more general models of robustness.

There are numerous tools to check robustness under parametric uncertainty; below we focus on randomized methods. This line of research goes back to pioneering papers by Stengel and Ray [74]. Within this approach, the uncertain parameters are assumed to have random rather than deterministic nature; for instance, they are assumed to be uniformly distributed over the respective intervals of uncertainty. Next, an acceptable tolerance ε, say $\varepsilon = 0.01$ is specified, and a check is performed if the resulting random family (of polynomials, matrices, transfer functions) is stable with probability no less than $(1-\varepsilon)$; see [77] for a comprehensive exposition of such a *randomized approach to robustness*.

In many of the NP-hard robustness problems, such a reformulation often leads to exact or approximate solutions. Moreover, the randomized approach has several attractive properties even in the situations where the deterministic solution is available. Indeed, the deterministic statements of robustness problems are minimax, hence, the answer is dictated by the "worst" element in the family, whereas these critical values of the uncertain parameters are rather unlikely to occur. Therefore, by neglecting a small risk of violation of the desired property (say, stability), the admissible domains of variation of the parameters may be considerably extended. This effect is known as the *probabilistic enhancement of robustness margins*; it is particularly tangible for the large number of the parameters. Another attractive property of the randomized approach is its low computational complexity which only slowly grows with increase in the number of uncertain parameters.

We illustrate some of these concepts and effects.

2.1 Volume of Violation and Approximate Feasibility

We consider robustness problems for systems described in terms of a design vector $x \in X \subseteq \mathbb{R}^n$ and a real uncertain parameter vector $q \in Q \subset \mathbb{R}^\ell$, where Q is a box. For such systems, the objective is to select $x \in X$ such that a given continuous performance specification

$$f(x,q) \leq 0 \tag{1}$$

is satisfied for all $q \in Q$. When such a design vector x exists, the triple (f, X, Q) is said to be *robustly feasible*.

In a number of situations, robust feasibility of $f(x,q) \leq 0$ is guaranteed if and only if $f(x,q^i) \leq 0$ for each of the vertices q^i of the ℓ-dimensional box Q, and we use the term *vertexization*. A typical example of a vertexization is the quadratic stability problem for the system with state space matrix $A(q) = A_0 + \sum_{i=1}^{\ell} A_i q_i$, where $A_i \in \mathbb{R}^{n\times n}$ are fixed and known, and the uncertainty parameter vector $q \in Q$. The goal is to find a symmetric candidate Lyapunov matrix $P = P(x)$ with entries $x_i \in \mathbb{R}$ viewed as the design variables, such that $P(x) \succ 0$ and the linear matrix inequality (LMI) $A^\top(q)P(x) + P(x)A(q) \prec 0$ holds for all $q \in Q$ (throughout the text, the signs $\succ$ and $\prec$ denote the positive and negative sign-definiteness of a symmetric matrix). Hence, with

$$f(x,q) = \lambda_{\max}\Big(A^\top(q)P(x) + P(x)A(q)\Big),$$

this strict feasibility design problem in x is reducible to the vertices q^i of Q. That is, the satisfaction of the Lyapunov inequality above for all $q \in Q$ is equivalent to $A^\top(q^i)P(x) + P(x)A(q^i) \prec 0$ for $i = 1, 2, \dots, N$. However, since $N = 2^\ell$, we see that the computational task can easily get out of hand. For example, with five states and ten uncertain parameters, the resulting LMI is of size greater than 5000 × 5000.

As an alternative to the computational burden associated with vertexization, it is often possible to introduce an *overbounding function* in such a way as to enable *convex programming* in order to test for robust feasibility; also, see Sect. 4 for a different approach to solving the feasibility problem for LMIs. Note also that a reduction to checking the vertices is rather an exception and is considered here for illustrative purposes, while the *overbounding techniques* may be applied to much broader classes of systems.

Specifically, given x, introduce the associated *violation set*

$$Q_{bad}(x) \doteq \{q \in Q : \ f(x, q) > 0\}$$

and estimate from above its volume. Equivalently, assuming that the uncertainty vector q is random, uniformly distributed over Q, we estimate from above the probability of violation for the performance specification.

In [4, 5], a computationally modest method for finding such overbounding functions is proposed and numerical examples are presented.

More delicate constructions are also described in [4, 5], where the notion of approximate feasibility is introduced. Namely, the triple (f, X, Q) is said to be *approximately feasible* if the following condition holds: Given any $\varepsilon > 0$, there exists some $x^\varepsilon \in X$ such that

$$\mathbf{Vol}\Big(\{q \in Q : \ f(x^\varepsilon, q) > 0\}\Big) < \varepsilon,$$

where $\mathbf{Vol}(\cdot)$ stands for the volume of a set. For such ε, x^ε is called an *ε-approximate solver*. So, instead of guaranteeing satisfaction of $f(x, q) \le 0$ for all $q \in Q$, we seek solution vectors x with associated violation set having volume less than any arbitrarily small prespecified level $\varepsilon > 0$.

We present a formal result on approximate feasibility in general terms; the details can be found in [4, 5]. First, we consider so-called *homogenizable* in x functions $f(x, q)$ and use their homogenized versions denoted by $f^+(x_0, x, q)$. In [4, 5] this requirement was shown to be not very much restrictive, covering quite a large class of functions. Next, the notion of *approximate feasibility indicator* (AFI) is introduced; in a sense, it is a convex generalization of the classical indicator function. For instance, a "natural" type of AFI is the exponential one, $\phi(\zeta) = e^\zeta$.

In the theorem to follow, the approximate feasibility indicator $\phi(\zeta)$ is used with argument $\zeta = f^+(x_0, x, q)$ in the determination of approximate feasibility.

Theorem 1 ([4, 5]) *Given the continuous homogenizable performance specification function $f(x, q)$, $X = \mathbb{R}^n$ and an approximate feasibility indicator $\phi(\cdot)$, define*

$$\Phi(x_0, x) \doteq \int_Q \phi(f^+(x_0, x, q))\mathrm{d}q$$

and

$$\Phi^* \doteq \inf_{x_0>0,x} \Phi(x_0, x).$$

Then the following holds:
(i) $\Phi^* = 0$ *implies approximate feasibility of* (f, X, Q)*;*
(ii) *For any* $x_0 > 0$ *and* $x \in \mathbb{R}^n$*,*

$$\mathbf{Vol}\left(Q_{bad}\left(\frac{x}{x_0}\right)\right) \le \Phi(x_0, x).$$

A similar idea of overbounding was presented in [6]. Multivariable polynomials $f(x)$ with parameter vector x restricted to a hypercube $X \in \mathbb{R}^n$ were considered, and the objective was to check the robust positivity of $f(x)$, i.e., to determine if $f(x) > 0$ for all $x \in X$. Again, instead of solving the original NP-hard problem, the authors proposed a straightforward procedure for the computation of an upper bound on the volume of violation by computing a respective *dilation integral* that depends on the degree k of a certain auxiliary polynomial, followed by a convex minimization in one scalar parameter. By increasing the degree k, the authors obtain a sequence of upper bounds ε_k which are shown to be “sharp” in the sense that they converge to zero whenever the positivity requirement is satisfied. Notably, that this dilation integral method applies to a general *polynomial dependence* on the variables.

2.2 Probabilistic Predictor

In the discussion above, the stochastic nature of the uncertain parameters was somewhat hidden; we just evaluated the bad portion of the uncertainty box. Assume now that the originally deterministic parameters are randomly distributed over the given uncertainty set Q. Then it seems natural to sample the uncertainty set Q and arrive at conclusions on the probability of robustness. In the control-related literature, these ideas have been first formulated in [74]; also see [3].

Together with numerous advantages of this approach, it also suffers serious drawbacks. First, it is usually desired to have any closed-form estimates of the robustness margin, rather than to rely on the results of simulations; moreover, in practical applications, such a simulation is often prohibited. Next, the sample size that guarantees high reliability of the result may happen to be rather large [76, 77], hence, simulations may be very time-consuming. On top of that, sampling in accordance with one or another distribution over a given set may be highly nontrivial [30, 57]. Finally, the results of Monte Carlo simulation heavily depend on the probabilistic distribution adopted and may lead to overly optimistic estimates of the robustness margin; the correct choice of the distribution is a nontrivial problem [2].

In this section, assuming the uniform distribution of the uncertain parameters over $q \in \gamma Q$, where $Q \subset \mathbb{R}^\ell$ is the uncertainty set and $\gamma \in \mathbb{R}$ is the scaling factor, we characterize the probability of stability of a system and evaluate the *probabilistic stability margin*

$$\gamma_{\max} := \sup\bigl\{\gamma : \mathsf{Prob}\{\text{the system is stable for } q \in \gamma Q\} \geq p\bigr\}, \quad p \in (0,\ 1),$$

where $\mathsf{Prob}(\cdot)$ denotes the probability of an event.

Without getting deep into the details, we describe the idea of the probabilistic approach to robustness as applied to polynomial families.

Since the early 1990s, numerous *graphical tests* for robust stability proved themselves to be efficient; these are based on the famous zero exclusion principle, which is formulated next. Consider the family of polynomials $p(s,q)$ which depend on the vector q of uncertain parameters confined to the connected set $Q \subset \mathbb{R}^\ell$. For a fixed $s = j\omega$, the set

$$V(\omega) \doteq \{p(j\omega, q) : q \in Q\}$$

is referred to as the *value set* of the family $p(s,q)$; it is the 2D image of Q under the mapping $p(j\omega, \cdot)$. Let the polynomial $p(s, q^0)$ be stable for some $q^0 \in Q$; then, for robust stability, the following condition is necessary and sufficient:

$$0 \notin V(\omega) \quad \text{for all} \quad \omega \in [0,\ \infty). \tag{2}$$

To exploit this result, one has to efficiently construct the set $V(\omega)$ and check condition (2). This is doable in a number of simple cases; however, for more or less involved dependence of $p(s,q)$ on q, this approach cannot be applied, since no closed-form description of the boundary of the value set is available, and checking condition (2) is complicated by the nonconvexity of $V\omega)$.

Taking the probabilistic point of view and letting q be random, uniformly distributed over Q, we consider the two-dimensional random variable

$$z_\omega = [\mathrm{Re}\, p(j\omega, q);\ \ \mathrm{Im}\, p(j\omega, q)]$$

and construct its confidence domain

$$V_{1-\varepsilon}(\omega) : \ \ \mathsf{Prob}\bigl\{z_\omega \in V_{1-\varepsilon}(\omega)\bigr\} \geq 1 - \varepsilon, \qquad \varepsilon > 0 \ \text{ is small.}$$

This set is referred to as a $100(1-\varepsilon)\%$ *probabilistic predictor* of the value set $V(\omega)$. The condition (2) now has to be checked for the predictor, rather than for the value set, hence, evaluating the probability of stability of the uncertain polynomial family.

Often, the construction of the predictor can be accomplished via using the central limiting behavior of the random vector z_ω. Indeed, if $p(s,q)$ depends affinely on q, and the q_is are mutually independent, the random vector z_ω is represented by the sum of independent random vectors, and if the number ℓ of the parameters is large enough, then, under the general assumptions on $p_i(s)$ it is well described by the two-dimensional Gaussian random vector with mean $\overline{z}_\omega = \mathsf{E} z_\omega$ and the covariance matrix $S = \mathsf{Cov}\, z_\omega$. Therefore, $V(\omega)$ may be approximated by the confidence ellipse

$$\mathcal{E}_\nu(\omega) \doteq \left\{ z \in \mathbb{R}^2 \colon \; (z - \overline{z}_\omega)^\top S^{-1} (z - \overline{z}_\omega) \le \nu^2 \right\},$$

where ν specifies the confidence level. In other words, if p_ν is the associated confidence probability, then for a given ω we have

$$\mathsf{Prob}\left\{ p(j\omega, q) \in \mathcal{E}_\nu(\omega) \right\} \approx \mathsf{p}_\nu = 1 - e^{-\nu^2/2}.$$

In a number of situations, it is possible to obtain a precise nonasymptotic distribution of the random vector z_ω and, respectively, a precise description of the probabilistic predictor.

We illustrate these ideas via the problem of robust stability of uncertain delay systems; i.e., those described by uncertain quasipolynomials, see [58]. In this case, the generic value set has a very complicated geometry; application of the zero exclusion principle is hardly possible, and we lean on the probabilistic approach.

Consider the delay system specified by the characteristic quasipolynomial

$$h(s, a, \tau) = a_0 + a_1 s + s^2 + 2se^{-\tau_1 s} + e^{-\tau_2 s}, \tag{3}$$

$$|a_0| \le \gamma, \quad |a_1| \le \gamma, \quad |1 - \tau_1| \le \gamma, \quad |2 - \tau_2| \le \gamma.$$

Here, both the coefficients and the delays are subject to interval uncertainty. The nominal system $h(s) = s^2 + 2se^{-s} + e^{-2s}$ is stable, $\max_k \mathrm{Re} s_k = -0.3181$, where s_k are the roots of the quasipolynomial $h(s)$ (the roots of $h(s)$ are the values of the Lambert function $W(x)e^{W(x)} = x$ at the point $x = -1$). For this system, the value of the radius of robustness cannot be found exactly, but the estimate $0.01 < \gamma_{\max} < 0.05$ is known from the literature. For the confidence level $\nu = 3$, the probabilistic approach gives $\gamma_\nu = 0.0275$, so that it fits well the deterministic estimate.

To illustrate, for a set of frequencies in $0 \le \omega \le 2$, Fig. 1a depicts the confidence ellipses $\mathcal{E}_\nu(\omega)$, $\nu = 3$, for the uncertainty range $\gamma = 0.0275$. Also, presented are the frequency responses $h(j\omega, q)$ for a number of sampled values of the uncertainty $q = (a_0, a_1, \delta\tau_1, \delta\tau_2)$ in the box $|q_i| \le \gamma$. The curves are seen to remain inside the "corridor" defined by the confidence ellipses. Figure 1b depicts the confidence ellipse $\mathcal{E}_\nu(\omega)$ for a "typical" $\omega = 1.3113$ together with sampled points $h(j\omega, q)$; the predictor is seen to approximate nicely the value set.

Probabilistic robustness techniques can be effectively exploited for robust control design [12, 39, 53, 54, 61, 77, 78].

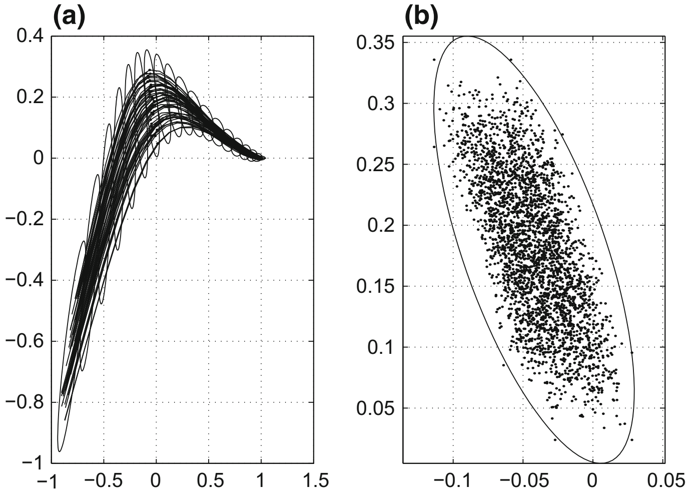


Fig. 1 **a** The plot of $h(j\omega, q)$ and confidence ellipses $\mathcal{E}_\nu(\omega)$, $\nu = 3$ for system (3). **b** Probabilistic predictor of the value set for $\omega = 1.3113$

2.3 *Probabilistic Enhancement of Robustness Margins*

It is important to note that, even for the values of $\mathsf{p}_\nu = \mathsf{p}$ close to unity, the ellipse $\mathcal{E}_\nu(\omega)$ is often considerably smaller than the value set $\mathbf{Vol}(\omega)$. Let us make use of the probabilistic counterpart of the zero exclusion principle (the origin does not belong to $\mathcal{E}_\nu(\omega)$ for all ω) and evaluate the *probabilistic stability margin* defined as

$$\gamma_{\mathsf{p}} \doteq \sup\bigl\{\gamma : 0 \notin \mathcal{E}_\nu(\omega) \text{ for all } \omega \in [0, \infty)\bigr\}.$$

It then usually happens that $\gamma_{\mathsf{p}} \gg \gamma_{\max}$, where $\gamma_{\max}$ is the deterministic stability margin. Hence, the uncertainty range may be considerably enlarged at the expense of neglecting low-probability events. This phenomenon is referred to as *probabilistic enhancement of classical robustness margins* [40]. Moreover, in accordance with the central limit theorem, this enlargement gets bigger as the number of uncertainties grow, and it is this case which is most problematic for deterministic methods. At the same time, the computational burden of probabilistic methods does not depend on the dimension of the vector of uncertain parameters. Indeed, putting the precise description of the value set aside, we make use of an approximation of it, which is defined by the two-dimensional covariance matrix.

We illustrate use of the probabilistic approach to the assessment of such an enhancement via the case of matrix uncertainty. Specifically, let us consider the uncertain matrix family

$$A = A_0 + \Delta, \qquad \Delta \in \gamma Q, \tag{4}$$

where $A_0 \in \mathbb{R}^{n\times n}$ is a known, Hurwitz stable matrix and Δ is its bounded perturbation confined to the ball in the Frobenius norm $\gamma Q = \{\Delta \in \mathbb{R}^{n\times n} : \|\Delta\|_F \le \gamma\}$; the goal

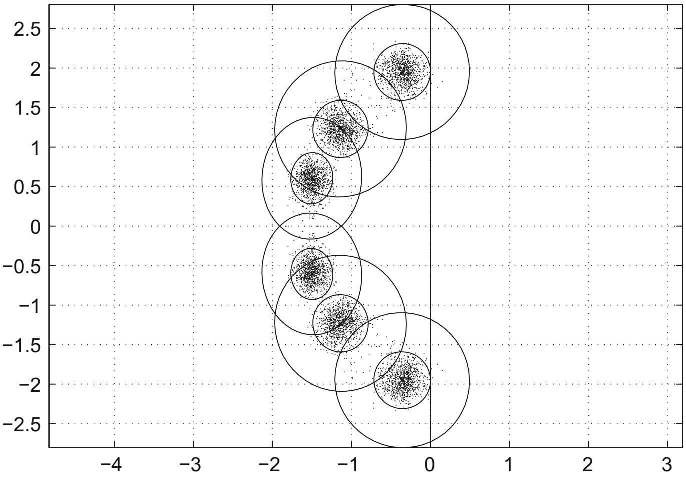


Fig. 2 The pseudospectrum of A_0, its linear approximation, and the probabilistic predictor

is to estimate the robust stability margin of A_0. To this end, we provide an approximate description of the *pseudospectrum* of A (4), the set of the eigenvalues of A for all admissible values of the uncertainty Δ.

For a generic case of simple complex eigenvalues $\lambda = \lambda(A_0) \in \mathbb{C}$, the perturbed eigenvalue $\lambda(A_0 + \Delta)$ is well described by the linear approximation

$$\tilde{\lambda} \;=\; \lambda + Rq, \quad R \in \mathbb{R}^{2\times\ell}, \quad \ell = n^2,$$

provided that γ is small enough. Here, $q \in \mathbb{R}^\ell$ is the vectorization of Δ, and the matrix R is defined by the left and right eigenvectors of λ.

It can be shown that, as q sweeps the ball γQ, the 2D-vector [Re $\tilde{\lambda}$, Im$\tilde{\lambda}$] sweeps the ellipse

$$\mathcal{E} := \left\{x \in \mathbb{R}^2 : \left(S^{-1}(x-\lambda),\, x-\lambda\right) \leq \gamma^2\right\}, \qquad S := RR^\top.$$

Now, assuming that the uncertainty q is random, uniformly distributed over the ball γQ, and specifying a confidence probability p, we make use of Lemma 2 (see Sect. 5.1) to shape an ellipsoidal probabilistic predictor $\mathcal{E}_\mathsf{p}$ of the ellipse $\mathcal{E}$.

A schematic illustration of the ideas above is given next. For a 6×6 stable matrix having $\ell = 36$ uncertain entries, quite an accurate upper bound $\overline{\gamma} = 0.3947$ of the stability margin can be found.

Let us specify $\mathsf{p} = 0.99$; then the constructions above yield $\hat{\gamma}_p = 0.7352$ as an estimate of the value of the probabilistic margin. In other words, the uncertainty radius is almost doubled, at the expense of admitting the 1%-probability of instability. To confirm these conclusions, we performed straightforward Monte Carlo simulations for $\gamma = \hat{\gamma}_p$, which resulted in the sampled probability of stability $p_{MC} = 0.9989$ (from a sample of $40,000$ points q). Figure 2 depicts the linear approximation of the pseudospectrum of A (larger ellipses) and its ellipsoidal probabilistic predictors (smaller ellipses, rightmost of them touch the imaginary axis), along with sampled values of the pseudospectrum.

Other examples relate to the probability of a polynomial with coefficients in a cube to be stable [46] and to the generation of random stable polynomials [69].

3 Randomization in Estimation

Usual assumptions on the noise in linear regression problems are that it is a sequence of independent zero-mean random variables (vectors). However in practical situations these assumptions are often violated which may strongly affect the performance of standard estimators. Therefore it is important to examine the possibility to estimate the regression parameters under minimal assumptions on the noise. It may appear surprising that the regression parameters can be consistently estimated in the case of biased, correlated and even nonrandom noise. However, it can be done under certain conditions when the inputs (regressors) are random. We consider a linear regression model

$$y_n = x_n^\top \theta + \xi_n \tag{5}$$

with the parameter vector $\theta \in \mathbb{R}^N$ to be estimated from the observations y_n, x_n, $n = 1, 2, \ldots$ It is assumed that the inputs x_n are zero-mean random vectors independent of the noise ξ_k. This assumption ensures "good" properties of estimators under extremely mild restrictions on the noise. The idea of using random inputs to eliminate bias was put forward by Fisher [22] as the randomization principle in the design of experiments. Besides settings of design type where regressors are randomized by the experimenter, random inputs arise in many applications of identification, filtering, recognition, etc. Having these applications in mind, we use the terms "inputs," "outputs," etc., rather than those traditional to the regression analysis (say, "regressors").

We follow the results in [25], see also [27]. Let us formulate the rigorous assumptions on the data for the regression problem (5).

(A) the inputs x_n are represented by a sequence of independent, identically distributed random vectors with symmetric distribution function, zero mean value $\mathsf{E}x_n = 0$, positive-definite covariance matrix $\mathsf{E}x_n x_n^\top = B \succ 0$, and a finite fourth moment $\mathsf{E}\|x_n\|^4 < \infty$; moreover, x_n is independent of $\{\xi_0, \xi_1, \ldots, \xi_n\}$.

(B) the noise ξ_n is mean-square bounded: $\mathsf{E}|\xi_n|^2 \leq \sigma^2$.

Theorem 2 *Under the assumptions above, the least square estimate θ_n of the true parameter θ is mean-square consistent, and the rate of convergence is given by*

$$\mathsf{E}(\theta_n - \theta)(\theta_n - \theta)^\top = \frac{\sigma^2}{n} B^{-1} + o\Big(\frac{1}{n}\Big). \tag{6}$$

If the inputs are deterministic and $B = \lim_{n\to\infty} \frac{1}{n} \sum_{i=1}^{\infty} x_i x_i^\top$, one can obtain a similar estimate for the least squares algorithm under the standard assumption that the noise is zero mean, $\mathsf{E}\xi_n = 0$. The principal contribution of Theorem 2 is the removal of this restrictive assumption.

A result similar to Theorem 2 holds true for the Polyak–Ruppert online averaging algorithm [64]:

$$\theta_n = \theta_{n-1} + \gamma_n x_n (y_n - \theta_{n-1}^\top x_n) \tag{7}$$

$$\hat{\theta}_n = (1 - n^{-1})\hat{\theta}_{n-1} + n^{-1}\theta_n, \tag{8}$$

where

$$\gamma_n/\gamma_{n+1} = 1 + o(\gamma_n); \tag{9}$$

for instance, $\gamma_n = 1/n^r$ for some $0 < r < 1$. It is proved in [25] that estimate (6) is true under assumptions (A), (B) for no-zero-mean noise.

The fruitful idea of randomizing the inputs is exploited in numerous problems of identification, control, optimization in the monographs [28, 29]. These results confirm the general conclusion: Randomization enables for a considerable relaxation of the standard assumptions on the noise. In Sect. 5, we focus on such approaches to stochastic optimization problems.

4 Feasibility

The problem of solving convex inequalities (also known as convex feasibility problem) is one of the basic problems of numerical analysis. It arises in numerous applications, including statistics, parameter estimation, pattern recognition, image restoration, tomography and many others, see, e.g., monographs and surveys [7, 15, 17] and references therein. Particular cases of the problem relate to solving of linear inequalities and to finding a common point of convex sets. The specific feature of some applications is a huge number of inequalities to be solved, while the dimensionality of the variables is moderate, see, e.g., the examples of applied problems below. Under these circumstances many known numerical methods are inappropriate. For instance, finding the most violated inequality may be a hard task; dual methods also cannot be applied due to large number of dual variables.

In this survey we follow mainly the paper [56] and focus on simple iterative methods which are applicable to the case of very large (and even infinite) number of inequalities. They are based on projection-like algorithms, originated in the works [1, 31, 36, 44]. There are many versions of such algorithms; they can be either parallel or non-parallel (row-action); in the latter case the order of projections is usually chosen as cyclical one or the-most-violated one, see [7, 15, 17]. All these methods are well suited for the finite (and not too large) number of constraints. The novelty of the method under consideration is its random nature, which allows to treat large-dimensional- and infinite-dimensional cases. Although the idea of exploiting stochastic algorithms for optimization problems with continua of constraints has been known in the literature [34, 51, 80], it led to much more complicated calculations than the proposed method. Another feature of the method is its finite termination property—under the strong feasibility assumption a solution is found after a finite number of steps with probability one. The version of a projection method for linear inequalities with this property has been proposed first by V.A. Yakubovich [81]. Below we survey the main results from [56]. Related contributions can be found in [13, 61].

Consider the general convex feasibility problem: find a point x in the set

$$C = \{x \in X\colon\ f(x,q) \le 0 \quad \forall q \in Q\}. \tag{10}$$

Here $X \subset \mathbb{R}^n$ is a convex closed set, $f(x,q)$ is convex in x for all $q \in Q$, where Q is an arbitrary set of indices (finite or infinite). Note that this formulation is similar to the robust feasibility problem (1) considered above. However, instead of finding its approximate solution or evaluating the volume of violation, we are aimed at finding a solution satisfying *all* inequalities, but using *randomized methods*.

Particular cases of problem (10) are:

1. *Finite number of inequalities*: $Q = \{1, \ldots m\}$.
2. *Semi-infinite problem*: $Q = [0, T] \subset \mathbb{R}^1$.
3. *Finding a common point of convex sets*: $f(x,q) = \mathrm{dist}(x, C_q) = \min_{y \in C_q} \|x - y\|$, where the sets $C_q := \{x \in X\colon\ f(x,q) \le 0 \ \text{ for a } \ q \in Q\} \subset \mathbb{R}^n$ are closed and convex and $C = \cap_{q\in Q} C_q$. Here, $\|x\|$ denotes the Euclidean norm of a vector.
4. *Linear inequalities:* $f(x,q) = a(q)^\top x - b(q)$.

We assume that a subgradient $\partial_x f(x,q)$ is available at any point $x \in X$ for all $q \in Q$ (we mean an arbitrary subgradient if the set of them is not a singleton).

The algorithm has the following structure. At the kth iteration, we generate randomly $q_k \in Q$; we assume that the q_k's are independent and identically distributed (i.i.d.) samples from some probabilistic distribution p_q on Q. Two key assumptions are adopted.

Assumption 1 (*strong feasibility*). The set C is nonempty and contains an interior point

$$\exists\, x^* \in C :\ \|x - x^*\| \le r \implies x \in C.$$

Here, $r > 0$ is a constant which is assumed to be known.

Assumption 2 (*distinguishability of feasible and infeasible points*). For $x \in X \setminus C$, the probability of generating a violated inequality is not vanishing:

$$\mathsf{Prob}\{f(x,q) > 0\} > 0.$$

This is the only assumption on the probability distribution p_q. For instance, if Q is a finite set and each element in Q is generated with nonzero probability, then Assumption 2 holds. The *feasibility algorithm* is then formulated as follows:

Algorithm 1: Given an initial point $x_0 \in X$, proceed as follows:

$$x_{k+1} = \pi_X\big(x_k - \lambda_k \partial_x f(x_k, q_k)\big), \tag{11}$$

$$\lambda_k = \begin{cases} \dfrac{f(x_k, q_k) + r\|\partial_x f(x_k, q_k)\|}{\|\partial_x f(x_k, q_k)\|^2} & \text{if } f(x_k, q_k) > 0; \\[2ex] 0 & \text{otherwise.} \end{cases} \tag{12}$$

Here, π_X is a projection operator onto X; that is, $\|x - \pi_X(x)\| = \mathrm{dist}(x, X)$. Hence, at every step, the calculation of a subgradient is performed just for one inequality, which is randomly chosen among all inequalities Q. Note that the value of r (the radius of a ball in the feasible set) is used in the algorithm; its modification for r unknown will be presented later. To explain the choice of the step-size λ_k in the algorithm, we consider the two particular cases.

1. *Linear inequalities:* $f(x, q) = a(q)^\top x - b(q)$, $X = \mathbb{R}^n$.
   Then we have $\partial_x f(x_k, q_k) = a_k$, where $f(x_k, q_k) = a_k^\top x_k - b_k$ and $a_k = a(q_k)$, $b_k = b(q_k)$, so that the algorithm takes the form

   $$x_{k+1} = x_k - \frac{(a_k^\top x_k - b_k)_+ + r\|a_k\|}{\|a_k\|^2} a_k$$

   for $(a_k^\top x_k - b_k)_+ \neq 0$, otherwise $x_{k+1} = x_k$; here, $c_+ = \max\{0, c\}$. For $r = 0$, the method coincides with the projection method for solving linear inequalities by Agmon–Motzkin–Shoenberg [1, 44].
2. *Common point of convex sets:* $f(x, q) = \mathrm{dist}(x, C_q)$, $C = \cap_{q \in Q} C_q$, $X = \mathbb{R}^n$.
   Then we have $\partial_x f(x_k, q_k) = \big(x_k - \pi_k(x_k)\big)/\rho_k$, where π_k denotes the projection onto the set $C_k = C_{q_k}$ and $\rho_k = \|x_k - \pi_k(x_k)\|$. The algorithm takes the form

   $$x_{k+1} = \pi_k(x_k) + \frac{r}{\varrho_k}\big(\pi_k(x_k) - x_k\big),$$

   provided that $x_k \notin C_k$; otherwise $x_{k+1} = x_k$. We conclude that, for $r = 0$, each iteration of the algorithm is the same as for the projection method for finding the intersection of convex sets [7, 31].

Having this in mind, the rule for selecting the step-size λ_k has a very natural explanation. Denote by y_{k+1} the point which is generated via the same formula as x_{k+1}, but with $r = 0$; assume also $X = \mathbb{R}^n$. Then, for the case of linear inequalities, y_{k+1} is the projection of x_k onto the half-space $\big\{x\colon\ a_k^\top x - b_k \leq 0\big\}$. Similarly, if we deal with finding a common point of convex sets, y_{k+1} is the projection of x_k onto the set C_k. It is easy to show that $\|x_{k+1} - y_{k+1}\| = r$. Thus the step in the algorithm is an (additively) over-relaxed projection; we perform an extra step (of length r) inside the current feasible set.

The idea of additive over-relaxation is due to V.A. Yakubovich who applied such a method to linear inequalities [81]. In the papers mentioned above, the order of sorting out the inequalities was either cyclic or the-most-violated one was taken, in contrast with the random order in the proposed algorithm.

Now we formulate the main result on the convergence of the algorithm.

Theorem 3 *Under Assumptions 1, 2, Algorithm 1 finds a feasible point in a finite number of iterations with probability one, i.e., with probability one there exists N such that $x_N \in C$ and $x_k = x_N$ for all $k \geq N$.*

We now illustrate how the general algorithm can be adapted to two particular important cases.

1. *Linear Matrix Inequalities* are one of the most powerful tools for model formulation in various fields of systems and control, see [10]. There exist well-developed techniques for solving such inequalities as well as for solving optimization problems subject to such inequalities (Semidefinite Programming, SDP). However in a number of applications (for instance, in robust stabilization and control), the number of LMIs is extremely large or even infinite, and such problems are beyond the applicability of the standard LMI tools. Let us cast these problems in the framework of the approach proposed above.

The space $\mathbb{S}_m$ of $m \times m$ symmetric real matrices equipped with the scalar product $< A, B >= \mathrm{tr}\, AB$ and the Frobenius norm, becomes a Hilbert space ($\mathrm{tr}(\cdot)$ denotes the trace of a matrix). Then we can define the projection A_+ of a matrix A onto the cone of positive semidefinite matrices. This projection can be found in explicit form. Indeed, if $A = RDR^\top$, $R^{-1} = R^\top$, is the eigenvector–eigenvalue decomposition of A and $D = \mathrm{diag}\,(d_1, \dots, d_m)$, then

$$A_+ = RD_+R^\top, \tag{13}$$

where $D_+ = \mathrm{diag}\,(d_1^+, \dots, d_m^+)$ and $d_i^+ = \max\{0, d_i\}$.

Linear matrix inequality is the expression of the form

$$A(x) = A_0 + \sum_{i=1}^{n} x_i A_i \preccurlyeq 0,$$

where $A_i \in \mathbb{S}_m, i = 0, 1, \dots, n$, are given matrices and $x = (x_1, \dots, x_n) \in \mathbb{R}^n$ is the vector variable. Another form of LMI was mentioned in Sect. 2; it is reducible to the *canonical form* above.

The general system of LMIs can be written as

$$A(x, q) = A_0(q) + \sum_{i=1}^{n} x_i A_i(q) \preccurlyeq 0 \qquad \forall q \in Q. \tag{14}$$

Here, Q is the set of indices which can be finite or infinite. The problem under consideration is to find an $x \in \mathbb{R}^n$ which satisfies LMIs (14). Our first goal is to convert these LMIs into a system of convex inequalities. For this purpose, introduce the scalar function

$$f(x, q) = \|A_+(x, q)\| \tag{15}$$

where $A(x, q)$ is given by (14) and A_+ is defined in (13).

Lemma 1 *The matrix inequalities* (14) *are equivalent to the scalar inequalities*

$$f(x, q) \leq 0 \qquad \forall q \in Q.$$

The function $f(x, q)$ is convex in x and its subgradient is given by

$$\partial_x f(x, q) = \frac{1}{f(x, q)} \begin{pmatrix} \operatorname{tr} A_1 A_+(x, q) \\ \vdots \\ \operatorname{tr} A_n A_+(x, q) \end{pmatrix}$$

if $f(x, q) > 0$; otherwise $\partial_x f(x, q) = 0$.

Hence, solving linear matrix inequalities can be converted into solving a convex feasibility problem.

2. *Solving linear equations.* This case has some peculiarities—the solution set is either a single point or a linear subspace, so that it never contains an interior point and Algorithm 1 with $r > 0$ does not converge. However it can be applied with $r = 0$; for a deterministic choice of the alternating directions it is precisely the Kaczmarz algorithm [36]. Its randomized version with equal probabilities for all equations has been proposed in [56]; it converges with linear rate. More recently, Strohmer and Vershynin [75] studied this method with the probabilities for choosing the equation $(a_i, x) = b_i$ being proportional to $\|a_i\|^2$. They proved that the rate of convergence depends on the condition number of the matrix A, but not on the number of equations. This result stimulated further research in [15, 16, 20, 26, 41].

5 Optimization

After the invention of the Monte Carlo (MC) paradigm by N. Metropolis and S. Ulam in the late 1940s [43], it has become extremely popular in numerous application areas such as physics, biology, economics, social sciences, and other areas. As far as mathematics is concerned, Monte Carlo methods proved to be exceptionally efficient in the simulation of various probability distributions, numerical integration, estimation of the mean values of the parameters, etc. [37, 67, 77]. More recent version of the approach, *Markov Chain Monte Carlo*, is often referred to as *MCMC revolution* [23]. The salient feature of MC approach to solution of various problems of this sort is that "often," it is dimension-free in the sense that, given N samples, the accuracy of the result does not depend on the dimension of the problem.

On the other hand, applications of the MC paradigm in the area of optimization are not that successful. In this regard, problems of global optimization deserve special attention. As explained in [82] (see beginning of Chapter 1.2), "*In global optimization, randomness can appear in several ways. The main three are: (i) the evaluations of the objective function are corrupted by random errors; (ii) the points x_i are chosen on the base of random rules, and (iii) the assumptions about the objective function are probabilistic.*" Pertinent to the exposition of this paper is only case (ii). Monte Carlo is the simplest, brute force example of randomness-based methods (in [82] it is referred to as "Pure Random Search"). With this method, one samples points uniformly in the feasible domain, computes the values of the objective function, and picks the record value as the output.

Of course, there are dozens of more sophisticated stochastic methods such as multistart, simulated annealing, genetic algorithms, evolutionary algorithms, etc.; e.g., see [24, 35, 52, 70, 72, 82] for an incomplete list of relevant references. However, most of these methods are heuristic in nature; often, they lack rigorous justification, and the computational efficiency is questionable. Moreover, there exist pessimistic results on "insolvability of global optimization problems." This phenomenon has first been observed as early as in the monograph [47] by A. Nemirovskii and D. Yudin, both in the deterministic and stochastic optimization setups (see Theorem, Section 1.6 in [47]). Specifically, the authors of [47] considered the minimax approach to the minimization of the class of Lipschitz functions and proved that, no matter what the optimization method is, it is possible to construct a problem which will require exponential (in the dimension) number of function evaluations. The "same" number of samples is required for the simplest MC method. Similar results can be found in [48], Theorem 1.1.2, where the construction of "bad" problems is exhibited. Below we present another example of such problems (with very simple objective functions, close to linear ones) which are very hard to optimize. Concluding this brief survey, we see that any advanced method of global optimization cannot outperform Monte Carlo when optimizing "bad" functions.

This explains our interest in the MC approach as applied to the optimization setup. In spite of the pessimistic results above, there might be a belief that, if Monte Carlo is applied to a "good" optimization problem (e.g., a convex one), the results would not be so disastrous. Our goal in this section is to blow up these optimistic expectations. We examine the "best" optimization problems (the minimization of a linear function on a ball) and estimate the accuracy of the Monte Carlo method. Unfortunately, the dependence on the dimension remains exponential, and practical solution of these simplest problems via such an approach is impossible for high dimensions.

The second part of the section is devoted to randomized algorithms for convex optimization. The efficiency of such an approach has been discovered recently; it became clear that advanced randomized coordinate descent and similar approaches for distributed optimization are strong competitors to deterministic versions of the methods.

5.1 Direct Monte Carlo in Optimization

In this subsection we show that *straightforward* use of Monte Carlo in optimization, both global and convex is highly inefficient in problems of high dimensions. The material is based on the results in [60].

Global optimization: A pessimistic example. We first present a simple example showing failure of stochastic global optimization methods in high-dimensional spaces. This example is constructed along the lines suggested in [47] (also, see [48], Theorem 1.1.2) and is closely related to one of the central problems discussed below, the minimization of a linear function over a ball in $\mathbb{R}^n$.

Consider an unknown vector $c \in \mathbb{R}^n$, $\|c\| = 1$, and the function

$$f(x) = \min\Big\{99 - c^\top x,\ \big(c^\top x - 99\big)/398\Big\}$$

to be minimized over the Euclidean ball $Q \subset \mathbb{R}^n$ of radius $r = 100$ and centered at the origin. Obviously, the function has one local minimum $x_1 = -100c$, with the function value $f_1 = -0.5$, and one global minimum $x^* = 100c$, with the function value $f^* = -1$. The objective function is Lipschitz with Lipschitz constant equal to 1, and max $f(x) -$ min $f(x) = 1$.

Any standard (not problem-oriented) version of stochastic global search (such as multistart, simulated annealing, etc.) will miss the domain of attraction of the global minimum with probability $1 - V^1/V^0$, where V^0 is the volume of the ball Q, and V^1 is the volume of the set $C = \{x \in Q : c^\top x \geq 99\}$. In other words, the probability of success is equal to

$$\mathsf{Prob} = \frac{V^1}{V^0} = \frac{1}{2} I\Big(\frac{2rh - h^2}{r^2}; \frac{n+1}{2}, \frac{1}{2}\Big),$$

where $I(x; a, b)$ is the regularized incomplete beta function with parameters a and b, and h is the height of the spherical cap C; in this example, $h = 1$. This probability quickly goes to zero as the dimension of the problem grows; say, for $n = 15$, it is of the order of 10^{-15}. Hence, any "advanced" method of global optimization will find the minimum with relative error not less than 50%; moreover, such methods are clearly seen to be no better than a straightforward Monte Carlo sampling. The same is true if our goal is to estimate the minimal value of the function f^* (not the minimum point x^*). Various methods based on ordered statistics of sample values (see Section 2.3 in [82]) fail to reach the set C with high probability, so that the prediction will be close to $f_1 = -0.5$ instead of $f^* = -1$.

Scalar convex optimization: Pessimistic results. Let Q denote the unit Euclidean ball in $\mathbb{R}^n$ and let $\xi^{(i)}\big|_1^N = \big\{\xi^{(1)}, \ldots, \xi^{(N)}\big\}$ be a multisample of size N from the uniform distribution $\xi \sim \mathscr{U}(Q)$.

Given the scalar-valued linear function

$$g(x) = c^\top x, \quad c \in \mathbb{R}^n, \tag{16}$$

defined on Q, estimate its maximum value from the multisample.

More specifically, let η^* be the true maximum of $g(x)$ on Q and let

$$\eta = \max\{g^{(1)}, \dots, g^{(N)}\}, \qquad g^{(i)} = g(\xi^{(i)}), \quad i = 1, \dots, N, \tag{17}$$

be the empirical maximum; we say that η approximates η^* *with accuracy at least* δ if

$$\frac{\eta^* - \eta}{\eta^*} \le \delta.$$

Then the problem is: *Given a probability level* $p \in]0,\ 1[$ *and accuracy* $\delta \in]0,\ 1[$, *determine the minimal length* $N_{\min}$ *of the multisample such that, with probability at least* p, *the accuracy of approximation is at least* δ *(i.e., with high probability, the empirical maximum nicely evaluates the true one).*

The results presented below are based on the following fact established in [59]; it relates to the probability distribution of a specific quadratic function of the random vector uniformly distributed on the Euclidean ball.

Lemma 2 ([59]) *Let the random vector* $\xi \in \mathbb{R}^n$ *be uniformly distributed on the unit Euclidean ball* $Q \subset \mathbb{R}^n$. *Assume that a matrix* $A \in \mathbb{R}^{m\times n}$ *has rank* $m \le n$. *Then the random variable*

$$\rho = \Big((AA^\top)^{-1}A\xi,\ A\xi\Big)$$

has the beta distribution $\mathscr{B}(\frac{m}{2},\ \frac{n-m}{2}+1)$ *with probability density function*

$$f_\rho(x) = \begin{cases} \dfrac{\Gamma(\frac{n}{2}+1)}{\Gamma(\frac{m}{2})\Gamma(\frac{n-m}{2}+1)}\, x^{\frac{m}{2}-1}(1-x)^{\frac{n-m}{2}} & \text{for } 0 \le x \le 1, \\ 0 & \text{otherwise}, \end{cases} \tag{18}$$

where $\Gamma(\cdot)$ *is the Euler gamma function.*

Alternatively, the numerical coefficient in (18) *writes*

$$\frac{\Gamma(\frac{n}{2}+1)}{\Gamma(\frac{m}{2})\Gamma(\frac{n-m}{2}+1)} = 1/B\Big(\frac{m}{2}, \frac{n-m}{2}+1\Big),$$

where $B(\cdot,\cdot)$ *is the beta function.*

We consider the scalar case (16) and discuss first a qualitative result that follows immediately from Lemma 2. Without loss of generality, let $c = (1,\ 0,\ \dots,\ 0)^\top$, so that the function $g(x) = x_1$ takes its values on the segment $[-1,\ 1]$, and the true maximum of $g(x)$ on Q is equal to 1 (respectively, -1 for the minimum) and is attained with $x = c$ (respectively, $x = -c$). Let us compose the random variable

$$\rho = g^2(\xi),$$

which is the squared first component ξ_1 of ξ. By Lemma 2 with $m = 1$ (i.e., $A = c^\top$), for the probability density function (pdf) of ρ we have

$$f_\rho(x) = \frac{\Gamma(\frac{n}{2}+1)}{\Gamma(\frac{1}{2})\Gamma(\frac{n+1}{2})} x^{-\frac{1}{2}}(1-x)^{\frac{n-1}{2}} := \beta_n\, x^{-\frac{1}{2}}(1-x)^{\frac{n-1}{2}}.$$

Straightforward analysis of this function shows that, as dimension grows, the mass of the distribution tends to concentrate closer to the origin, meaning that the random variable (r.v.) ρ is likely to take values which are far from the maximum, equal to unity.

We next state the following rigorous result [60].

Theorem 4 *Let ξ be a random vector uniformly distributed over the unit Euclidean ball $Q \subset \mathbb{R}^n$ and let $g(x) = x_1$, $x \in Q$. Given $p \in]0,\ 1[$ and $\delta \in]0,\ 1[$, the minimal sample size $N_{\min}$ that guarantees, with probability at least p, for the empirical maximum of $g(x)$ to be at least a δ-accurate estimate of the true maximum, is given by*

$$N_{\min} = \frac{\ln(1-p)}{\ln\Big[\frac{1}{2} + \frac{1}{2} I\big((1-\delta)^2;\, \frac{1}{2},\, \frac{n+1}{2}\big)\Big]}, \tag{19}$$

where $I(x; a, b)$ is the regularized incomplete beta function with parameters a and b.

Clearly, a correct notation should be $N_{\min} = \lceil \cdot \rceil$, i.e., rounding toward the next integer; we omit it, but it is implied everywhere in the sequel.

Numerical values of the function $I(x; a, b)$ can be computed via use of the MATLAB routine `betainc`. For example, with the modest values $n = 10$, $\delta = 0.05$, and $p = 0.95$, formula (19) gives $N_{\min} \approx 8.9 \cdot 10^6$, and this quantity grows quickly as the dimension n increases.

Since we are interested in small values of δ, i.e., in x close to unity, a "closed-form" lower bound for $N_{\min}$ can be computed as stated below.

Corollary 1 *In the conditions of Theorem 4*

$$N_{\min} > N_{\text{appr}} = \frac{\ln(1-p)}{\ln\Big[1 - \frac{\beta_n}{n+1}\frac{1}{1-\delta}\big(2\delta - \delta^2\big)^{(n+1)/2}\Big]},$$

where $\beta_n = \frac{\Gamma(\frac{n}{2}+1)}{\Gamma(\frac{1}{2})\Gamma(\frac{n+1}{2})} = 1/B(\frac{1}{2}, \frac{n+1}{2})$.

Further simplification of the lower bound can be obtained

$$N_{\text{appr}} > \widetilde{N}_{\text{appr}} = -\frac{\ln(1-p)}{\sqrt{2\pi(n+1)}\frac{1}{1-\delta}\big(2\delta - \delta^2\big)^{(n+1)/2}}.$$

The lower bounds obtained above are quite accurate; for instance, with $n = 10$, $\delta = 0.05$, and $p = 0.95$, we have $N_{\min} \approx 8.8694 \cdot 10^6$, while $N_{\text{appr}} \approx 8.7972 \cdot 10^6$ and $\widetilde{N}_{\text{appr}} = 8.5998 \cdot 10^6$.

The moral of this subsection is that, for high dimensions, a straightforward use of Monte Carlo sampling cannot be considered as a tool for finding extreme values of a function, even in the convex case.

5.2 Randomized Methods

On the other hand, exploiting randomized methods in different forms can be highly efficient; in many cases they are strong competitors of deterministic algorithms.

Unconstrained minimization. We start with *random search* methods for unconstrained minimization

$$\min f(x), \quad x \in R^n.$$

Probably the first publication relates to the 1960s [42, 65]. The idea was to choose a random direction in the current point and make a step resulting in decrease of the objective function. Rigorous results on convergence of some random search algorithms were obtained in [19]. Nevertheless the practical experiments with similar methods were mostly disappointing, and they did not attract much attention (excluding global optimization, see above). For convex problems the situation has changed recently, when the dimension of problems under consideration became very large (n is of the order 10^6) or when distributed optimization problems arose $(f(x) = \sum_{i=1}^N f_i(x_i),\ x = (x_1, \dots, x_N),\ N$ is large). We survey some results in this direction first.

The basic algorithm of random search can be written as

$$x_{k+1} = x_k - \gamma_k \frac{\hat{f}(x_k + \mu_k u_k) - \hat{f}(x_k)}{\mu_k} u_k, \tag{20}$$

where x_k is a k-th approximation to the solution x^*, u_k is a random vector, γ_k, μ_k are step-sizes, and $\hat{f}(x_k)$ is a measured value of $f(x_k)$; either $\hat{f}(x_k) = f(x_k)$ (deterministic setup) or $\hat{f}(x_k) = f(x_k) + \xi_k$, ξ_k being a random noise (stochastic optimization). Algorithm (20) requires one calculation of the objective function per iteration, its symmetric version

$$x_{k+1} = x_k - \gamma_k \frac{\hat{f}(x_k + \mu_k u_k) - \hat{f}(x_k - \mu_k u_k)}{2\mu_k} u_k, \tag{21}$$

uses two calculations. The strategy of choosing step-sizes depends on smoothness of $f(x)$ and on the presence of errors ξ_k in function evaluation. The following result is adaptation of more general theorems in [62, 63] for C^2 functions.

Theorem 5 *Consider the problem of unconstrained minimization of $f(x)$, where $f(x)$ is strongly convex, twice differentiable, with gradient satisfying the Lipschitz condition. Suppose u_k are random i.i.d. uniformly distributed in the cube $||u||_\infty \le 1$. Noises ξ_k are independent of $u_1, \dots, u_k$ and have bounded second moment $\mathsf{E}|\xi_i|^2 \le \sigma^2$. The step-size satisfies the following conditions: $\gamma_k = a/k$, $\mu_k = \mu/k^4$, a is large enough. Then the iterations x_k in algorithms (20), (21) converge to the minimum point x^* in mean-square and*

$$\mathsf{E}\|x_k - x^*\|^2 = O(1/\sqrt{k}).$$

It is worth mentioning that randomization of directions u_k allows to remove the assumption $\mathsf{E}x_k = 0$, which is standard in stochastic optimization methods [38]; a similar effect for estimation is exhibited in Theorem 2. If compared with the classical Kiefer–Wolfowitz (KW) method, algorithms (20), (21) are less laborious: they require just one or two function evaluations per iteration vs n or $2n$ in the KW-method. On the other hand, asymptotic rate of convergence is the same: $O(1/\sqrt{n})$. More details about convergence, various forms, computational experience of such algorithms can be found in the publications of J. Spall (e.g., [73]); he names the algorithms SPSA (Simultaneous Perturbation Stochastic Approximation). The pioneering research on the algorithms are due to Yu. Ermoliev [21] and H. Kushner [38].

Now we focus on purely deterministic version of problem (5), where measurements of the objective function do not contain errors: $\hat{f}(x_k) = f(x_k)$. As we mentioned above, the interest to such methods grew enormously when very high-dimensional problems became appealing due to such applications as machine learning and neural networks. The interest has been triggered with Yu. Nesterov's paper [49]. Roughly speaking, the approach of [49] is as follows. It is assumed that the Lipschitz constants L_i for partial derivatives $\partial f/\partial x_i$ are known (and they can be easily estimated for quadratic functions). Then, at the kth iteration, the index $i = \alpha$ is chosen with probability proportional to L_i, and new iteration is obtained by changing coordinate $i\alpha$ with step-size $(1/L_\alpha)\partial f/\partial x_\alpha$. Yu. Nesterov provides sharp estimates on the rate of convergence and also presents the accelerated version of the algorithm. These theoretical results supported with intensive numerical experiments for huge-scale problems confirm advantages of the random coordinate descent. This line of research found numerous applications in distributed optimization [9, 45, 66]. The titles of many publications (e.g., recent one [33]) confirm advantages of randomized algorithms.

Randomization techniques are also helpful for minimization of nonsmooth convex functions, when the only data available are the values of the function $f(x)$ at an arbitrary point. The idea of the following algorithm is due to A. Gupal [32], also see [55], Section 6.5.2. In contrast with algorithm (21), we generate a random point $\tilde{x}_k$ in the neighborhood of the current iteration point x_k and then make a step similar to (21) from this point. Thus the algorithm is written as

$$x_{k+1} = x_k - \gamma_k \frac{f(\tilde{x}_k + \mu_k u_k) - f(\tilde{x}_k - \mu_k u_k)}{2\mu_k} u_k, \tag{22}$$

$$\tilde{x}_k = x_k + \alpha_k h_k \tag{23}$$

where u_k, h_k are independent random vectors uniformly distributed in the cube $\|u\|_\infty \leq 1$, while $\alpha_k, \gamma_k, \mu_k$ are scalar step-sizes. It can be seen that randomization step with h_k is equivalent to smoothing of the original function, thus algorithm similar to (21) is applied to the smoothed function. By adjusting the parameters α_k, γ_k, μ_k, we arrive at the convergence result.

Theorem 6 *Let $f(x)$ be convex, and let a unique minimum point x^* exist. Let the step-sizes satisfy the conditions*

$$\sum_{k=1}^{\infty} \gamma_k = \infty,\ \sum_{k=1}^{\infty} \gamma_k^2 \leq \infty,\ \gamma_k/\alpha_k \to 0,\ \mu_k/\alpha_k \to 0,\ \alpha_k \to 0,\ |\alpha_k - \alpha_{k+1}|/\gamma_k \to 0.$$

Then $x_k \to x^*$ with probability one.

This result guarantees convergence of the algorithm to the minimum point. However it does not provide effective strategies for choosing parameters, neither it estimates the rate of convergence. Above-mentioned problems are deeply investigated in [50]. The authors apply Gaussian smoothing technique (i.e., the vectors u_k are Gaussian) and present randomized methods for various classes of functions (smooth and nonsmooth) for different situations (gradient or gradient-free oracles). The versions of the algorithms with the best rate of convergence are indicated.

To conclude, we remind that there exist no-zero-order deterministic methods for minimization of nondifferentiable convex functions, so that randomized methods provide the only option.

Constrained minimization. There are various problem formulations related to randomized methods for optimization in the presence of constraints.

One of them is closely related to feasibility problem (10), but now we are looking to the feasible point which minimizes an objective function

$$\min(c, x) \quad f(x, q) \leq 0 \quad \forall q \in Q. \tag{24}$$

Here we have taken the objective function to be linear without loss of generality. Constraint functions $f(x, q)$ are convex in the variable $x \in \mathbb{R}^n$ for all values of the parameters q. Numerous examples of constraints of this form were discussed in Sect. 4. Such problems are closely related to *robust optimization*, see [8] and Sect. 2. A randomized approach to the problem consists of a random choice of N parameters $q_1, \ldots, q_N$ from the set Q and solving the convex optimization problem with a finite number of constraints

$$\min_{x \in C}(c, x) \quad f(x, q_i) \leq 0 \quad i = 1, \ldots, N. \tag{25}$$

We suppose that this problem can be solved with high accuracy (e.g., if $f(x,q)$ are linear in x, then (25) is LP), and denote the solution by x_N. Such an approach has been proposed in [11]; the authors answer the following question: *How many samples (N) need to be drawn in order to guarantee that the resulting randomized solution violates only a small portion of the constraints?* They assume that there is some probability measure on Q which defines the probability of violation of constraints $V(x)$ for arbitrary x. The main result in [11] states

Theorem 7 $\mathsf{E}\, V(x_N) \le \dfrac{n}{N+1}$.

Of course this result says nothing about the accuracy of the randomized solution (i.e., how close x_N is to the true solution x^* or how small $(c, x_N - x^*)$ is. However, it provides much useful information. Some related results can be found in Sect. 2 above.

Another type of constrained optimization problems reads as

$$\min(c,x), \quad x \in Q, \tag{26}$$

where $Q \subset \mathbb{R}^n$ is a closed bounded set (convex or nonconvex) such that it is hard to solve explicitly the problem above, and projection on Q is also unavailable. Then a possible option is to sample random points in Q and take the best point having the minimal value of the objective function. It is exactly the "direct Monte-Carlo" we have considered in Sect. 2 and found it to be inefficient. However, another approach, based on cutting plane ideas, might be more promising. We assume that a so-called *boundary oracle* is available, that is for an $x \in Q$ and $y \in \mathbb{R}^n$, the quantities

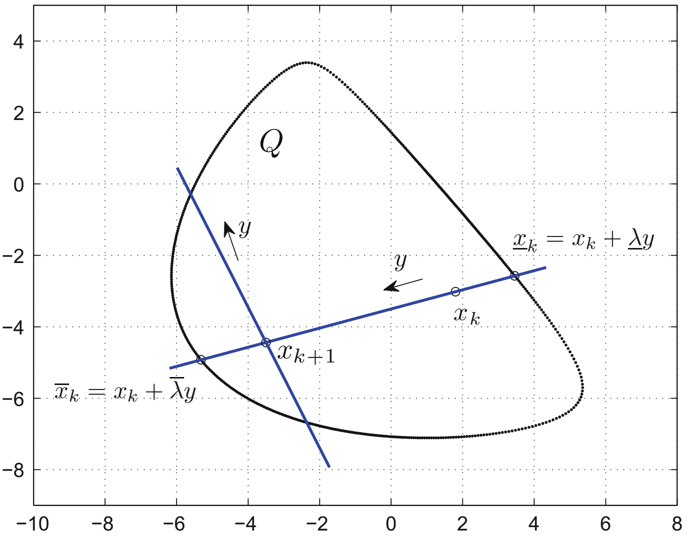
$$\underline{\lambda} = \arg\max\{\lambda \ge 0 \colon\ x - \lambda y \in Q\}, \qquad \overline{\lambda} = \arg\max\{\lambda \ge 0 \colon\ x + \lambda y \in Q\},$$

can be found efficiently. Numerous examples of sets with known boundary oracles can be found in [30, 68, 71]. Then, starting with some known $x_0 \in Q$, we proceed sampling in Q by using the technique described below.

Hit-and-Run algorithm (HR). For $x_k \in Q$, take a direction vector y uniformly distributed on the unit sphere; the oracle returns $\underline{x}_k = x_k - \underline{\lambda} y$ and $\overline{x}_k = x_k + \overline{\lambda} y$. Then, draw x_{k+1} uniformly distributed on $[\underline{x}_k, \overline{x}_k]$. Repeat. Schematically, this algorithm is illustrated in Fig. 3.

This technique was proposed in [71, 79]; under mild assumptions on Q, the distribution of the random point x_k was proved to approach the uniform distribution on Q. Instead of using the "direct Monte-Carlo," we now apply the randomized cutting plane algorithm, following the ideas of [18, 57].

A cutting plane algorithm. Start with $X_0 = Q$. For X_k, generate $3N$ points $x_k, \underline{x}_k, \overline{x}_k, k = 1, \dots, N$, by the HR algorithm and find $f_k = \min(c,x)$, where the minimum is taken over these $3N$ points. Proceed to the new set $X_{k+1} = X_k \bigcap \{x \colon\ (c,x) \le f_k\}$ and the initial point $x_0 = \arg\min(c,x)$, where the minimum is also taken over the $3N$ points mentioned above.



Q
y
y
$\underline{x}_k = x_k + \underline{\lambda} y$
x_k
x_{k+1}
$\overline{x}_k = x_k + \overline{\lambda} y$
4
2
0
−2
−4
−6
−8
−10
−8
−6
−4
−2
0
2
4
6
8

Fig. 3 The idea of the HR algorithm

Rigorous results on the rate of convergence of such an algorithm are lacking. For the idealized analog of it (with the points x "truly" uniformly distributed in X_k), the results on convergence can be found in [18, 57]. Moreover, the algorithm presented above includes the boundary points $\underline{x}_k$, $\overline{x}_k$; this essentially improves the convergence, since the minimum in the original problem (26) is attained at a boundary point. Numerical experiments in [18, 57] confirm a nice convergence if the set Q is not too "flat."

6 Conclusions

We have covered in this chapter several topics—in robustness, estimation, control, feasibility, constrained and unconstrained optimization—where the ideas of randomization can be applied and moreover can provide better results than deterministic methods. We could see that the situation with regard to effectiveness of randomized methods is not completely clarified; e.g., some straightforward attempts to apply Monte Carlo for optimization do not work for high dimensions. On the other hand, the only approach to minimization of nonsmooth convex functions with zero-order oracle (i.e., only function values are available) is based on randomization. We hope that the survey will stimulate further interest toward this exciting field of research.

Acknowledgements Financial support for this work was provided by the Russian Science Foundation through project no. 16-11-10015.

References

1. Agmon, S.: The relaxation method for linear inequalities. Canad. J. Math. **6**, 382–393 (1954)
2. Barmish, B.R., Lagoa, C.M.: The uniform distribution: A rigorous justification for its use in robustness analysis. Math. Control Sign. Syst. **10**(3), 203–222 (1997)
3. Barmish, B., Polyak, B.: A new approach to open robustness problems based on probabilistic prediction formulae. In: Proc. 13th World Congress of IFAC. San Francisco, **H**, 1–6 (1996)
4. Barmish, B.R., Shcherbakov, P.S.: On avoiding vertexization of robustness problems: The approximate feasibility concept. In: Proc. 39th Conference on Decision and Control, Sydney, Australia (2000)
5. Barmish, B.R., Shcherbakov, P.S.: On avoiding vertexization of robustness problems: The approximate feasibility concept. IEEE Transa Autom. Control **47**(5), 819–824 (2002)
6. Barmish, B.R., Shcherbakov, P.S., Ross, S.R., Dabbene, F.: On positivity of polynomials: The dilation integral method. IEEE Transa Autom. Control **54**(5), 965–978 (2009)
7. Bauschke, H.H., Borwein, J.M.: On projection algorithms for solving convex feasibility problems. SIAM Review **38**(3):367–426 (1996)
8. Ben-Tal, A., Nemirovski, A.: Robust convex optimization. Math. Oper. Res. **23**(4), 769–805 (1998)
9. Bertsekas, D.P., Tsitsiklis, J.N.: Parallel and Distributed Computation. Prentice Hall Inc. (1989)
10. Boyd, S., El Ghaoui, L., Feron, E., Balakrishnan, V.: Linear Matrix Inequalities in Systems and Control Theory. SIAM Publ., Philadelphia (1994)
11. Calafiore, G., Campi, M.C.: Uncertain convex programs: Randomized solutions and confidence levels. Math. Prog. **201**(1), 25–46 (2005)
12. Calafiore, G., Campi, M.: The scenario approach to robust control design. IEEE Trams. Autom. Control **45**(5), 742–753 (2006)
13. Calafiore G., Polyak, B.: Stochastic algorithms for exact and approximate feasibility of robust LMIs. IEEE Trans. Autom. Control. **46**(11), 1755–1759 (2001)
14. Campi, M.; Why is resorting to fate wise? A critical look at randomized algorithms in systems and control. Eur. J. Control **16**(5), 419430 (2010)
15. Censor, Y., Cegielski, A.: Projection methods: An annotated bibliography of books and reviews. Optimization: A Journal of Math. Progr. Oper. Res. **64**(11), 2343–2358 (2015)
16. Censor, Y., Herman, G.T., Jiang, M.: A note on the behavior of the randomized Kaczmarz algorithm of Strohmer and Vershynin. J. Fourier Anal. Appl. **15**(4), 431–436 (2009)
17. Censor, Y., Zenios, S.A.: Parallel Optimization: Theory, Algorithms, and Applications. New York, NY, USA: Oxford University Press; (1997)
18. Dabbene, F., Scherbakov, P.S., Polyak, B.T.: A randomized cutting plane method with probabilistic geometric convergence. SIAM J. Optimiz.**20**(6), 3185–3207 (2010)
19. Dorea, C.: Expected number of steps of a random optimization method. J.Optimiz. Th. Appl. **39**(2), 165–171 (1983)
20. Eldar, E., Needell, D.: Acceleration of randomized Kaczmarz method via the Johnson Lindenstrauss Lemma Numerical Algorithms **58**(2), 163–177 (2011)
21. Ermoliev, Yu., Wets, R. (eds.): Numerical Techniques for Stochastic Optimization. Springer (1988)
22. Fisher, R.A.: The Design of Experiments. Oliver and Boyd, Edinburgh (1935)
23. Gilks, W.R., Richardson, S., Spiegelhalter, D.J.: Markov Chain Monte Carlo in Practice. Chapman and Hall, London (1996)
24. Goldberg, D.: Genetic Algorithms in Search, Optimization and Machine Learning. Addison-Wesley, Reading, MA (1989)
25. Goldensluger, A, Polyak, B.: Estimation of regression parameters with arbitrary noise. Math. Meth. Stat. **2**(1), 18–29 (1993)
26. Gower, M., Richtarik, P.: Randomized iterative methods for linear systems. SIAM J. Matr. Anal. Appl. **36**(4)1660–1690 (2015)
27. Granichin, O.: Estimating the parameters of linear regression in an arbitrary noise. Autom. Remote Control **63**(1), 25–35 (2002)

28. Granichin, O., Polyak, B.: Randomized Algorithms for Estimation and Optimization under Almost Arbitrary Noises. Nauka, Moscow (2003) (in Russian)
29. Granichin, O., Volkovich, Z., Toledano-Kitai, D.: Randomized Algorithms in Automatic Control and Data Mining. Springer, Berlin-Heidelberg (2015)
30. Gryazina, E.N., Polyak, B.: Random sampling: Billiard Walk algorithm. Eur. J. Oper. Res. **238**(2), 497–504 (2014)
31. Gubin, L., Polyak, B., Raik, E.: The method of projections for finding the common point of convex sets. USSR Comput. Math. Math. Phys. **7**(6), 1-24 (1967)
32. Gupal, A.: A method for the minimization of almost-differentiable functions. Cybernetics. (1), 115–117 (1977)
33. Gürbüzbalaban, M. Ozdaglar, A., Parrilo, P.: Why random reshuffling beats stochastic gradient descent, arXiv:1510.08560v2 [math.OC], May 1, 2018.
34. Heunis, A.J.: Use of Monte Carlo method in an algorithm which solves a set of functional inequalities. J. Optim. Theory Appl. **45**(1), 89–99 (1984)
35. Horst, R., Pardalos, Panos M. (eds.): Handbook of Global Optimization, vol. **1**. Kluwer, Dordrecht (1995)
36. Kaczmarz, S.: Angenäherte Aufslösung von Systemen linearer Gleichungen. Bull. Intern. Acad. Polon. Sci., Lett. A. 355–357 (1937). English translation: Approximate solution of systems of linear equations. Int. J. Control **57**(6), 1269–1271 (1993)
37. Kroese, D.P., Taimre, T., Botev,Z.I.: Handbook of Monte Carlo Methods. John Wiley and Sons, New York (2011)
38. Kushner, H.J., Clark, D.S.: Stochastic Approximation Methods for Constrained and Unconstrained Systems. Vol. 26 of Applied Mathematical Sciences. Springer, New York (1978)
39. Lagoa, C.M., Li, X., Sznaier, M.: Probabilistically constrained linear programs and risk-adjusted controller design. SIAM J. Optimiz. **15**(3), 938–951 (2005)
40. Lagoa, C.M., Shcherbakov, P.S., Barmish, B.R.: Probabilistic enhancement of classical robustness margins: The unirectangularity concept. Syst. Control Lett. **35**(1), 31–43 (1998)
41. Leventhal, D., Lewis, A.S.: Randomized methods for linear constraints: Convergence rates and conditioning. Math. Oper. Res. **35**(3) 641–654 (2010)
42. Matyas, J.: Random optimization. Autom. Remote Control **26**(2), 246–253 (1965)
43. Metropolis, N., Ulam S.: The Monte Carlo method. J. Amer. Stat. Assoc. **44**(247), 335–341 (1949)
44. Motzkin, T.S., Shoenberg, I.J.: The relaxation method for linear inequalities. Canad. J. Math. **6**, 393–404 (1954)
45. Nedic, A.: Random algorithms for convex minimization problems. Math. Progr. **129**(2), 225–253 (2011)
46. Nemirovskii, A.S, Polyak, B.T.: Necessary conditions for the stability of polynomials and their use. Autom. Remote Control **55**(11), 1644–1649 (1994)
47. Nemirovski, A., Yudin, D.B.: Problem Complexity and Method Efficiency in Optimization. John Wiley and Sons, New York (1983)
48. Nesterov, Yu.: Introductory Lectures on Convex Optimization: A Basic Course. Klüwer (2004)
49. Nesterov, Y.: Efficiency of coordinate descent methods on huge-scale optimization problems. SIAM J. Optimiz. **22**(2), 341–362 (2012)
50. Nesterov, Yu., Spokoiny, V.: Random gradient-free minimization of convex functions. Foundations Comput. Math. **17**(2), 527–566 (2017)
51. Novikova, N.M.: Stochastic quasi-gradient method for minimax seeking. USSR Comp. Math. Math. Phys. **17**, 91–99 (1977)
52. Pardalos, Panos M., Romeijn, H. Edwin (eds.): Handbook of Global Optimization, vol. **2**. Kluwer, Dordrecht (2002)
53. Petrikevich, Ya. I.: Randomized methods of stabilization of the discrete linear systems. Autom. Remote Control **69**(11), 1911–1921 (2008)
54. Petrikevich, Ya.I., Polyak, B.T., Shcherbakov, P.S.: Fixed-order controller design for SISO systems using Monte Carlo technique. In: Proc. 9th IFAC Workshop “Adaptation and Learning in Control and Signal Processing” (ALCOSP’07) St.Petersburg, Russia (2007)

55. Polyak, B.T.: Introduction to Optimization. Optimization Software, New York (1987)
56. Polyak, B.: Random algorithms for solving convex inequalities. In: Butnariu, D., Censor, Y., Reich, S. (eds.) Inherently Parallel Algorithms in Feasibility and Optimization and Their Applications, pp. 409–422. Elsevier (2001)
57. Polyak, B.T., Gryazina, E.N.: Randomized methods based on new Monte Carlo schemes for control and optimization. Ann. Oper. Res. **189**(1), 342–356 (2011)
58. Polyak, B.T., Shcherbakov, P.S.: A probabilistic approach to robust stability of time delay systems. Autom. Remote Control **57**(12), 1770–1779 (1996)
59. Polyak, B.T., Shcherbakov, P.S.: Random spherical uncertainty in estimation and robustness. IEEE Trans. Autom Control **45**(11), 2145–2150 (2000)
60. Polyak, B., Shcherbakov, P.: Why does Monte Carlo Fail to Work Properly in High-Dimensional Optimization Problems? J. Optim. Th. Appl. **173**(2), 612–627 (2017)
61. Polyak, B.T., Tempo, R.: Probabilistic robust design with linear quadratic regulators. Syst. Control Lett. **43**(5), 343–353 (2001)
62. Polyak, B.T., Tsybakov, A.B.: Optimal order of accuracy for search algorithms in stochastic optimization. Problems Inform. Transmiss. **26**(2), 126–133 (1990)
63. Polyak, B.T., Tsybakov, A.B.: On stochastic approximation with arbitrary noise (the KW case). In: Khas'minskii, R.Z. (ed.) Topics in Nonparametric Estimation. Advances in Soviet Math. **12**, 107–113 (1992)
64. Polyak, B., Yuditskij A.: Acceleration of stochastic approximation procedures by averaging. SIAM J. on Control Optimiz. **30**(4), 838–855 (1992)
65. Rastrigin, L.A.: Statistical Search Method. Nauka, Moscow (1968) in Russian)
66. Richtárik, P., Tacáč, M.: Iteration complexity of randomized block-coordinate descent methods for minimizing a composite function. Math. Progr. 2014, **144**(1–2), 1–38 (2014)
67. Robert, C.P., Casella, G.: Monte Carlo Statistical Methods. Springer-Verlag, New York (1999)
68. Shcherbakov, P.: Boundary oracles for control-related matrix sets. In: Proc. 19th Int. Symp. "Mathematical Theory of Networks and Systems" (MTNS-2010), Budapest, Hungary, Jul 5–9, 2010, pp. 665–670.
69. Shcherbakov, P., Dabbene, F.: On the generation of random stable polynomials. Eur. J. Control **17**(2), 145–159 (2011)
70. Simon, D.: Evolutionary Optimization Algorithms. Wiley, New York (2013)
71. Smith, R.L.: Efficient Monte Carlo procedures for generating points uniformly distributed over bounded regions. Oper. Res. **32**(6), 1296–1308 (1984)
72. Solis, F.J., Wets, R.J-B.: (1981). Minimization by random search techniques. Math. Oper. Res. **6**(1), 19–13 (1981)
73. Spall, J.C.: Introduction to Stochastic Search and Optimization: Estimation, Simulation, and Control. Vol. 64 of Wiley-Interscience series in discrete mathematics and optimization. John Wiley and Sons, Hoboken, NJ (2003)
74. Stengel, R.F., Ray L.R.: Stochastic robustness of linear time-invariant control systems. IEEE Trans. Autom. Control **36**(1), 82–87 (1991)
75. Strohmer, T., Vershynin, R.: A randomized Kaczmarz algorithm with exponential convergence. J. Fourier Anal. Appl. **15**(2), 262–278 (2009)
76. Tempo, R., Bai, Er-Wei, Dabbene, F.: Probabilistic robustness analysis: Explicit bounds for the minimum number of samples Syst. Control Lett. **30**(5), 237–242 (1997)
77. Tempo, R., Calafiore, G., Dabbene, F.: Randomized Algorithms for Analysis and Control of Uncertain Systems, with Applications. Springer, London (2013)
78. Tremba, A., Calafiore, G., Dabbene, F., Gryazina, E., Polyak, B., Shcherbakov, P., Tempo, R.: RACT: Randomized algorithms control toolbox for MATLAB. In: Proc. 17th World Congress of IFAC, Seoul, pp. 390–395 (2008)
79. Turchin, V.F.: On the computation of multidimensional integrals by the Monte-Carlo method. Theory of Probability and its Applications, **16**(4), 720–724 (1972)
80. Volkov, Y.V., Zavriev, S.K.: A general stochastic outer approximation method. SIAM J. Control Optimiz. **35**(4), 1387–1421 (1997)

81. Yakubovich, V.A.: Finite terminating algorithms for solving countable systems of inequalities and their application in problems of adaptive systems Doklady AN SSSR **189**, 495–498 (1969) (in Russian)
82. Zhigljavsky, A., Žhilinskas, A.: Stochastic Global Optimization. Springer Science+Business Media, New York (2008)

Stabilization of Deterministic Control Systems Under Random Sampling: Overview and Recent Developments

Aneel Tanwani, Debasish Chatterjee and Daniel Liberzon

Abstract This chapter addresses the problem of stabilizing continuous-time deterministic control systems via a sample-and-hold scheme under random sampling. The sampling process is assumed to be a Poisson counter, and the open-loop system is assumed to be stabilizable in an appropriate sense. Starting from as early as mid-1950s, when this problem was studied in the Ph.D. dissertation of R.E. Kalman, we provide a historical account of several works that have been published thereafter on this topic. In contrast to the approaches adopted in these works, we use the framework of piecewise deterministic Markov processes to model the closed-loop system, and carry out the stability analysis by computing the extended generator. We demonstrate that for any continuous-time robust feedback stabilizing control law employed in the sample-and-hold scheme, the closed-loop system is asymptotically stable for all large enough intensities of the Poisson process. In the linear case, for increasingly large values of the mean sampling rate, the decay rate of the sampled process increases monotonically and converges to the decay rate of the unsampled system in the limit. In the second part of this article, we fix the sampling rate and address the question of whether there exists a feedback gain which asymptotically stabilizes the system in mean square under the sample-and-hold scheme. For the scalar linear case, the answer is in the affirmative and a constructive formula is provided here. For systems with dimension greater than 1 we provide an answer for a restricted class of linear systems, and we leave the solution corresponding to the general case as an open problem.

A. Tanwani (✉)
LAAS-CNRS, University of Toulouse, 31400 Toulouse, France
e-mail: atanwani@laas.fr

D. Chatterjee
Systems and Control Engineering, IIT Bombay, Mumbai 400076, India
e-mail: dchatter@iitb.ac.in

D. Liberzon
Coordinated Science Laboratory, Department of Electrical and Computer Engineering, University of Illinois at Urbana-Champaign, Urbana, IL 61820, USA
e-mail: liberzon@illinois.edu

© Springer Nature Switzerland AG 2018

T. Başar (ed.), *Uncertainty in Complex Networked Systems*,
Systems & Control: Foundations & Applications,
https://doi.org/10.1007/978-3-030-04630-9_6

1 Introduction

This chapter addresses the problem of stabilization of sampled-data control systems under random sampling. Let $(\tau_n)_{n\in\mathbb{N}}$ denote a monotonically increasing sequence in $[0, +\infty[$ with $\tau_0 := 0$. Consider a nonlinear control system

$$\dot{x}(t) = f\big(x(t), u(t)\big), \qquad x(0) \text{ given}, \quad t \geqslant 0, \tag{1}$$

where $f : \mathbb{R}^d \times \mathbb{R}^m \to \mathbb{R}^d$ is a continuously differentiable map, and the *control process* $t \mapsto u(t)$ is constant on each $[\tau_n, \tau_{n+1}[$ for each n. The corresponding solution $\big(x(t)\big)_{t\geqslant 0}$ of (1) is referred to as the *state process*. We shall comment on the precise properties of the solutions of (1) momentarily. Control systems where the control process gets updated at the discrete time instants $(\tau_n)_{n\in\mathbb{N}}$ are referred to as *sampled-data control systems* [2, 10, 24], and typically arise when implementing controllers using a computer [8, 18], or in the context of networked control systems [20, 33, 47].

Since any admissible control process $t \mapsto u(t)$ defined above can be written as

$$u(t) = \sum_{k=0}^{+\infty} u(\tau_k) \cdot 1_{[\tau_k, \tau_{k+1}[}(t) \quad \text{for } t \geqslant 0, \tag{2}$$

it is clear that the two key ingredients of sampled-data control systems are the sampling times $(\tau_k)_{k\in\mathbb{N}}$ and the control values $\big(u(\tau_k)\big)_{k\in\mathbb{N}}$. Different classes of these two ingredients are possible: the former may be periodic [35, 36, 46], state-dependent [7, 22, 38, 42] or random [23, 24]; and the latter may be a random sequence generated by a randomized Markovian policy as defined in [1] or just a feedback from the states at the sampling instants [20, 23], etc. One of the fundamental problems of interest is to provide a description of these two components (often in the form of an algorithm) that results in stability of the closed-loop system. Different approaches have been developed for the necessary analysis depending on how the sampling instants $(\tau_n)_{n\in\mathbb{N}}$ are chosen: see [2] for an overview of classical tools in linear systems with periodic sampling, the papers [30, 36, 37] provide tools specifically suited for nonlinear systems, and the approaches used for optimizing certain performance criterion can be found in [9, 10]. In this article, we are interested in the situation where the sampling times are generated *randomly*. Formally, we define N_t to be the number of sampling instants before (and including) time t as

$$N_t := \sup\big\{n \in \mathbb{N} \big| \tau_n \leq t\big\} \quad \text{for } t \geqslant 0, \tag{3}$$

and stipulate that the *sampling process* $(N_t)_{t\geqslant 0}$ is a continuous-time stochastic process satisfying the basic requirement

$$\tau_{N_t} \xrightarrow[t\uparrow+\infty]{} +\infty \text{ almost surely.} \tag{4}$$

It is assumed that there is an underlying probability triplet $(\Omega, \mathcal{F}, \mathsf{P})$, sufficiently rich, that provides the substrate for these processes (i.e., each random variable considered here is defined on $(\Omega, \mathcal{F}, \mathsf{P})$), and in the sequel we shall denote the mathematical expectation with respect to the probability measure P by $\mathsf{E}[\cdot]$.

Due to our assumptions on the random sequence $\left(\tau_k\right)_{k\in\mathbb{N}}$ and the right-hand side f of (1), it follows that, P-almost everywhere on the sample space Ω, Carathéodory solutions of (1) exist for a sufficiently small interval of time containing $t = 0$. In addition, we *assume* that solutions of (1) exist for all times. Typically, the sampling process $\left(N_t\right)_{t\geqslant 0}$ is constructed by means of a renewal process [4, 20]: independent and identically distributed positive random variables $\left(S_n\right)_{n\in\mathbb{N}^*}$ are defined on $(\Omega, \mathcal{F}, \mathsf{P})$,[1] with the probability distribution function of S_1 being $\mathsf{F}_{\text{hld}}(t) := \mathsf{P}(S_1 \le t)$ for $t \geqslant 0$, and the sequence $\left(\tau_n\right)_{n\in\mathbb{N}}$ is defined according to $\tau_0 := 0$ and $\tau_k := \sum_{\ell=1}^{k} S_\ell$ for $k \in \mathbb{N}^*$. The random variable S_n is the nth *holding time*.

Typical control problems in this setting consist of the design of controllers (feedbacks) for stabilization [23, 49], optimal control [3, 10], state estimation[2] [32, 41], etc., and we will study the problem of stabilization in this article. A mapping $t \mapsto x(t)$ that satisfies (1) in the preceding setting is, naturally, a stochastic process, and consequently, a library of different notions of stochastic stability are available to us [25, 26]. We will restrict our attention mostly to the particularly important property of stability in the mean and mean-square—especially well-studied in the context of linear models [11, 28]—in the sequel.

Finally, we note a connection with the work of Roberto Tempo, to whom this article is dedicated, and his coworkers on randomized algorithms in control theory [43]. That work asks the question of how many random samples *in space* are needed to obtain a sufficient guarantee that a property of interest holds over the whole space, whereas here we are asking how frequently we should sample randomly *in time* so that the feedback is still stabilizing.

2 Connections with Piecewise Deterministic Markov Processes

This section serves the purpose of demonstrating that sampled-data control systems under random sampling can be readily recast as piecewise deterministic Markov processes (PDMPs); consequently, typical control problems can be immediately addressed under this rather general and well-established umbrella framework [13, 14].

To start our discussion, we recall that the sequence of holding times $\left(S_n\right)_{n\in\mathbb{N}^*}$ is, typically, independent and identically distributed. The assumption of S_1 being

[1] For us $\mathbb{N}^* := \mathbb{N} \setminus \{0\}$.

[2] In contrast to the continuous-time systems given in (1), the references indicated here in the context of state estimation problems deal with discrete times linear systems, and the arrival of observations is modeled as a random process.

an exponential random variable with a given positive intensity λ is fairly common, and the resulting sampling process $\left(N_t\right)_{t\geqslant 0}$ is, consequently, a Poisson process with intensity λ. Recall [39, Theorem 2.3.2] that the Poisson process of intensity $\lambda > 0$ is a continuous-time random process $\left(N_t\right)_{t\geqslant 0}$ taking values in $\mathbb{N}^*$, with $N_0 = 0$, for every $n \in \mathbb{N}^*$ and $0 =: t_0 < t_1 < \cdots < t_n < +\infty$ the increments $\{N_{t_k} - N_{t_{k-1}}\}_{k=1}^n$ are independent, and $N_{t_k} - N_{t_{k-1}}$ is distributed as a Poisson-$\lambda(t_k - t_{k-1})$ random variable for each k. The Poisson process is among the most well-studied processes, and standard results (see, e.g., [39, §2.3]) show that it is memoryless and Markovian. Nevertheless, the resulting state process $\left(x(t)\right)_{t\geqslant 0}$ obtained as a solution of (1) under Poisson sampling is *not* controlled Markovian in general. Recall that an $\mathbb{R}^\nu$-valued random process $\left(\widetilde{x}(t)\right)_{t\geqslant 0}$ controlled by an $\mathbb{R}^m$-valued random process $\left(\widetilde{u}(t)\right)_{t\geqslant 0}$ is *controlled Markov* [19, §III.6] if for every $t, h > 0$ and every Borel set $\mathcal{S} \subset \mathbb{R}^\nu$ we have

$$\mathsf{P}\left(\widetilde{x}(t+h) \in \mathcal{S} \middle| \widetilde{x}(s), \widetilde{u}(s) \text{ for } s \in [0,t]\right) = \mathsf{P}\left(\widetilde{x}(t+h) \in \mathcal{S} \middle| \widetilde{x}(t), \widetilde{u}(t)\right).$$

Indeed, suppose that we intend to employ feedback at sampling instants so that $u(t) = u(\tau_{N_t}) = \kappa\big(x(\tau_{N_t})\big)$ for some measurable map κ, fix $t, t' > 0$, and suppose that the history $\big\{\big(x(s), u(s)\big) \big| s \in [0,t]\big\}$ up to time t is available to us. Of course, any finite k samples may have occured during $[t, t+t']$. If $k = 0$, then $x(\tau_{N_t})$ is not needed to find the conditional distribution of $x(t+t')$ given $\big\{\big(x(s), u(s)\big) \big| s \in [0,t]\big\}$. If $k = 1$, then the conditional distribution of $x(t+t')$ depends on the value of $x(\tau_{N_t})$: since $\tau_{N_t+1} \in]t, t+t']$, the control action at τ_{N_t+1} depends on $x(\tau_{N_t+1})$, and influences $x(t+t')$. A similar reasoning holds for all $k \geqslant 2$.

The controlled Markovian property is extremely desirable in practice, and to arrive at a controlled Markov process in the context of (6), we proceed to adjoin an additional random vector by enlarging the state space. Corresponding to the state process $\left(x(t)\right)_{t\geqslant 0}$ that solves (1), we define the continuous-time *last-sample process* $\left(x(\tau_{N_t})\right)_{t\geqslant 0}$; at each time t, $x(\tau_{N_t})$ is the value of the vector of states at the last sampling time immediately preceding t. In other words, $\mathbb{R}^d$-valued process $\left(x(\tau_{N_t})\right)_{t\geqslant 0}$ attains the value of the states at each sampling instant and stays constant over the corresponding holding time. It turns out to be convenient to introduce the continuous-time *error* process $\left(e(t)\right)_{t\geqslant 0}$ defined by

$$e(t) := x(t) - x(\tau_{N_t}) \quad \text{for } t \geqslant 0. \tag{5}$$

With the joint stochastic process $\left(x(t), e(t)\right)_{t\geqslant 0}$ taking values in $\mathbb{R}^d \times \mathbb{R}^d$, we write the system of interest as a stochastic process described by the ordinary differential equation

$$\begin{pmatrix} \dot{x}(t) \\ \dot{e}(t) \end{pmatrix} = \begin{pmatrix} f\big(x(t), u(t)\big) \\ f\big(x(t), u(t)\big) \end{pmatrix} \quad \text{for almost all } t \geqslant 0, \tag{6a}$$

and at each sampling time τ_{N_t} the process $\big(x(t), e(t)\big)_{t\geqslant 0}$ is reset according to

$$\begin{pmatrix} x(\tau_{N_t}) \\ e(\tau_{N_t}) \end{pmatrix} = \begin{pmatrix} x(\tau_{N_t}^-) \\ 0 \end{pmatrix} \quad \text{with the convention that } x(\tau_0^-) = x_0. \tag{6b}$$

It is readily observed that the joint process $\big(x(t), e(t)\big)_{t\geqslant 0}$ is controlled Markovian. We sometimes abbreviate the right-hand side of (6a) by

$$\mathbb{R}^d \times \mathbb{R}^m \ni (x, u) \mapsto F(x, u) := \begin{pmatrix} f(x, u) \\ f(x, u) \end{pmatrix} \in \mathbb{R}^d \times \mathbb{R}^d.$$

We shall be concerned exclusively with *feedback* controls in this article. In other words, we stipulate that there exists some measurable map

$$\mathbb{R}^d \times \mathbb{R}^d \ni (x, e) \mapsto \kappa(x, e) \in \mathbb{R}^m$$

such that our control process becomes, in the notation of (2),

$$u(t) = \sum_{k=0}^{+\infty} \kappa\big(x(\tau_k), e(\tau_k)\big) 1_{[\tau_k, \tau_{k+1}[}(t) \quad \text{for } t \geqslant 0.$$

In other words, with κ substituted into (6a), our *closed-loop* system becomes

$$\begin{pmatrix} \dot{x}(t) \\ \dot{e}(t) \end{pmatrix} = \begin{pmatrix} f\Big(x(t), \kappa\big(x(\tau_{N_t}), e(\tau_{N_t})\big)\Big) \\ f\Big(x(t), \kappa\big(x(\tau_{N_t}), e(\tau_{N_t})\big)\Big) \end{pmatrix} \quad \text{for almost all } t \geqslant 0, \tag{7}$$

while the reset map (6b) stays intact.

With the class of admissible feedback control processes as described above, the description (6b)–(7) provides the basic ingredients to transit to the framework of PDMPs. Indeed, we see readily that the standard conditions for a PDMP [14, (24.8), p. 62] hold for the joint process $\big(x(t), e(t)\big)_{t\geqslant 0}$ described by (6b)–(7) with

- the vector field $\mathfrak{X}$ in [14, §24] being the map $(x, e) \mapsto F\big(x, \kappa(x, e)\big)$,
- the jump rate λ in [14, §24] being a nonnegative measurable function such that $\mathsf{F}_{\text{hld}}(t) = \exp\big(\int_0^t \lambda(s)\,\mathrm{d}s\big)$, which can be readily derived for particular cases of probability distribution functions F_{hld} as in [14, p. 37], and
- the stochastic kernel Q for the reset map in [14, §24, p. 58] is the Dirac measure $Q\big(B; (x, e)\big) := \delta_{\{(x,0)\}}(B) = 1_B(x, 0)$ for every Borel subset $B \subset \mathbb{R}^d \times \mathbb{R}^d$ in the context of (6b)–(7).

In this chapter, we will work exclusively under the assumption that the controller has access to perfect state measurements at sampling times. While, in general, it is of interest to consider feedbacks which depend on the measurement error at sampling times $e(\tau_{N_t})$, we can drop the dependence of feedback κ on $e(\tau_{N_t})$ in the case of perfect measurements since $e(\tau_{N_t}) = 0$, for each $t \in [0, +\infty[$, in such cases. In the sequel, we shall employ the feedback exclusively as a function of $x(\tau_{N_t})$, which is described in (5) by the difference between $x(t)$ and $e(t)$, i.e, we shall employ some measurable map $\kappa' : \mathbb{R}^d \to \mathbb{R}^m$ such that $\kappa(x, e) = \kappa'(x - e)$ for all $(x, e) \in \mathbb{R}^d \times \mathbb{R}^d$; we shall abuse notation and continue to use the symbol κ for κ' since there is no risk of confusion.

Remark 2.1 As a consequence of the preceding discussion, we observe that the techniques in [14, Chapters 4, 5] (including several results on stability and optimal control) carry over at once to the setting of sampled-data control systems under random sampling as special cases. In particular, the so-called *extended generator* of the PDMP (6b)–(7) is a particularly useful device for the purposes of analyzing stability and optimality, and we shall look at it in greater detail below in the context of stability.

The *extended generator* of the joint process $\big(x(t), e(t)\big)_{t\geqslant 0}$ is the linear operator $\psi \mapsto \mathcal{L}\psi$ defined by

$$\begin{aligned}&\mathbb{R}^d \times \mathbb{R}^d \ni (y, z) \mapsto \mathcal{L}\psi(y, z) := \\ &\qquad \lim_{h\downarrow 0} \frac{1}{h}\Big(\mathsf{E}\left[\psi\big(x(t+h), e(t+h)\big)\big|x(t) = y, e(t) = z\right] - \psi(y, z)\Big) \in \mathbb{R} \qquad (8)\end{aligned}$$

for all maps $\psi : \mathbb{R}^d \times \mathbb{R}^d \to \mathbb{R}$ such that the limit is defined everywhere. It is possible to directly write down the extended generator of $\big(x(t), e(t)\big)_{t\geqslant 0}$ from [14, (26.15), p. 70]. We provide the following Proposition catering to the most standard special case of sampling process being Poisson; a direct proof of Proposition 2.2 is included in Appendix A for completeness.

Proposition 2.2 *If the sampling process $\big(N_t\big)_{t\geqslant 0}$ is Poisson with intensity $\lambda > 0$, then the joint process $\big(x(t), e(t)\big)_{t\geqslant 0}$ described above is Markovian. Moreover, for any function $\mathbb{R}^d \times \mathbb{R}^d \ni (y, z) \mapsto \psi(y, z) \in [0, +\infty[$ with at most polynomial growth as $\|(y, z)\| \to +\infty$, we have*

$$\mathcal{L}\psi(y, z) = \big\langle \nabla_y\psi(y, z) + \nabla_z\psi(y, z), f\big(y, \kappa(y - z)\big)\big\rangle + \lambda\big(\psi(y, 0) - \psi(y, z)\big). \qquad (9)$$

We submit that this extended generator serves as an important tool in most control-theoretic problems associated with this class of randomly sampled-data systems. In particular, (9) provides the following *Dynkin's formula*

$$\mathsf{E}\left[\psi(x(t), e(t))\right] = \mathsf{E}\left[\psi(x(0), e(0))\right] + \mathsf{E}\left[\int_0^t \mathcal{L}\psi(x(s), e(s))\,\mathrm{d}s\right], \qquad (10)$$

which allows us to establish connections with definitive results on stability.

In the sequel, while we provide an account of stability results obtained by different means in prior works, the focus is on using the extended generator to obtain conditions under which the sampled-data systems are asymptotically stable.

3 Lower Bounds on the Sampling Rate

We employ the tools from the previous section to study the following qualitative property of the closed-loop system (7)–(6b). The closed-loop system (7)–(6b) is *globally exponentially stable in the second moment* [25, Chapter 1, p. 23] if there exist two constants $C, \mu > 0$ such that

$$\text{for every } x(0) \in \mathbb{R}^d \text{and } t \geqslant 0, \quad \mathsf{E}\left[\|x(t)\|^2 \,\middle|\, x(0)\right] \le C\,\|x(0)\|^2\,\mathrm{e}^{-\mu t}.$$

This particular property of stochastic stability is standard, and says that, on an average, the square norm of the system states converges exponentially fast to 0 uniformly from every initial condition.

As a first step in obtaining conditions which guarantee this property, we specify the class of feedback controls in (7). The natural candidates for feedback controls, for which we solve the sampled-data problem, are the ones which asymptotically stabilize the system when the measurements of the state are available in continuous time (without sampling), and possess some robustness properties with respect to errors in the measurement of state. To attribute these properties to the feedback law $\kappa : \mathbb{R}^d \to \mathbb{R}^m$ appearing in (7), it is assumed that there is a function $U : \mathbb{R}^d \to [0, +\infty[$ such that

(L1) there exist $\underline{\alpha}, \overline{\alpha} > 0$ satisfying

$$\underline{\alpha}|x|^2 \le U(x) \le \overline{\alpha}|x|^2 \quad \text{for all } x \in \mathbb{R}^d;$$

(L2) there exist $\alpha, \gamma > 0$ which satisfy

$$\langle \nabla U(x), f(x, \kappa(x-e))\rangle \le -\alpha\,U(x) + \gamma U(e) \quad \text{for all } (x, e) \in \mathbb{R}^d \times \mathbb{R}^d;$$

(L3) there exist $\chi_x > 0$, $\chi_e \in \mathbb{R}$ satisfying

$$\langle \nabla U(e), f(x, \kappa(x-e))\rangle \le \chi_x\,U(x) + \chi_e U(e) \quad \text{for all } (x, e) \in \mathbb{R}^d \times \mathbb{R}^d.$$

Restricting our attention to such a class of controllers, we are interested in addressing the following problem:

Problem 1 Consider the system (7)–(6b) with $\left(N_t\right)_{t\geqslant 0}$ in (3) a Poisson process of intensity λ. If the feedback law $\kappa : \mathbb{R}^d \to \mathbb{R}^m$ is such that (L1)–(L3) hold for some function $U : \mathbb{R}^d \to [0, +\infty[$, does there exist $\lambda > 0$ such that the closed-loop system (7)–(6b) is globally exponentially stable in the second moment?

It is noted that, between two consecutive updates in the controller value, the process $\left(x, e\right)$ follows the differential equation

$$\dot{x} = f(x, \kappa(x - e)) \tag{11a}$$

$$\dot{e} = f(x, \kappa(x - e)). \tag{11b}$$

Assumptions (L1)–(L2) basically characterize the existence of a feedback controller which renders the system (11a) input-to-state stable (ISS) with respect to measurement errors e. Assumption (L3) is introduced to bound the growth of the error e which satisfies (11b). The notion of ISS, pioneered in [40], has been instrumental in the synthesis of control laws for nonlinear systems under actuation and measurement errors. While the general formulation of ISS property would involve nonlinear gains, here we choose to work with linear gains to simplify the presentation. Sampled-data problems in the deterministic setting, where the objective is to find upper bounds on the sampling period that guarantee asymptotic stability, employing feedback controllers with aforementioned robustness properties, have been studied in [36]. In fact, such tools have also been useful in a more general framework where errors in measurements may result from sources other than sampling (see, e.g., [30]). For our purposes, the existence of such robust static controllers allows us to compute a lower bound on the mean sampling rate that solves Problem 1.

Proposition 3.1 *Assume that there exist $\kappa : \mathbb{R}^d \to \mathbb{R}^m$ and $U : \mathbb{R}^d \to \mathbb{R}_{\geqslant 0}$ such that (L1), (L2), and (L3) hold. If the sampling process $\big(N_t\big)_{t\geqslant 0}$ is Poisson with intensity $\lambda > 0$, then for each $\lambda > 0$ and $\delta \in [0, 1[$ satisfying*

$$\lambda > \chi_e + \frac{\gamma\chi_x}{\delta\alpha} \tag{12}$$

the system (7)–(6b) is exponentially stable in the second moment.

Proof Let us define the function $V : \mathbb{R}^d \times \mathbb{R}^d \to [0, +\infty[$

$$V(x, e) = U(x) + \beta U(e),$$

where $\beta > 0$ is to be specified momentarily. From Proposition 2.2 it follows that

$$\begin{aligned}\mathcal{L}V(x, e) &= \langle \nabla U(x) + \beta\nabla U(e), f(x, \kappa(x - e))\rangle - \lambda\beta U(e)\\ &\leq -\alpha U(x) + \gamma U(e) + \beta\chi_x U(x) + \beta\chi_e U(e) - \lambda\beta U(e).\end{aligned}$$

Pick $\delta \in [0, 1[$ and select $\beta = \delta\alpha\chi_x^{-1}$. Then for any $\lambda > 0$ satisfying (12), there exists $0 < \varepsilon < 1$ such that

$$\lambda\beta > \chi_e\beta + \gamma + \varepsilon\alpha(1 - \delta)\beta$$

so that

$$\mathcal{L}V(x, e) \leq -\varepsilon\alpha(1 - \delta)(U(x) + \beta U(e)) = -\varepsilon\alpha(1 - \delta)V(x, e).$$

Exponential stability in the second moment of the process $\big(x(t), e(t)\big)_{t\geqslant 0}$ now follows from Dynkin's formula (10). □

The main point of Proposition 3.1 is to show that, for controllers with certain robustness properties, the sampled-data system with random sampling is exponentially stable with large enough sampling rate, and this is done by using the extended generator for the controlled Markovian process $\big(x(t), e(t)\big)_{t\geqslant 0}$. This result can be generalized in several ways. Instead of requiring quadratic bounds on the function U in (L1), if for some $\underline{\alpha} > 0$, $p \geqslant 1$, $U(x)$ is lower (respectively, upper) bounded by $\underline{\alpha}|x|^p$ (resp. $\overline{\alpha}|x|^p$) for each $x \in \mathbb{R}^n$, then exponential stability in pth mean can be established. Other than the Poisson process, it is also possible to consider a different random process to determine the sampling instants. This of course changes the formula for the extended generator. Another level of generalization arises from introducing a diffusion term in the system dynamics (1), which would require us to work with a weaker notion of a solution, and consequently, the assumptions on function U need to be strengthened to be able to compute the extended generator. Stability analysis using extended generator for impulsive renewal systems with diffusion term in the differential equation has been carried out in [23].

So far, we have adopted a general approach to address the control of sampled-data nonlinear systems. Most of the results in the literature on stabilization with random sampling have been presented in the context of linear systems, and with the exception of [23], extended generators have not appeared elsewhere. We now focus our attention on linear systems: An overview of different approaches is presented and our eventual goal is to establish equivalence between some of these approaches and the extended generator approach for the case of Poisson sampling. In the process, we establish what may be regarded as a converse Lyapunov theorem for (6b)–(7) when the underlying renewal process $\big(N_t\big)_{t\geqslant 0}$ is Poisson with fixed intensity $\lambda > 0$.

4 Randomly Sampled Linear Systems: A Random Walk Down the History Lane

4.1 System Description

In the remainder of this chapter, we will restrict our attention to randomly sampled-data control of linear systems described by

$$\dot{x}(t) = Ax(t) + Bu(t), \quad x(0) = x_0 \text{ given}, \quad t \geqslant 0, \tag{13}$$

with the input u given by

$$u(t) = Kx(\tau_{N_t}) \quad \text{for all } t \geqslant 0,$$

where the pair (A, B) is assumed to be stabilizable, and the feedback gain K is assumed to be fixed a priori. With $\big(N_t\big)_{t\geqslant 0}$ the sampling process for the above control system, the resulting stochastic system for the joint process $\big(x(t), e(t)\big)_{t \geqslant 0}$ is described by

$$\begin{pmatrix} \dot{x}(t) \\ \dot{e}(t) \end{pmatrix} = \begin{pmatrix} A + BK & -BK \\ A + BK & -BK \end{pmatrix} \begin{pmatrix} x(t) \\ e(t) \end{pmatrix} \quad \text{for almost all } t \geqslant 0, \tag{14a}$$

and the reset equation at the sampling times is

$$\begin{pmatrix} x(\tau_{N_t}) \\ e(\tau_{N_t}) \end{pmatrix} = \begin{pmatrix} I & 0 \\ 0 & 0 \end{pmatrix} \begin{pmatrix} x(\tau_{N_t}^-) \\ e(\tau_{N_t}^-) \end{pmatrix}. \tag{14b}$$

For this class of systems, lower bounds on the sampling rates required for stability can be computed more explicitly. Also, this case has been studied in the literature over several epochs, and we provide an overview of the approaches that have been used for analyzing the stability of randomly sampled linear systems. To simplify notation, let us abbreviate the system in (14) as

$$\begin{aligned} \dot{\overline{x}} &= F\overline{x} \\ \overline{x}(\tau_{N_t}) &= G\,\overline{x}(\tau_{N_t}^-) \end{aligned} \tag{15}$$

where $\overline{x} := (x^\top, e^\top)^\top \in \mathbb{R}^{\overline{n}}$ and $\overline{n} = 2d$.

4.2 Early Efforts

It may appear surprising that the investigations into control of linear sampled-data control systems under random sampling started as early as the late 1950s. Indeed, Rudolf Kalman in his Ph.D. dissertation [24] studied sample-and-hold schemes for linear time-invariant control systems under random sampling. In particular, he studied several stochastic stability notions for both linear scalar systems and systems of higher dimensions: the definitions of *stability almost surely*, *stability in the mean*, *stability in mean-square*, and *stability in the mean sampling period* appear in his thesis. It is interesting to note that the key steps in his work were to first understand the asymptotic behaviour of the process $\big(x(\tau_k)\big)_{k\in\mathbb{N}}$ as $k \to +\infty$, and thereafter to derive certain inferences about the continuous-time process $\big(x(t)\big)_{t\geqslant 0}$. Only asymptotically stable system matrices A were considered by Kalman; this peculiar assumption was perhaps a natural consequence of his proof technique. The operator-theoretic approach à la extended generators pioneered by Dynkin [16, 17] was relatively less known at the time of Kalman's graduation.

About a decade later, Oskar Leneman at MIT published a sequence of short articles on control of linear time-invariant sample-and-hold systems under random sampling. Chief among this sequence is [29], where Leneman claimed that certain calculations in [24] did not quite lead to correct results. He focussed attention on scalar problems in [29], and derived his results following the same route as that of

Kalman: first getting estimates of the behavior of the sampled process $\big(x(\tau_k)\big)_{k\in\mathbb{N}}$, followed by inferring stability of the underlying continuous-time process $\big(x(t)\big)_{t\geqslant 0}$ via lengthy calculations involving some integral transform calculus. Once again, only scalar asymptotically stable systems were considered. Related problems of stability of linear control systems under random sample-and-hold schemes were almost concurrently investigated by Harold Kushner and his collaborators [27], and their techniques were also similar to those in [29]. To the best of our knowledge, it seems that this early period focussed attention only on open-loop asymptotically stable systems; even neutrally stable linear systems were perhaps considered too difficult to handle via these techniques. An admittedly speculative reason for this may have been that even for Poisson sampling, it was not clear how to deal with remarkably long holding times (that appear with probability 1) during which the process may deviate very far away from a given compact set since the right-hand side of the x-subsystem of (14) is *affine* in x during the holding times.

To anyone attempting to follow the footprints of Leneman, it is not difficult to appreciate the tediousness of the calculations involved in transitioning from estimates of the behavior of the sampled process. (In fact, [29] skips quite a few details and provides the readers with just the key steps of his proofs.) The first part of deriving estimates for the sampled process $\big(x(\tau_k)\big)_{k\in\mathbb{N}}$ is relatively simple:

Lemma 4.1 *Let $-\infty < t' < t'' < +\infty$. If $A \in \mathbb{R}^{d\times d}$, then*

$$\int_{t'}^{t''} \mathrm{e}^{tA}\,\mathrm{d}t = \frac{\big(\mathrm{e}^{t''A} - \mathrm{e}^{t'A}\big)}{A},$$

where the object on the right-hand side is defined by

$$\frac{\big(\mathrm{e}^{t''A} - \mathrm{e}^{t'A}\big)}{A} := \sum_{k=1}^{+\infty} \frac{(t'')^k - (t')^k}{k!} A^{k-1}.$$

Proof On the one hand, if $A \in \mathbb{R}^{d\times d}$ is non-singular, then (see also [6, p. 47])

$$\begin{aligned}\int_{t'}^{t''} \mathrm{e}^{tA}\,\mathrm{d}t &= \int_{t'}^{t''} \sum_{k=0}^{+\infty} \frac{A^k}{k!} t^k\,\mathrm{d}t = \sum_{k=0}^{+\infty} \frac{A^k}{(k+1)!}\big((t'')^{k+1} - (t')^{k+1}\big)\\ &= A^{-1}(\mathrm{e}^{t''A} - \mathrm{e}^{t'A})\\ &= \sum_{k=1}^{+\infty} \frac{(t'')^k - (t')^k}{k!} A^{k-1} = \frac{\big(\mathrm{e}^{t''A} - \mathrm{e}^{t'A}\big)}{A},\end{aligned}$$

where we have carried out the interchange of the summation and the integral under the shadow of Tonelli's theorem [15, Theorem 4.4.5]. In particular, we observe that the map

$$\mathbb{R}^{d\times d} \ni A \mapsto \frac{\big(\mathrm{e}^{t''A} - \mathrm{e}^{t'A}\big)}{A} \in \mathbb{R}^{d\times d}$$

is continuous. On the other hand, if $A \in \mathbb{R}^{d\times d}$ is singular, we pick a sequence of matrices $\big(A_n\big)_{n\in\mathbb{N}^*}$ with $A_n := A + \varepsilon_n I$ and $\varepsilon_n \downarrow 0$, such that each A_n is nonsingular. (For instance, we employ a similarity transformation to obtain the upper-triangular complex-Jordan form J of A; the eigenvalues of A are on the diagonal of J and since A is singular, there is at least one 0 on the diagonal of J; we pick the sequence $\varepsilon_n \downarrow 0$ such that $J + \varepsilon_n I$ is nonsingular for each n — this is possible since the spectrum of A is a finite set.) Since $A_n \xrightarrow[n\to+\infty]{} A$, we apply the assertion to the nonsingular matrix A_n instead of A, and the general formula follows at once from continuity. □

To simplify some calculations below, we assume that $A \in \mathbb{R}^{d\times d}$ is non-singular. Starting from (13) with a given initial condition $x(0)$, and

$$u(t) = Kx(\tau_i) \quad \text{whenever } t \in [\tau_i, \tau_{i+1}[, \quad i \in \mathbb{N}, \tag{16}$$

we arrive at

$$x(t) = \Big(\mathrm{e}^{(t-\tau_i)A} + \mathrm{e}^{tA}A^{-1}\big(\mathrm{e}^{-\tau_i A} - \mathrm{e}^{-tA}\big)BK\Big)x(\tau_i) \quad \text{for } t \in [\tau_i, \tau_{i+1}[, \tag{17}$$

or equivalently,

$$x(t) = \Big(\mathrm{e}^{(t-\tau_i)A}\big(I + A^{-1}BK\big) - A^{-1}BK\Big)x(\tau_i) \quad \text{for all } t \in [\tau_i, \tau_{i+1}[. \tag{18}$$

By continuity of solutions,

$$x(\tau_{i+1}) = \Big(\mathrm{e}^{(\tau_{i+1}-\tau_i)A}\big(I + A^{-1}BK\big) - A^{-1}BK\Big)x(\tau_i),$$

which is a recursive formula for the states at consecutive sampling instants. Multiplying out, for any $N \in \mathbb{N}^*$,

$$x(\tau_N) = \prod_{i=0}^{N-1}\Big(\mathrm{e}^{(\tau_{i+1}-\tau_i)A}\big(I + A^{-1}BK\big) - A^{-1}BK\Big)x(0), \tag{19}$$

where we remember that the product is directed.

In the scalar case ($d = 1$), by independence of the holding times,

$$\begin{aligned}\mathsf{E}\big[x(\tau_N)\big|x(0)\big] &= \mathsf{E}\left[\prod_{i=0}^{N-1}\Big(\mathrm{e}^{(\tau_{i+1}-\tau_i)A}\big(1 + A^{-1}BK\big) - A^{-1}BK\Big)x(0)\big|x(0)\right]\\ &= \prod_{i=0}^{N-1}\mathsf{E}\Big[\mathrm{e}^{(\tau_{i+1}-\tau_i)A}\big(1 + A^{-1}BK\big) - A^{-1}BK\Big]x(0)\end{aligned}$$

$$= \prod_{i=0}^{N-1}\Big(\mathsf{E}\big[\mathrm{e}^{(\tau_{i+1}-\tau_i)A}\big]\big(1 + A^{-1}BK\big) - A^{-1}BK\Big)x(0).$$

The quantity $\mathsf{E}[\mathrm{e}^{(\tau_{i+1}-\tau_i)A}]$ is simply the moment generating function $\mathcal{M}_S$ (if it exists) of $(\tau_{i+1} - \tau_i)$ evaluated at $A \in \mathbb{R}$, denoted hereafter by $\mathcal{M}_S(A)$.[3] Therefore,

$$\mathsf{E}\left[x(\tau_N)\middle|x(0)\right] = \prod_{i=0}^{N-1}\Big(\mathcal{M}_S(A)\big(1 + A^{-1}BK\big) - A^{-1}BK\Big)x(0).$$

For convergence of the product on the right-hand side to 0 as $N \to +\infty$, it is necessary and sufficient that

$$|\mathcal{M}_S(A)(A + BK) - BK| < |A|, \tag{20}$$

from which we can immediately arrive at the range of permissible K's. The question of designing stabilizing feedback gains K is addressed in detail in Sect. 6; Merely assuming that $A + BK = A(1 + A^{-1}BK)$ is Hurwitz stable may not be enough!

Remark 4.2 (*$A + BK$ Hurwitz is necessary for the scalar case*) In the scalar case and an unstable open-loop system (that is, $A > 0$), if we select the feedback gain K such that $A + BK > 0$, then the condition (20) will not be satisfied. Indeed, $\mathcal{M}_S(A) > 1$ for every $A > 0$ whenever the former exists.

The multidimensional case is similar to the scalar one: by independence of the holding times,

$$\begin{aligned}
\mathsf{E}\left[x(\tau_N)\middle|x(0)\right] &= \mathsf{E}\left[\prod_{i=0}^{N-1}\Big(\mathrm{e}^{(\tau_{i+1}-\tau_i)A}\big(I + A^{-1}BK\big) - A^{-1}BK\Big)x(0)\middle|x(0)\right]\\
&= \prod_{i=0}^{N-1}\mathsf{E}\Big[\mathrm{e}^{(\tau_{i+1}-\tau_i)A}\big(I + A^{-1}BK\big) - A^{-1}BK\Big]x(0) \qquad (21)\\
&= \prod_{i=0}^{N-1}\Big(\mathsf{E}\big[\mathrm{e}^{(\tau_{i+1}-\tau_i)A}\big]\big(I + A^{-1}BK\big) - A^{-1}BK\Big)x(0).
\end{aligned}$$

The matrix $\mathsf{E}\big[\mathrm{e}^{(\tau_{i+1}-\tau_i)A}\big]$ is well defined whenever $\mathcal{M}_S(\|A\|) = \mathsf{E}\big[\mathrm{e}^{(\tau_{i+1}-\tau_i)\|A\|}\big]$ exists; this follows from a standard application of the dominated convergence theorem [15, Theorem 4.3.5]. Now, the necessary and sufficient condition for convergence of the product on the right-hand side to 0 as $N \to +\infty$ is that

[3] Recall that the moment generating function $\mathcal{M}_X$, if it exists, of a random variable X is the function $\mathbb{R} \ni \xi \mapsto \mathcal{M}_X(\xi) := \mathsf{E}[\mathrm{e}^{\xi X}] \in \mathbb{R}$. The moment generating function may only be defined on a subset of $\mathbb{R}$, of course.

$$A^{-1}\Bigl(\mathsf{E}\bigl[\mathrm{e}^{(\tau_{i+1}-\tau_i)A}\bigr](A+BK)-BK\Bigr) \quad \text{is Schur stable.} \tag{22}$$

It is evident that straightforward calculations are enough to arrive at necessary and sufficient conditions for stability in the mean of the sampled process $\bigl(x(\tau_k)\bigr)_{k\in\mathbb{N}}$. A similar calculation can be carried out for $\bigl(\, \|x(\tau_k)\|\,\bigr)_{k\in\mathbb{N}}$ to arrive at convergence in mean-square of the process $\bigl(\, \|x(\tau_k)\|\,\bigr)_{k\in\mathbb{N}}$.

However, the preceding calculations do not shed much light on the inter-sample behavior of $\bigl(x(t)\bigr)_{t\geqslant 0}$. The transition from stability of the sampled process to that of $\bigl(x(t)\bigr)_{t\geqslant 0}$ is a nontrivial matter. A tiny calculation in this direction is to check whether the process $\bigl(x(\tau_{N_t})\bigr)_{t\geqslant 0}$ is stable, and to this end, our assumption (4) provides the necessary support, and one concludes that $\mathsf{E}\bigl[x(\tau_{N_t})\big|x(0)\bigr] \xrightarrow[t\to+\infty]{} 0$. The next natural step is to compute $\mathsf{E}\bigl[x(t)\big|x(0)\bigr]$ for a given time t, and finally to take the limit (if it exists), as $t \to +\infty$. However, at this stage matters start to become rather tedious and complicated. Indeed, if we proceed as Leneman does in [29], for the quadratic function $\mathbb{R}^d \ni x \mapsto \varphi(x) := \frac{1}{2}\langle x, Qx\rangle \in [0, +\infty[$ where $Q \in \mathbb{R}^{d\times d}$ is some symmetric and positive-definite matrix,

$$\begin{aligned}\mathsf{E}\bigl[\varphi\bigl(x(t)\bigr)\big|x(0)\bigr] &= \mathsf{E}\left[\varphi\bigl(x(t)\bigr)\sum_{k=0}^{+\infty} 1_{[\tau_k,\tau_{k+1}[}(t)\Big|x(0)\right]\\ &= \sum_{k=0}^{+\infty}\mathsf{E}\bigl[\varphi\bigl(x(t)\bigr)1_{[\tau_k,\tau_{k+1}[}(t)\big|x(0)\bigr]\end{aligned}$$

where the second equality follows by the monotone convergence theorem. Since $1_{[\tau_k,\tau_{k+1}[}(t) = 1$ if and only if $N_t = k$ and 0 otherwise, each summand on the right-hand side can be manipulated as

$$\mathsf{E}\bigl[\varphi\bigl(x(t)\bigr)1_{[\tau_k,\tau_{k+1}[}(t)\big|x(0)\bigr] = \mathsf{P}\bigl(N_t = k\big|x(0)\bigr)\ \mathsf{E}\bigl[\varphi\bigl(x(t)\bigr)\big|x(0), N_t = k\bigr].$$

If the sampling process $\bigl(N_t\bigr)_{t\geqslant 0}$ is Poisson with intensity λ, we have the expression $\mathsf{P}\bigl(N_t = k\big|x(0)\bigr) = \mathrm{e}^{-\lambda t}\frac{(\lambda t)^k}{k!}$ since the sampling process is independent of the state process, but for more general sampling (renewal) processes, such expressions are difficult to arrive at. Even if $\bigl(N_t\bigr)_{t\geqslant 0}$ is Poisson-λ, it is still not simple to compute the second term $\mathsf{E}\bigl[\varphi\bigl(x(t)\bigr)\big|x(0), N_t = k\bigr]$. Indeed, one would naturally proceed, for the specific case of φ defined above, by employing (19) and then (17) and separating out terms consisting of terms involving $x(\tau_k)$ and $(t-\tau_k)$. The (quadratic) terms consisting only of $x(\tau_k)$ can be dealt with as discussed above, and those containing $(t-\tau_k)$ would need the probability distribution of $(t-\tau_k)$. By all indications, Leneman's calculations (which are not explicitly provided in [29]) completed the preceding steps for the case of $d = 1$ and asymptotically stable A. It should be evident that for sampling processes more general than Poisson, this route quickly becomes intractable.

4.3 New Generation, Same Problem

Skipping a few decades, we arrive at [31] which presents stability conditions for several sampling routines, one of which is random sampling. Instead of computing $\mathsf{E}\left[\varphi\big(x(t)\big)\big|x(0)\right]$ exactly, the authors of [31] obtain an upper bound and provide conditions which make this upper bound converge to zero asymptotically. However, the conditions given in their main result on random sampling [31, Theorem 5] are seen to hold only for open-loop stable systems. To see this, consider the scalar system

$$\dot{x} = ax + u$$

and by choosing $u = \kappa x(\tau_{N_t})$, we consider the system $\dot{x} = F\dot{x}$, where

$$F := \begin{pmatrix} a+\kappa & -\kappa \\ a+\kappa & -\kappa \end{pmatrix}.$$

Employing the transformation $T = \left(\begin{smallmatrix} 1 & 0 \\ 1 & 1 \end{smallmatrix}\right)$ and defining $\overline{x} = T\overline{z}$, we see that

$$\mathrm{e}^{Ft} = \begin{pmatrix} (1-\kappa/a)\mathrm{e}^{at} + \kappa/a & 0 \\ -(1-\kappa/a)\mathrm{e}^{at} - \kappa/a & 0 \end{pmatrix}.$$

Let

$$\overline{M} := \begin{pmatrix} 1 & 0 \\ 0 & 0 \end{pmatrix} \mathrm{e}^{Ft} \begin{pmatrix} 1 & 0 \\ 0 & 0 \end{pmatrix} = \begin{pmatrix} ((1-\kappa/a)\mathrm{e}^{at} + \kappa/a) & 0 \\ 0 & 0 \end{pmatrix}.$$

According to [31, Theorem 5], the sufficient condition for asymptotic stability in the second moment is

$$\left\| \mathsf{E}\left[\overline{M}^{\top}\overline{M}\right] \right\| < 1.$$

However, for Poisson sampling with intensity λ, it is seen that

$$\begin{aligned} \mathsf{E}\left[((1-\kappa/a)\mathrm{e}^{at} + \kappa/a)^2\right] &= \lambda \int_0^{+\infty} ((1-\kappa/a)\mathrm{e}^{at} + \kappa/a)^2 \mathrm{e}^{-\lambda t}\,\mathrm{d}t \\ &= (1-\kappa/a)^2 \frac{\lambda}{\lambda - 2a} + \frac{\kappa^2}{a^2} + 2(\kappa/a)(1-\kappa/a)\frac{\lambda}{\lambda - a}. \end{aligned}$$

Note that the term on right-hand side is greater than 1 for each $\lambda > 2a$.[4] In fact, it is a decreasing function of λ, and

$$\lim_{\lambda \to +\infty} \mathsf{E}\left[((1-\kappa/a)\mathrm{e}^{at} + \kappa/a)^2\right] = 1.$$

[4] The necessity of the condition $\lambda > 2a$ for scalar linear systems with Poisson sampling is discussed in Sect. 6.2.

This shows that, even in such simple cases, we do not get $\left\| \mathsf{E}\left[\overline{M}^\top \overline{M}\right] \right\| < 1$ for arbitrarily large values of λ. This demonstrates the conservatism in the sufficient condition proposed in [31, Theorem 5], and hence it can be presumed that the problem of computing $\mathsf{E}\left[\varphi\big(x(t)\big)\middle|x(0)\right]$ did not get a positive response until the first decade of this century. One positive response to this question has been provided in [4], which we treat in greater detail in the next section. The authors of [4] provide necessary and sufficient conditions for mean-square stability of linear systems under random sampling for a rather general class of random processes. We examine closely and comment on their main result in Sect. 5.1. The techniques involved in [4] are quite different from the ones that are mainstream.

Before moving on, we mention a couple of additional references dealing with random sampling. The article [3] deals with control under random sampling: An optimal control problem with a quadratic instantaneous cost for linear controlled diffusions was studied in this particular work, but under the assumption that there are only finitely many sampling instants. The authors of the recent article [49] also limited their scope to a Lyapunov stable matrix A.

The preceding efforts involve hands-on calculations that are specific to linear system models and/or specific (and simple) sampling processes, with the exception of [4]. The connection between PDMPs and sampled-data control under random sampling discussed in Remark 2.1 immediately opens up the possibility of employing generator-based ideas in this context; our agenda for the next section will focus on this connection closely. In particular, we shall demonstrate in Sect. 5.2 that the main results of [4] can also be derived by employing the extended generator (8).

5 Equivalence of Different Stability Conditions for Linear PDMPs

Turning our attention to (15), and looking at this joint system with state $\overline{x} = (x^\top, e^\top)^\top$, it is possible to find necessary and sufficient conditions for asymptotic stability in second moment by computing $\mathsf{E}\left[\psi\big(\overline{x}(t)\big)\middle|\overline{x}(0)\right]$ for system (15), with ψ quadratic in $\overline{x}$. This is done in an explicit manner in [4], where the authors use the recursive Volterra integral equation to compute $\mathsf{E}\left[\|\overline{x}(t)\|^2 \middle|\overline{x}(0)\right]$. Another tool for analyzing the stability in second moment for system (15) was already revealed in Sect. 3 in the form of extended generator. After providing a quick overview of how $\mathsf{E}\left[\|\overline{x}(t)\|^2 \middle|\overline{x}(0)\right]$ is computed, we show the equivalence between the two approaches, which essentially establishes a converse Lyapunov theorem for (15) with Poisson renewal process.

5.1 Volterra Integral Approach

To analyze stability in second moment for system (15), it is observed that we can write [4, Proposition 6]

$$\mathsf{E}\left[\overline{x}^\top(t)\overline{Q}\overline{x}(t)\right] = \overline{x}_0^\top W(t)\overline{x}_0 \tag{23}$$

where the matrix-valued function $W : [0, +\infty[\to \mathbb{R}^{\overline{n}\times\overline{n}}$ satisfies the Volterra integral equation

$$W(t) = \mathcal{K}(W)(t) + H(t), \tag{24}$$

with $H(t) = \mathrm{e}^{F^\top t}\overline{Q}\mathrm{e}^{Ft}\mathrm{e}^{-\lambda t}$ for some positive-definite and symmetric $\overline{Q} \in \mathbb{R}^{\overline{n}\times\overline{n}}$, and $\lambda > 0$ being the intensity of the Poisson sampling process $\left(N_t\right)_{t\geqslant 0}$ so that the jump times τ_{N_t} in (14) have the property that $\left(\tau_{N_t} - \tau_{N_t-1}\right) \sim \mathrm{Exp}(\lambda)$. In (24), the operator $\mathcal{K} : \mathcal{C}^1(\mathbb{R}_{\geqslant 0}; \mathbb{R}^{\overline{n}\times\overline{n}}) \to \mathcal{C}^1(\mathbb{R}_{\geqslant 0}; \mathbb{R}^{\overline{n}\times\overline{n}})$ is given by

$$\mathcal{K}(W)(t) := \lambda \int_0^t \mathrm{e}^{F^\top s} G^\top W(t-s) G \mathrm{e}^{Fs} \mathrm{e}^{-\lambda s}\,\mathrm{d}s. \tag{25}$$

Due to (23), stability of (14) can be formulated in terms of the asymptotic properties of the matrix-valued function $W(t)$. In [4, Theorem 3], depending upon the stability notion under consideration, several conditions are provided which are equivalent to convergence of W in appropriate norms. For example, conditions for stochastic stability are equivalent to absolute convergence of $\int_0^{+\infty} W(s)\,ds$, and the conditions given for mean-square stability are equivalent to $W(t) \to 0$.

5.2 Connections Between the Extended Generator and Volterra Integral Techniques

In Sect. 3, we used the extended generator to obtain sufficient conditions for stability of nonlinear PDMPs. In case of linear systems (14), the same approach can be adopted while restricting attention to quadratic test functions. Since we now have a characterization of stability in terms of the function W given in (24), it is natural to ask whether we can establish necessary conditions for stability in second moment using the extended generator. To show that these approaches are equivalent for linear dynamics (15) and Poisson renewal processes, we have the following result.

Theorem 5.1 *Consider system (14) with $(N_t)_{t\geqslant 0}$ a Poisson process of intensity $\lambda > 0$. The following statements are equivalent:*

(S1) *System (14) is exponentially stable in second moment.*

(S2) *There exists a symmetric positive-definite matrix $\overline{Q} \in \mathbb{R}^{\overline{n} \times \overline{n}}$ such that the matrix-valued function W satisfying (24), with $H(t) = \mathrm{e}^{F^\top t}\overline{Q}\mathrm{e}^{Ft}\mathrm{e}^{-\lambda t}$, converges to zero exponentially as $t \to +\infty$.*

(S3) *There exists a symmetric, positive- definite matrix $\overline{P} \in \mathbb{R}^{\overline{n} \times \overline{n}}$ such that*

$$F^\top \overline{P} + \overline{P}\, F + \lambda(G^\top \overline{P}\, G - \overline{P}) < 0. \tag{26}$$

If we let $\psi(\overline{x}) := \overline{x}^\top \overline{P}\overline{x}$, then using the expression for $\mathcal{L}\psi(x, e)$ in (9), the inequality (26) is equivalently written as $\mathcal{L}\psi(x, e) < 0$, for each $(x, e) \in \mathbb{R}^{d \times d}$. A condition similar to (26) has also appeared in [5, Theorem 7]. Note that the result of Theorem 5.1 is of independent interest as it proves a converse Lyapunov theorem for a class of linear PDMPs which are exponentially stable in second moment. Establishing converse Lyapunov theorems for stochastic hybrid systems, in general, was identified as an open problem in [44, Section 8.4, Open Problem 4], and Theorem 5.1 provides a result in this direction for a particular class of stochastic hybrid systems. The nontrivial aspect of the proof of Theorem 5.1 relies on constructing $\overline{P}$ using the expression for W in (24).

Proof The equivalence between (S1) and (S2) follows directly from (23), where the latter is derived in [4, Proposition 6]. In the sequel, we prove the equivalence between (S2) and (S3), and for our purposes it is useful to recall that, using the properties of Volterra integral equation, W can be explicitly described by the expression

$$W(t) := \sum_{j=1}^{+\infty} \mathcal{K}^j(H)(t) + H(t). \tag{27}$$

Now, let us assume that (S3) holds, and from there we show that there is a matrix $\overline{Q}$ such that W satisfying (24), with $H(t) = \mathrm{e}^{F^\top t}\overline{Q}\mathrm{e}^{Ft}\mathrm{e}^{-\lambda t}$, converges to zero as t goes to infinity. Let $\overline{P}$ be the symmetric, positive-definite matrix satisfying (26), so there exists $\alpha > 0$ such that

$$F^\top \overline{P} + \overline{P}F + \lambda(G^\top \overline{P}G - \overline{P}) + \alpha\overline{P} < 0.$$

Take $\overline{Q} = \overline{P}$. Multiplying the last inequality by $\mathrm{e}^{F^\top t}$ from left, e^{Ft} from right, and the scalar $\mathrm{e}^{-\lambda t}$, we get

$$F^\top H + HF + \lambda(J - H) < -\alpha H \tag{28}$$

where we recall that $H(t) = \mathrm{e}^{F^\top t}\overline{Q}\mathrm{e}^{Ft}\mathrm{e}^{-\lambda t}$, and

$$J(t) := \mathrm{e}^{F^\top t} G^\top \overline{Q} G \mathrm{e}^{Ft}\mathrm{e}^{-\lambda t}.$$

With this choice of $\overline{Q}$ and H, let W be the function obtained from solving (27). To see that W converges to zero exponentially, we need the following lemma:

Lemma 5.2 *For the continuously differentiable matrix-valued function W given in* (27)*, it holds that*

$$\frac{\mathrm{d}}{\mathrm{d}t}W(t) = \sum_{j=1}^{+\infty} \mathcal{K}^j(F^\top H + HF - \lambda H)(t) + \sum_{j=0}^{+\infty} \lambda \mathcal{K}^j(J)(t) \\ + F^\top H(t) + H(t)F - \lambda H(t). \tag{29}$$

The proof of this lemma is given in Appendix B. Combined with (28), and using the expression for W in (27), this lemma immediately yields

$$\dot{W}(t) \leq -\alpha W(t)$$

from which the exponential convergence of W follows.

Next, we show that (S2) implies the existence of matrix $\overline{P}$ such that (S3) holds. For this implication to hold, the important relation that we need to develop is

$$\frac{\mathrm{d}}{\mathrm{d}t}W(t) = F^\top W(t) + W(t)F - \lambda W(t) + \lambda G^\top W(t)G, \quad t \geqslant 0. \tag{30}$$

Indeed, if (30) holds, then by letting,

$$\overline{P} := \lim_{t\to+\infty} \int_0^t W(s)\,\mathrm{d}s,$$

it is seen that

$$\begin{aligned} F^\top\overline{P} + \overline{P}F + \lambda(G^\top\overline{P}G - \overline{P}) &= \lim_{t\to+\infty} \int_0^t \frac{\mathrm{d}}{\mathrm{d}s}W(s)\,\mathrm{d}s, \\ &= \lim_{t\to+\infty} W(t) - W(0) \\ &= -\overline{Q} \end{aligned}$$

where we used the fact that $\lim_{t\to+\infty} W(t) = 0$ because of (S2). The limit in the definition of the matrix $\overline{P}$ is well-defined because W converges to zero exponentially. The matrix $\overline{P}$ is also seen to be symmetric and positive definite. To show this, we first observe from (27) that, for each $s \geqslant 0$, $W(s)$ is symmetric and $W(s) \geqslant H(s)$. Suppose, ad absurdum, that $\overline{P}$ is not positive definite; then, there exists $\overline{x} \in \mathbb{R}^{\overline{n}}$, $\overline{x} \neq 0$, such that

$$\begin{aligned} 0 = \overline{x}^\top\overline{P}\overline{x} &= \lim_{t\to+\infty} \int_0^t \overline{x}^\top W(s)\overline{x}\,\mathrm{d}s \\ &\geqslant \lim_{t\to+\infty} \int_0^t \overline{x}^\top H(s)\overline{x}\,\mathrm{d}s = \lim_{t\to+\infty} \int_0^t \overline{x}^\top \mathrm{e}^{sF^\top}\overline{Q}\mathrm{e}^{sF}\mathrm{e}^{-\lambda s}\overline{x}\,\mathrm{d}s. \end{aligned}$$

Since $\overline{Q}$ is positive definite, the last inequality suggests that $\mathrm{e}^{sF}\overline{x} = 0$ for every $s \geqslant 0$, and hence $\overline{x} = 0$; a contradiction.

So, the focus in the remainder of the proof is on proving (30). We already have an expression for $\frac{\mathrm{d}}{\mathrm{d}t}W$ in Lemma 5.2. To simplify the terms on the right-hand side of (29), we introduce the following lemma:

Lemma 5.3 *For each $j \geqslant 1$, we have*

$$\begin{aligned}\mathcal{K}^j(F^\top H + HF - \lambda H) + \lambda\mathcal{K}^{j-1}(J)(t) = \lambda G^\top \mathcal{K}^{j-1}(H)(t)G \\ + F^\top \mathcal{K}^j(H)(t) + \mathcal{K}^j(H)(t)F - \lambda\mathcal{K}^j(H)(t).\end{aligned} \tag{31}$$

Again, the proof of this lemma is provided in Appendix B. Combining the statements of Lemmas 5.2 and 5.3, we get

$$\begin{aligned}\frac{\mathrm{d}}{\mathrm{d}t}W(t) = \sum_{j=1}^{+\infty} \lambda G^\top \mathcal{K}^{j-1}(H)(t)G + F^\top \mathcal{K}^j(H)(t) + \mathcal{K}^j(H)(t)F - \lambda\mathcal{K}^j(H)(t) \\ + F^\top H(t) + H(t)F - \lambda H(t).\end{aligned} \tag{32}$$

On the other hand, it follows from the expression for W in (27) that

$$\begin{aligned}F^\top W(t) + W(t)F - \lambda W(t) = \sum_{j=1}^{+\infty} F^\top \mathcal{K}^j(H)(t) + \mathcal{K}^j(H)(t)F - \lambda\mathcal{K}^j(H)(t) \\ + F^\top H(t) + H(t)F - \lambda H(t).\end{aligned} \tag{33}$$

Substituting (33) in (32), and using the notation $\mathcal{K}^0$ to denote the identity operator, we get

$$\frac{\mathrm{d}}{\mathrm{d}t}W(t) = F^\top W(t) + W(t)F - \lambda W(t) + \lambda G^\top \left(\sum_{j=1}^{+\infty} \mathcal{K}^{j-1}(H)(t)\right) G.$$

The desired Eq. (30) now follows by recalling the definition of W from (27). □

5.3 Exponential Stability Under Random Sampling

Now that we have established the necessary and sufficient conditions for stability of the randomly sampled-data system (14) in Theorem 5.1, we can obtain refined estimates on the mean sampling rate λ for stability in second moment to solve Problem 1. We will only work out the estimates that can be obtained from the statement

(S3). A direct way to obtain a lower bound on the mean sampling rate is by solving the inequality (26) in λ and $\overline{P}$, for a given $K \in \mathbb{R}^{m\times d}$. But, since (26) is a bilinear matrix inequality, and hence nonconvex, it is difficult to obtain analytical bounds on λ for feasibility. To overcome this issue, we choose to work with a block diagonal $\overline{P}$ and proceed with computing the lower bounds on λ analytically with such $\overline{P}$. We fix K to be any matrix which makes $A + BK$ Hurwitz, and with this assumption, we show that by choosing λ large enough as a function of the matrices A, B, K, the resulting system is asymptotically stable in second moment.

Theorem 5.4 *Consider the system (14), with $\left(N_t\right)_{t\geqslant 0}$ a Poisson process of intensity λ. Assume that there exist $\alpha > 0$, a matrix $K \in \mathbb{R}^{d\times m}$ and a symmetric positive-definite matrix $P \in \mathbb{R}^{d\times d}$ satisfying*

$$(A + BK)^\top P + P(A + BK) \le -\alpha P. \tag{34}$$

For $\mathbb{R}^d \ni y \mapsto V(y) := \langle y, Py\rangle$, there exist constants C_0, C_1, such that

$$\begin{aligned} &\textit{for every } \rho \in\,]0, \alpha[,\ \textit{for every } \lambda > \rho + C_0 + \frac{C_1}{(\alpha - \rho)}, \\ &\textit{for every } x(0) \in \mathbb{R}^d, \textit{and for every } t \geqslant 0 \end{aligned} \tag{35}$$

we have

$$\mathsf{E}\left[V(x(t))\middle|x(0)\right] \le\ V(x(0)) \exp{(-\rho t)}. \tag{36}$$

In particular, for all $\lambda > 0$ sufficiently large, the closed-loop system (14) is globally exponentially stable in the second moment.

Remark 5.5 It is seen from the statement of the theorem that, even if we choose the decay rate ρ to be close to α, it is possible to achieve it by choosing the sampling rate λ to be sufficiently large. In other words, with faster sampling rates, we approach the performance of the continuous-time system.

Remark 5.6 In the proof of Theorem 5.4, we compute the constants C_0 and C_1 in (35) as functions of the matrices A, B, K and P satisfying (34). By letting $\widetilde{Y} := P^{1/2}BKP^{-1/2}$, and $\widetilde{A} := P^{1/2}AP^{-1/2}$, it turns out that we can choose

$$C_0 := \sigma_{\max}\big(-\widetilde{Y} - \widetilde{Y}^\top\big) \quad \text{and} \tag{37a}$$

$$C_1 := \sigma_{\max}\Big(\big(\widetilde{Y}^\top - \widetilde{Y} - \widetilde{A}\big)\big(\widetilde{Y} - \widetilde{Y}^\top - \widetilde{A}^\top\big)\Big), \tag{37b}$$

where, for a given matrix M, $\sigma_{\max}(M)$ denotes the maximum eigenvalue of a matrix M. In fact, it is possible to show that the claim of Theorem 5.4 holds whenever

$$\lambda - \rho > \sigma_{\max}\Big(\frac{1}{\alpha - \rho}\big(\widetilde{Y}^\top - \widetilde{Y} - \widetilde{A}\big)\big(\widetilde{Y} - \widetilde{Y}^\top - \widetilde{A}^\top\big) - \widetilde{Y} - \widetilde{Y}^\top\Big).$$

Corollary 5.7 *Let $K = -R^{-1}B^\top P$, where R and P are symmetric positive-definite matrices which satisfy, for some $\alpha > 0$, the relation*

$$\Big(A + \frac{\alpha}{2}I\Big)^\top P + P\Big(A + \frac{\alpha}{2}I\Big) - 2PBR^{-1}B^\top P \le 0. \tag{38}$$

For each $\rho \in]0, \alpha[$, if λ satisfies (35) with

$$C_0 := 2\sigma_{\max}\big(P^{1/2}BR^{-1}B^\top P^{1/2}\big) \quad \textit{and}$$
$$C_1 := \sigma_{\max}\big(P^{1/2}AP^{-1}A^\top P^{1/2}\big),$$

then (36) holds.

The bounds in Corollary 5.7 are obtained by observing that the choice of $K = -R^{-1}B^\top P$ leads to $\widetilde{Y} = \widetilde{Y}^\top$, which simplifies the expression for C_0 and C_1 to some extent.

Proof of Theorem 5.4 We choose a quadratic function $\psi : \mathbb{R}^d \times \mathbb{R}^d \to \mathbb{R}_{\geqslant 0}$ of the form

$$(x, e) \mapsto \psi(x, e) := \langle x, P_x x\rangle + \langle e, P_e e\rangle\,, \tag{39}$$

where P_x and P_e are symmetric positive-definite matrices. Using (9) from Proposition 2.2, we obtain

$$\begin{aligned}
\mathcal{L}\psi(x, e) &= \big\langle (P_x + P_x^\top)x + (P_e + P_e^\top)e, (A + BK)x - BKe\big\rangle \\
&\quad + \lambda\left(\langle x, P_x x\rangle - \langle e, P_e e\rangle - \langle x, P_x x\rangle\right) \\
&= \big\langle (P_e + P_e^\top)e, (A + BK)x - BKe\big\rangle \\
&\qquad + \big\langle (P_x + P_x^\top)x, (A + BK)x - BKe\big\rangle - \lambda\,\langle e, P_e e\rangle \\
&= -\lambda\,\langle e, P_e e\rangle + \big\langle x, P_x(A + BK)x + (A + BK)^\top P_x x\big\rangle \\
&\quad - \big\langle e, (P_e BK + K^\top B^\top P_e)e\big\rangle + 2\,\langle e, P_e(A + BK)x\rangle \\
&\quad - 2\,\langle x, P_x BKe\rangle\,.
\end{aligned}$$

Letting $P_x = P_e = P$ and $A_K := A + BK$, we get

$$\begin{aligned}
&\mathcal{L}\psi(x, e) \\
&= -\left\langle \begin{pmatrix} x \\ e\end{pmatrix}, \begin{pmatrix} -PA_K - A_K^\top P & PBK - A_K^\top P \\ -PA_K + K^\top B^\top P & \lambda P + PBK + K^\top B^\top P\end{pmatrix}\begin{pmatrix} x \\ e\end{pmatrix}\right\rangle \\
&\le -\left\langle \begin{pmatrix} x \\ e\end{pmatrix}, \underbrace{\begin{pmatrix} \alpha P & PBK - A_K^\top P \\ -PA_K + K^\top B^\top P & \lambda P + PBK + K^\top B^\top P\end{pmatrix}}_{=:M(\lambda)}\begin{pmatrix} x \\ e\end{pmatrix}\right\rangle.
\end{aligned}$$

We next analyze the matrix $M(\lambda)$ and show that for λ large enough, $M(\lambda)$ is positive definite and see how the minimum eigenvalue of $M(\lambda)$ varies with λ. We first write

$M(\lambda)$ as

$$M(\lambda) := M_0 + M_1(\lambda)$$

where for a fixed $\rho \in]0, \alpha[$,

$$M_0 := \begin{pmatrix} \rho P & 0 \\ 0 & \rho P \end{pmatrix} \tag{40}$$

and

$$M_1(\lambda) := \begin{pmatrix} (\alpha - \rho)P & PBK - A_K^\top P \\ -PA_K + K^\top B^\top P & (\lambda - \rho)P + PBK + K^\top B^\top P \end{pmatrix}.$$

Using Schur complements [48, §7.4] and introducing the notation $Y := PBK$ it is seen that

$$\begin{aligned} & M_1(\lambda) \geqslant 0 \\ \Leftrightarrow\ & (\lambda - \rho)P + Y + Y^\top \geqslant \frac{\big(Y^\top - Y - PA\big)P^{-1}\big(Y - Y^\top - A^\top P\big)}{\alpha - \rho}. \end{aligned}$$

Let $P^{1/2}$ denote the positive square root of P. Also, let $\widetilde{Y} := P^{1/2} BKP^{-1/2}$, and $\widetilde{A} := P^{1/2} AP^{-1/2}$. Then, conjugation by $P^{-1/2}$ yields

$$\begin{aligned} & M_1(\lambda) \geqslant 0 \\ \Leftrightarrow\ & (\lambda - \rho)I + \widetilde{Y} + \widetilde{Y}^\top \geqslant \frac{1}{\alpha - \rho}\big(\widetilde{Y}^\top - \widetilde{Y} - \widetilde{A}\big)\big(\widetilde{Y} - \widetilde{Y}^\top - \widetilde{A}^\top\big). \end{aligned}$$

Using Weyl's inequality [21, Theorem 4.3.1], we obtain

$$\begin{aligned} & \sigma_{\max}\left(\frac{1}{\alpha - \rho}\big(\widetilde{Y}^\top - \widetilde{Y} - \widetilde{A}\big)\big(\widetilde{Y} - \widetilde{Y}^\top - \widetilde{A}^\top\big) - (\widetilde{Y} + \widetilde{Y}^\top)\right) \\ & \le \sigma_{\max}\left(\frac{1}{\alpha - \rho}\big(\widetilde{Y}^\top - \widetilde{Y} - \widetilde{A}\big)\big(\widetilde{Y} - \widetilde{Y}^\top - \widetilde{A}^\top\big)\right) + \sigma_{\max}(-\widetilde{Y} - \widetilde{Y}^\top) \\ & = \frac{1}{\alpha - \rho}\sigma_{\max}\left(\big(\widetilde{Y}^\top - \widetilde{Y} - \widetilde{A}\big)\big(\widetilde{Y} - \widetilde{Y}^\top - \widetilde{A}^\top\big)\right) + \sigma_{\max}(-\widetilde{Y} - \widetilde{Y}^\top) \\ & =: \frac{1}{\alpha - \rho}C_1 + C_0 \end{aligned}$$

where we introduced the constants C_0, C_1 given in (37). It is now observed that $M_1 \geqslant 0$ for each $\lambda > \rho + C_0 + C_1/(\alpha - \rho)$, and hence

$$\mathcal{L}\psi(x, e) \le -\left\langle \begin{pmatrix} x \\ e \end{pmatrix}, M_0 \begin{pmatrix} x \\ e \end{pmatrix} \right\rangle = -\rho\,\psi(x, e).$$

The assertion of Theorem 5.4 follows. □

It must be noted that the condition (35) is only sufficient for stability in second moment because in the notation of (S3) of Theorem 5.1, the proof was worked out by choosing $\overline{P} = \begin{pmatrix} P & 0 \\ 0 & P \end{pmatrix}$. This choice indeed makes our estimates of λ conservative. In the next section, we study stability of closed-loop systems for smaller values of λ by addressing the converse question of designing static feedbacks for linear systems.

6 Converse Question and Feedback Design

In contrast to finding lower bounds on the sampling rate for a given feedback law in previous sections, we are now interested in designing the feedback laws for a fixed sampling rate. The problem of interest is thus formalized as follows:

Problem 2 Consider the system (14), with $(N_t)_{t\geqslant 0}$ a Poisson process of intensity λ. If $\lambda > 0$ is given, does there exist a matrix $K \in \mathbb{R}^{m\times d}$ such that (14) is globally exponentially stable in second moment?

Preparatory to addressing this problem, we first observe that the search space for the feedback gain K is constrained by the sampling rate even in the setting of deterministic sampling—see Sect. 6.1 for the relevant discussion. Moreover, in the setting of Poisson sampling, there is a lower bound on the sampling rate that must be satisfied for the expectation to be well defined; see Sect. 6.2 for the corresponding details. These two observations are then employed to provide a partial answer to Problem 2.

6.1 Using the Scalar Deterministic Case as a Guideline

Before addressing this question with random sampling, let us have a quick look at the deterministic sampling case and observe how one would choose a feedback gain in that case. Consider the scalar system

$$\dot{x}(t) = ax(t) + u(t), \quad t \geqslant 0,$$

with a given $a > 0$. Our objective is to asymptotically stabilize this system at the origin, and the state measurements are available only periodically at $(\tau_i)_{i\in\mathbb{N}^*} \subset [0, +\infty[$, where $\tau_{i+1} - \tau_i = T$ for some fixed $T > 0$; in other words, $\tau_n = nT$. We aim to design a controller $u(t) = \kappa x(\tau_{N_t})$, with an appropriately chosen κ depending on the sampling period T. Elementary calculations yield

$$
\begin{aligned}
x(\tau_{i+1}) &= \exp(aT)x(\tau_i) + \int_{\tau_i}^{\tau_{i+1}} \exp(a(\tau_{i+1}-s))\kappa x(\tau_i)\,\mathrm{d}s \\
&= \Big(\exp(aT) + \frac{\kappa}{a}\big(\exp(aT)-1\big)\Big)x(\tau_i),
\end{aligned}
$$

and for a fixed sampling period $T > 0$, the closed-loop system is asymptotically stable if and only if the sequence $(x(\tau_n))_{n\in\mathbb{N}^*}$ converges to 0. The latter holds if and only if

$$
\left|\exp(aT) + \frac{\kappa}{a}\big(\exp(aT)-1\big)\right| < 1,
$$

or equivalently, if and only if

$$
-a\Big(\frac{\exp(aT)+1}{\exp(aT)-1}\Big) < \kappa < -a.
$$

We observe two key facts:

- The inequality $\kappa < -a$ is necessary for the stability of the continuous-time system. The other inequality gives a lower bound on the value of κ, and shows that for a fixed sampling rate, one can *not* choose $|\kappa|$ to be very large.
- On the one hand, as T goes to zero (the case of fast sampling), this lower bound goes to $-\infty$. On the other hand, as T grows large (the case of slow sampling), this lower bound approaches $-a$ from below, and the admissible set of the stabilizing gain κ becomes smaller.

In dimensions larger than 1, the problem of selecting a suitable control gain K becomes more delicate, as we shall momentarily see.

6.2 *Necessary Lower Bounds for the Sampling Rate*

We turn our attention back to the system

$$
\dot{x}(t) = Ax(t) + BKx(\tau_{N_t}), \qquad x(0) \text{ given}, \quad t \geqslant 0, \tag{41}
$$

where we recall that $\big(N_t\big)_{t\geqslant 0}$ defined in (3) is a Poisson process of intensity λ which determines the sampling times. We assume for the sake of simplicity that A is in its complex-Jordan normal form and that it is non-singular. It can be easily verified that, for each sample path, and $i \in \mathbb{N}^*$, we have

$$
x(\tau_{i+1}) = A^{-1}\Big(\mathrm{e}^{A(\tau_{i+1}-\tau_i)}\big(A+BK\big) - BK\Big)x(\tau_i). \tag{42}
$$

If the linear system (41) is exponentially stable in the second moment, then the discrete-time system (42) must also be exponentially stable in the second moment,[5] and therefore, there exist [34, Theorem 9.4.2] a symmetric positive definite matrix $P_{\mathrm{d}} \in \mathbb{R}^{d\times d}$ and $\gamma \in [0,1[$ such that for each $i \in \mathbb{N}^*$,

$$\mathsf{E}\left[\langle x(\tau_{i+1}), P_{\mathrm{d}}x(\tau_{i+1})\rangle \,\middle|\, x(\tau_i)\right] \le \gamma \mathsf{E}\left[\langle x(\tau_i), P_{\mathrm{d}}x(\tau_i)\rangle \,\middle|\, x(\tau_i)\right].$$

With $A_K := (A+BK)$ and $\tilde{P}_{\mathrm{d}} := A^{-1}P_{\mathrm{d}}A^{-1}$, time-invariance of the data leads to

$$\begin{aligned} A_K^\top \mathsf{E}\left[\mathrm{e}^{SA^\top}\tilde{P}_{\mathrm{d}}\mathrm{e}^{SA}\right]A_K - A_K^\top \mathsf{E}\left[\mathrm{e}^{SA^\top}\right]\tilde{P}_{\mathrm{d}}BK \\ - (BK)^\top \tilde{P}_{\mathrm{d}}\mathsf{E}\left[\mathrm{e}^{SA}\right]A_K + (BK)^\top\tilde{P}_{\mathrm{d}}BK \le \gamma A\tilde{P}_{\mathrm{d}}A, \end{aligned}$$

where S is an exponential random variable with parameter λ. The matrix on the left-hand side is well defined if and only if $\mathsf{E}\left[\mathrm{e}^{SA^\top}\tilde{P}_{\mathrm{d}}\mathrm{e}^{SA}\right]$ and $\mathsf{E}\left[\mathrm{e}^{SA}\right]$ are well-defined.

The (j,k)th entry of the matrix $\mathsf{E}\left[\mathrm{e}^{SA^\top}\tilde{P}_{\mathrm{d}}\mathrm{e}^{SA}\right]$ is

$$\mathsf{E}\left[\sum_{\ell=1}^{d}\sum_{m=1}^{d}(\mathrm{e}^{SA^\top})_{j\ell}\big(\tilde{P}_{\mathrm{d}}\big)_{\ell m}(\mathrm{e}^{SA})_{mk}\right].$$

Since e^{SA} is in the block-diagonal form with the eigenvalues of A on the diagonal, this expectation is of the form $\mathsf{E}\left[p_{jk}(S)\mathrm{e}^{S(\sigma_j+\sigma_k)}\right]$ for $1 \le j,k \le d$, where σ_j, σ_k are the jth and kth diagonal entries (eigenvalues) of A, and $p_{jk}(\cdot)$ is a polynomial of degree at most $2d$. This expectation is finite only if $\lambda > \Re\sigma_j + \Re\sigma_k$, and therefore, $\mathsf{E}\left[\mathrm{e}^{SA^\top}\tilde{P}_{\mathrm{d}}\mathrm{e}^{SA}\right]$ is well-defined whenever $\lambda > 2\max\{\Re\sigma_j(A) \mid j = 1,\dots,d\}$. Similarly, $\mathsf{E}\left[\mathrm{e}^{SA}\right]$ is well-defined only for $\lambda > \max\{\Re\sigma_j(A) \mid j=1,\dots,d\}$.

We conclude from this discussion that

$$\lambda > 2\max\{\Re\sigma_j(A) \mid j = 1,\dots,d\}$$

is a necessary condition for asymptotic stability in the second moment of the sampled process $\big(x(\tau_n)\big)_{n\in\mathbb{N}^*}$, and seek to resolve the following conjecture:

Conjecture 6.1 *Consider the system* (41)*, where* $\big(N_t\big)_{t\geqslant 0}$ *is a Poisson process of given intensity* $\lambda > 0$*. For each* $\lambda > 2\max\{\Re\sigma_j(A) \mid j=1,\dots,d\}$*, there exists a feedback matrix* $K \in \mathbb{R}^{m\times d}$ *such that* (41) *is globally asymptotically stable in the second moment.*

[5] The definition of exponential stability in the second moment for the discrete-time case is analogous to the continuous-time version that we have quoted above.

6.3 The Scalar Case with Poisson Sampling

We proceed to verify that the Conjecture 6.1 holds in the scalar case.

Proposition 6.2 *Conjecture 6.1 holds when the system dimension $d = 1$.*

Proof Without loss of generality, we look at the scalar plant

$$\dot{x}(t) = ax(t) + u(t)$$

with $a > 0$ and are interested in choosing the scalar feedback gain κ such that $u(t) = \kappa x(\tau_{N_t})$, $t \geqslant 0$, results in mean-squared asymptotic stability. Recalling that $e(t) = x(t) - x(\tau_{N_t})$ for $t \geqslant 0$, we pick

$$\psi(x, e) := px^2 + e^2$$

for some $p > 0$ to be specified later. Using (9), we get

$$\mathcal{L}\psi(x, e) = -\left\langle \begin{pmatrix} x \\ e \end{pmatrix}, \underbrace{\begin{pmatrix} -2(a+\kappa)p & p\kappa - (a+\kappa) \\ p\kappa - (a+\kappa) & \lambda + 2\kappa \end{pmatrix}}_{=:M} \begin{pmatrix} x \\ e \end{pmatrix} \right\rangle.$$

If we show that there exist $p > 0$ and $\kappa < 0$ such that M is positive definite, our proof will be complete. Toward this end, we first look at the determinant of M:

$$\begin{aligned} \det(M) &= -2p(a+\kappa)(\lambda + 2\kappa) - (a+\kappa)^2 - p^2\kappa^2 + 2p\kappa(a+\kappa) \\ &= -(p+1)^2(a+\kappa)^2 + 2p(a+\kappa)(ap + a - \lambda) - a^2p^2. \end{aligned}$$

Defining $\theta := -(a+\kappa)$, we observe that $\det(M) > 0$ if and only if

$$(p+1)^2\theta^2 - 2p\theta(ap + a - \lambda) + a^2p^2 < 0.$$

The left-hand side of the inequality is a convex function of θ, and it attains its global minimum at

$$\theta^* = \frac{p(\lambda - a(1+p))}{(p+1)^2}.$$

It is then readily verified that the value of $\det(M)$ with $\theta = \theta^*$ is

$$\det(M_{\theta=\theta^*}) = \frac{p^2(\lambda - a(p+1))^2}{(p+1)^2} - a^2p^2,$$

so that $\det(M_{\theta=\theta^*}) > 0$ whenever

$$0 < p < \frac{\delta}{2a}, \quad \text{where } \delta := \lambda - 2a. \tag{43}$$

Fixing $\theta = \theta^*$ and letting p satisfy (43), we next look at the trace of M:

$$\begin{aligned}\operatorname{trace}(M_{\theta=\theta^*}) &= \lambda - 2a + 2\theta^*(p-1)\\ &= \delta + 2p\frac{\lambda - a(p+1)}{(p+1)^2}(p-1).\end{aligned}$$

Since $\operatorname{trace}(M_{\theta=\theta^*})$ is a continuous function of p and $\operatorname{trace}(M) = \delta > 0$ when $p = 0$, it follows that for $p > 0$ sufficiently small, it is possible to make both $\operatorname{trace}(M)$ and $\det(M)$ strictly positive. The resulting feedback law is

$$\kappa = -a - \frac{p(2a - \delta - a(1+p))}{(p+1)^2},$$

with $p > 0$ chosen such that $\operatorname{trace}(M_{\theta=\theta^*}) > 0$. The proof is complete. □

Remark 6.3 In the proof of Proposition 6.2 we selected the function ψ from (39) with $P_x = p$ and $P_e = 1$. An interesting observation is that if we select $P_x = P_e$ (as we did in the proof of Theorem 5.4), and λ is fixed, it is not possible to choose a feedback gain K such that $\mathcal{L}\psi(x,e) < 0$. To see this, we observe again in the scalar case that by letting $p_x = p_e = p$,

$$\mathcal{L}\psi(x,e) = -\left\langle \begin{pmatrix} x \\ e \end{pmatrix}, \begin{pmatrix} -2(a+\kappa)p & -ap \\ -ap & (\lambda + 2\kappa)p \end{pmatrix} \begin{pmatrix} x \\ e \end{pmatrix} \right\rangle.$$

We can choose $\kappa < -a$ so that both the diagonal terms of the matrix become negative, and by looking at the determinant of the matrix, it is seen that $\mathcal{L}\psi(x,e) < 0$, if and only if,

$$2\theta(\lambda - 2a - 2\theta) > a^2,$$

where $\theta = -(a+\kappa) > 0$. For a given value of a, one can find $\lambda > 2a$, such that the foregoing inequality is infeasible, regardless of the values of θ, or κ.

6.4 The Multidimensional Case

We employ the guidelines from the previous subsections to address Conjecture 6.1 for systems with dimension greater than 1. As already mentioned, our results here are not quite complete, and we require an additional assumption on the class of linear control systems:

Assumption 1 The matrix pair (A, B) is such that, there exist positive-definite matrices R and P, which solve the algebraic Riccati equation

$$A^\top P + PA - 2PBR^{-1}B^\top P = -\alpha P, \tag{44}$$

and $(A - BR^{-1}B^\top P)$ is Hurwitz. Moreover, the matrix P has the property that for some $C > 0$ and $p > \frac{2}{3}$,

$$\lim_{\alpha \downarrow 0} \frac{\sigma_{\max}(P)}{\alpha^p} \le C. \tag{45}$$

Assumption 1 requires that $\sigma_{\max}(P) = O(\alpha^p)$ when $\alpha \downarrow 0$. There exist linear systems that satisfy this Assumption; indeed, consider A and B given by

$$A = \begin{pmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 0 \end{pmatrix}, \quad B = \begin{pmatrix} 1 & 0 \\ 0 & 0 \\ 0 & 1 \end{pmatrix}, \tag{46}$$

and choose $R = 2\,I$, with I denoting the identity matrix (of appropriate dimension). Then (44) admits a unique solution P, with $(A - BR^{-1}B^\top P)$ Hurwitz, and the (i, j)th entry of P has the form

$$[P]_{ij} = \frac{p_{ij}(\alpha)}{1 + \alpha^4}$$

where p_{ij} are functions satisfying $\lim_{\alpha \downarrow 0} \frac{p_{ij}(\alpha)}{\alpha} = 0$ when $(i, j) \neq (3, 3)$, and for $(i, j) = (3, 3)$ we have $\lim_{\alpha \downarrow 0} \frac{p_{ij}(\alpha)}{\alpha} = 3$. A crisp characterization of the class of systems that satisfy Assumption 1 is under investigation.

Remark 6.4 System (46) is a particular example of null-controllable systems where the eigenvalues of A are on the imaginary axis. In general, we do not expect Assumption 1 to hold for systems with eigenvalues of A in open right-half complex plane. This can be seen for the scalar systems $\dot{x} = ax + u$, for which the solution of (44) with $R = 1$ is $p = 2a + \alpha$, and clearly (45) holds only with $a = 0$ for $0 < p \le 1$.

The following Theorem provides a recipe for designing feedback controllers under Assumption 1.

Theorem 6.5 *Consider the system* (41) *where* $\left(N_t\right)_{t \geqslant 0}$ *is a Poisson process of given intensity* $\lambda > 0$*, and suppose that Assumption 1 holds. Then there exists* $\alpha > 0$ *(sufficiently small) such that the feedback gain*

$$K = -R^{-1}B^\top P \quad \textit{with } P \textit{ solving} (44)$$

renders the system (41) *globally asymptotically stable in the second moment.*

Proof of Theorem 6.5 For $\alpha > 0$ we let P denote the solution of (44), and choose

$$\psi(x, e) := \eta_e \langle e, Pe \rangle + \eta_x \langle x, Px \rangle \quad \text{for } (x, e) \in \mathbb{R}^d \times \mathbb{R}^d,$$

where the positive scalars η_e, η_x will be specified later. The expression in (9) with the above choice of ψ yields

$$\mathcal{L}\psi(e,x) = -\left\langle \begin{pmatrix} x \\ e \end{pmatrix}, M(\alpha,\lambda)\begin{pmatrix} x \\ e \end{pmatrix} \right\rangle,$$

where

$$M(\alpha,\lambda) := \begin{pmatrix} -\eta_x(A_K^\top P + PA_K) & \eta_x PBK - \eta_e A_K^\top P \\ \eta_x K^\top B^\top P - \eta_e PA_K & \eta_e(\lambda P + PBK + K^\top B^\top P) \end{pmatrix},$$

in which $A_K = A + BK$, and dependence on α is through the matrix P. It follows that M is positive definite if M_0 and M_1 are positive definite, where

$$M_0 := \begin{pmatrix} \frac{\alpha}{2}\eta_x P & -\eta_e A_K^\top P \\ -\eta_e PA_K & \frac{\lambda}{2}\eta_e P \end{pmatrix}$$
$$M_1 := \begin{pmatrix} \frac{\alpha}{2}\eta_x P & \eta_x PBK \\ \eta_x K^\top B^\top P & \frac{\lambda}{2}\eta_e P + \eta_e PBK + \eta_e K^\top B^\top P \end{pmatrix}.$$

We first treat M_0. Using Schur complements [48, §7.4] followed by conjugation with $\eta_e^{-1/2}P^{-1/2}$, we get

$$M_0 > 0 \Leftrightarrow \frac{\lambda}{2}I > 2\frac{\eta_e}{\eta_x\alpha}P^{1/2}(A+BK)P^{-1}(A+BK)^\top P^{1/2}. \tag{47}$$

In view of Assumption 1, for a $p > \frac{2}{3}$ satisfying $\sigma_{\max}(P) = O(\alpha^p)$, we pick $\varepsilon > 0$ such that $0 < \varepsilon < p - \frac{2}{3}$, and select $\eta_e, \eta_x > 0$ such that

$$\frac{\eta_e}{\eta_x} = O(\alpha^{1+\varepsilon}). \tag{48}$$

By letting $\alpha \downarrow 0$, we see that $\sigma_{\max}(P^{1/2}) = O(\alpha^{p/2})$, which also yields that $P^{1/2}(A + BK)P^{-1}(A+BK)^\top P^{1/2} = O(1)$. Thus, the term on the right-hand side of the inequality (47) is bounded by $O(\alpha^\varepsilon)$. This shows that for α sufficiently small, $M_0 > 0$.

We next analyze M_1. Substituting $K = -R^{-1}B^\top P$ into M_1, using Schur complements [48, §7.4], and conjugating by $\eta_e^{-1/2}P^{-1/2}$, we get

$$\begin{aligned} &M_1 > 0 \quad \Leftrightarrow \\ &\frac{\lambda}{2}I > 2\alpha P^{1/2}BR^{-1}B^\top P^{1/2} + 2\frac{\eta_x}{\eta_e\alpha}P^{1/2}BR^{-1}B^\top P^2BR^{-1}B^\top P^{1/2}. \end{aligned} \tag{49}$$

Letting $\alpha \downarrow 0$, in view of Assumption 1 we have $\sigma_{\max}(P) = O(\alpha^p)$. The first term on the right-hand side is $O(\alpha^{p+1})$. For our choice of η_e and η_x in (48), we get

$$\frac{\eta_x}{\eta_e\alpha} = O(\alpha^{-2-\varepsilon}). \tag{50}$$

This way, the second term on the right-hand side of the inequality (49) is $O(\alpha^{3p-2-\varepsilon})$, which under the assumption $p > \frac{2}{3} + \varepsilon$, converges to zero as $\alpha \downarrow 0$. We conclude that M_1, and hence $M = M_0 + M_1$, are positive definite for sufficiently small $\alpha > 0$.

7 Conclusions

This chapter provided an overview on the problem of stabilization of deterministic control systems under random sampling. Although the problem was first introduced almost 60 years ago, the earlier efforts did not create many inroads. The use of modern tools from the literature on stochastic systems has indeed brought a constructive solution to this problem. In particular, this chapter provided the solution to this problem using the extended generator and Volterra integral techniques, and also developed connections between these two approaches. One particular question that needs further investigation is the design of feedback laws for fixed sampling rates. In this direction, Conjecture 6.1 is shown to hold for scalar systems and to some extent for multidimensional systems under a strong assumption. Investigating design techniques for constructing feedback gains in linear case for given sampling rates is indeed relevant for several applications.

As it is naturally the case, the problem has been studied with more depth in the case of linear systems which lead to Theorem 5.1 and quantitative estimates in Theorem 5.4. Extending such results for the case when the sampling process in not necessarily Poisson, but governed by some other distribution needs to be investigated. In general, one can also apply the extended generator approach to the case where transition rates are state dependent and locally bounded [20], but the stability conditions need to be worked out more explicitly for such cases. Another set of problems that emerges from these results is to develop their analogue counterparts for nonlinear systems. It is not immediately clear how the Volterra integral technique used in Theorem 5.1 could be generalized in nonlinear setting. Hence, it needs to be seen whether a converse Lyapunov theorem can be proven for nonlinear PDMPs. Also, at this moment, Theorem 5.4 shows that faster sampling in the limit leads to the same convergence rate as one obtains for the unsampled system. To extend this line of thought, we are currently looking into whether for randomly sampled processes, the expected value of the random variable at each time converges to the value of the function obtained as a solution to the unsampled process, as the mean sampling rate grows.

While this chapter addressed the problem of stabilization with random sampling using static time-invariant state feedback controllers, one can also explore the possibility of considering dynamic controllers with output feedback. Going beyond the realm of conventional dynamic controllers, more recently in [45], the authors work with discontinuous, or hybrid controllers, and consider the effect of random perturbations in communication of discrete and continuous state to the controller. Addressing similar questions, as the ones confronted in this chapter, for a more general class of controllers is likely to bring significant contributions to the currently active field of stochastic hybrid systems [12, 20, 44].

Acknowledgements The authors thank V. S. Borkar, A. Ferrante, J. P. Hespanha, H. Ishii, D. E. Quevedo, A. Teel, G. Yin, and S. Yüksel for helpful discussions and pointers to relevant literature.

Appendix

A: Proof of Proposition 2.2

Proof The fact that $\big(x(t), e(t)\big)_{t\geqslant 0}$ is Markovian follows from the observation that the future of $x(t)$ depends on $x(\tau_{N_t})$ and, therefore, equivalently on $e(t)$.

Let $\mathbb{R}^d \times \mathbb{R}^d \ni (y, z) \mapsto \psi(y, z) \in \mathbb{R}$ denote a function with at most polynomial growth as $\|(y, z)\| \to +\infty$. Since the system under consideration is well-posed, we have, for $h > 0$ small,

$$\begin{aligned}
&\mathsf{E}\left[\psi\big(x(t+h), e(t+h)\big)\middle|x(t) = y, e(t) = z\right]\\
&= \mathsf{E}\Big[\psi\big(x(t+h), e(t+h)\big)\big(1_{\{N_{t+h}=N_t\}} + 1_{\{N_{t+h}=1+N_t\}}\\
&\qquad + 1_{\{N_{t+h}-N_t\geqslant 2\}}\big)\ \big|\ x(t), e(t)\Big].
\end{aligned} \tag{51}$$

We now compute the conditional probability distribution of $\big(x(t+h), e(t+h)\big)$ for small $h > 0$ given $\big(x(t), e(t)\big)$. Since the sampling process is independent of the joint process $\big(x(\tau_{N_t}), x(t)\big)_{t\geqslant 0}$, by definition of the sampling (Poisson) process we have, for $h \downarrow 0$,

$$\begin{cases}
\mathsf{P}\left(N_{t+h} - N_t = 0\middle|N_t, e(t), x(t)\right) = 1 - \lambda h + o(h),\\
\mathsf{P}\left(N_{t+h} - N_t = 1\middle|N_t, e(t), x(t)\right) = \lambda h + o(h),\\
\mathsf{P}\left(N_{t+h} - N_t \geqslant 2\middle|N_t, e(t), x(t)\right) = o(h).
\end{cases}$$

Using these expressions we develop (51) further for $h \downarrow 0$ as

$$\begin{aligned}
&\mathsf{E}\left[\psi\big(x(t+h), e(t+h)\big)\middle|x(t) = y, e(t) = z\right]\\
&= \mathsf{E}\Big[\psi\big(x(t+h), e(t+h)\big)\big(1_{\{N_{t+h}=N_t\}} + 1_{\{N_{t+h}=1+N_t\}}\big)\ \big|\ x(t), e(t)\Big] + o(h)\\
&= \mathsf{E}\left[\psi\big(x(t+h), e(t+h)\big)\middle|x(t), e(t), N_{t+h} = N_t\right]\cdot\big(1 - \lambda h + o(h)\big)\\
&\quad + \mathsf{E}\left[\psi\big(x(t+h), e(t+h)\big)\middle|x(t), e(t), N_{t+h} = 1 + N_t\right]\big(\lambda h\big) + o(h).
\end{aligned} \tag{52}$$

The two significant terms on the right-hand side of (52) are now computed separately. For the event $N_{t+h} = N_t$, given $x(t) = y$, $e(t) = z$, we have for $h \downarrow 0$,

$$\begin{aligned}
\psi\big(x(t+h), e(t+h)\big) &= \psi\big(y, z\big) + h\left\langle\nabla_y\psi\big(y, z\big), f\left(y, \kappa\big(x(\tau_{N_t})\big)\right)\right\rangle\\
&\quad + h\left\langle\nabla_z\psi\big(y, z\big), f\left(y, \kappa\big(x(\tau_{N_t})\big)\right)\right\rangle + o(h),
\end{aligned}$$

leading to the first term on the right-hand side of (52) having the estimate

$$\begin{aligned}
&\mathsf{E}\left[\psi\big(x(t+h), e(t+h)\big)\middle|N_{t+h} = N_t, x(t) = y, e(t) = z\right] \cdot \big(1 - \lambda h + o(h)\big)\\
&= \psi\big(y, z\big) + h\left\langle \nabla_y \psi\big(y, z\big) + \nabla_z \psi\big(y, z\big), f\left(y, \kappa\big(x(\tau_{N_t})\big)\right)\right\rangle\\
&\qquad - (\lambda h)\psi\big(y, z\big) + o(h) \quad \text{for } h \downarrow 0.
\end{aligned}$$

Concerning the second term on the right-hand side of (52), we observe that conditional on $N_{t+h} = 1 + N_t$, the probability distribution of $\tau_{N_{t+h}}$ is [39, Theorem 2.3.7] uniform over $[t, t+h[$ by definition of the sampling (Poisson) process, i.e.,

$$\mathsf{P}\left(\tau_{N_{t+h}} \in [s, s+s'[\middle|N_{t+h} = 1 + N_t\right) = \frac{1}{h}s' \quad \text{for } [s, s+s'[\subset [t, t+h[.$$

Since the sampling process is independent of the state process, the preceding conditional probability is equal to

$$\mathsf{P}\left(\tau_{N_{t+h}} \in [s, s+s'[\middle|N_{t+h} = 1 + N_t, x(t) = y, e(t) = z\right).$$

We define $\theta \in [0, 1[$ such that $\tau_{N_{t+h}} = t + \theta h, x(t) = y, e(t) = z$; then θ is uniformly distributed on $[0, 1[$ given $N_{t+h} = 1 + N_t$. We also have, conditioned on the same event,

$$e(\tau_{N_{t+h}}) = e(t + \theta h) = 0,$$

and

$$x(\tau_{N_{t+h}}) = x(t + \theta h) = x(t) + \theta h f\left(x(t), \kappa\big(x(\tau_{N_t})\big)\right) + o(h).$$

The above expressions then lead to, conditioned on the event $N_{t+h} = 1 + N_t, x(t) = y, e(t) = z$ and for $h \downarrow 0$,

$$\begin{aligned}
x(t+h) &= x(t+\theta h) + (1-\theta)hf\big(x(t+\theta h), \kappa(x(t+\theta h))\big) + o(h)\\
&= x(t) + \theta h f\left(x(t), \kappa\big(x(\tau_{N_t})\big)\right) + (1-\theta)hf\big(x(t+\theta h), \kappa(x(t+\theta h))\big) + o(h)\\
&= x(t) + \theta h f\left(x(t), \kappa\big(x(\tau_{N_t})\big)\right) + (1-\theta)hf\big(x(t), \kappa(x(t))\big) + o(h).
\end{aligned}$$

Similarly, it can be verified directly from the differential equation governing e that conditioned on the same event,

$$e(t+h) = (1-\theta)hf(x(t), \kappa(x(t))) + o(h) \quad \text{for } h0.$$

Therefore, for $h \downarrow 0$,

$$\begin{aligned}
&\mathsf{E}\left[\psi\big(x(t+h), e(t+h)\big)\middle|x(t) = y, e(t) = z, N_{t+h} = 1 + N_t\right] \cdot (\lambda h)\\
&= \int_0^1 \psi\Big(y + \theta h f\left(x(t), \kappa\big(x(\tau_{N_t})\big)\right) + (1-\theta)hf\big(x(t), \kappa(x(t))\big) + o(h),\\
&\qquad (1-\theta)hf\big(x(t), \kappa(x(t))\big) + o(h)\Big)\,\mathrm{d}\theta \cdot (\lambda h)
\end{aligned}$$

$$
\begin{aligned}
&= \int_0^1 \Big(\psi\big(y,0\big) + h \big\langle \nabla_y \psi\big(y,0\big), \theta h f\big(x(t), \kappa\big(x(\tau_{N_t})\big)\big)\big) + (1-\theta) h f\big(x(t), k(x(t),0)\big\rangle \\
&\quad + h \big\langle \nabla_z \psi\big(y,0\big), (1-\theta) h f\big(x(t), \kappa(x(t))\big)\big\rangle + o(h) \Big)\, \mathrm{d}\theta \cdot (\lambda h) \\
&= \Big(\psi\big(y,0\big) + O(h) \Big) \cdot (\lambda h) \\
&= (\lambda h) \psi\big(y,0\big) + o(h).
\end{aligned}
$$

Putting everything together, we arrive at

$$
\begin{aligned}
&\mathsf{E}\psi\big(x(t+h), e(t+h)\big)\big| x(t) = y, e(t) = z \\
&= \psi(y,z) + h\Big(\big\langle \nabla_y \psi(y,z) + \nabla_z \psi(y,z), f\big(y, \kappa\big(y-z\big)\big)\big\rangle \Big) \\
&\quad - (\lambda h)\big(\psi(y,z) - \psi(y,0)\big) + o(h).
\end{aligned}
$$

Substituting these expressions in (8), we see that for each $(y,z) \in \mathbb{R}^d \times \mathbb{R}^d$,

$$
\begin{aligned}
\mathcal{L}\psi(y,z) = \big\langle \nabla_y \psi(y,z) + \nabla_z \psi(y,z), f\big(y, \kappa\big(y-z\big)\big)\big)\big\rangle \\
- \lambda\big(\psi(y,z) - \psi(y,0)\big),
\end{aligned}
$$

as asserted. □

B: Proofs of Lemmas 5.2 and 5.3

Proof of Lemma 5.2 The desired expression for $\frac{\mathrm{d}}{\mathrm{d}t} W(t)$ is obtained by differentiating

$$
W(t) = \sum_{j=0}^{+\infty} \mathcal{K}^j(H)(t)
$$

where we recall that $\mathcal{K}$ is given in (25) and $\mathcal{K}^0$ is the identity operator. To do so, we basically compute $\frac{\mathrm{d}}{\mathrm{d}t}\mathcal{K}^j(H)(t)$ for each $j \geqslant 0$. Since $\mathcal{K}^0(H)(t) = H(t)$, we first observe that

$$
\frac{\mathrm{d}}{\mathrm{d}t} H(t) = F^\top H(t) + H(t) F - \lambda H(t).
$$

Similarly, we compute

$$
\begin{aligned}
\frac{\mathrm{d}}{\mathrm{d}t}\mathcal{K}(H)(t) &= \lambda \mathrm{e}^{tF^\top} G^\top (H)(0) G \mathrm{e}^{tF} \mathrm{e}^{-\lambda t} \\
&\quad + \lambda \int_0^t \mathrm{e}^{sF^\top} G^\top \left(\frac{\mathrm{d}}{\mathrm{d}t} H(t-s) \right) G \mathrm{e}^{sF} \mathrm{e}^{-\lambda s}\, \mathrm{d}s \\
&= \lambda J(t) + \mathcal{K}(F^\top H + HF - \lambda H)(t).
\end{aligned}
$$

Next, to compute $\frac{\mathrm{d}}{\mathrm{d}t}\mathcal{K}^j(H)(t)$, for $j \geqslant 2$, we use the induction principle. Let us assume that, for some $j \geqslant 2$,

$$\frac{\mathrm{d}}{\mathrm{d}t}\mathcal{K}^{j-1}(H)(t) = \lambda\mathcal{K}^{j-2}(J)(t) + \mathcal{K}^{j-1}(F^\top H + HF - \lambda H)(t).$$

It then follows that

$$\begin{aligned}\frac{\mathrm{d}}{\mathrm{d}t}\mathcal{K}^j(H)(t) &= \lambda \mathrm{e}^{tF^\top} G^\top \mathcal{K}^{j-1}(H)(0) G \mathrm{e}^{tF}\mathrm{e}^{-\lambda t} \\ &\quad + \lambda \int_0^t \mathrm{e}^{sF^\top} G^\top \frac{\mathrm{d}}{\mathrm{d}t}\mathcal{K}^{j-1}(H)(t-s) G \mathrm{e}^{sF}\mathrm{e}^{-\lambda s}\,\mathrm{d}s \\ &= \lambda\mathcal{K}^{j-1}(J)(t) + \mathcal{K}^j(F^\top H + HF - \lambda H)(t).\end{aligned}$$

Using this last expression and recalling the definition of W from (27), we obtain

$$\frac{\mathrm{d}}{\mathrm{d}t}W(t) = \sum_{j=1}^{+\infty}\mathcal{K}^j(F^\top H + HF + \lambda(J-H))(t) + (F^\top H + HF + \lambda(J-H))(t),$$

which is the desired statement. □

Proof of Lemma 5.3 We first verify the desired expression (31) for $j = 1$. It is seen that

$$\begin{aligned}\lambda J(t) - \lambda G^\top H(t) G &= \lambda \mathrm{e}^{F^\top t} G^\top Q G \mathrm{e}^{Ft}\mathrm{e}^{-\lambda t} - \lambda G^\top H(t) G \\ &= \lambda \int_0^t \frac{\partial}{\partial s}\left(\mathrm{e}^{F^\top s} G^\top H(t-s) G \mathrm{e}^{Fs}\mathrm{e}^{-\lambda s}\right)\mathrm{d}s \\ &= F^\top \mathcal{K}(H)(t) + \mathcal{K}(H)(t)F - \lambda\mathcal{K}(H)(t) \\ &\quad + \lambda \int_0^t \left(\mathrm{e}^{F^\top s} G^\top \frac{\partial}{\partial s} H(t-s) G \mathrm{e}^{Fs}\mathrm{e}^{-\lambda s}\right)\mathrm{d}s \\ &= F^\top \mathcal{K}(H)(t) + \mathcal{K}(H)(t)F - \lambda\mathcal{K}(H)(t) \\ &\quad - \mathcal{K}(F^\top H + HF - \lambda H)(t),\end{aligned}$$

and hence (31) holds for $j = 1$.

Proceeding by induction, we assume that for some $j \geqslant 1$

$$\begin{aligned}F^\top\mathcal{K}^j(H)(t) + \mathcal{K}^j(H)(t)F - \lambda\mathcal{K}^j(H)(t) &= -\lambda G^\top \mathcal{K}^{j-1}(H)(t) G \\ &\quad + \mathcal{K}^j(F^\top H + HF - \lambda H) + \lambda\mathcal{K}^{j-1}(J)(t). \qquad (53)\end{aligned}$$

We then observe that

$$-\lambda G^\top \mathcal{K}^j(H(t)) G = \lambda \int_0^t \frac{\partial}{\partial s_j}\left(\mathrm{e}^{F^\top s_j} G^\top \mathcal{K}^j(H)(t-s_j) G \mathrm{e}^{Fs_j}\mathrm{e}^{-\lambda s_j}\right)\mathrm{d}s_j \qquad (54)$$

because $\mathcal{K}^j(H)(0) = 0$ for each $j \geqslant 1$. To compute the expression in the integrand on the right-hand side, we observe that

$$\frac{\partial}{\partial s_j}\mathcal{K}^j(H)(t-s_j) = -\lambda\mathcal{K}^{j-1}(J)(t-s_j) - \mathcal{K}^j(F^\top H + HF - \lambda H)(t-s_j),$$

which results in

$$\begin{aligned}\frac{\partial}{\partial s_j}&\big(\mathrm{e}^{F^\top s_j}G^\top\mathcal{K}^j(H)(t-s_j)G\mathrm{e}^{Fs_j}\mathrm{e}^{-\lambda s_j}\big)\mathrm{d}s_j\\ &= -\lambda\mathrm{e}^{F^\top s_j}G^\top\mathcal{K}^{j-1}(J)(t-s)G\mathrm{e}^{Fs_j}\mathrm{e}^{-\lambda s_j}\\ &\quad - \mathrm{e}^{F^\top s_j}G^\top\mathcal{K}^j(F^\top H + HF - \lambda H)(t-s_j)G\mathrm{e}^{Fs_j}\mathrm{e}^{-\lambda s_j}\\ &\quad + F^\top(\mathrm{e}^{F^\top s_j}G^\top\mathcal{K}^j(H)(t-s_j)G\mathrm{e}^{Fs_j}\mathrm{e}^{-\lambda s_j})\\ &\quad + (\mathrm{e}^{F^\top s_j}G^\top\mathcal{K}^j(H)(t-s_j)G\mathrm{e}^{Fs_j}\mathrm{e}^{-\lambda s_j})F\\ &\quad - \lambda(\mathrm{e}^{F^\top s_j}G^\top\mathcal{K}^j(H)(t-s_j)G\mathrm{e}^{Fs_j}\mathrm{e}^{-\lambda s_j}).\end{aligned}$$

Substituting this last equality in (53), we get

$$\begin{aligned}\lambda\mathcal{K}^j(J(t)) - \lambda G^\top\mathcal{K}^j(H(t))G = F^\top\mathcal{K}^{j+1}(H)(t) + \mathcal{K}^{j+1}(H)(t)F - \lambda\mathcal{K}^{j+1}(H)(t)\\ - \mathcal{K}^{j+1}(F^\top H + HF - \lambda H)(t),\end{aligned}$$

and the assertion follows. □

References

1. A. Arapostathis, V. S. Borkar, E. Fernandez-Gaucherand, M. K. Ghosh, and S. I. Marcus. Discrete-time controlled Markov processes with average cost criterion: A survey. *SIAM Journal on Control and Optimization*, 31(2):282–344, 1993.
2. J. Ackermann. *Sampled-Data Control Systems: Analysis and Synthesis, Robust System Design*. Springer-Verlag, Heidelberg, 1985.
3. M. Adès, P. E. Caines, and R. P. Malhamé. Stochastic optimal control under Poisson-distributed observations. *IEEE Transactions on Automatic Control*, 45(1):3–13, 2000.
4. D. J. Antunes, J. P. Hespanha, and C. J. Silvestre. Volterra integral approach to impulsive renewal systems: Application to networked control. *IEEE Transactions on Automatic Control*, 57(3):607–619, 2012.
5. D. Antunes, J. Hespanha, and C. Silvestre. Stability of networked control systems with asynchronous renewal links: An impulsive systems approach. *Automatica*, 49(2):402–413, 2013.
6. A. A. Agrachev and Yu. L. Sachkov. *Control Theory from the Geometric Viewpoint*, volume 87 of *Encyclopaedia of Mathematical Sciences*. Springer-Verlag, Berlin, 2004. Control Theory and Optimization, II.
7. K. J. Åström. Event based control. In *Analysis and Design of Nonlinear Control Systems*, pages 127–148. Springer, 2008.
8. K. J. Åström and B. M. Wittenmark. *Computer Controlled Systems: Theory and Design*. Prentice-Hall, Inc. Upper Saddle River, New Jersey, 3rd edition, 1997.

9. T. Başar and P. Bernhard. H_∞-*Optimal Control and Related Minimax Design Problems: A Dynamic Game Approach*. Birkhäuser, Basel, 2nd edition, 2008.
10. T. Chen and B. A. Francis. *Optimal Sampled-Data Control Systems*. Springer-Verlag, London, 1995.
11. O. L. V. Costa, M. D. Fragoso, and M. G. Todorov. *Continuous-Time Markov Jump Linear Systems*. Probability and its Applications (New York). Springer, Heidelberg, 2013.
12. C. G. Cassandras and J. Lygeros, editors. *Stochastic Hybrid Systems*. CRC Press, Taylor & Francis Group, Boca Raton, FL, 2007.
13. M. H. A. Davis. Piecewise-deterministic Markov processes: a general class of nondiffusion stochastic models. *Journal of the Royal Statistical Society. Series B. Methodological*, 46(3):353–388, 1984. With discussion.
14. M. H. A. Davis. *Markov Models and Optimization*, volume 49 of *Monographs on Statistics and Applied Probability*. Chapman & Hall, London, 1993.
15. R. M. Dudley. *Real Analysis and Probability*, volume 74 of *Cambridge Studies in Advanced Mathematics*. Cambridge University Press, Cambridge, 2002. Revised reprint of the 1989 original.
16. E. B. Dynkin. Infinitesimal operators of Markov random processes. *Doklady Akademii Nauk SSSR*, 105:206–209, 1955.
17. E. B. Dynkin. Markov processes and semi-groups of operators. *Akademija Nauk SSSR. Teorija Verojatnosteĭ i ee Primenenija*, 1:25–37, 1956.
18. G. F. Franklin, J. D. Powell, and M. Workman. *Digital Control of Dynamic Systems*. Addison-Wesley, Reading, MA, 3rd edition, 1997.
19. W. H. Fleming and H. M. Soner. *Controlled Markov Processes and Viscosity Solutions*, volume 25 of *Stochastic Modelling and Applied Probability*. Springer, New York, 2 edition, 2006.
20. J. P. Hespanha. Modeling and analysis of networked control systems using stochastic hybrid systems. *Annual Reviews in Control*, 38(2):155–170, 2014.
21. R. A. Horn and C. R. Johnson. *Matrix Analysis*. Cambridge University Press, 2nd edition, 2013.
22. W. P. M. H. Heemels, K. H. Johansson, and P. Tabuada. An introduction to event-triggered and self-triggered control. In *Proc. 51st IEEE Conf. Decision & Control*, pages 3270–3285, 2012.
23. J. P. Hespanha and A. R. Teel. Stochastic impulsive systems driven by renewal processes. In *Proc. 17th International Symposium on Mathematical Theory of Networked Systems*, pages 606–618. 2006.
24. R. Kalman. *Analysis and synthesis of linear systems operating on randomly sampled data*. PhD thesis, Department of Electrical Engineering, Columbia University, New York, USA, 1957.
25. R. Khasminskii. *Stochastic Stability of Differential Equations*, volume 66 of *Stochastic Modelling and Applied Probability*. Springer, Heidelberg, 2nd edition, 2012. With contributions by G. N. Milstein and M. B. Nevelson.
26. F. Kozin. A survey of stability of stochastic systems. *Automatica*, 5(1):95–112, 1969.
27. H. Kushner and L. Tobias. On the stability of randomly sampled systems. *IEEE Transactions on Automatic Control*, 14(4):319–324, 1969.
28. H. J. Kushner. *Stochastic Stability and Control*. Mathematics in Science and Engineering, Vol. 33. Academic Press, New York-London, 1967.
29. O. Leneman. Random sampling of random processes: Mean-square behavior of a first order closed-loop system. *IEEE Transactions on Automatic Control*, 13(4):429–432, 1968.
30. D. Liberzon, D. Nešić, and A. R. Teel. Lyapunov-based small-gain theorems for hybrid systems. *IEEE Transactions on Automatic Control*, 59(6):1395–1410, 2014.
31. L. Montestruque and P. Antsaklis. Stability of model-based networked control systems with time-varying transmission times. *IEEE Transactions on Automatic Control*, 49(9):1562–1572, 2004.
32. A. S. Matveev and A. V. Savkin. The problem of state estimation via asynchronous communication channels with irregular transmission times. *IEEE Transactions on Automatic Control*, 48(4):670–676, 2003.
33. A. S. Matveev and A. V. Savkin. *Estimation and Control over Communication Networks*. Birkhauser, Boston, MA, 2009.

34. S. Meyn and R. L. Tweedie. *Markov Chains and Stochastic Stability*. Cambridge University Press, Cambridge, 2nd edition, 2009. With a prologue by Peter W. Glynn.
35. D. Nešić and L. Grüne. Lyapunov-based continuous-time nonlinear controller redesign for sampled-data implementation. *Automatica*, 41(7):1143–1156, 2005.
36. D. Nešić and A. R. Teel. Input–output stability properties of networked control systems. *IEEE Transactions on Automatic Control*, 49(10):1650–1667, 2004.
37. D. Nešić, A. R. Teel, and P. V. Kokotović. Sufficient conditions for stabilization of sampled-data nonlinear systems via discrete-time approximations. *Systems & Control Letters*, 38(4):259–270, 1999.
38. D. E. Quevedo, V. Gupta, W. J. Ma, and S. Yüksel. Stochastic stability of event-triggered anytime control. *IEEE Transactions on Automatic Control*, 59(12):3373–3379, 2014.
39. Y. Suhov and M. Kelbert. *Probability and Statistics by Example, vol II; Markov Chains: A Primer in Random Processes and their Applications*. Cambridge University Press, Cambridge, 2008.
40. E. D. Sontag. Smooth stabilization implies coprime factorization. *IEEE Transactions on Automatic Control*, 34(4):435–443, 1989.
41. B. Sinopoli, L. Schenato, M. Franceschetti, K. Poolla, M. I. Jordan, and S. S. Sastry. Kalman filtering with intermittent observations. *IEEE Transactions on Automatic Control*, 49(9):1453–1464, 2004.
42. P. Tabuada. Event-triggered real-time scheduling of stabilizing control tasks. *IEEE Transactions on Automatic Control*, 52(9):1680–1685, 2007.
43. R. Tempo, G. Calafiore, and F. Dabbene. *Randomized Algorithms for Analysis and Control of Uncertain Systems*. Springer, London, 2nd edition, 2012.
44. A. R. Teel, A. Subbaraman, and A. Sferlazza. Stability analysis for stochastic hybrid systems: A survey. *Automatica*, 50:2435–2456, 2014.
45. A. Tanwani and A. R. Teel. Stabilization with event-driven controllers over a digital communication channel with random transmissions. In *Proc. 56th IEEE Conference on Decision and Control*, pages 6063–6068. 2017.
46. G. C. Walsh, O. Beldiman, and L. G. Bushnell. Asymptotic behavior of nonlinear networked control systems. *IEEE Transactions on Automatic Control*, 46(7):1094–1097, 2001.
47. S. Yüksel and T. Başar. *Stochastic Networked Control Systems*, volume 10. Springer, New York, NY, 2013.
48. F. Zhang. *Matrix Theory*. Universitext. Springer, New York, 2nd edition, 2011. Basic results and techniques.
49. P. Zhao, L. Y. Wang, and G. Yin. Controllability and adaptation of linear time-invariant systems under irregular and Markovian sampling. *Automatica*, 63:92–100, 2016.

Robust Design Through Probabilistic Maximization

T. Alamo, J. M. Manzano and E. F. Camacho

Abstract In this chapter, we study randomized maximization methods for robust design under uncertainty. In particular, we show how order statistics can be used to derive novel design schemes. We provide concentration inequalities that allow us to guarantee that the obtained design parameters meet some probabilistic specifications. The proposed methodology addresses the robust design problem without relying on any convexity assumption or precise knowledge of the probabilistic distribution of the underlying uncertainty. Moreover, the required sample complexity does not depend on the dimension of the design problem. We also propose schemes that allow us to obtain one level of probability guarantees.

Keywords Randomized algorithms · Probabilistic robustness · Uncertain systems

1 Introduction

In a robustness problem, the controller parameters and auxiliary variables are parameterized by means of a decision variable vector θ, which is denoted as *design parameter* and is restricted to a set Θ. Furthermore, the uncertainty w is bounded in the set $\mathcal{W}$ and represents one of the admissible uncertainty realizations. We also consider a real measurable function $f : \Theta \times \mathcal{W} \to \mathbb{R}$ that serves to formulate the specific design problem under consideration. For example, if w represents the uncertain parameters that characterize the dynamics of a given plant and θ the parameters of the controller, $f(\theta, w)$ could represent the L_2 gain corresponding to the pair

T. Alamo (✉) · J. M. Manzano · E. F. Camacho
Departamento de Ingeniería de Sistemas y Automática, Escuela Superior de Ingenieros,
Universidad de Sevilla, Camino de los Descubrimientos s/n, 41092 Seville, Spain
e-mail: talamo@us.es

J. M. Manzano
e-mail: manzano@us.es

E. F. Camacho
e-mail: efcamacho@us.es

© Springer Nature Switzerland AG 2018

T. Başar (ed.), *Uncertainty in Complex Networked Systems*,
Systems & Control: Foundations & Applications,
https://doi.org/10.1007/978-3-030-04630-9_7

(θ, w) (see [36]). Similar examples in which a function $f(\cdot, \cdot)$ is used to evaluate the performance of a given design parameter θ under a specific realization of the uncertainty w can be found in [33, 37, 39].

In a deterministic setting, we could try to compute the worst-case situation by means of a maximization over the uncertain parameter set $\mathcal{W}$. That is, given $\hat{\theta} \in \Theta$ we could try to compute

$$\gamma_{max}(\hat{\theta}) = \max_{w \in \mathcal{W}} f(\hat{\theta}, w).$$

This sort of maximization problems often belong to the family of NP-hard problems and their exact solution is normally unaffordable from a computational point of view [13]. Sometimes a conservative approach is applied to obtain, not $\gamma_{max}(\hat{\theta})$, but an upper bound of its value. However, the obtained upper bounds might be highly conservative.

In order to circumvent these issues, randomized algorithms can be used [37]. This is the approach that we follow in this chapter. Randomized algorithms have been successful in addressing different robust control problems of convex [6, 16, 22, 30] and non-convex nature [4, 25]. See, [2, 27, 28, 34, 35], for recent applications.

We consider a probabilistic measure $\Pr_{\mathcal{W}}$ in $\mathcal{W}$. Given $\hat{\theta} \in \Theta$, violation level $\eta \in (0, 1)$ and failure level $\delta \in (0, 1)$, the objective of this chapter is to propose a methodology to compute $\gamma \in \mathbb{R}$ in such a way that, with probability at least $1 - \delta$,

$$\Pr_{\mathcal{W}}\{f(\hat{\theta}, w) > \gamma\} \leq \eta.$$

We notice that this is a two-level probabilistic constraint because it considers not only the violation level η, but also the probability δ of not obtaining γ such that the constraint $\Pr_{\mathcal{W}}\{f(\hat{\theta}, w) > \gamma\} \leq \eta$ is satisfied. This problem has been addressed in [36] (see also [11]). The following property summarizes the main contribution of [36].

Property 1 *Given $\theta \in \Theta$, the measurable function $f : \Theta \times \mathcal{W} \to \mathbb{R}$, $\eta \in (0, 1)$ and $\delta \in (0, 1)$, suppose that*

$$N \geq \frac{1}{\eta} \ln \frac{1}{\delta}.$$

Suppose also that $\mathbf{w} = \{w^{(1)}, w^{(2)}, \ldots, w^{(N)}\}$ are i.i.d samples drawn from a given probability distribution $\Pr_{\mathcal{W}}$. Denote

$$\gamma_N = J(\theta, \mathbf{w}) = \max_{i=1,\ldots,N} f(\theta, w^{(i)}).$$

Then, with probability at least $1 - \delta$,

$$\Pr_{\mathcal{W}}\{f(\theta, w) > \gamma_N\} \leq \eta.$$

The proof of this result can be found in [36]. We also provide an alternative proof in this chapter. Property 1 is stated in terms of two probability levels. The first one is the probability of violation $\eta \in (0, 1)$ and the second one is the failure level $\delta \in (0, 1)$. The sample complexity N in Property 1 grows linearly with $\frac{1}{\eta}$ and with the natural logarithm of $\frac{1}{\delta}$. This means that affordable values for N are obtained even for very small values of the failure level δ.

In this chapter, we propose some generalizations to Property 1 and show how to use them in the context of control of uncertain systems. We also provide some results that depend only on one level of probability (failure level δ). Except for a reference to a corollary in [6], all the presented results are proved from scratch.

The chapter is organized as follows. In Sect. 2 we show, by means of an illustrative example, how to use Property 1 in the context of robust control. In Sect. 3 we propose a generalization of the notion of max function borrowed from the field of order statistics. We present a two-level of probability result in Sect. 4 that generalizes the results presented in [36]. In Sect. 5, a one level of probability result is provided. The relationship between the hypergeometric distribution and the problems addressed in this chapter is made explicit in Sect. 6. Some conclusions are given in Sect. 7. The chapter ends with four appendices that cover the most technical developments required to prove several of the results presented in this chapter.

2 Motivational Example: Bounding the Error

Suppose that the dynamics of a system are given by

$$x(k+1) = h(x(k), u(k)) + w(k),$$

where $w(k) \in \mathbb{R}^{n_w}$ is a random vector with probability distribution $\Pr_{\mathcal{W}}$. We assume that $w(k)$ is probabilistically independent from $w(k-1)$. That is, for every j and k, with $j < k$, we have that

$$\mathbf{w} = \{w(j), w(j+1), \ldots, w(k)\}$$

is a sequence of independent identically distributed (i.i.d.) disturbance-error terms. We notice that given the sequences $x(k)$ and $u(k)$, the sequence of error terms can be obtained directly from $w(k) = x(k+1) - h(x(k), u(k))$. Suppose that we obtain S i.i.d. error terms

$$\{\tilde{w}^{(1)}, \tilde{w}^{(2)}, \ldots, \tilde{w}^{(S)}\}.$$

We could derive an ellipsoidal outer bound $\mathcal{E} = \{\, w \ : \ \|w\|_P \leq \gamma \,\}$, where $P > 0$ and $\|w\|_P = \sqrt{w^\top P w}$, for this collection of S “learning points” choosing

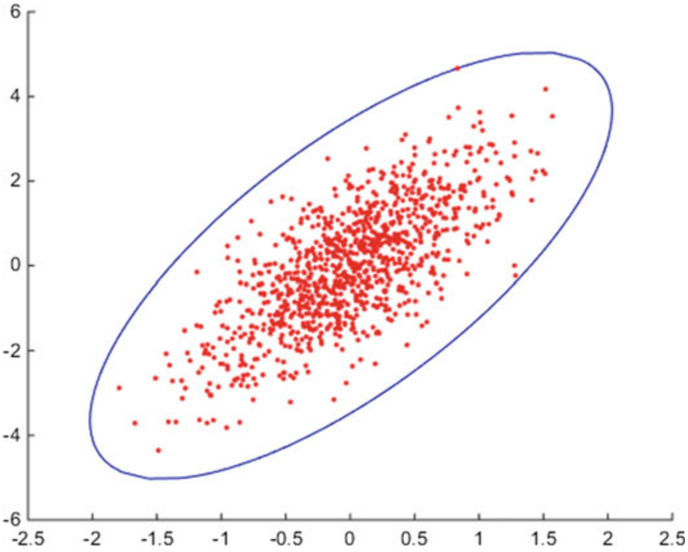


Fig. 1 Learning points and naive ellipsoidal bound

$$P = \left(\frac{1}{S} \sum_{i=1}^{S} \tilde{w}^{(i)} (\tilde{w}^{(i)})^\top \right)^{-1}, \tag{1}$$

$$\gamma = \max_{i=1,\ldots,S} \sqrt{(\tilde{w}^{(i)})^\top P \tilde{w}^{(i)}}. \tag{2}$$

We notice that P corresponds to the inverse of the empirical covariance matrix. The matrix P often provides a reasonable approximation of the geometry of a given cloud of points (especially if the points are generated by a Gaussian distribution) [12, 31]. In Fig. 1, a randomly generated set of S learning points are displayed ($S = 1000$). The points correspond to a multivariate Gaussian distribution. Also, the ellipsoidal outer bound obtained by means of Eqs. (1) and (2) is shown. We notice that this “naive” approach, although it is able to outer bound the cloud of learning points, does not guarantee that additional i.i.d. “test points” are contained in the naive ellipsoidal bound.

Suppose now that we are interested in bounding the error term by means of an expression of the form

$$\Pr_{\mathcal{W}}\{\|w\|_P > \gamma\} \leq \eta, \tag{3}$$

where $\eta \in (0, 1)$ and $\Pr_{\mathcal{W}}$ denotes the probabilistic measure related to the disturbance error terms. We could, given a collection of i.i.d. learning samples, compute the smallest ellipsoidal bound that contains all (or almost every) learning samples. Using the results of convex scenario [15, 17, 18], we could obtain the number S of learning samples required to guarantee, with high probability, that the obtained ellipsoidal bound meets the probabilistic specifications given by (3). We notice that the number of decision variables required to parameterize matrix P grows with the square of the dimension of the space. This could be an issue because the number of required samples for convex scenario grows with the number of decision variables [3, 6]. Alternative sample complexity bounds could be obtained using results from statistical learning theory [4, 38, 39].

In order to circumvent the negative impact that the dimensionality of the design space has in the number of samples required to guarantee the probabilistic specifications, one can resort to probabilistic validation approaches. See, for example, [6, 7, 20, 21, 32]. In this case, we could obtain $P > 0$ and γ from a learning set and validate them by means of an additional validation set. The validation step could consist of obtaining a collection of N i.i.d. validation samples $\{w^{(1)}, \ldots, w^{(N)}\}$ and checking if the constraints

$$\|w^{(i)}\|_P \leq \gamma,\ i = 1, \ldots, N$$

are satisfied. Here we could allow some violations of these constraints to increase the possibility of passing the validation step (see, for example, [6, 10]). We notice that this approach has the main drawback that the candidate pair (P, γ) could fail to pass the validation step. Therefore, one is forced to include a mechanism to provide a sequence of candidate solutions that are tested in a sequential manner until one of them satisfies the exit condition (validation test). As it is detailed in [6, section 7], this exit condition can lead to feasibility issues. Moreover, the recursive generation of candidate solutions increases the sample complexity of the validation set [6, 32].

In order to avoid the incorporation of a validation test, we could proceed in a different manner. We could obtain $P > 0$ from a learning data set (by means of Eq. (1)) and derive γ from an additional one. That is, denoting $f(\theta, w) = w^\top P w$ and $\Theta = \{\ P \in \mathbb{R}^{n_w \times n_w}\ :\ P > 0\ \}$ we have that given $\theta = P > 0$ and a collection of N i.i.d. error terms

$$\{w^{(1)}, w^{(2)}, \ldots, w^{(N)}\},$$

that have not been used to compute P, we could obtain

$$\gamma_N = \max_{i=1,\ldots,N} \|w^{(i)}\|_P = \max_{i=1,\ldots,N} f(\theta, w^{(i)}). \tag{4}$$

It is clear that if N is large, γ_N constitutes a probabilistic upper bound for $\|w\|_P$.

In Fig. 2, the histogram of the weighted norm $\|w^{(i)}\|_P, i = 1, \ldots, N$, for an additional collection of $N = 1000$ i.i.d. error terms, is displayed. We could use Property 1 to obtain the number N of i.i.d. samples required to guarantee, with probability no smaller than $1 - \delta$, that γ_N satisfies

$$\text{Pr}_{\mathcal{W}}\{f(\theta, w) > \gamma_N\} = \text{Pr}_{\mathcal{W}}\{\|w\|_P > \gamma_N\} \leq \eta.$$

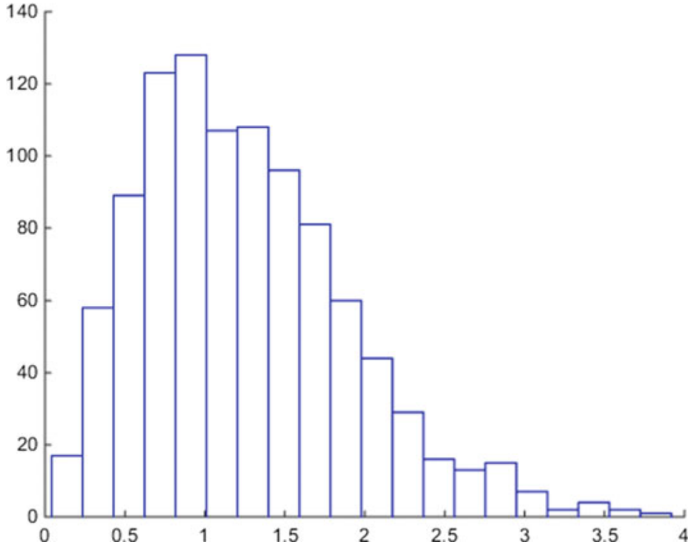


Fig. 2 Histogram of the weighted norm $\|w^{(i)}\|_P$, $i = 1, \ldots, N$, for the error terms in the additional learning set

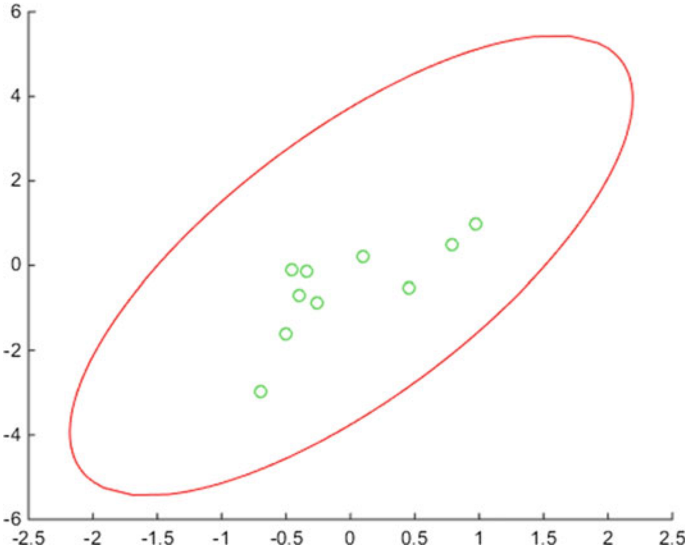


Fig. 3 Probabilistic ellipsoidal bound and additional test points $w^{(i)}$, $i = N + 1, \ldots, N + 10$

We notice that this approach circumvents the issue of the dimension of the design space since the number of required samples does not depend on it. Another advantage is that this approach always provides a probabilistic outer bound (does not depend on an exit condition). In Fig. 3, an ellipsoidal bound for the error terms is displayed.

Matrix P has been obtained by means of a learning set (1000 samples) using Eq. (1) and the radius γ_N as the maximum radius obtained from an additional maximization set of 1000 samples (see Eq. (4)). Ten additional test points are also shown in the figure.

Suppose that the support of $\Pr_{\mathcal{W}}$ is not finite. That is, given any scalar $\hat{\gamma}$ and $P > 0$, the probability $\Pr_{\mathcal{W}}\{\|w\|_P > \hat{\gamma}\}$ is strictly larger than zero. In this case, for every $\hat{\gamma}$, the probability $\Pr_{\mathcal{W}^N}\{\gamma_N > \hat{\gamma}\}$ tends to 1 when N tends to infinity. Since this is valid for every $\hat{\gamma}$ we conclude that if the support of $\Pr_{\mathcal{W}}$ is not finite then, with probability 1, we have

$$\lim_{N\to\infty} \gamma_N = \infty.$$

This means that the probabilistic upper bound obtained by means of Property 1 degrades when $\frac{1}{\eta}\log\frac{1}{\delta}$ is large because it eventually becomes too conservative.

In real systems, the support of $\Pr_{\mathcal{W}}$ is usually finite. That is, there is a finite scalar $\hat{\gamma}$ such that $\Pr_{\mathcal{W}}\{\|w\|_P \le \hat{\gamma}\} = 1$. In this case γ_N is upper bounded, with probability one, by $\hat{\gamma}$. This means that when N tends to infinity, γ_N tends to the smallest value $\hat{\gamma}$ satisfying $\Pr_{\mathcal{W}}\{\|w\|_P \le \hat{\gamma}\} = 1$. This behavior is convenient when considering hard constraints on w because the probabilistic upper bound approaches (from below) the largest possible value of $\|w\|_P$.

A very different situation occurs when we consider soft constraints (chance constraint setting). In this case, we are interested in the minimum value of γ for which the probabilistic constraint $\Pr_{\mathcal{W}}\{\|w\|_P > \gamma\} \le \eta$ is satisfied. We notice that γ_N can provide an overly conservative estimation because when N tends to infinity, γ_N tends to the largest possible value of $\|w\|_P$ regardless of the value of η.

In order to address this conservativeness issue, one can discard the largest values of a given collection of N samples in the computation of the probabilistic upper bound of $f(\theta, w) = \sqrt{w^\top P w}$. In this way, the max function is replaced by a generalized max function that leads to less conservative results. Next section formalizes this concept.

3 Generalization of the Max Function

We now present a generalization of the notion of the maximum of a collection of scalars. This generalization is borrowed from the field of order statistics [1, 8], and will allow us to reduce the conservativeness that follows from the use of the standard notion of max function. We assume that $\mathbf{v}$ is a real vector of N components. That is,

$$\mathbf{v} = [v^{(1)}, v^{(2)}, \dots, v^{(N)}]^\top \in \mathbb{R}^N.$$

We denote by $\mathbf{v}_+$ the vector obtained rearranging the values of the components of $\mathbf{v}$ in a non-increasing order. That is,

$$\mathbf{v}_+ = [v_+^{(1)}, v_+^{(2)}, \ldots, v_+^{(N)}]^\top \in \mathbb{R}^N,$$

where

$$v_+^{(1)} \geq v_+^{(2)} \geq \ldots \geq v_+^{(N-1)} \geq v_+^{(N)}.$$

Clearly,

$$v_+^{(1)} = \max_{1\leq i \leq N} v^{(i)}, \quad v_+^{(N)} = \min_{1\leq i \leq N} v^{(i)}.$$

Furthermore, $v_+^{(2)}$ denotes the second largest value in $\mathbf{v}$ and $v_+^{(N-1)}$ the second smallest one. We notice that $v_+^{(i)}$ could be equal to $v_+^{(i+1)}$ if vector $\mathbf{v}$ has components with repeated values. The inequalities

$$v_+^{(i)} \geq v_+^{(j)}, \quad j = i, \ldots, N,$$

imply that at most $i-1$ components of $\mathbf{v}$ have values larger than $v_+^{(i)}$. Moreover, $v_+^{(i)}$ is larger than or equal to $N-i+1$ components of vector $\mathbf{v}$. We are now in a position to introduce the notion of generalized max function.

Definition 1 (*Generalized max function*) Given vector

$$\mathbf{v} = [v^{(1)}, v^{(2)}, \ldots, v^{(N)}]^\top \in \mathbb{R}^N,$$

and the integer r with $1 \leq r \leq N$ we denote

$$\phi(\mathbf{v}, r) = v_+^{(r)}$$

where vector $\mathbf{v}_+ = [v_+^{(1)}, v_+^{(2)}, \ldots, v_+^{(N)}]^\top \in \mathbb{R}^N$ is obtained by rearranging the values of the components of $\mathbf{v}$ in a non-increasing order. That is,

$$v_+^{(1)} \geq v_+^{(2)} \geq \ldots \geq v_+^{(N-1)} \geq v_+^{(N)}.$$

The notation proposed in this chapter is similar to the one used in the field of order statistics [1, 8]. Given a set of N scalars

$$\{v^{(1)}, v^{(2)}, \ldots, v^{(N)}\},$$

it is standard in order statistics to denote the smallest one by $v_{1:N}$, the second smallest one by $v_{2:N}$, and so on so forth until $v_{N:N}$ which denotes the largest one. In this way, given $r \geq 1$ we have that $v_{r:N}$ satisfies that no more than $r-1$ elements of $\{v_1, v_2, \ldots, v_N\}$ are strictly smaller than $v_{r:N}$. Furthermore, $v_{N:N}$ is the largest element. With the notation of order statistics we have $\phi(\mathbf{v}, r) = v_+^{(r)} = v_{N-r+1:N}$. We notice that the notation $\phi(\mathbf{v}, r)$ does not need to make explicit N, the number of components of v. This provides a more compact notation and will be used in the remaining part of the chapter.

The following definition introduces the notion of empirical generalized maximum.

Definition 2 (*Empirical generalized maximum*) Given $\theta \in \Theta$, the integers r and N with $1 \leq r \leq N$ and $\mathbf{w} = \{w^{(1)}, \ldots, w^{(N)}\}$, we define the empirical generalized maximum for the pair $(\theta, \mathbf{w})$ as

$$J_r^N(\theta, \mathbf{w}) = \phi(\begin{bmatrix} f(\theta, w^{(1)}) \\ f(\theta, w^{(2)}) \\ \vdots \\ f(\theta, w^{(N)}) \end{bmatrix}, r).$$

Clearly, if $\mathbf{w}$ has been obtained sampling N elements from $\mathcal{W}$ we have that $J_r^N(\theta, \mathbf{w})$ has a probabilistic nature. One of the main objectives of the chapter henceforth is to analyze the effects of N and r on the probabilistic behavior of $J_r^N(\theta, \mathbf{w})$ when used as a chance constrained estimation of the maximum of $f(\theta, w)$ in $\mathcal{W}$.

The notion of empirical generalized maximum allows us to generalize the results of [36]. This is precisely the main objective of the next section.

4 Two Level of Probability Results

Before introducing the main result of this section, we present some definitions and notations.

Definition 3 (*Probability of violation*) Given the pair $(\theta, \gamma) \in \Theta \times \mathbb{R}$ and the measurable function $f : \Theta \times \mathcal{W} \to \mathbb{R}$, we define the probability of violation $E(\theta, \gamma)$ as

$$E(\theta, \gamma) = \Pr_{\mathcal{W}}\{f(\theta, w) > \gamma\}.$$

We notice that the probability constraint

$$E(\theta, \gamma) = \Pr_{\mathcal{W}}\{f(\theta, w) > \gamma\} \leq \eta$$

means that if we draw M i.i.d. samples $\{w^{(1)}, \ldots, w^{(M)}\}$ and denote by s the number of samples $w^{(i)}$ for which $f(\theta, w^{(i)}) > \gamma$, then the ratio s/M is upper bounded by η, with probability 1, if M tends to infinity. Therefore, the constraint $E(\theta, \gamma) \leq \eta$ can be given an asymptotic interpretation.

Definition 4 (*Probability of asymptotic failure*) Given $\theta \in \Theta$, the measurable function $f : \Theta \times \mathcal{W} \to \mathbb{R}$, the violation level $\eta \in (0, 1)$ and the integers N, r, with $1 \leq r \leq N$, we define the probability of asymptotic failure $P_\eta(N, r)$ as

$$P_\eta(N, r) = \Pr_{\mathcal{W}^N}\{E(\theta, J_r^N(\theta, \mathbf{w})) > \eta\}.$$

That is, the probability of asymptotic failure is the probability of drawing $\mathbf{w} \in \mathcal{W}^N$ according to $\Pr_{\mathcal{W}^N}$ and obtaining $\Pr_{\mathcal{W}}\{f(\theta, w) > \gamma_{N,r}\} > \eta$, where $\gamma_{N,r} = J_r^N(\theta, \mathbf{w})$. The probability of asymptotic failure depends on the particular choices of θ and $f(\cdot,\cdot)$. With a slight abuse of notation, we do not make explicit this dependence in the notation $P_\eta(N, r)$ because, as it will be shown in this section, an upper bound for the probability of asymptotic failure that only depends on η, N, and r can be obtained.

We notice that Property 1 can be rewritten in terms of the probability of asymptotic failure for the particular case $r = 1$. That is, we could rewrite Property 1 as

Property 2 *Given $\theta \in \Theta$, the measurable function $f : \Theta \times \mathcal{W} \to \mathbb{R}$ and $\eta \in (0, 1)$, $\delta \in (0, 1)$, we have*

$$P_\eta(N, 1) = \Pr_{\mathcal{W}^N}\{E(\theta, J_1^N(\theta, \mathbf{w})) > \eta\} \le \delta$$

provided that

$$N \ge \frac{1}{\eta} \ln \frac{1}{\delta}.$$

We are now in a position to present a generalization of Property 1 that allows us to better address chance constrained problems. The result is written in terms of $J_r^N(\theta, \mathbf{w})$.

Property 3 *Given $\theta \in \Theta$, the measurable function $f : \Theta \times \mathcal{W} \to \mathbb{R}$, the integers r and N with $1 \le r \le N$ and $\eta \in (0, 1)$, we have*

$$P_\eta(N, r) = \Pr_{\mathcal{W}^N}\{E(\theta, J_r^N(\theta, \mathbf{w})) > \eta\} \le \sum_{i=0}^{r-1} \binom{N}{i} \eta^i (1-\eta)^{N-i}. \tag{5}$$

Moreover, given $\delta \in (0, 1)$, the probability of asymptotic failure $P_\eta(N, r)$ is no larger than δ, provided that

$$N \ge \frac{1}{\eta}\left(r - 1 + \ln \frac{1}{\delta} + \sqrt{2(r-1)\ln \frac{1}{\delta}}\right). \tag{6}$$

The first claim of the property is proved in Sect. 6. The second one is derived directly from the first claim and corollary 1 in [6]. We notice that inequality (5) can also be derived from classical results from order statistics [1, 8]. Another possibility to prove inequality (5) is to resort to convex scenario [15, 19]. Property 3 is stated in terms of two probability levels. The first one is the probability of violation η and the second one is the failure level δ. In the next section, we present a result that relies only on one level of probability (failure level δ).

5 One Level of Probability Results

As commented in the previous section, the probability constraint

$$\mathrm{Pr}_{\mathcal{W}}\{f(\theta, w) > \gamma\} \le \eta$$

can be given an asymptotic interpretation. In some relevant control design problems, one is not interested in an asymptotic result but in a non-asymptotic one. We enumerate now some examples:

- **Adaptive Control** [9, 26]: Suppose that, in an adaptive control setting, the design parameter vector $\theta(k)$ is updated every sample time k. Suppose also that

  $$\mathbf{w}_a = \{w^{(k-N+1)}, \ldots, w^{(k-1)}, w^{(k)}\}$$

  represents the last N sampled (i.i.d.) realizations of uncertainty. Given the positive integer r, we could consider $J_r^N(\theta(k), \mathbf{w}_a)$ as a probabilistic upper bound for $f(\theta(k), w)$. Because of the adaptive nature of $\theta(k)$, we focus only on the constraint $J_r^N(\theta(k), \mathbf{w}_a) \ge f(\theta(k), w(k+1))$, where $w(k+1)$ denotes the uncertain future realization of $w \in \mathcal{W}$ at sample time $k+1$. That is, we are interested in bounding

  $$\mathrm{Pr}_{\mathcal{W}^{N+1}}\{J_r^N(\theta(k), \mathbf{w}_a) < f(\theta(k), w^{(k+1)})\}.$$

  This problem has been addressed in the field of order statistics [1, 8] and in the context of randomized algorithms [18]. For example, for the particular case $r = 1$, Proposition 4 in [18] states that

  $$\mathrm{Pr}_{\mathcal{W}^{N+1}}\{J_1(\theta(k), \mathbf{w}_a) < f(\theta(k), w^{(k+1)})\} \le \frac{1}{N+1}.$$

- **Iterative Learning Control** [14]: Consider, for example, a chemical batch process in which before running each batch, the design control parameter is updated taking into consideration information of the past batches. Suppose that $\mathbf{w}_a = \{w^{(k-N+1)}, \ldots, w^{(k-1)}, w^{(k)}\}$ represents the last N sampled (i.i.d.) realizations of uncertainty. In this setting, given the design parameter vector $\theta(k)$ and the integer r, we are interested in characterizing the probability of violation of the constraints

  $$f(\theta(k), w^{(k+i)}) \le J_r^N(\theta(k), \mathbf{w}_a), \;\; i = 1, \ldots, M,$$

  where M denotes the number of uncertainty realizations required to model the next batch. Let us state the problem in a more formal way. Suppose that

  $$\mathbf{w}_b = \{w^{(k+1)}, w^{(k+2)}, \ldots, w^{(k+M)}\}$$

represents the next M future i.i.d. realizations of the uncertainty. Given the positive integer s, we are interested in bounding the probability

$$\Pr_{\mathcal{W}^{N+M}}\{J_r^N(\theta(k), \mathbf{w}_a) < J_s^M(\theta(k), \mathbf{w}_b)\},$$

which is the probability that more than s future samples out of a total of M do not satisfy the constraint $f(\theta(k), w) \leq J_r^N(\theta(k), \mathbf{w}_a)$.

In order to formalize the non-asymptotic result presented in this section, we introduce in the following definition the notion of probability of non-asymptotic failure.

Definition 5 (*Probability of non-asymptotic failure*) Given the positive integers r, s, N, and M with $r \leq N$ and $s \leq M$, suppose that

$$\mathbf{w} = \{w^{(1)}, w^{(2)}, \ldots, w^{(N+M)}\}$$

is drawn according to $\Pr_{\mathcal{W}^{N+M}}$. Denote

$$\begin{aligned}\mathbf{w}_a &= \{w^{(1)}, \ldots, w^{(N)}\},\\ \mathbf{w}_b &= \{w^{(N+1)}, \ldots, w^{(N+M)}\}.\end{aligned}$$

We define the probability of non-asymptotic failure $P_s^M(N, r)$ as

$$P_s^M(N, r) = \Pr_{\mathcal{W}^{N+M}}\{J_r^N(\theta, \mathbf{w}_a) < J_s^M(\theta, \mathbf{w}_b)\}.$$

Next property constitutes one of the main contributions of the chapter. It provides an upper bound to the probability of non-asymptotic failure.

Property 4 *Given $\theta \in \Theta$, the measurable function $f : \Theta \times \mathcal{W} \to \mathbb{R}$, the integers r, s, N, M with $1 \leq r \leq N$ and $1 \leq s \leq M$, we have*

$$P_s^M(N, r) \leq \sum_{i=0}^{r-1} \frac{\binom{N}{i}\binom{M}{q-i}}{\binom{N+M}{q}}, \tag{7}$$

where $q = r + s - 1$.

The inequality (7) is proved in Sect. 6. This result can also be derived using results from the field of order statistics [1, 8]. We notice that the binomial coefficients appearing in inequality (7) are defined as

$$\binom{n}{m} = \begin{cases} 0 & \text{if } n < m \\ \frac{n!}{m!(n-m)!} & \text{otherwise.} \end{cases}$$

Property 4 can be used to derive numerically the sample complexity N required to guarantee that the probability $P_s^M(N, r)$ does not exceed a given failure level $\delta \in (0, 1)$ for specific values of r, M and s. The following property provides an explicit sample complexity bound for the particular case $s = 1$. Notice that this case is specially relevant in engineering problems for moderate values of M. The non-failure situation for the choice $s = 1$ corresponds to

$$J_r^N(\theta, [w^{(1)}, \dots, w^{(N)}]^\top) \geq f(\theta, w^{(N+j)}), \quad j = 1, \dots, M.$$

That is, the probability of non-asymptotic failure for $s = 1$ is

$$P_1^M(N, r) = \Pr_{\mathcal{W}^{N+M}}\{J_r^N(\theta, [w^{(1)}, \dots, w^{(N)}]^\top) < \max_{j=1,\dots,M} f(\theta, w^{(N+j)})\}.$$

The following property shows how to explicitly choose N in order to guarantee that the probability of non-asymptotic failure $P_1^M(N, r)$ is no larger than a given failure level $\delta \in (0, 1)$.

Property 5 *Given $\theta \in \Theta$, the measurable function $f : \Theta \times \mathcal{W} \to \mathbb{R}$, the positive integers r, N and M, with $r \leq N$, we have*

$$P_1^M(N, r) \leq 1 - \left(1 - \frac{r}{N+1}\right)^M. \tag{8}$$

Moreover, given failure level $\delta \in (0, 1)$,

$$P_1^M(N, r) \leq \delta,$$

provided that

$$N \geq r - 1 + \frac{rM}{\delta}.$$

Proof See Appendix C.

The bounds on the probability of failure (both in its asymptotic and non-asymptotic versions) allow one to assess the probabilistic performance of a given design vector $\theta \in \Theta$. Therefore, the bounds provided in Properties 3 and 4 provide a way to address the robustness analysis. In order to pass from a probabilistic analysis framework to a design one, different schemes are possible. For example, if the design parameter set Θ has finite cardinality, then the "finite families for design" approach can be adopted (see [5, 6, 29]). Another possibility is to resort to sequential schemes as in [6, 20, 32] or to randomization in the design space [24].

6 Probabilistic Maximization by Sampling

We prove in this section the first claim of Property 4. We show that the probability of non-asymptotic failure $P_s^M(N,r)$ can be bounded by the hypergeometric distribution:

$$P_s^M(N,r) \leq \sum_{i=0}^{r-1} \frac{\binom{N}{i}\binom{M}{r+s-1-i}}{\binom{N+M}{r+s-1}}.$$

Once this inequality is proved, we will use it to derive the first claim of Property 3:

$$P_\eta(N,r) \leq \sum_{i=0}^{r-1} \binom{N}{i} \eta^i (1-\eta)^{N-i}.$$

In order to relate the probability of non-asymptotic failure with the hypergeometric distribution we first introduce some notations. We will denote by $\mathcal{C}_N^{N+M}$ the set representing all the different ways of drawing, without replacement, N elements out of a set of $N+M$. This is stated in a formal way in the following definition.

Definition 6 Given integers N and M, we say that the set of integers $I = \{i_1, i_2, \ldots, i_N\}$ belongs to $\mathcal{C}_N^{N+M}$ if and only if

$$1 \leq i_1 < i_2 < \ldots < i_{N-1} < i_N \leq N+M.$$

Moreover, given $I = \{i_1, i_2, \ldots, i_N\} \in \mathcal{C}_N^{N+M}$, we denote by I^c the complement of I in $1, \ldots, N+M$. That is, $I_c = \{j_1, j_2, \ldots, j_M\}$ with

$$1 \leq j_1 < j_2 < \ldots < j_{M-1} < j_M \leq N+M$$

and $I \bigcap I^c = \emptyset$.

Every element $I \in \mathcal{C}_N^{N+M}$ provides a different way of choosing N components from $\mathbf{v} \in \mathbb{R}^{N+M}$. Therefore, the cardinality of $\mathcal{C}_N^{N+M}$ is equal to

$$\binom{N+M}{N} = \frac{(N+M)!}{M!N!}.$$

The following definition introduces a notation that allows us to select from vector $\mathbf{v} \in \mathbb{R}^{N+M}$ a vector $\mathbf{z} \in \mathbb{R}^N$ composed by N components from $\mathbf{v} \in \mathbb{R}^{N+M}$.

Definition 7 Given $I = \{i_1, i_2, \ldots, i_N\} \in \mathcal{C}_N^{N+M}$ and vector

$$\mathbf{v} = [v^{(1)}, v^{(2)}, \ldots, v^{(N+M)}]^\top \in \mathbb{R}^{N+M},$$

we denote by

$$\mathbf{z} = \mathbf{v}(I)$$

the vector

$$\mathbf{z} = [z^{(1)}, z^{(2)}, \ldots, z^{(N)}]^\top \in \mathbb{R}^N,$$

where $z^{(j)} = v^{(i_j)}$, $j = 1, \ldots, N$.

It is clear that, given $\mathbf{v} \in \mathbb{R}^{N+M}$, $1 \leq r \leq N$, and a random element I of $\mathcal{C}_N^{N+M}$,

$$\phi(\mathbf{v}(I), r)$$

provides probabilistic information about the $N + M$ components of vector $\mathbf{v}$. We characterize in this section the values of integers N and r such that $\phi(\mathbf{v}(I), r)$ upper bounds with high probability the remaining components of $\mathbf{v}$, that is, the components of $\mathbf{v}(I^c)$. In particular, given $1 \leq s \leq M$, we obtain the probability that the generalized max $\phi(\mathbf{v}(I), r)$ does not upper bound the generalized maximum $\phi(\mathbf{v}(I^c), s)$. In the following property, the probability

$$\Pr_{\mathcal{C}_N^{N+M}}\{\phi(\mathbf{v}(I), r) < \phi(\mathbf{v}(I^c), s)\}$$

is bounded by means of the hypergeometric distribution.

Property 6 *Given the positive integers N, r, M, and s, with $1 \leq r \leq N$ and $1 \leq s \leq M$, we have*

$$\Pr_{\mathcal{C}_N^{N+M}}\{\phi(\mathbf{v}(I), r) < \phi(\mathbf{v}(I^c), s)\} \leq \sum_{i=0}^{r-1} \frac{\binom{N}{i}\binom{M}{q-i}}{\binom{N+M}{q}}, \quad \forall \mathbf{v} \in \mathbb{R}^{N+M},$$

where $q = r + s - 1$.

The proof of this property can be found in Appendix A.

We now present the main contribution of this section, which is the relationship between the probability of non-asymptotic failure and the hypergeometric distribution,

Property 7 *Given $\theta \in \Theta$, the measurable function $f : \Theta \times \mathcal{W} \to \mathbb{R}$, the integers r, s, N, M with $1 \leq r \leq N$ and $1 \leq s \leq M$, we have*

$$P_s^M(N, r) \leq \sum_{i=0}^{r-1} \frac{\binom{N}{i}\binom{M}{q-i}}{\binom{N+M}{q}},$$

where $q = r + s - 1$.

Proof Given the positive integers N, r, M, and s, with $1 \leq r \leq N$ and $1 \leq s \leq M$, suppose that

$$\mathbf{w} = \{w^{(1)}, w^{(2)}, \ldots, w^{(N+M)}\}$$

is drawn according to $\Pr_{\mathcal{W}^{N+M}}$. Let

$$\begin{aligned} \mathbf{w}_a &= \{w^{(1)}, \ldots, w^{(N)}\}, \\ \mathbf{w}_b &= \{w^{(N+1)}, \ldots, w^{(N+M)}\}. \end{aligned}$$

We have, by Definition 5, that the probability of non-asymptotic failure $P_s^M(N, r)$ is equal to

$$\Pr_{\mathcal{W}^{N+M}}\{J_r^N(\theta, \mathbf{w}_a) < J_s^M(\theta, \mathbf{w}_b)\}.$$

If we now define

$$\begin{aligned} \mathbf{v}_{\mathbf{w}} &= [f(\theta, w^{(1)}), f(\theta, w^{(2)}), \ldots, f(\theta, w^{(N+M)})]^\top \\ &= [v_1, v_2, \ldots, v_N, v_{N+1}, \ldots, v_{N+M}]^\top \\ \mathbf{v}_a(\mathbf{w}) &= [v_1, v_2, \ldots, v_N]^\top \\ \mathbf{v}_b(\mathbf{w}) &= [v_{N+1}, v_{N+2}, \ldots, v_{N+M}]^\top. \end{aligned}$$

we obtain, by Definition 2,

$$\begin{aligned} J_r^N(\theta, \mathbf{w}_a) &= \phi(\mathbf{v}_a(\mathbf{w}), r) \\ J_s^M(\theta, \mathbf{w}_b) &= \phi(\mathbf{v}_b(\mathbf{w}), s). \end{aligned}$$

Therefore,

$$\begin{aligned} P_s^M(N, r) &= \Pr_{\mathcal{W}^{N+M}}\{J_r^N(\theta, \mathbf{w}_a) < J_s^M(\theta, \mathbf{w}_b)\} \\ &= \Pr_{\mathcal{W}^{N+M}}\{\phi(\mathbf{v}_a(\mathbf{w}), r) < \phi(\mathbf{v}_b(\mathbf{w}), s)\}. \end{aligned}$$

Since the elements of $\mathbf{w}$ are i.i.d., the order in which they are drawn does not affect the probability. That is,

$$\begin{aligned} P_s^M(N, r) &= \Pr_{\mathcal{W}^{N+M}}\{\phi(\mathbf{v}_a(\mathbf{w}), r) < \phi(\mathbf{v}_b(\mathbf{w}), s)\} \\ &= \Pr_{\mathcal{W}^{N+M}, \mathcal{C}_N^{N+M}}\{\phi(\mathbf{v}_{\mathbf{w}}(I), r) < \phi(\mathbf{v}_{\mathbf{w}}(I^c), s)\}. \end{aligned}$$

In view of Property 6 we have that, for every $\mathbf{v}_{\mathbf{w}} \in \mathbb{R}^{N+M}$,

$$\Pr_{\mathcal{C}_N^{N+M}}\{\phi(\mathbf{v}_{\mathbf{w}}(I), r) < \phi(\mathbf{v}_{\mathbf{w}}(I^c), s)\} \leq \sum_{i=0}^{r-1} \frac{\binom{N}{i}\binom{M}{q-i}}{\binom{N+M}{q}}.$$

Since the previous inequality is valid for every $\mathbf{v}_{\mathbf{w}} \in \mathbb{R}^{N+M}$ we obtain that the probability of non-asymptotic failure can be bounded by an expression that does not depend on the particular probability measure $\Pr_{\mathcal{W}}$. That is,

$$\begin{aligned} P_s^M(N,r) &= \Pr_{\mathcal{W}^{N+M},\mathcal{C}_N^{N+M}}\{\phi(\mathbf{v}_{\mathbf{w}}(I),r) < \phi(\mathbf{v}_{\mathbf{w}}(I^c),s)\} \\ &\leq \sum_{i=0}^{r-1} \frac{\binom{N}{i}\binom{M}{q-i}}{\binom{N+M}{q}}. \end{aligned}$$

■

We now prove the first claim of Property 3. That is, we prove the following property.

Property 8 *Given $\theta \in \Theta$, the measurable function $f : \Theta \times \mathcal{W} \to \mathbb{R}$, the integers r and N with $1 \leq r \leq N$ and $\eta \in (0,1)$, we have*

$$P_\eta(N,r) = \Pr_{\mathcal{W}^N}\{E(\theta, J_r^N(\theta,\mathbf{w})) > \eta\} \leq \sum_{i=0}^{r-1} \binom{N}{i} \eta^i (1-\eta)^{N-i}.$$

Proof Suppose that M is an integer and that $s(M)$ denotes the smallest integer no smaller than ηM. That is, $s(M) = \lceil \eta M \rceil$. Suppose now that

$$\mathbf{w} = \{w^{(1)}, w^{(2)}, \ldots, w^{(N+M)}\}$$

is drawn according to $\Pr_{\mathcal{W}^{N+M}}$. Let

$$\begin{aligned} \mathbf{w}_a &= \{w^{(1)}, \ldots, w^{(N)}\}, \\ \mathbf{w}_b &= \{w^{(N+1)}, \ldots, w^{(N+M)}\}. \end{aligned}$$

Suppose that $\gamma \in \mathbb{R}$ satisfies

$$\gamma \geq J_{s(M)}^M(\theta, \mathbf{w}_b).$$

Then, the fraction of the components of the vector

$$\mathbf{v}_b = [f(\theta, w^{(N+1)}), f(\theta, w^{(N+2)}), \ldots, f(\theta, w^{(N+M)})]^\top \in \mathbb{R}^M$$

strictly larger than γ is no larger than

$$\frac{s(M)-1}{M} = \frac{\lceil \eta M \rceil - 1}{M} < \frac{\eta M + 1 - 1}{M} = \eta.$$

From this we infer that if M tends to infinity, then the inequality

$$\gamma \geq J^M_{s(M)}(\theta, \mathbf{w}_b)$$

implies, with probability 1, that

$$E(\theta, \gamma) = \Pr_{\mathcal{W}}\{f(\theta, w) \leq \gamma\} \leq \eta.$$

If we make now γ equal to $J^N_r(\theta, \mathbf{w}_a)$, we obtain that

$$J^N_r(\theta, \mathbf{w}_a) \leq \lim_{M\to\infty} J^M_{s(M)}(\theta, \mathbf{w}_b)$$

implies

$$E(\theta, J^N_r(\theta, \mathbf{w}_a)) \leq \eta.$$

Therefore, the probability of asymptotic failure can be bounded by the probability of the event

$$\Pr_{\mathcal{W}^{N+M}}\{J^N_r(\theta, \mathbf{w}_a) > \lim_{M\to\infty} J^M_{s(M)}(\theta, \mathbf{w}_b)\} = \lim_{M\to\infty} P^M_{s(M)}(N, r)$$

That is,

$$P_\eta(N, r) \leq \lim_{M\to\infty} P^M_{s(M)}(N, r).$$

From Property 4, we have

$$\begin{aligned} P_\eta(N, r) &\leq \lim_{M\to\infty} P^M_{s(M)}(N, r) \\ &\leq \lim_{M\to\infty} \sum_{i=0}^{r-1} \frac{\binom{N}{i}\binom{M}{r+s(M)-1-i}}{\binom{N+M}{r+s(M)-1}}, \\ &= \sum_{i=0}^{r-1} \binom{N}{i} \eta^i (1-\eta)^{N-i}. \end{aligned}$$

We notice that the last inequality is due to the asymptotic convergence of the hypergeometric distribution to the binomial one (see Property 11 in Appendix D). ■

7 Conclusions

In this chapter we proposed a novel methodology, based on probabilistic maximization, to address robust design problems. This technique allows one to assess the probabilistic performance of a given design parameter vector. The proposed approach is well suited for non-asymptotic schemes as adaptive control or iterative learning control. Concentration inequalities that allow one to guarantee that the obtained design parameters meet some probabilistic specifications are presented. The proposed methodology does not rely on any convexity assumption or precise knowledge of the probabilistic distribution of the underlying uncertainty. Moreover, the required sample complexity does not depend on the dimension of the design problem.

Acknowledgements This chapter has been written in the memory of Roberto Tempo, with whom we had a strong relationship not only on the academic side, but also on the personal one. During these years, Roberto was always an inspiration because of his positive and enthusiastic attitude. We are indebted to him for sharing with us his vision on robust control. He was one of the pioneers of the field of randomized algorithms [11, 36]. The monograph "Randomized Algorithms for Analysis and Control of Uncertain Systems" [37], written by Roberto Tempo and co-workers, laid the foundations of randomized algorithms in control.

The authors acknowledge MEyC Spain (contract DPI2016-76493-C3-1-R), the University of Sevilla (VI PPITUS), and the European Research Council (Advanced Research Grant 769051 OCONTSOLAR).

Appendices

A: Properties of the Generalized Max Function

We now prove Property 6, which is rewritten here for the convenience of the reader.

Property 9 *Given the positive integers N, r and s with $1 \leq r \leq N$ and $1 \leq s \leq M$*

$$\Pr_{\mathcal{C}_N^{N+M}}\{\phi(\mathbf{v}(I), r) < \phi(\mathbf{v}(I^c), s)\} \leq \sum_{i=0}^{r-1} \frac{\binom{N}{i}\binom{M}{q-i}}{\binom{N+M}{q}}, \quad \forall \mathbf{v} \in \mathbb{R}^{N+M},$$

where $q = r + s - 1$.

Proof Denote by

$$\mathbf{v}_+ = [v_+^{(1)}, v_+^{(2)}, \ldots, v_+^{(N+M)}]^\top \in \mathbb{R}^{N+M}$$

the vector composed by the components of $\mathbf{v} \in \mathbb{R}^{N+M}$ ordered in non-increasing order. Denote also by $\mathbf{z}$ the first $q = r + s - 1$ components of $\mathbf{v}_+$. That is,

$$\mathbf{z} = [v_+^{(1)}, v_+^{(2)}, \ldots, v_+^{(q)}]^\top \in \mathbb{R}^q.$$

Notice that, by construction, the smallest component of $\mathbf{z}$ is equal to v_+^q. Given $I \in \mathcal{C}_N^{M+N}$, suppose that at least r components of $\mathbf{v}(I)$ are included among the components of $\mathbf{z}$. This implies that

$$\phi(\mathbf{v}(I), r) \geq v_+^q. \tag{9}$$

Moreover, under the assumption that at least r components of $\mathbf{v}(I)$ are included in $\mathbf{z}$, no more than $q - r = s - 1$ components of $\mathbf{v}(I^c)$ are included in $\mathbf{z}$. This means that

$$\phi(\mathbf{v}(I^c), s) \leq v_+^q. \tag{10}$$

We conclude from inequalities (9) and (10) that if r or more components of $\mathbf{v}(I)$ are included in $\mathbf{z}$, then

$$\phi(\mathbf{v}(I), r) \geq v_+^q \geq \phi(\mathbf{v}(I^c), s).$$

Therefore, the probability of non-asymptotic failure is no larger than the probability that fewer than r elements of $\mathbf{v}(I)$ are included in $\mathbf{z}$. Given i with $0 \leq i \leq r - 1$, the probability that $\mathbf{v}(I)$ has exactly i components in $\mathbf{z}$ is given by

$$\frac{\begin{pmatrix} N \\ i \end{pmatrix} \begin{pmatrix} M \\ q-i \end{pmatrix}}{\begin{pmatrix} N+M \\ q \end{pmatrix}}.$$

Therefore, the probability

$$\text{Pr}_{\mathcal{C}_N^{N+M}} \{\phi(\mathbf{v}(I), r) < \phi(\mathbf{v}(I^c), s)\}$$

is bounded by

$$\sum_{i=0}^{r-1} \frac{\begin{pmatrix} N \\ i \end{pmatrix} \begin{pmatrix} M \\ q-i \end{pmatrix}}{\begin{pmatrix} N+M \\ q \end{pmatrix}}.$$

■

B: Inequalities for the Natural Logarithm

The following lemma states well-known bounds for the the natural logarithm. Such inequalities are frequently used in the context of probability theory [23].

Lemma 1 *Suppose that* $z \in [0, 1)$. *Then,*

$$z \leq z + \frac{z^2}{2} \leq \ln\left(\frac{1}{1-z}\right) \leq z + \frac{z^2}{2(1-z)} \leq \frac{z}{1-z}.$$

Proof The first inequality is trivial. From the well-known Taylor series expansion for the natural logarithm (see, for example, [23, II.8]) we have

$$\ln\left(\frac{1}{1-z}\right) = \sum_{k=1}^{\infty} \frac{z^k}{k}.$$

Since $z \geq 0$, we have

$$\ln\left(\frac{1}{1-z}\right) \geq \sum_{k=1}^{2} \frac{z^k}{k} = z + \frac{z^2}{2}.$$

This proves the second inequality. We now prove the third one:

$$\begin{aligned}\ln\left(\frac{1}{1-z}\right) &= \sum_{k=1}^{\infty} \frac{z^k}{k} = z + \sum_{k=2}^{\infty} \frac{z^k}{k} \\ &\leq z + \frac{z^2}{2} \sum_{k=0}^{\infty} z^k = z + \frac{z^2}{2(1-z)}.\end{aligned}$$

We notice that in the last equality we used the equality $\sum\limits_{k=0}^{\infty} z^k = \frac{1}{1-z}$. The fourth inequality follows directly from

$$z + \frac{z^2}{2(1-z)} = z\left(1 + \frac{z}{2(1-z)}\right) = z\left(\frac{2-z}{2(1-z)}\right) \leq \frac{z}{1-z}.$$

■

C: Non-asymptotic Failure ($s = 1$)

We now prove in this appendix Property 5, which is rewritten here for the convenience of the reader.

Property 10 *Given $\theta \in \Theta$, the measurable function $f : \Theta \times \mathcal{W} \to \mathbb{R}$, the positive integers r, N and M, with $r \leq N$, we have*

$$P_1^M(N, r) \leq 1 - \left(1 - \frac{r}{N+1}\right)^M. \qquad (11)$$

Moreover, given $\delta \in (0, 1)$,

$$P_1^M(N, r) \leq \delta,$$

provided that

$$N \geq r - 1 + \frac{rM}{\delta}.$$

Proof We first show that inequality (11) is satisfied for the particular case $M = 1$. That is, we will prove that $P_1^1(N, r) \leq 1 - (1 - \frac{r}{N+1})^1 = \frac{r}{N+1}$. From Property 4 and $M = s = 1$, we have

$$P_1^1(N, r) \leq \sum_{i=0}^{r-1} \frac{\begin{pmatrix} N \\ i \end{pmatrix} \begin{pmatrix} 1 \\ r-i \end{pmatrix}}{\begin{pmatrix} N+1 \\ r \end{pmatrix}}.$$

Moreover, since

$$\begin{pmatrix} 1 \\ r-i \end{pmatrix} = \begin{cases} 1 & \text{if } r-i=1 \\ 0 & \text{if } r-i>1 \end{cases},$$

we obtain that

$$\begin{aligned} P_1^1(N, r) &\leq \sum_{i=0}^{r-1} \frac{\begin{pmatrix} N \\ i \end{pmatrix} \begin{pmatrix} 1 \\ r-i \end{pmatrix}}{\begin{pmatrix} N+1 \\ r \end{pmatrix}} = \sum_{i=r-1}^{r-1} \frac{\begin{pmatrix} N \\ i \end{pmatrix} \begin{pmatrix} 1 \\ r-i \end{pmatrix}}{\begin{pmatrix} N+1 \\ r \end{pmatrix}} = \frac{\begin{pmatrix} N \\ r-1 \end{pmatrix} \begin{pmatrix} 1 \\ 1 \end{pmatrix}}{\begin{pmatrix} N+1 \\ r \end{pmatrix}} \\ &= \left(\frac{N!}{(r-1)!(N-r+1)!} \right) \left(\frac{r!(N-r+1)!}{(N+1)!} \right) = \frac{r}{N+1}. \end{aligned}$$

That is,

$$P_1^1(N, r) \leq \frac{r}{N+1}. \tag{12}$$

We now prove that if inequality (11) is satisfied for $M - 1 \geq 1$, then it is also satisfied for M. That is, we assume now that

$$P_1^{M-1}(N, r) \leq 1 - \left(1 - \frac{r}{N+1}\right)^{M-1}, \tag{13}$$

and use this assumption to prove inequality (11). Suppose that

$$\mathbf{w} = [w^{(1)}, w^{(2)}, \ldots, w^{(N+M)}]^\top$$

is drawn according to $\Pr_{\mathcal{W}^{M+N}}$. Given integers i and j with $1 \leq i \leq j \leq M$, denote by A_i^j the event

$$J_r^N(\theta, [w^{(1)}, \ldots, w^{(N)}]^\top) \geq f(\theta, w^{(N+k)}), \quad k = i, \ldots, j.$$

With this notation, we have that the event A_1^M is the complement of the event

$$\begin{aligned} J_r^N(\theta, [w^{(1)}, \ldots, w^{(N)}]^\top) &< \max_{j=1,\ldots,M} f(\theta, w^{(N+j)}) \\ &= J_1^M(\theta, [w^{(N+1)}, \ldots, w^{(N+M)}]^\top). \end{aligned}$$

That is,

$$\Pr_{\mathcal{W}^{N+M}}\{A_1^M\} = 1 - P_1^M(N, r). \tag{14}$$

Similarly, we have

$$\begin{aligned} \Pr_{\mathcal{W}^{N+M}}\{A_M^M\} &= \Pr_{\mathcal{W}^{N+M}}\{J_r^N(\theta, [w^{(1)}, \ldots, w^{(N)}]^\top) \geq f(\theta, w^{(N+M)})\} \\ &= \Pr_{\mathcal{W}^{N+M}}\{J_r^N(\theta, [w^{(1)}, \ldots, w^{(N)}]^\top) \geq f(\theta, w^{(N+1)})\} \\ &= 1 - P_1^1(N, r). \end{aligned}$$

We now show how to obtain a bound on $P_1^M(N, r)$ from the bound on $P_1^{M-1}(N, r)$. Notice that

$$\begin{aligned} 1 - \Pr_1^M(N, r) &= \Pr_{\mathcal{W}^{N+M}}\{A_1^M\} \\ &= \Pr_{\mathcal{W}^{N+M}}\{A_1^{M-1} \text{ and } A_M^M\} \\ &= \Pr_{\mathcal{W}^{N+M}}\{A_1^{M-1}\}\Pr_{\mathcal{W}^{N+M}}\{A_M^M | A_1^{M-1}\}. \end{aligned} \tag{15}$$

We notice that the event A_1^{M-1} provides statistical evidence for the satisfaction of the inequality

$$J_r(N, r) \geq f(\theta, w^{(N+M)}).$$

That is,

$$\Pr_{\mathcal{W}^{N+M}}\{A_M^M | A_1^M\} \geq \Pr_{\mathcal{W}^{N+M}}\{A_M^M | \text{ not } A_1^M\}.$$

From this inequality we infer

$$\begin{aligned} \Pr_{\mathcal{W}^{N+M}}\{A_M^M\} &= \Pr_{\mathcal{W}^{N+M}}\{A_1^M\}\Pr_{\mathcal{W}^{N+M}}\{A_M^M | A_1^M\} \\ &\quad + \Pr_{\mathcal{W}^{N+M}}\{\text{ not } A_1^M\}\Pr_{\mathcal{W}^{N+M}}\{A_M^M | \text{ not } A_1^M\} \\ &\leq \Pr_{\mathcal{W}^{N+M}}\{A_1^M\}\Pr_{\mathcal{W}^{N+M}}\{A_M^M | A_1^M\} \\ &\quad + \Pr_{\mathcal{W}^{N+M}}\{\text{ not } A_1^M\}\Pr_{\mathcal{W}^{N+M}}\{A_M^M | A_1^M\} \\ &= (\Pr_{\mathcal{W}^{N+M}}\{A_1^M\} + \Pr_{\mathcal{W}^{N+M}}\{\text{ not } A_1^M\})\Pr_{\mathcal{W}^{N+M}}\{A_M^M | A_1^M\} \\ &= \Pr_{\mathcal{W}^{N+M}}\{A_M^M | A_1^M\}. \end{aligned}$$

Thus,

$$\Pr_{\mathcal{W}^{N+M}}\{A_M^M | A_1^M\} \geq \Pr_{\mathcal{W}^{N+M}}\{A_M^M\}.$$

From this and Eq. (15), we obtain,

$$\begin{aligned}1 - \mathrm{Pr}_1^M(N, r) &= \mathrm{Pr}_{\mathcal{W}^{N+M}}\{A_1^{M-1}\}\mathrm{Pr}_{\mathcal{W}^{N+M}}\{A_M^M | A_1^{M-1}\}\\ &\geq \mathrm{Pr}_{\mathcal{W}^{N+M}}\{A_1^{M-1}\}\mathrm{Pr}_{\mathcal{W}^{N+M}}\{A_M^M\}\\ &= (1 - P_1^{M-1}(N, r))(1 - P_1^1(N, r)).\end{aligned}$$

We conclude, in view of this and inequalities (12) and (13), that

$$\begin{aligned}1 - \mathrm{Pr}_1^M(N, r) &\geq (1 - \frac{r}{N+1})^{M-1}(1 - \frac{r}{N+1})\\ &= (1 - \frac{r}{N+1})^M.\end{aligned}$$

That is,

$$\mathrm{Pr}_1^M(N, r) \leq 1 - \left(1 - \frac{r}{N+1}\right)^M. \tag{16}$$

This proves the first claim of the property. We now prove the second one. In view of inequality (16), a sufficient condition for $P_1^M(N, r) \leq \delta$ is

$$1 - \left(1 - \frac{r}{N+1}\right)^M \leq \delta.$$

Equivalently,

$$1 - \delta \leq \left(1 - \frac{r}{N+1}\right)^M,$$

$$\ln(1 - \delta) \leq M \ln\left(1 - \frac{r}{N+1}\right).$$

$$\ln\left(\frac{1}{1 - \delta}\right) \geq M \ln\left(\frac{1}{1 - \frac{r}{N+1}}\right). \tag{17}$$

Since $\frac{r}{N+1} \in (0, 1)$, we infer from Lemma 1 (see Appendix B) that

$$\ln\left(\frac{1}{1 - \frac{r}{N+1}}\right) \leq \frac{\frac{r}{N+1}}{1 - \frac{r}{N+1}} = \frac{r}{N + 1 - r}.$$

Therefore, a sufficient condition for inequality (17) is

$$\ln\left(\frac{1}{1 - \delta}\right) \geq \frac{rM}{N + 1 - r}. \tag{18}$$

We have also from Lemma 1 that

$$\delta \le \ln\left(\frac{1}{1-\delta}\right).$$

Therefore, a sufficient condition for inequality (18), and consequently for inequality $P_1^M(N,r) \le \delta$ is

$$\delta \ge \frac{rM}{N+1-r},$$

$$N+1-r \ge \frac{rM}{\delta},$$

$$N \ge r-1+\frac{rM}{\delta}.$$

■

D: Asymptotic Behavior of the Hypergeometric Distribution

It is well known that the hypergeometric distribution can be approximated by a binomial distribution when the total population tends to infinity [23, II.11]. Next property states the asymptotic approximation of the hypergeometric distribution by a binomial one.

Property 11 *Suppose that the integers r and N, with* $1 \le r \le N$*, are given. Suppose also that the sequence* $s(1)$,$s(2)$, ..., $s(M)$ *satisfies*

$$\lim_{M\to\infty} \frac{s(M)}{M} = \eta \in (0,1).$$

Then,

$$\lim_{M\to\infty} \sum_{i=0}^{r-1} \frac{\binom{N}{i}\binom{M}{q(M)-i}}{\binom{N+M}{q(M)}} = \sum_{i=1}^{r-1} \binom{N}{i} \eta^i (1-\eta)^{N-i},$$

where $q(M) = r + s(M) - 1$.

Proof In order to simplify the notation, we do not make explicit the dependence with respect to M in $q(M)$.

$$\begin{aligned} \Upsilon_i &= \binom{N+M}{q}^{-1} \binom{M}{q-i} \\ &= \left(\frac{q!(N+M-q)!}{(N+M)!}\right)\left(\frac{M!}{(q-i)!(M-q+i)!}\right) \end{aligned}$$

$$
\begin{aligned}
&= \frac{q!}{(q-i)!}\left(\frac{M!}{(N+M)!}\right)\left(\frac{(N+M-q)!}{(M-q+i)!}\right)\\
&= \frac{q!}{(q-i)!}\left(\frac{(N+M-q)(N+M-q-1)\ldots(M-q+i+1)}{(N+M)(N+M-1)\ldots(M+1)}\right)\\
&= \frac{q!}{(q-i)!}\left(\frac{\prod\limits_{j=1}^{N-i} N+M-q+1-j}{\prod\limits_{j=1}^{N} N+M+1-j}\right)\\
&= \frac{q!}{(q-i)!}\left(\frac{\prod\limits_{j=1}^{N-i} N+M-q+1-j}{\prod\limits_{j=1}^{N-i} N+M+1-j \prod\limits_{j=N-i+1}^{N} N+M+1-j}\right)\\
&= \left(\prod_{j=1}^{i} q-j+1\right)\left(\frac{\prod\limits_{j=1}^{N-i} N+M-q+1-j}{\prod\limits_{j=1}^{N-i}(N+M+1-j)\prod\limits_{j=1}^{i} M+j}\right)\\
&= \prod_{j=1}^{i}\frac{q-j+1}{M+j}\prod_{j=1}^{N-i}\frac{N+M-q+1-j}{N+M+1-j} \qquad (19)
\end{aligned}
$$

Recall that $q = q(M) = r + s(M) - 1$. Since r and N are given (bounded) integers we have from the assumptions of the property that

$$
\begin{aligned}
\lim_{M\to\infty}\frac{q(M)}{M} &= \lim_{M\to\infty}\frac{r+s(M)-1}{M} = \eta\\
\lim_{M\to\infty}\frac{q(M)-j+1}{M+j} &= \lim_{M\to\infty}\frac{q(M)}{M} = \eta, \quad j = 1,\ldots,r-1,\\
\lim_{M\to\infty}\frac{N+M-q(M)+1-j}{N+M+1-j} &= \lim_{M\to\infty}\left(1-\frac{q(M)}{M}\right) = 1-\eta, \quad j = 1,\ldots,N.
\end{aligned}
$$

Therefore, from equality (19), we obtain

$$
\lim_{M\to\infty}\Upsilon_i = \prod_{j=1}^{i}\eta\prod_{j=1}^{N-i}(1-\eta) = \eta^i(1-\eta)^{N-i}.
$$

We conclude

$$\lim_{M\to\infty}\sum_{i=0}^{r-1}\frac{\binom{N}{i}\binom{M}{q(M)-i}}{\binom{N+M}{q(M)}} = \lim_{M\to\infty}\sum_{i=0}^{r-1}\binom{N}{i}\Upsilon_i$$

$$= \sum_{i=0}^{r-1}\binom{N}{i}\eta^i(1-\eta)^{N-i}.$$

■

References

1. M. Ahsanullah, V.B. Nevzorov, and M. Shakil. *An introduction to Order Statistics*. Atlantis Press, Paris, 2013.
2. M. Alamir. On probabilistic certification of combined cancer therapies using strongly uncertain models. *Journal of Theoretical Biology*, 384:59–69, 2015.
3. T. Alamo, R. Tempo, and E.F. Camacho. Improved sample size bounds for probabilistic robust control design: A pack-based strategy. In *Proceedings of the 2007 IEEE Conference on Decision and Control*, pages 6178–6183, 2007.
4. T. Alamo, R. Tempo, and E.F. Camacho. Randomized strategies for probabilistic solutions of uncertain feasibility and optimization problems. *IEEE Transactions on Automatic Control*, 54(11):2545–2559, 2009.
5. T. Alamo, R. Tempo, and A. Luque. On the sample complexity of randomized approaches to the analysis and design under uncertainty. In *Proceedings of the American Control Conference*, Baltimore, USA, 2010.
6. T. Alamo, R. Tempo, A. Luque, and D.R. Ramirez. Randomized methods for design of uncertain systems: sample complexity and sequential algorithms. *Automatica*, 52:160–172, 2015.
7. T. Alamo, R. Tempo, D.R. Ramirez, and E.F. Camacho. A sequentially optimal randomized algorithm for robust LMI feasibility problems. In *Proceedings of the European Control Conference*, Kos, Greece, 2007.
8. B.C Arnold, N. Balakrishnan, and H.N. Nagaraja. *A First Course in Order Statistics*. John Wiley and Sons, New York, 1992.
9. K.J. Åström and B. Wittenmark. *Adaptive Control*. Addison Wesley, second edition, 1995.
10. E.W. Bai, H. Cho, R. Tempo, and Y. Ye. Optimization with few violated constraints for linear bounded error parameter estimation. *IEEE Transactions on Automatic Control*, 47(4):1067–1077, 2002.
11. E.W. Bai, R. Tempo, and Y. Ye. Worst-case properties of the uniform distribution and randomized algorithms for robustness analysis. *Mathematics of Control, Signals and Systems*, 11(4):183–196, 1998.
12. C.M. Bishop. *Pattern Recognition and Machine Learning*. Springer, 2006.
13. V. Blondel and J.N. Tsitsiklis. A survey of computational complexity results in systems and control. *Automatica*, 36:1249–1274, 2000.
14. D.A. Bristow, M. Tharayil, and A.G. Alleyne. A survey of iterative learning control. *IEEE Control Systems Magazine*, 26(3):96–114, 2006.
15. G. Calafiore. Random convex programs. *SIAM Journal of Optimization*, 20:3427–3464, 2010.
16. G. Calafiore and M.C. Campi. The scenario approach to robust control design. *IEEE Transactions on Automatic Control*, 51(5):742–753, 2006.

17. M.C. Campi, G. Calafiore, and S. Garatti. Interval predictor models: Identification and reliability. *Automatica*, 45:382–392, 2009.
18. M.C. Campi and G.C. Calafiore. Notes on the scenario design approach. *IEEE Transactions on Automatic Control*, 54(2):382–385, 2009.
19. M.C. Campi and S. Garatti. The exact feasibility of randomized solutions of robust convex programs. *SIAM Journal of Optimization*, 19:1211–1230, 2008.
20. M. Chamanbaz, F. Dabbene, R. Tempo, V. Venkataramanan, and Q.-G. Wang. Sequential randomized algorithms for convex optimization in the presence of uncertainty. *IEEE Transactions of Automatic Control*, 61(9):2565–2571, 2016.
21. F. Dabbene, P.S. Shcherbakov, and B.T. Polyak. A randomized cutting plane method with probabilistic geometric convergence. *SIAM Journal of Optimization*, 20:3185–3207, 2010.
22. D.P. de Farias and B. Van Roy. The linear programming approach to approximate dynamic programming. *Operations Research*, 51:850–865, 2003.
23. W. Feller. *An Introduction to Probability Theory and its Applications*. John Wiley and Sons, New York, third edition, 1968.
24. Y. Fujisaki and Y. Kozawa. *Probabilistic robust controller design: probable near minimax value and randomized algorithms*. In G. Calafiore and F. Dabbene, editors, Probabilistic and Randomized Methods for Design under Uncertainty, Springer, London, 2006.
25. S. Grammatico, X. Zhang, K. Margellos, P. Goulart, and J. Lygeros. A scenario approach for non-convex control design. *IEEE Transactions on Automatic Control*, 61(2):334–345, 2016.
26. L. Ljung. *System Identification*. Prentice Hall, Upper Saddle River, NJ, second edition, 1999.
27. M. Lorenzen, F. Dabbene, R. Tempo, and F. Allgöwer. Constraint-tightening and stability in stochastic model predictive control. *IEEE Transactions on Automatic Control*, 62(7):3165–3177, 2017.
28. M. Lorenzen, F. Dabbene, R. Tempo, and F. Allgöwer. Stochastic MPC with offline uncertainty sampling. *Automatica*, 81:176–183, 2017.
29. J. Luedtke and S. Ahmed. A sample approximation approach for optimization with probabilistic constraints. *SIAM Journal on Optimization*, 2(19):674–699, 2008.
30. K. Margellos, P. Goulart, and J. Lygeros. On the road between robust optimization and the scenario approach for chance constrained optimization problems. *IEEE Transactions on Automatic Control*, 59(8):2258–2263, 2014.
31. K.P. Murphy. *Machine Learning: A Probabilistic Perspective*. The MIT Press, 2012.
32. Y. Oishi. Polynomial-time algorithms for probabilistic solutions of parameter-dependent linear matrix inequalities. *Automatica*, 43:538–545, 2007.
33. I.R. Petersen and R. Tempo. Robust control of uncertain systems: Classical results and recent developments. *Automatica*, 50:1315–1335, 2014.
34. P. Pflaum, M. Alamir, and M.Y. Lamoudi. Battery sizing for PV power plants under regulations using randomized algorithms. *Renewable Energy*, 113:596–607, 2017.
35. P. Pflaum, M. Alamir, and M.Y. Lamoudi. Probabilistic energy management strategy for EV charging stations using randomized algorithms. *IEEE Transactions on Control Systems Technology*, 26(3):1099–1106, 2018.
36. R. Tempo, E.W. Bai, and F. Dabbene. Probabilistic robustness analysis: explicit bounds for the minimum number of samples. *Systems & Control Letters*, 30:237–242, 1997.
37. R. Tempo, G. Calafiore, and F. Dabbene. *Randomized Algorithms for Analysis and Control of Uncertain Systems, with Applications*. Springer-Verlag, London, second edition, 2013.
38. V.N. Vapnik. *Statistical Learning Theory*. John Wiley and Sons, New York, 1998.
39. M. Vidyasagar. *A Theory of Learning and Generalization*. Springer, London, 1997.

Compressive Sensing and Algebraic Coding: Connections and Challenges

Mathukumalli Vidyasagar and Mahsa Lotfi

Abstract Compressive sensing refers to the reconstruction of high dimensional but low-complexity objects from relatively few measurements. Examples of such objects include: high dimensional but sparse vectors, large images with very few sharp edges, and high-dimensional matrices of low rank. One of the most popular methods for reconstruction is to solve a suitably constrained ℓ_1-norm minimization problem, otherwise known as basis pursuit (BP). In this approach, a key role is played by the measurement matrix, which converts the high dimensional but sparse vector (for example) into a low-dimensional real-valued measurement vector. The widely used sufficient conditions for guaranteeing that BP recovers the unknown vector are the restricted isometry property (RIP), and the robust null space property (RNSP). It has recently been shown that the RIP implies the RNSP. There are two approaches for generating matrices that satisfy the RIP, namely, probabilistic and deterministic. Probabilistic methods are older. In this approach, the measurement matrix consists of samples of a Gaussian or sub-Gaussian random variable. This approach leads to measurement matrices that are "order optimal," in that the number of measurements required is within a constant factor of the optimum achievable. However, in practice, such matrices have no structure, which leads to enormous storage requirements and CPU time. Recently, the emphasis has shifted to the use of sparse binary matrices, which require less storage and are much faster than randomly generated matrices. A recent trend has been the use of methods from algebraic coding theory, in particular, expander graphs and low-density parity-check (LDPC) codes,

This research was supported by the National Science Foundation, USA under Award #ECCS-1306630, and by the Department of Science and Technology, Government of India.

M. Vidyasagar (✉)
Indian Institute of Technology Hyderabad, Kandi, India
e-mail: m.vidyasagar@iith.ac.in; m.vidyasagar@utdallas.edu

M. Vidyasagar
The University of Texas at Dallas, Richardson, TX, USA

M. Lotfi
Erik Jonsson School of Engineering and Computer Science, The University of Texas at Dallas, Richardson, TX 75080, USA
e-mail: mafilot@gmail.com

© Springer Nature Switzerland AG 2018

T. Başar (ed.), *Uncertainty in Complex Networked Systems*,
Systems & Control: Foundations & Applications,
https://doi.org/10.1007/978-3-030-04630-9_8

to construct sparse binary measurement matrices. In this chapter, we will first briefly summarize the known results on compressed sensing using both probabilistic and deterministic approaches. In the first part of the chapter, we introduce some new constructions of sparse binary measurement matrices based on low-density parity-check (LDPC) codes. Then, we describe some of our recent results that lead to the fastest available algorithms for compressive sensing in specific situations. We suggest some interesting directions for future research.

1 Introduction

Compressed sensing refers to the recovery of high dimensional but low-complexity entities from a limited number of measurements. The most widely studied applications of compressed sensing are the recovery of a high dimensional but low-rank matrix from a small number of linear projections of the matrix, and vector recovery, which is the problem studied in this chapter. Specifically, the problem is to recover a vector $x \in \mathbb{R}^n$ where only $k \ll n$ components are significant and the rest are either zero or small, from a set of linear measurements $y = Ax$ where $A \in \mathbb{R}^{m \times n}$. A variant is when $y = Ax + \eta$ where η denotes measurement noise, and a prior bound of the form $\|\eta\| \leq \varepsilon$ is available. By far, the most popular solution methodology for this problem is *basis pursuit* in which an approximation $\hat{x}$ to the unknown vector x is constructed via

$$\hat{x} := \operatorname*{argmin}_{z} \|z\|_1 \text{ s.t. } \|y - Az\| \leq \varepsilon. \tag{1}$$

The basis pursuit approach (with $\eta = 0$ so that the constraint in (1) becomes $y = Az$) was proposed in [1, 2], but without guarantees on its performance. During the mid-2000s, several papers derived sufficient conditions on the measurement matrix A under which basis pursuit leads to the recovery of all sufficiently sparse vectors. Some of the more prominent papers are [3–7]. In particular, it is shown in [3] that if the measurement matrix A consists of mn samples of a zero-mean, unit-variance Gaussian or sub-Gaussian random variable, normalized by $1/\sqrt{m}$, then basis pursuit leads to the recovery of sparse vectors with high probability (with respect to the process of generating A). In this approach, the number of measurements m that is required is $O(k \log(n/k))$. Subsequent research showed that *any* algorithm requires $\Omega(k \log(n/k))$ measurements. See [8] for an early result, and [9] for a simpler and more explicit version of this bound. Thus, during the early days of compressed sensing theory, random Gaussian matrices were considered to be "order optimal" in the sense that the number of measurements is within a fixed universal constant of the minimum required.

In recent times, there has been a lot of interest in the use of *sparse binary* measurement matrices for compressed sensing. One of the main advantages of this approach is that it allows one to connect compressed sensing to fields such as graph theory and algebraic coding theory. Random matrices are dense, and each element needs to be stored to high precision. In contrast, sparse binary matrices require less storage both because they are sparse, and also because every nonzero element equals one. For this reason, binary matrices are also said to be "multiplication-free." As a result, popular compressed sensing approaches such as (1) can be applied effectively for far larger matrices, and with greatly reduced CPU time, when A is a sparse binary matrix instead of a random Gaussian matrix.

Previously, the best available bounds for the number of measurements required by a binary matrix are $m = O(\max\{k^2, \sqrt{n}\})$. This bound is improved here to $k\sqrt{n}$. Note that there is no O symbol in the formula. This contrasts with $m = O(k \log(n/k))$ for random Gaussian matrices. However, in the latter case, the O symbol hides a very large constant. It is shown here that for values of $n < 10^5$ of thereabouts, the *known* bounds with binary matrices are in fact *smaller* than with random Gaussian matrices.

The preceding discussion refers to the case where a particular matrix A is "guaranteed" to recover *all* sufficiently sparse vectors. A parallel approach is to study conditions under which "most" sparse vectors are recovered. Specifically, in this approach, n, m are fixed, and k is varied from 1 to m. For each choice of k, a large number of vectors with exactly k nonzero components are generated at random, and the fraction that is recovered accurately is computed. Clearly, as k is increased, this fraction decreases. But the phenomenon of interest is known as "phase transition." One might expect that the fraction of recovered randomly generated vectors equals 1 when k is sufficiently small, and decreases gradually to 0 as k approaches m. In reality, there is a *sharp boundary* below which almost all k-sparse vectors are recovered, and above which almost no k-sparse vectors are recovered. This has been established theoretically for the case where A consists of random Gaussian samples in [6, 10–12]. A very general theory is derived in [13], where the measurement matrix still consists of random Gaussians, but the objective function is changed from the ℓ_1-norm to an arbitrary convex function. In a recent paper [14], phase transitions are studied *empirically* for several classes of *deterministic* measurement matrices, and it is verified that there is essentially no difference between the phase transitions with random Gaussian matrices.

Now we describe the organization of the chapter as well as its contributions. Part I of the chapter, specifically Sects. 2–5, contains background material, but also includes some improvements over known results. Specifically, Sect. 2 gives a precise definition of compressed sensing. Sections 3 and 7 discuss two of the most popular sufficient conditions for achieving compressed sensing, namely, the restricted isometry property (RIP) and the robust null space property (RNSP), respectively. The relationship between the two is discussed in Sect. 5. Then, we review the literature on the construction of binary matrices for compressed sensing in Sect. 6.

Part II of the chapter presents some original contributions on the construction of binary measurement matrices where the number of measurements is "nearly optimal." In Sect. 7, we derive a sufficient condition for a binary matrix to satisfy the RNSP. This condition improves the best-known bounds by a factor of roughly $3\sqrt{3}/2 \approx 2.6$. In Sect. 8 we derive a universal lower bound on the number m of measurements that is needed to satisfy the sufficient condition derived in Sect. 7.

It is shown that the number of measurements is minimized when the bipartite graph associated with the measurement matrix has girth 6. In Sect. 9, we present a class of binary matrices that have girth 6, which includes as special cases (i) a construction from low-density parity-check (LDPC) coding theory known as array codes and (ii) another construction based on Euler squares, the matrices in this class come close to meeting the lower bound on the number of measurements derived in Sect. 8. This is the justification for the phrase "nearly optimal" in the title of the chapter.

In Part III of the chapter, we present a new algorithm for compressed sensing that is *noniterative*, i.e., does not involve any optimization. Consequently, it is hundreds of times faster than basis pursuit. The new algorithm is based on the theory of expander graphs, and is able to accommodate exactly sparse as well as nearly sparse vectors, and also "shot noise," that is, noise with bounded support.

Finally, in Part IV of the chapter, we focus on "statistical" recovery, that is, the recovery of "most" sparse vectors as opposed to all sparse vectors. In this case, the number of measurements can be reduced substantially. The currently best-known results from the literature are presented in Sect. 16. In Sect. 17, we discuss the phase transition behavior of the basis pursuit formulation when this class of binary matrices is used. In Sect. 18, we present some numerical examples. The last section containing numerical examples is Sect. 19, where the performance of the noniterative algorithm is illustrated.

On the basis of these examples, it is possible to conclude that: (i) there is no discernible difference between the phase transition behavior with random Gaussian matrices compared to the binary matrices proposed in [15], and the class of matrices proposed here. On the other hand, the time of execution using our class of binary matrices is 1,000 times faster, if not more, compared to random Gaussian matrices. On the basis of the material presented here, we believe that the class of binary matrices proposed here are a viable alternative to, and possibly a replacement for, random Gaussian measurement matrices.

Part I: Preliminaries

2 Definition of Compressed Sensing

Let $\Sigma_k \subseteq \mathbb{R}^n$ denote the set of k-sparse vectors in $\mathbb{R}^n$, that is

$$\Sigma_k := \{x \in \mathbb{R}^n : \|x\|_0 \leq k\},$$

where, as is customary, $\|\cdot\|_0$ denotes the number of nonzero components of x. Given a norm $\|\cdot\|$ on $\mathbb{R}^n$, the k**-sparsity index** of x with respect to that norm is defined by

$$\sigma_k(x, \|\cdot\|) := \min_{z \in \Sigma_k} \|x - z\|.$$

Now, we are in a position to define the compressed sensing problem precisely. Note that $A \in \mathbb{R}^{m \times n}$ is called the measurement matrix and $\Delta : \mathbb{R}^m \to \mathbb{R}^n$ is called the "decoder map."

Definition 1 The pair (A, Δ) is said to achieve **exact sparse recovery of order** k if

$$\Delta(Ax) = x, \ \forall x \in \Sigma_k. \tag{2}$$

The pair (A, Δ) is said to achieve **stable sparse recovery of order** k and indices p, q if there exists a constant C such that

$$\|\Delta(Ax) - x\|_p \le C\sigma_k(x, \|\cdot\|_q), \ \forall x \in \mathbb{R}^n. \tag{3}$$

The pair (A, Δ) is said to achieve **robust sparse recovery of order** k and indices p, q (and norm $\|\cdot\|$) if there exist constants C and D such that, for all $\eta \in \mathbb{R}^m$ with $\|\eta\| \le \varepsilon$, it is the case that

$$\|\Delta(Ax + \eta) - x\|_p \le C\sigma_k(x, \|\cdot\|_q) + D\varepsilon, \ \forall x \in \mathbb{C}^n. \tag{4}$$

It is obvious that robust sparse recovery implies stable sparse recovery, which in turn implies exact sparse recovery. The above definitions apply to general norms. In this chapter, and indeed in much of the compressed sensing literature, the emphasis is on the case where $q = 1$ and $p \in [1, 2]$. However, the norm on η is still arbitrary.

3 Approaches to Compressed Sensing—I: RIP

Next, we present some sufficient conditions for basis pursuit as defined in (1) to achieve robust or stable sparse recovery. There are two widely used sufficient conditions, namely, the restricted isometry property (RIP) and the stable or robust null space property (SNSP or RNSP). We begin by discussing the RIP.

Definition 2 A matrix $A \in \mathbb{R}^{m \times n}$ is said to satisfy the **restricted isometry property (RIP)** of order k with constant δ if

$$(1 - \delta)\|u\|_2^2 \le \|Au\|_2^2 \le (1 + \delta)\|u\|_2^2, \ \forall u \in \Sigma_k. \tag{5}$$

The RIP is formulated in [3]. In that same paper, another constant called the "restricted orthogonality constant" is also introduced, but it is no longer used. Also, some authors define the RIP constant to be *the smallest* constant such that (5) holds. However, it is now known that determining *the smallest* constant such that (5) holds is NP-hard. Thus, the emphasis in contemporary theory is on the following question: Given integers n, k, and a constant δ, can we determine an integer m and a matrix $A \in \mathbb{R}^{m \times n}$ such that (5) holds?

It is shown in a series of papers [3, 4, 7] that the RIP of A is sufficient for (A, Δ_{BP}) to achieve robust sparse recovery. Now, we present the best known, and indeed the "best possible," result relating RIP and robust recovery.

Theorem 1 *If A satisfies the RIP of order tk with constant $\delta_{tk} < \sqrt{(t-1)/t}$ for some $t \geq 4/3$, or $\delta_{tk} < t/(4-t)$ for $t \in (0, 4/3)$, then (A, Δ_{BP}) achieves robust sparse recovery. Moreover, both bounds are tight.*

The first bound is proved in [16] while the second bound is proved in [17]. Note that both bounds are equal when $t = 4/3$. Hence, the theorem provides a continuous tight bound on δ_{tk} for all $t > 0$.

This theorem raises the question as to how one may go about designing measurement matrices that satisfy the RIP. There are two popular approaches, one probabilistic and one deterministic. In the probabilistic method, the measurement matrix A equals $(1/\sqrt{m})\Phi$ where Φ consists of mn independent samples of a Gaussian variable, or more generally, a sub-Gaussian random variable. In this chapter, we restrict our attention to the case where A consists of Gaussian samples, and refer the reader to [18] for the more general case of sub-Gaussian samples. The relevant bound on m to ensure that A satisfies the RIP with high probability is given next; it is a fairly straightforward modification of [18, Theorem 9.27].

Theorem 2 *Suppose an integer k and real numbers $\delta, \xi \in (0, 1)$ are specified, and that $A = (1/\sqrt{m})\Phi$, where $\Phi \in \mathbb{R}^{m \times n}$ consists of independent samples of a normal Gaussian random variable X. Define*

$$g = 1 + \frac{1}{\sqrt{2 \ln(en/k)}}, \eta = \frac{\sqrt{1+\delta} - 1}{g}. \tag{6}$$

Then, A satisfies the RIP of order k with constant δ with probability $\geq 1 - \xi$ provided

$$m \geq \frac{2}{\eta^2} \left(k \ln \frac{en}{k} + \ln \frac{2}{\xi} \right). \tag{7}$$

Proof The proof of this theorem is given in very sketchy form, as it follows that of [18, Theorem 9.27]. In that theorem, it is shown that, if the measurement matrix $A \in \mathbb{R}^{m \times n}$ consists of independent samples of Gaussian random variables, and if

$$m \geq \frac{2}{\eta^2} \left(k \log \frac{en}{k} + \ln \frac{2}{\xi} \right),$$

where η satisfies

$$\delta \leq 2g\eta + g^2\eta^2,$$

then A satisfies the RIP of order k with constant δ, with probability $\geq 1 - \xi$. Now, the above equation can be rewritten as

$$\delta + 1 \leq 1 + 2g\eta + g^2\eta^2 = (1 + g\eta)^2.$$

Rearranging this equation leads to (6).

Equation (7) leads to an upper bound of the form $m = O(k \log(n/k))$ for the number of measurements that suffice for the random matrix to satisfy the RIP with high probability. It is shown in [9, Theorem 3.1] that *any* algorithm that achieves stable sparse recovery requires $m = O(k \log(n/k))$ measurements. See [8, Theorem 5.1] for an earlier version. For the convenience of the reader, we restate the latter theorem. Note that it is assumed in [9] that $p = q = 1$, but the proof requires only that $p = q$. In order to state the theorem, we introduce the entropy with respect to an arbitrary integer θ. Suppose $\theta \geq 2$ is an integer. Then, the **θ-ary entropy** $H_\theta : (0, 1) \to (0, 1]$ is defined by

$$H_\theta(u) := -u \log_\theta \frac{u}{\theta - 1} - (1 - u) \log_\theta (1 - u). \tag{8}$$

If $\theta = 2$, this is just the usual Shannon entropy (to the base 2) of a binary random variable assuming the values 0 and 1 with probabilities u and $1 - u$, respectively. Elementary calculus shows that the θ-ary entropy assumes its maximum value when $u = (\theta - 1)/\theta =: u^*$, and that $H(u^*) = 1$. Moreover, $H_\theta(\cdot)$ is monotonic on either side of u^*.

Theorem 3 *Suppose $A \in \mathbb{R}^{m \times n}$ and that, for some map $\Delta : \mathbb{R}^m \to \mathbb{R}^n$, the pair (A, Δ) achieves stable k-sparse recovery with constant C. Define $\theta = \lfloor n/k \rfloor$. Then,*[1]

$$\begin{aligned} m &\geq \frac{1 - H_\theta(1/2)}{\log(4 + 2C)} k \log \theta & (9) \\ &= \frac{1}{\log(4 + 2C)} \frac{\log(\theta / 2\sqrt{\theta - 1})}{\log \theta} k \log \theta & (10) \\ &\approx \frac{1}{2(\log(4 + 2C))} k \log \left\lfloor \frac{n}{k} \right\rfloor \ \textit{if}\ n \gg k. & (11) \end{aligned}$$

Because robust k-sparse recovery implies stable k-sparse recovery, the bound in (11) applies also to robust k-sparse recovery. Note that the expression (9) is the one presented in [9, Theorem 3.1], but the second and third expressions follow readily.

Comparing Theorems 2 and 3 shows that $m = O(k \log(n/k))$ measurements are both necessary and sufficient for robust k-sparse recovery. For this reason, the probabilistically generated measurement matrices are considered to be "order optimal." However, this statement is misleading because the O symbol in the upper bound hides a very large constant, as shown next.

[1] Note that the base of the logarithm does not matter because it cancels out between the two log terms.

Example 1 Suppose $n = 22,201 = 149^2$ and $k = 69$, which is a problem instance studied later in Sect. 18. Then, the upper and lower bounds from Theorems 2 and 3 imply that

$$14 \leq m \leq 44,345.$$

Thus, the spread between the upper and lower bounds is more than three orders of magnitude. Also, the upper bound for the number of measurements is *more* than the dimension n.

There is another factor as well. As can be seen from Theorem 2, probabilistic methods lead to measurement matrices that satisfy the RIP *only with high probability*, that can be made close to one but never exactly equal to one. Moreover, as shown in [19], once a matrix has been generated, it is NP-hard to test whether *that particular matrix* satisfies the RIP.

These observations have led the research community to explore deterministic methods to construct matrices that satisfy the RIP. A popular approach is based on coherence of a matrix.

Definition 3 Suppose $A \in \mathbb{R}^{m \times n}$ is column normalized, so that $\|a_j\|_2 = 1$ for all $j \in [n]$, where a_j denotes the j-column of A. Then, the **coherence** of A is denoted by $\mu(A)$ and is defined as

$$\mu(A) := \max_{i \neq j} |\langle a_i, a_j \rangle|. \tag{12}$$

The following result is an easy consequence of the Gershgorin circle theorem.

Lemma 1 *A matrix $A \in \mathbb{R}^{m \times n}$ satisfies the RIP of order k with constant*

$$\delta_k = (k-1)\mu, \tag{13}$$

provided that $(k-1)\mu < 1$, or equivalently, $k < 1 + 1/\mu$.

4 Approaches to Compressed Sensing—II: RNSP

An alternative to the RIP approach to compressed sensing is provided by the stable (and robust) null space property. The SNSP is formulated in [20], while, to the best of the authors' knowledge, the RNSP is formulated for the first time in [21]; see also [18, Definition 4.17].

Definition 4 Suppose $A \in \mathbb{R}^{m \times n}$ and let $\mathcal{N}(A)$ denote the null space of A. Then, A is said to satisfy the **stable null space property (SNSP)** of order k with constant $\rho < 1$ if, for every set $S \subseteq [n]$ with $|S| \leq k$, we have that

$$\|v_S\|_1 \leq \rho \|v_{S^c}\|_1, \ \forall v \in \mathcal{N}(A). \tag{14}$$

The matrix A is said to satisfy the **robust null space property (RNSP)** of order k for the norm $\| \cdot \|$ with constants $\rho < 1$ and $\tau > 0$ if, for every set $S \subseteq [n]$ with $|S| \leq k$, we have that

$$\|h_S\|_1 \leq \rho \|h_{S^c}\|_1 + \tau \|Ah\|, \ \forall h \in \mathbb{R}^n. \tag{15}$$

It is obvious that RNSP implies the SNSP.

The utility of these definitions is brought out in these theorems.

Theorem 4 (See [18, Theorem 4.12]) *Suppose A satisfies the stable null space property of order k with constant ρ. Then, the pair (A, Δ_{BP}) achieves stable k-sparse recovery with*

$$C = 2\frac{1+\rho}{1-\rho}. \tag{16}$$

Theorem 5 (See [18, Theorem 4.22]) *Suppose A satisfies the robust null space property of order k for the norm $\| \cdot \|$ with constants ρ and τ. Then, the pair (A, Δ_{BP}) achieves robust k-sparse recovery with*

$$C = 2\frac{1+\rho}{1-\rho}, D = \frac{4\tau}{1-\rho}. \tag{17}$$

5 Relationship Between RIP and RNSP

Until recently, the twin approaches of RIP and RNSP had proceeded along parallel tracks. However, it is shown in [22, Theorem 9] that if A satisfies the RIP of order tk with constant $\delta_{tk} < \sqrt{(t-1)/t}$ for some $t > 1$, then it satisfies the RSNP of order k. The specific result is the following:

Theorem 6 *Given integers k, n, and a real number $t > 1$, suppose that the matrix A satisfies the RIP of order tk with constant $\delta_{tk} = \delta < \sqrt{(t-1)/t}$. Define*

$$\nu := \sqrt{t(t-1)} - (t-1). \tag{18}$$

Then, A satisfies the RNSP with constants

$$\rho = c/a < 1, \tau = b\sqrt{k}/a^2, \tag{19}$$

where

$$\begin{aligned} a &:= [\nu(1-\nu) - \delta(0.5 - \nu + \nu^2)]^{1/2} \\ &= \frac{[(1-\delta) - (1+\delta)(1-2\nu)^2]^{1/2}}{2}, \end{aligned} \tag{20}$$

$$b := \nu(1-\nu)\sqrt{1+\delta}, \tag{21}$$

$$c := \left[\frac{\delta \nu^2}{2(t-1)} \right]^{1/2}. \tag{22}$$

As stated in Theorem 1, $\delta_{tk} < \sqrt{(t-1)/t}$ is the weakest sufficient condition in terms of RIP for robust sparse recovery, whenever $t > 4/3$. Taken in conjunction with Theorem 6, it follows that it is *not* possible to obtain weaker sufficient conditions using the RIP approach than using the RNSP approach.

Note that if A has coherence μ, then by Lemma 1, we have that $\delta_{tk} \leq (tk-1)\mu$ for all t. Next, by Theorem 6, basis pursuit achieves robust k-sparse recovery, whenever

$$(tk-1)\mu < \sqrt{\frac{t-1}{t}} \tag{23}$$

for *any* $t > 1$. So, let us ask: What is an "optimal" choice of t? To answer this question, we neglect the 1 in comparison to tk, and rewrite the above inequality as

$$k\mu < \sqrt{\frac{t-1}{t^3}}.$$

Thus, we get the best bound by maximizing the right side with respect to t. It is an easy exercise in calculus to show that the maximum is achieved with $t = 1.5$, and the corresponding bound $\sqrt{(t-1)/t} = 1/\sqrt{3}$. Hence, by combining with Lemma 1, we can derive the following bound.

Theorem 7 *Suppose $A \in \mathbb{R}^{m \times n}$ has coherence μ. Then, (A, Δ_{BP}) achieves robust k-sparse recovery, whenever*

$$(1.5k-1)\mu < 1/\sqrt{3}, \tag{24}$$

or equivalently

$$k < \left\lfloor \frac{2}{3\sqrt{3}\mu} + \frac{2}{3} \right\rfloor. \tag{25}$$

Moreover, the bound is nearly optimal when applying Theorem 6.

If we retain the term $tk - 1$ instead of replacing it by tk, we would get a more complicated expression for the optimal value of t. However, it can be verified that if (24) is satisfied, then so is (23).

6 Binary Matrices for Compressed Sensing: A Review

In this section, we present a brief review of the use of binary matrices as measurement matrices in compressed sensing. The first construction of a binary matrix that satisfies the RIP is due to DeVore and is given in [15]. The DeVore matrix has dimensions

$q^2 \times q^{r+1}$, where q is a power of a prime number, and $r \geq 2$ is an integer, has exactly q elements of 1 in each column, and has coherence $\mu \leq r/q$. This construction is generalized to algebraic curves in [23], but does not seem to offer much of an advantage over that in [15]. A construction that leads to matrices of order $2^m \times 2^{m(m+1)/2}$ based on Reed–Muller codes is proposed in [24]. Because the number of measurements is restricted to be a power of 2, this is not a very practical method. A construction in [25] is based on a method to generate Euler squares from nearly a century ago [26]. The resulting binary matrix has dimensions $lq \times q^2$, where q is an *arbitrary* integer, making this perhaps the most versatile construction. The integer l is bounded as follows: Let $q = 2^{r_0} p_1^{r_1} \ldots p_s^{r_s}$ be the prime number decomposition of q. Then $l + 1 \leq \min\{2^{r_0}, p_1^{r_1}, \ldots, p_s^{r_s}\}$. In particular, if q is itself a power of a prime, we can have $l = q - 1$. Each column of the resulting binary matrix has exactly l ones and the matrix has coherence $1/l$. All of these matrices can be used to achieve robust k-sparse recovery via the basis pursuit formulation, by combining Lemma 1 with Theorem 1. Another method found in [27] constructs binary matrices using the Chinese remainder theorem, and achieves *probabilistic* recovery.

There is another property that is sometimes referred to as the ℓ_1-RIP, introduced in [28–30], which makes a connection between expander graphs and compressed sensing. However, while this approach readily leads to stable k-sparse recovery, it does not lend itself readily to *robust* k-sparse recovery. One of the main contributions of [31] is to show that the construction of [15] can also be viewed as a special case of an expander graph construction proposed in [32].

Yet another direction is initiated in [33], in which a general approach is presented for generating binary matrices for compressed sensing using algebraic coding theory. In particular, it is shown that binary matrices which, when viewed as elements over the binary field $\mathbb{F}_2$, have good properties in decoding, will also be good measurement matrices when viewed as matrices of real numbers. In particular, several notions of "pseudoweights" are introduced, and it is shown that these pseudoweights can be related to the satisfaction of the stable (but not robust) null space property of binary matrices. These bounds are improved in [34] to prove the stable null space property under weaker conditions than in [33].

Part II: Binary Matrices Based on LDPC Codes

7 Robust Null Space Property of Binary Matrices

In this section, we commence presenting the new results of this chapter on identifying a class of binary matrices for compressed sensing that has a nearly optimal number of measurements. Note that the contents of this part of the chapter are taken from [35].

Suppose $A \in \{0, 1\}^{m \times n}$ with $m < n$. Then, A can be viewed as the biadjacency matrix of a bipartite graph with n input (or "left") nodes and m output (or "right") nodes. Such a graph is said to be **left regular** if each input node has the same degree, say d_l. This is equivalent to saying that each column of A contains exactly d_l ones. Given a bipartite graph with E edges, n input nodes and m output nodes, define the "average left degree" of the graph as $\bar{d_l} = E/n$ and $\bar{d_r} = E/m$. Note that these average degrees need not be integers. Then, it is clear that $n\bar{d_l} = m\bar{d_r}$. The **girth** of a graph is defined as the length of the shortest cycle. Note that the girth of a bipartite graph is always an even number, and in "simple" graphs (not more than one edge between any pair of vertices), the girth is at least four.

Hereafter, in this chapter, we will not make a distinction between a binary matrix, and the bipartite graph associated with the matrix. Specifically, the columns correspond to the "left" nodes while the rows correspond to the "right" nodes. So an expression such as "A is a left-regular binary matrix of degree d_l" means that the associated bipartite graph is left regular with degree d_l. This usage will permit us to avoid some tortuous sentences.

Theorems 8 and 9 are the starting point for the contents of this section.

Theorem 8 (See [34, Theorem 2]) *Suppose $A \in \{0, 1\}^{m\times n}$ is left regular with left degree d_l, and suppose that the maximum inner product between any two columns of A is λ. Then, for every $v \in \mathcal{N}(A)$, we have that*

$$|v_i| \leq \frac{\lambda}{2d_l}\|v\|_1, \ \forall i \in [n], \tag{26}$$

where $[n]$ denotes $\{1, \dots, n\}$.

If the matrix A has girth 6 or more, then the maximum inner product between any two columns of A is at most equal to one. In such a case, it is possible to improve the bound (26).

Theorem 9 (See [34, Theorem 3]) *Suppose $A \in \{0, 1\}^{m\times n}$ and that A has girth $g \geq 6$. Then, for every $v \in \mathcal{N}(A)$, we have that*

$$|v_i| \leq \frac{\|v\|_1}{C'}, \ \forall i \in [n], \tag{27}$$

where, if $g = 4t + 2$, then

$$C' := 2\sum_{i=0}^{t}(d_l - 1)^i, \tag{28}$$

and if $g = 4t$, then

$$C' := 2\sum_{i=0}^{t-1}(d_l - 1)^i, \tag{29}$$

Note that Theorem 9 is an improvement over Theorem 8 only when the girth of the graph is ≥ 10. If the girth equals 6, then C' as defined in (28) becomes $C' = 2$, and the bound in (27) becomes the same as that in (26) after noting that $\lambda = 1$. Similarly, if $g = 8$, then $\mathscr{C}'$ in (29) also becomes just $C' = 2$.

In [34], the bounds (26) and (27) are used to derive sufficient conditions for the matrix A to satisfy the *stable* null space property. However, it is now shown that the same two bounds can be used to infer the *robust* null space property of A. This is a substantial improvement, because with such an A matrix, basis pursuit would lead to *robustness against measurement noise*, which is not guaranteed with the SNSP. We derive our results through a series of preliminary results.

Lemma 2 *Suppose $A \in \mathbb{R}^{m \times n}$, and let $\| \cdot \|$ be any norm on $\mathbb{R}^m$. Suppose there exist constants $\alpha > 2$, $\beta > 0$ such that*

$$|h_i| \leq \frac{\|h\|_1}{\alpha} + \beta \|Ah\|, \ \forall i \in [n], \ \forall h \in \mathbb{R}^n. \tag{30}$$

Then, for all $k < \alpha/2$, the matrix A satisfies the RNSP of order k. Specifically, whenever $S \subseteq [n]$ with $|S| \leq k$, (15) *holds with*

$$\rho = \frac{k}{\alpha - k}, \tau = \frac{\alpha k \beta}{\alpha - k}. \tag{31}$$

Proof Let $S \subseteq [n]$ with $|S| \leq k$ be arbitrary. Then,

$$\begin{aligned} \|h_S\|_1 &= \sum_{i \in S} |h_i| \\ &\leq \frac{k}{\alpha} \|h\|_1 + k\beta \|Ah\| \\ &= \frac{k}{\alpha} (\|h_S\|_1 + \|h_{S^c}\|_1) + k\beta \|Ah\|. \end{aligned}$$

Therefore,

$$\left(1 - \frac{k}{\alpha} \right) \|h_S\|_1 \leq \frac{k}{\alpha} \|h_{S^c}\|_1 + k\beta \|Ah\|,$$

or

$$\|h_S\|_1 \leq \frac{k}{\alpha - k} \|h_{S^c}\|_1 + \frac{\alpha k \beta}{\alpha - k} \|Ah\|,$$

which is the desired conclusion.

Next, let $A \in \mathbb{R}^{m \times n}$ be arbitrary and let $\| \cdot \|$ be any norm on $\mathbb{R}^n$. Let $\mathcal{N}(A) \subseteq \mathbb{R}^n$ denote the null space of A, and let $\mathcal{N}^\perp := [\mathcal{N}(A)]^\perp$ denote the orthogonal complement of $\mathcal{N}(A)$ in $\mathbb{R}^n$. Then, for all $u \in \mathcal{N}^\perp$, it is easy to see that

$$\|u\|_2 \leq \frac{1}{\sigma_{\min}} \|Au\|_2,$$

where $\sigma_{\min}$ is the smallest nonzero singular value of A. Because all norms on a finite-dimensional space are equivalent, there exists a constant c that depends only on the norm $\| \cdot \|$ on $\mathbb{R}^m$ such that

$$\|y\|_2 \le c\|y\|, \ \forall y \in \mathbb{R}^m. \tag{32}$$

In particular, $\|y\|_2 \le \|y\|_1$, so we can take $c = 1$ in this case. Therefore, by Schwarz' inequality, we get

$$\|u\|_1 \le \sqrt{n}\|u\|_2 \le \frac{c\sqrt{n}}{\sigma_{\min}}\|Au\|, \ \forall u \in \mathcal{N}^\perp. \tag{33}$$

Now, we can state the main result of this section.

Theorem 10 *Suppose $A \in \{0, 1\}^{m\times n}$ is left regular with left degree d_l, and let λ denote the maximum inner product between any two columns of A (and observe that $\lambda \le d_l$). Next, let $\sigma_{\min}$ denote the smallest nonzero singular value of A, and for an arbitrary norm $\| \cdot \|$ on $\mathbb{R}^m$, choose the constant c such that (32) holds. Then, A satisfies (30) with*

$$\alpha = \frac{2d_l}{\lambda}, \beta = \left(\frac{\lambda}{2d_l} + 1\right)\frac{c\sqrt{n}}{\sigma_{\min}}. \tag{34}$$

Consequently, for all $k < \alpha/2 = d_l/\lambda$, A satisfies the RNSP of order k with

$$\rho = \frac{\lambda k}{2d_l - \lambda k}, \tau = \frac{2d_l k}{2d_l - \lambda k}\beta. \tag{35}$$

Proof Let $h \in \mathbb{R}^n$ be arbitrary, and express h as $h = v + u$, where $v \in \mathcal{N}(A)$ and $u \in \mathcal{N}^\perp$. Then, clearly

$$|h_i| = |v_i + u_i| \le |v_i| + |u_i|, \ \forall i \in [n].$$

We will bound each term separately.

As shown in Theorem 8, we have that

$$\begin{aligned} |v_i| &\le \frac{\lambda}{2d_l}\|v\|_1 \\ &\le \frac{\lambda}{2d_l}(\|h\|_1 + \|u\|_1) \\ &\le \frac{\lambda}{2d_l}\|h\|_1 + \frac{\lambda c\sqrt{n}}{2d_l\sigma_{\min}}\|Au\| \\ &= \frac{\lambda}{2d_l}\|h\|_1 + \frac{\lambda c\sqrt{n}}{2d_l\sigma_{\min}}\|Ah\|, \end{aligned}$$

where the last step follows from the fact that $Ah = Au$ because $Av = \mathbf{0}$. Next,

$$|u_i| \leq \|u\|_1 \leq \frac{c\sqrt{n}}{\sigma_{\min}}\|Ah\|, \ \forall i \in [n].$$

Combining these two inequalities shows that

$$|h_i| \leq |v_i| + |u_i| \leq \frac{\lambda}{2d_l}\|h\|_1 + \left(\frac{\lambda}{2d_l} + 1\right)\frac{c\sqrt{n}}{\sigma_{\min}}\|Ah\|.$$

This establishes (34). Now (35) follows from Lemma 2, specifically (31).

Theorem 11 *Suppose $A \in \{0,1\}^{m\times n}$ is left regular with left degree d_l, and has girth of at least six. Define the constant C' as in (28) or (29) as appropriate. Then, for all $k < C'/2$, the matrix A satisfies the RNSP of order k, with constants*

$$\rho = \frac{k}{C' - k}, \tau = \frac{C'k}{C' - k}\beta. \tag{36}$$

The proof of Theorem 11 is entirely analogous to that of Theorem 10, with the bound in Theorem 9 replacing that in Theorem 8. Therefore, the proof is omitted.

The results in Theorem 10 lead to sharper bounds for the sparsity count compared to using RIP and coherence bounds. This is illustrated next.

Example 2 Suppose $A \in \{0,1\}^{m\times n}$ is left regular with degree d_l and with the inner product between any two columns bounded by λ. Then, it is easy to see that the coherence μ of A is bounded by λ/d_l. Therefore, if we use Theorem 7, then it follows that $(A, \Delta_{\mathrm{BP}})$ achieves robust k-sparse recovery, whenever

$$k < \left\lfloor \frac{2d_l}{3\sqrt{3}\lambda} + \frac{2}{3} \right\rfloor.$$

In contrast, if we use Theorem 10, it follows that $(A, \Delta_{\mathrm{BP}})$ achieves robust sparse recovery, whenever $k < d_l/\lambda$, which is an improvement by a factor of roughly $3\sqrt{3}/2 \approx 2.6$.

8 Lower Bounds on the Number of Measurements

Theorem 9 shows that, for a fixed left degree d_l, as the girth of the graph corresponding to A becomes larger, so does the constant C'. Therefore, as the girth of A increases, so does the upper bound on k as obtained from Theorem 11. This suggests that, for a given left degree d_l and number of input nodes n, it is better to choose graphs of large girth. However, as shown next, as the girth of a graph is increased, the number of measurements m also increases. As shown below, the "optimal" choice for the girth is actually six.

To establish this statement, let us define

$$\bar{k} := \begin{cases} (d_l - 1)^t & \text{if } g = 4t + 2, \\ (d_l - 1)^{t-1} & \text{if } g = 4t. \end{cases} \tag{37}$$

It is recognized that $\bar{k}$ is just the last term in the summation in (28) and (29). Now, if the actual sparsity count k satisfies $k \leq \bar{k}$, then it follows from Theorem 9 that the pair (A, Δ_{BP}) achieves robust k-sparse recovery. As stated before, if we choose the matrix A to have higher and higher girth, the bound $\bar{k}$ also becomes higher. So, the question, therefore, becomes: What happens to m, the number of measurements, as the girth is increased? The answer is given next.

Theorem 12 *Suppose $A \in \{0, 1\}^{m \times n}$ is d_l-left regular graph with $m \leq n$, and that every row and every column of A contain at least two ones. If the girth g of A equals $4t + 2$, then*

$$m \geq \bar{k}^{2/(t+1)} n^{t/(t+1)}, \tag{38}$$

whereas if $g = 4t$ for $t \geq 2$, then

$$m \geq \bar{k}^{(2t-1)/[t(t-1)]} n^{(t-1)/t}. \tag{39}$$

The proof of Theorem 12 is based on the following result [36, Equations (1) and (2)]:

Theorem 13 *Suppose $A \in \{0, 1\}^{m \times n}$ with $m < n$. Suppose, further, that in the bipartite graph associated with A, every node has degree ≥ 2.[2] Let E denote the total number of edges of the graph, and define $\bar{d}_l = E/n$, $\bar{d}_r = E/m$ to be the average left-node degree and average right-node degree, respectively. Suppose, finally, that the graph has girth $g = 2r$. Then,*

$$m \geq \sum_{i=0}^{r-1} (\bar{d}_l - 1)^{\lceil i/2 \rceil} (\bar{d}_r - 1)^{\lfloor i/2 \rfloor}. \tag{40}$$

It is important to note that the above theorem does not require any assumptions about the underlying graph (e.g., regularity). The only assumption is that every node has degree two or more, so as to rule out trivial cases. Usually, such theorems are used to find upper bounds on the girth of a bipartite graph in terms of the numbers of its nodes and edges (as in Theorem 14 below). However, we turn it around here and use the theorem to find a lower bound on m, given the integers n and g.

Note that if $g = 4$, then $r = 2$ and the bound (40) becomes $m \geq \bar{d}_l$, which is trivial. In fact, m has to exceed the *maximum* degree of any left node. However, for $g \geq 6$, the bound in (40) is meaningful.

[2] This is equivalent to the requirement that every row and every column of A contains at least two ones.

Proof (*Of Theorem* 12) The bound (40) implies that m is no smaller than the last term in the summation, that is

$$m \geq \bar{d}_l^{\lceil (r-1)/2 \rceil} \bar{d}_r^{\lfloor (r-1)/2 \rfloor}. \tag{41}$$

Because A is assumed to be left regular, actually $\bar{d}_l = d_l$, but we do not make use of this, and will carry the symbol $\bar{d}_l$ throughout. By definition, we have that $\bar{d}_r = (n\bar{d}_l)/m$. Therefore, if $n \geq m$, then it follows that

$$\bar{d}_r - 1 = \frac{n\bar{d}_l}{m} - 1 \geq \frac{n\bar{d}_l}{m} - \frac{n}{m} = \frac{n}{m}(\bar{d}_l - 1).$$

Therefore, (41) implies that

$$m \geq (\bar{d}_l - 1)^{\alpha} \left(\frac{n}{m} \right)^{\lfloor (r-1)/2 \rfloor}, \tag{42}$$

where

$$\alpha = \lceil (r-1)/2 \rceil + \lfloor (r-1)/2 \rfloor.$$

Now, we treat the cases $g = 4t + 2$ and $g = 4t$ separately. If $g = 4t + 2$, then $r = g/2 = 2t + 1$, so that

$$\lceil (r-1)/2 \rceil = \lfloor (r-1)/2 \rfloor = t, \alpha = 2t.$$

Therefore, (42) becomes

$$m \geq (\bar{d}_l - 1)^{2t} \left(\frac{n}{m} \right)^t = \bar{k}^2 \left(\frac{n}{m} \right)^t.$$

This can be rearranged as

$$m^{t+1} \geq n^t \bar{k}^2,$$

or

$$m \geq \bar{k}^{2/(t+1)} n^{t/(t+1)},$$

which is (38). In case $g = 4t$, the proof proceeds along entirely parallel lines and is omitted.

It is obvious from (38) that the lower bound is minimized (for a fixed choice of n and $\bar{k}$) with $t = 1$ or $g = 6$. Similarly, the lower bound in (39) is minimized when $t = 2$ or $g = 8$. Higher values of g would lead to more measurements being required. We can also compare $g = 6$ with $g = 8$ and show that $g = 6$ is better. Let us substitute $t = 1$ in (38) and $t = 2$ in (39). This gives

$$m \geq \begin{cases} \bar{k} n^{1/2} & \text{if } g = 6, \\ \bar{k}^{3/2} n^{1/2} & \text{if } g = 8. \end{cases} \tag{43}$$

If we wish to have fewer measurements than the dimension of the unknown vector, we can set $m < n$. Substituting this requirement into (43) leads to

$$\bar{k} < n^{1/2} \text{ if } g = 6, \bar{k} < n^{1/3} \text{ if } g = 8.$$

Hence, graphs of girth 6 are preferable to graphs of girth 8, because the upper limit on the recoverable sparsity count $\bar{k}$ is higher with a graph of girth 6 than with a graph of girth 8.

9 Construction of Nearly Optimal Graphs of Girth 6

The discussion of the preceding section suggests that we must look for bipartite graphs of girth 6 where the integer m satisfies the bound (40) with the $\geq$ replaced by an equality, or at least, close to it. In this section, we prove a general result to the effect that a class of binary matrices has girth 6. Then, we give two specific constructions. The first of these is based on array codes which are a part of low-density parity-check (LDPC) codes, and the second is based on Euler squares. The first construction is easier to explain, but the second one gives far more flexibility in terms of the number of measurements.

Here is the general theorem.

Theorem 14 *Suppose $A \in \{0, 1\}^{lq \times q^2}$ for some integers $4 \leq l \leq q - 1$. Suppose, further, that*

1. *$\bar{d}_l \geq l$, where $\bar{d}_l$ is the average left degree of A.*
2. *The maximum inner product between any two columns of A is one.*
3. *Every row and every column of A have at least two ones.*

Then the girth of A is six.

Remark: Before proving the theorem, let us see how closely such a matrix satisfies the inequality (40). In the constructions below, we have that $\bar{d}_l = d_l = l$, $g = 6$, and $r = 3$. Therefore, the bound in (40) becomes

$$m \geq 1 + (l - 1) + (l - 1)(q - 1) = q(l - 1) + 1.$$

Since $m = lq$, we see that the actual value of m exceeds the lower bound for m by a factor of $l/(l - 1)$ (after neglecting the last term of -1 on the right side). Note that there is no guarantee that the lower bound in Theorem 10 is actually achievable. So, the class of matrices proposed above (if they could actually be constructed), can be said to be "near optimal." In applying this theorem, we would choose q such that $n \leq q^2$, and choose any desired $l \leq q - 1$. With such a measurement matrix, basis pursuit will achieve robust k-sparse recovery up to $k < l$, that is, $k < \sqrt{n}$, more or less.

Proof Let g denote the girth of A. Then, Condition (2) implies that $g \geq 6$. Condition (3) implies that the bound (40) applies with $m = lq$, $n = q^2$, and $n/m = q/l$. Let $g = 2r$, and define

$$\alpha = \lceil (r-1)/2 \rceil + \lfloor (r-1)/2 \rfloor, \beta = \lfloor (r-1)/2 \rfloor.$$

Then, the inequality (40) implies that

$$lq \geq (\bar{d}_l - 1)^{\alpha} (q/l)^{\beta} \geq (l-1)^{\alpha} (q/l)^{\beta}.$$

This can be rewritten as

$$(l-1)^{\alpha} \frac{q^{\beta-1}}{l^{\beta+1}} \leq 1. \tag{44}$$

Note that $g \geq 6$, so that $r \geq 3$, due to Condition (2). We study two cases separately.

Case (1): $g = 4t$ for some $t \geq 2$. In this case,

$$(r-1)/2 = t - 1/2, \lceil (r-1)/2 \rceil = t, \lfloor (r-1)/2 \rfloor = t - 1,$$

$$\alpha = 2t - 1, \beta = t - 1.$$

Therefore, (44) becomes

$$(l-1)^{2t-1} \frac{q^{t-2}}{l^t} \leq 1, \tag{45}$$

or

$$q^{t-2}(l-1)^{t-1} \leq \left(\frac{l}{l-1} \right)^t \leq 2^t,$$

because $l/(l-1) \leq 2$ for $l \geq 2$. Also

$$q^{t-2}(l-1)^{t-1} \geq q^{t-2}(l-1)^{t-2} = [q(l-1)]^{t-2}.$$

Combining these inequalities gives

$$[q(l-1)]^{t-2} \leq 2^t,$$

or

$$\left[\frac{q(l-1)}{2}\right]^{t-2} \leq 2^2 = 4. \tag{46}$$

It is now shown that (46) cannot hold if $t \geq 3$. If $t \geq 3$, then

$$\frac{q(l-1)}{2} \leq \left[\frac{q(l-1)}{2}\right]^{t-2} \leq 4,$$

or $q(l-1) \leq 8$. However, $q \geq 5$ and $l-1 \geq 3$, so this inequality cannot hold. Now, let us consider the possibility that $g = 8$, i.e., that $t = 2$. In this case, (45) becomes

$$(l-1)^3 \frac{1}{l^2} \leq 1, \text{ or } (l-1)^3 \leq l^2.$$

This inequality can hold only for $l = 1, 2, 3$ and not if $l \geq 4$. Hence, A cannot have girth $4t$ for any $t \geq 2$.

Case (2): $g = 4t + 2$ for some $t \geq 1$. In this case,

$$\lceil (r-1)/2 \rceil = \lfloor (r-1)/2 \rfloor = t, \alpha = 2t, \beta = t.$$

So, (44) becomes

$$(l-1)^{2t} \frac{q^{t-1}}{l^{t+1}} \leq 1. \tag{47}$$

As before, this can be rewritten as

$$q^{t-1}(l-1)^{t-1} \leq \left(\frac{l}{l-1}\right)^{t+1} \leq 2^{t+1},$$

or

$$\left[\frac{q(l-1)}{2}\right]^{t-1} \leq 2^2 = 4. \tag{48}$$

This inequality can hold if $t = 1$ because the left side equals 1. However, if $t > 1$, then (48) implies that

$$\frac{q(l-1)}{2} \leq \left[\frac{q(l-1)}{2}\right]^{t-1} \leq 4,$$

or $q(l-1) \leq 8$, which is impossible. Hence, (48) implies that $t = 1$, or that $g = 6$.

Now, we present two explicit constructions of binary matrices that satisfy the conditions of Theorem 14.

The first construction is taken from the theory of low-density parity-check (LDPC) codes, and is a generalization of [37]. This type of construction for low-density parity-check codes (LDPC) was first introduced in [38]. Let q be a prime number, and let $P \in \{0,1\}^{q \times q}$ be any "fixed-point free" permutation of $[q]$. In [37], P is taken as the shift permutation matrix defined by $P_{i,i-1} = 1$ and the rest zeros, where $i-1$ is interpreted modulo q. Then, $P^q = I$, the identity matrix. Now, let $l < q$ be any integer, and define the matrix $H(q,l) \in \{0,1\}^{lq \times q^2}$ as the block-partitioned matrix $[M_{ij}], i \in [l], j \in [q]$, where

$$M_{ij} = P^{(i-1)(j-1)}. \tag{49}$$

More elaborately, the matrix $H(q,l)$ is given by

$$H(q,l) = \begin{bmatrix} I & I & I & \ldots & I \\ I & P & P^2 & \ldots & P^{q-1} \\ I & P^2 & P^4 & \ldots & P^{2(q-1)} \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ I & P^{l-1} & P^{2(l-1)} & \ldots & P^{(l-1)(q-1)} \end{bmatrix}. \tag{50}$$

The matrix $H(q,l)$ is biregular, with left (column) degree l and right (row) degree q. It is rank deficient, having rank $(q-1)l+1$. In principle, we could drop the redundant rows, but that would destroy the left regularity of the matrix, thus rendering the theory in this chapter inapplicable. (However, the resulting matrix would still be right regular.) Moreover, due to the fixed-point-free nature of P, it follows that the inner product between any two columns of $H(q,l)$ is at most equal to one.

It is shown in [37, Proposition 1] that $H(q,l)$ has girth 6, but that follows from Theorem 14.

The second construction is based on Euler squares. In [26], a general recipe is given for constructing generalized Euler squares. This is used in [25] to construct an associated binary matrix of order $lq \times q^2$ where q is any arbitrary integer (in contrast with the construction of [37] which requires q to be a prime number), such that the maximum inner product between any two columns is at most equal to one. Again, by Theorem 14, such matrices have girth 6 and are thus nearly optimal for compressed sensing. The upper bound on l is defined as follows: Let $q = 2^{r_0} p_1^{r_1} \ldots p_s^{r_s}$ be the prime number decomposition of q. Then $l < \min\{2^{r_0}, p_1^{r_1}, \ldots, p_s^{r_s}\}$. In particular, if q is a prime or a power of a prime, then we can have $l < q-1$. It is easy to verify that if q is a prime, then the construction in [25] is the same as the array code construction of [37] with permuted columns. For the case, where q is a prime power, the construction is more elaborate and is not pursued further here.

Example 3 In this example, we compare the number of samples required when using the DeVore construction and a matrix that satisfies the hypotheses of Theorem 14, such as the array code matrix or the Euler square matrix. The conclusions are that: (i) When $k < \sqrt{n}/4$, the Devore construction requires fewer measurements than the array code, whereas when $\sqrt{n}/4 < k < \sqrt{n}$, the array code type of matrix requires fewer measurements. (ii) When $k > \sqrt{n}/2$, the DeVore construction requires more measurements than n, the dimension of the unknown vector, whereas the array code construction has $m < n$ whenever $k < \sqrt{n}$.

To see this, recall that the DeVore construction produces a matrix of dimensions $q^2 \times q^{r+1}$ with the maximum inner product between columns equal to r, and each column contains q ones. So if we choose $r = 2$, then λ in Theorem 11 equals 2, while $d_l = l$. Consequently the DeVore matrix satisfies the RNSP of order k whenever $k < q/2$, and the number of measurements m_D equals $q^2 = 4k^2$, Thus, $m_D < n$ requires that $4k^2 < n$, or $k < \sqrt{n}/2$. In contrast, a matrix of the type discussed in Theorem 14 has dimensions $lq \times q^2$ where $n = q^2$ and $l = k + 1$. For this class of matrices, we have $\lambda = 1$ and $d_l = l$. This matrix satisfies the RNSP whenever $k = l - 1 < q$, and the number of measurements equals $lq = (k + 1)q$. Now, $4k^2 < kq$ if and only if $k < q/4 = \sqrt{n}/4$. Also $m_A = (k + 1)q < n = q^2$ whenever $k + 1 < q = \sqrt{n}$. Here, in the interests of simplicity, we ignore the fact that q has to be a prime number in both cases, and various rounding up operations.

Part III: Applications of Graph Theory to Compressed Sensing

10 Compressive Sensing Using Expander Graphs

This part of the chapter is taken from [31].

A recent development is the application of ideas from algebraic coding theory to compressive sensing. In [39], a method called "sudocodes" is proposed, which is based on low-density parity-check (LDPC) codes, which are well established in coding theory. The sudocodes method can recover sparse signals with high probability. Motivated by this method, Xu and Hassibi in [40] proposed a method based on expander graphs, which are a special type of bipartite graph. For the convenience of the reader, the definition of an expander graph is recalled next.

The object under study is an *undirected bipartite graphs*, consisting of a set $\mathcal{V}_I$ of input vertices, a set $\mathcal{V}_O$ of output vertices, and an edge set $\mathcal{E} \subseteq \mathcal{V}_O \times \mathcal{V}_I$, where $(i, j) \in \mathcal{E}$ if and only if there is an edge between node $i \in \mathcal{V}_O$ and node $j \in \mathcal{V}_I$. The corresponding matrix $A \in \{0, 1\}^{|\mathcal{V}_O| \times |\mathcal{V}_I|}$ is called the **biadjacency matrix** of the bipartite graph. The graph is said to be **left regular** of degree D, or D**-left regular**, if every input node has degree D. This is equivalent to requiring that every column of the biadjacency matrix A has exactly D elements equal to 1. Given an input vertex $j \in \mathcal{V}_I$, let $\mathcal{N}(i) \subseteq \mathcal{V}_O$ denote the set of its neighbors, defined as

$$\mathcal{N}(j) := \{i \in \mathcal{V}_O : (i, j) \in \mathcal{E}\}.$$

Given set of input vertices $S \subseteq \mathcal{V}_I$, the set of its neighbors $\mathcal{N}(S) \subseteq \mathcal{V}_O$ is defined as

$$\mathscr{N}(S) := \bigcup_{j \in S} \mathscr{N}(j) = \{i \in \mathscr{V}_O : \exists j \in S \text{ s.t. } (i, j) \in \mathscr{E}\}.$$

Definition 5 A D-left-regular bipartite graph $(\mathscr{V}_I, \mathscr{V}_O, \mathscr{E})$ is said to be a $(K, 1-\beta)$-**expander** for some integer K and some number $\beta \in (0, 1)$ if, for every $S \subseteq V_I$ with $|S| \leq K$, we have that $|N(S)| \geq (1-\beta)D|S|$.

In [40], Xu and Hassibi introduce a new signal recovery algorithm in which the biadjacency matrix of an expander graph with $\beta \leq 1/4$ is used as the measurement matrix. It is referred to here as the "Expander Recovery Algorithm." Xu and Hassibi show that their algorithm recovers an unknown k-sparse vector x *exactly* in $O(k \log n)$ iterations. Subsequently, their method was updated in [41] by increasing the expansion factor from $1 - 1/4 = 3/4$ to $1 - \varepsilon$ in which $\varepsilon < \frac{1}{4}$. With this change, it is shown that the number of recovery iterations required is $O(k)$. However, the number of measurements is more than in the Xu–Hassibi algorithm.

Expander Recovery Algorithm

```
1: Initialize $x - 0_{n\times 1}$
2: if $Y = Ax$ then return output $x$ and exit
3: else
4:     find a variable $x_j$ such that at least $(1-2\varepsilon)$D of the measurements it participates in have identical gap $g$
5:     $x_j \leftarrow x_j + g$ and go to step 2
6: end if
```

In the algorithm above, the term g is called the *gap* and it determines the amount of information of the unknown signal that is missing in the estimate. The gap is defined as following:

$$g_i = y_i - \sum_{j=1}^{n} A_{ij} x_j$$

in which x, y, and A are the unknown signal, the measurement vector, and the measurement matrix, respectively.

11 The New Algorithm

Now, we present our new algorithm, and show that it can exactly recover sparse signals in a single pass, without any iterations. Then, we analyze the performance of the algorithm when the true but unknown vector is not exactly sparse, and/or the measurement is corrupted by noise. The performance of our algorithm is compared with those of ℓ_1-norm minimization and expander graph algorithms in the next section.

12 The New Algorithm

Suppose a matrix $A \in \{0, 1\}^{m \times n}$ has the following properties, referred to as the **main assumption**:

1. Every column a_j of A has precisely q entries of 1 and $m - q$ entries of 0.
2. If a_j, a_t are distinct columns of A, then $\langle a_j, a_t \rangle \leq r - 1$.

Suppose $x \in \Sigma_k$ is a k-sparse n-dimensional vector, and define $y = Ax$ to be the measurement vector. For a given index $j \in [n]$, let $\{v_1(j), \ldots, v_q(j)\} \subseteq [m]$ denote the q rows such that $a_{ij} = 1$. For an index $j \in [n]$, the **reduced measurement vector** $\bar{y}_j \in \mathbb{R}^q$ is defined as

$$\bar{y}_j := [y_{v_1(j)} \ldots y_{v_q(j)}]^\top.$$

Note that $\bar{y}_j$ is the vector consisting of the q measurements in which the component x_j participates.

The main result is given next. Recall that $\|v\|_0$ denotes the number of nonzero components of a vector v.

Theorem 15 *Suppose $x \in \Sigma_k$, $y = Ax$. Then,*

1. If $j \notin \text{supp}(x)$, then $\|\bar{y}_j\|_0 \leq k(r - 1)$.
2. If $j \in \text{supp}(x)$, then $\bar{y}_j$ contains at least $q - (k - 1)(r - 1)$ components that are all equal to x_j.

Proof For $t \in [n]$, let $\mathbf{e}_t \in \mathbb{R}^n$ denote the tth canonical basis vector, which has a 1 as its tth element, and zeros elsewhere, and let $\mathbf{1}_q \in \mathbb{R}^q$ denote the column vector consisting of all ones. Then, we can write:

$$x = \sum_{t \in \text{supp}(x)} x_t \mathbf{e}_t,$$

$$y = Ax = \sum_{t \in \text{supp}(x)} x_t A \mathbf{e}_t = \sum_{t \in \text{supp}(x)} x_t a_t,$$

where a_t denotes the tth column of A. Therefore, for a fixed $j \in [n]$ and $l \in [q]$, we have that

$$y_{v_l(j)} = \sum_{t \in \text{supp}(x)} x_t (a_t)_{v_l(j)}.$$

Letting l range over $[q]$ shows that

$$\bar{y}_j = \sum_{t \in \text{supp}(x)} x_t (\overline{a_t})_j, \tag{51}$$

where $(\overline{a_t})_j$ is the reduced vector of a_t consisting of $(a_t)_{v_1(j)}, \ldots, (a_t)_{v_q(j)}$.

Proof of (1): Suppose $j \notin \text{supp}(x)$. Then, $j \neq t$ for all $t \in \text{supp}(x)$. Therefore, according to item (ii) of the main assumption, we have that $\langle a_j, a_t \rangle \leq r - 1$. Recall that $v_1(j), \ldots, v_q(j)$ are the row indices of column j that contain a 1. Therefore, for a fixed index $t \neq j$, the number of 1's in the set $\{(a_t)_{v_1(j)}, \ldots, (a_t)_{v_q(j)}\}$ equals the inner product $\langle a_j, a_t \rangle$ and thus cannot exceed $r - 1$. Therefore, for a fixed index $t \in \text{supp}(x)$, the vector $x_t(\overline{a_t})_j$ contains no more than $r - 1$ nonzero entries. Substituting this fact into (51) shows that $\bar{y}_j$ is the sum of at most k vectors (because x is k-sparse), each of which has no more than $r - 1$ nonzero entries. Therefore, $\|\bar{y}_j\|_0 \leq k(r - 1)$.

Proof of (2): Suppose $j \in \text{supp}(x)$. Then, we can write

$$\bar{y}_j = \sum_{t \in \text{supp}(x)} x_t(\overline{a_t})_j \tag{52}$$

$$= x_j \mathbf{1}_q + \sum_{t \in \text{supp}(x) \setminus \{j\}} x_t(\overline{a_t})_j, \tag{53}$$

because the "reduced vector" $(\overline{a_j})_j$ consists of q 1's, as denoted by $\mathbf{1}_q$. By the same reasoning as in the proof of (1), it follows that

$$\left\| \sum_{t \in \text{supp}(x) \setminus \{j\}} x_t(\overline{a_t})_j \right\|_0 \leq (k - 1)(r - 1).$$

Therefore, at least $q - (k - 1)(r - 1)$ terms in $\bar{y}_j$ equal x_j.

In view of Theorem 15, we can formulate an algorithm for the recovery of k-sparse vectors, as follows:

New Recovery Algorithm

```
for $j \in [n]$ do
    Construct the reduced measurement vector $\bar{y}_j$.
    Find the number of the elements of $\bar{y}_j$ that are nonzero; call it $\nu$.  ▷ (In implementation, we find the number of elements that are greater than some tolerance $\delta$.)
    if $\nu > q/2$ then
        Find a group of $q/2$ elements in $\bar{y}_j$ that are equal; call this value $\theta_j$.  ▷ (In implementation, we allow some tolerance here.)
        $\hat{x}_j = \theta_j$.
    else
        $\hat{x}_j = 0$
    end
end
```

Note that there is no iterative process involved in the recovery— the estimate $\hat{x}$ is generated after *a single pass* through all n indices.

Theorem 16 *If x is k-sparse, and A satisfies the main assumption with $q > 2k(r - 1)$, then $\hat{x} = x$.*

Proof Note $q > 2k(r-1)$ implies that

$$k(r-1) < q/2, q-(k-1)(r-1) > q-k(r-1) > q/2.$$

Therefore, by Statement 1 of Theorem 15, it follows that if $j \notin \text{supp}(x)$, then $\|\bar{y}_j\|_0 \leq k(r-1) < q/2$. Taking the contrapositive shows that if $\|\bar{y}_j\|_0 \geq q/2$, then $j \in \text{supp}(x)$. Therefore, by Statement 2 of Theorem 15, it follows that at least $q-(k-1)(r-1) > q-k(r-1) > q/2$ elements of $\bar{y}_j$ equal x_j.

Next, we present the extension of our basic algorithm to the cases of a sparse signal with measurement noise, and a nearly sparse signal.

13 Recovery of Sparse Signals with Measurement Noise

In previous work, the model for noisy measurements is that $y = Ax + \eta$ where there is a prior bound of the form $\|\eta\|_2 \leq \varepsilon$. If $x \in \Sigma_k$, then $\sigma_k(x, \|\cdot\|_1) = 0$. Therefore, if robust sparse recovery is achieved, then the bound in (4) becomes $\|\hat{x} - x\|_2 \leq D\varepsilon$. However, our approach draws its inspiration from coding theory, wherein it is possible to recover a transmitted signal correctly provided the transmission is not corrupted in too many places. Therefore, our noise model is that $\|\eta\|_0 \leq M$. In other words, it is assumed that a maximum of M components of the "true" measurement Ax are corrupted by additive noise, but there are no assumptions regarding the *magnitude* of the error signal η. In this case, it is shown that, by increasing the number of measurements, it is possible to recover the true sparse vector x *perfectly*.

Theorem 17 *Suppose $x \in \Sigma_k$, and that $y = Ax + \eta$ where $\|\eta\|_0 \leq M$. Suppose further that the matrix A satisfies the main assumption. Then,*

1. *If $j \notin \text{supp}(x)$, then $\bar{y}_j$ contains no more than $k(r-1) + M$ nonzero components.*
2. *If $j \in \text{supp}(x)$, then $\bar{y}_j$ contains at least $q - [(k-1)(r-1) + M]$ components that are all equal to x_j.*
3. *Suppose the new recovery algorithm is applied with a measurement matrix A that satisfies the main assumption with $q > 2[k(r-1) + M]$. Then, $\hat{x} = x$.*

Proof Suppose $x \in \Sigma_k$ and let $y = Ax + \eta$ where A satisfies the main assumption and $\|\eta\|_0 \leq M$. Let $u = Ax$ denote the uncorrupted measurement. For a fixed index $j \in [n]$, let $\bar{y}_j \in \mathbb{R}^q$ denote the reduced measurement vector, consisting of the components $y_{v_1(j)}$ through $y_{v_q(j)}$, and define $\bar{u}_j \in \mathbb{R}^q$ and $\bar{\eta}_j \in \mathbb{R}^q$ analogously.

First suppose $j \notin \text{supp}(x)$. Then, it follows from Item (1) of Theorem 15 that $\|\bar{u}_j\|_0 \leq k(r-1)$. Moreover, because η has no more than M nonzero components and $\bar{\eta}_j$ is a subvector of η, it follows that $\|\bar{\eta}_j\|_0 \leq M$. Therefore,

$$\|\bar{y}_j\|_0 = \|\bar{u}_j + \bar{\eta}_j\|_0 \leq \|\bar{u}_j\|_0 + \|\bar{\eta}_j\|_0 \leq k(r-1) + M.$$

This is Item (1) above. Next, suppose that $j \in \mathrm{supp}(x)$. Then, it follows from Item (1) of Theorem 15 that at least $q - (k-1)(r-1)$ elements of $\bar{u}_j$ equal x_j. Because $\|\bar{\eta}_j\|_0 \leq M$, it follows that at least $q - (k-1)(r-1) - M$ components of $\bar{y}_j$ equal x_j. This is Item (2) above. Finally, if $q > 2k(r-1) + 2M$, it follows as in the proof of Theorem 16 that $\hat{x} = x$.

Note that the assumption on the noise signal η can be modified to $\|\bar{\eta}_j\|_0 \leq M$ for each $j \in [n]$. In other words, instead of assuming that η has no more than M nonzero components, one can assume that every reduced vector $\bar{\eta}_j$ has no more than E nonzero components.

14 Recovery of Nearly Sparse Signals

As before, if $x \notin \Sigma_k$, then let $x_d \in \mathbb{R}^n$ denote the projection of x onto its k largest components, and let $x_r = x - x_d$. We refer to x_d, x_r as the dominant part and the residual, respectively. Note that, for any $p \in [1, \infty]$, we have that the sparsity index $\sigma_k(x, \|\cdot\|_p)$ equals $\|x_r\|_p$. To (nearly) recover such a vector, we modify the new recovery algorithm slightly. Let δ be a specified threshold.

Modified Recovery Algorithm

```
for $j \in [n]$ do
    Construct the reduced measurement vector $\bar{y}_j$.
    Find the number of the elements of $\bar{y}_j$ that are greater than $\delta$ in magnitude; call it $\nu$.
    if $\nu > q/2$ then
        Find a group of $q/2$ elements in $\bar{y}_j$ such that the difference between the largest and smallest elements is no larger than $2\delta$; Let $\theta_j$ denote the average of these numbers.
        $\hat{x}_j = \theta_j$.
    else
        $\hat{x}_j = 0$
    end
end
```

Theorem 18 *Suppose $x \in \mathbb{R}^n$ and that $\sigma_k(x, \|\cdot\|_1) \leq \delta$. Write $x = x_d + x_r$ where x_d is the dominant part of x consisting of its k largest components, and $x_r = x - x_d$ is the residual. Let $y = Ax$ where A satisfies the main assumption with $q > 2k(r-1)$, and apply the modified recovery algorithm. Then, (i)* $\mathrm{supp}(\hat{x}) = \mathrm{supp}(x_d)$ *and (ii)* $\|\hat{x} - x_d\|_\infty \leq \delta$.

Remark: If ℓ_1-norm minimization is used to recover a *nearly sparse* vector using (1), then the resulting estimate $\hat{x}$ need not be sparse, and second, the support set of the dominant part of $\hat{x}$ need not equal the support set of the dominant part of x.

Proof Write $x = x_d + x_r$ where x_d consists of the dominant part of x and x_r consists of the residual part. By assumption, $\|x_r\|_1 \leq \delta$. Note that the measurement y equals

$Ax = Ax_d + Ax_r$. Let $u = Ax_d$ and observe that $x_d \in \Sigma_k$. Further, observe that, because the matrix A is binary, we have that the induced matrix norm

$$\|A\|_{1\to\infty} := \sup_{v\neq 0} \frac{\|Av\|_\infty}{\|v\|_1} = \max_{i,j} |a_{ij}| = 1.$$

Therefore, $\|Ax_r\|_\infty \leq \|x_r\|_1 \leq \delta$. Now, by Item (1) of Theorem 15, we know that if $j \notin \text{supp}(x_d)$, then no more than $k(r-1)$ components of the reduced vector $\bar{u}_j$ are nonzero. Therefore, then no more than $k(r-1)$ components of the reduced vector $\bar{y}_j$ have magnitude more than δ. By Item (2) of Theorem 15, we know that if $j \in \text{supp}(x_d)$, then at least $q-(k-1)(r-1)$ components of $\bar{u}_j$ equal x_j. Therefore, at least $q-(k-1)(r-1)$ components of $\bar{y}_j$ lie in the interval $[x_j-\delta, x_j+\delta]$. Finally, if $q > 2k(r-1)$, then there is only one collection of $q-(k-1)(r-1) > q/2$ components of the reduced vector $\bar{y}_j$ that lie in an interval of width 2δ. The true x_j lies somewhere within this interval, and we can set $\hat{x}_j$ equal to the midpoint of the interval containing all of these components. In this case, $|\hat{x}_j - x_j| \leq \delta$. Because this is true for all $j \in \text{supp}(x_d)$, it follows that (i) $\text{supp}(\hat{x}) = \text{supp}(x_d)$ and (ii) $\|\hat{x} - x_d\|_\infty \leq \delta$.

Finally, it is easy to combine the two proof techniques and to establish the following theorem for the case where x is not exactly sparse and the measurements are noisy.

Theorem 19 *Suppose $x \in \mathbb{R}^n$ and that $\sigma_k(x, \|\cdot\|_1) \leq \delta$. Write $x = x_d + x_r$ where x_d is the dominant part of x consisting of its k largest components, and $x_r = x - x_d$ is the residual. Let $y = Ax + \eta$ where $\|\eta\|_0 \leq M$, and A satisfies the main assumption with $q > 2k(r-1) + 2M$. Apply the modified recovery algorithm. Then, (i)* $\text{supp}(\hat{x}) = \text{supp}(x_d)$ *and (ii)* $\|\hat{x} - x_d\|_\infty \leq \delta$.

15 Construction of a Binary Measurement Matrix

The results presented until now show that the key to the procedure is the construction of a binary matrix A that satisfies the main assumption. In this subsection, it is shown that previous work by DeVore [15] provides a simple recipe for constructing a binary matrix with the desired properties. Note that [15] was the first paper to propose a completely deterministic procedure for constructing a matrix that satisfies the restricted isometry property. It is shown, in this section, that DeVore's matrix is also a special case of the biadjacency matrix of an expander graph. Therefore, the DeVore matrix acts as a bridge between two distinct compressed sensing algorithms.

We now describe the construction in [15]. Suppose q is a prime number or a power of a prime number, and let $\mathbb{F}_q$ denote the finite field with q elements. Suppose a is a polynomial of degree $r-1$ or less with coefficients in $\mathbb{F}_q$, and define its "graph" as the set of all pairs $(x, a(x))$ as x varies over $\mathbb{F}_q$. Now, construct a vector $u_a \in \{0, 1\}^{q^2 \times 1}$ by setting the entry in row (i, j) to 1 if $j = a(i)$, and to zero otherwise. To illustrate, suppose $q = 3$, so that $\mathbb{F}_q = \{0, 1, 2\}$ with arithmetic modulo 3. Let $r = 4$, and let $a(x) = 1 + 2x + x^2 + x^3$. With this choice, we have that $a(0) = 1, a(1) = 2$, and $a(2) = 2$. The corresponding 9×1 column vector has 1's in positions $(0, 1)$, $(1, 2)$, $(2, 2)$, and zeros elsewhere. This construction results in a $q^2 \times 1$ column vector u_a that consists of q blocks of size $q \times 1$, each of which contains a single 1 and $q - 1$ zeros. Therefore, u_a contains q elements of 1 and the rest equal to zero.

Now, let $\Pi_{r-1}(\mathbb{F}_q)$ denote the set of all polynomials of degree $r - 1$ or less with coefficients in $\mathbb{F}_q$. In other words,

$$\Pi_{r-1}(\mathbb{F}_q) := \left\{ a(x) = \sum_{i=0}^{r-1} a_i x^i, a_i \in \mathbb{F}_q \right\}.$$

Note that $\Pi_{r-1}(\mathbb{F}_q)$ contains precisely q^r polynomials, because each of the r coefficients can assume q different values.[3] Now, define

$$A := [u_a, a \in \Pi_{r-1}(\mathbb{F}_q)] \in \{0, 1\}^{q^2 \times q^r}. \tag{54}$$

The following theorem from [15] shows that the matrix A constructed as above satisfies the main assumption, and also the RIP with appropriately chosen constants.

Theorem 20 (See [15, Theorem 3.1]) *For the matrix $A \in \{0, 1\}^{q^2 \times q^r}$ defined in (54), we have that*

$$\langle u_a, u_b \rangle \leq r - 1 \tag{55}$$

whenever a, b are distinct polynomials in $\Pi_{r-1}(\mathbb{F}_q)$. Consequently, if we define the column-normalized matrix $A' = (1/\sqrt{q})A$, then A satisfies the RIP of order k with constant $\delta_k \leq ((k - 1)(r - 1))/q$.

Next, it is shown that the DeVore construction is a special case of a method given in [32] for constructing expander graphs. The construction in [32] is as follows: Let $h \geq 2$ be any integer. Then, the map $\Gamma : \mathbb{F}_q^r \times \mathbb{F}_q \to \mathbb{F}^{s+1}$ is defined as

$$\Gamma(f, y) := [y, f(y), f^h(y), f^{h^2}(y), \dots, f^{h^{s-1}}(y)]. \tag{56}$$

An alternate way to express the function Γ is

$$\Gamma(f, y) = [y, (f^{h^i}(y), i = 0, \dots, s - 1)].$$

In the definition of the function Γ, y ranges over $\mathbb{F}_q$ as the "counter," and the above graph is left regular with degree q. The set of input vertices is $\mathbb{F}_q^r$, consisting of polynomials in some indeterminate Y with coefficients in $\mathbb{F}_q$ of degree no larger than

[3] If the leading coefficient of a polynomial is zero, then the degree would be less than r.

$r-1$. The set of input vertices has cardinality q^r. The set of output vertices is $\mathbb{F}^{s+1}$ and each output vertex is an $(s+1)$-tuple consisting of elements from $\mathbb{F}_q$. The set of output vertices has cardinality q^{s+1}. Note that the graph is q-left regular in that every input vertex has exactly q outgoing edges.

Theorem 21 (See [32, Theorem 3.3]) *For every pair of integers h, s, the bipartite graph defined in (56) is a $(h^s, 1-\beta)$-expander with*

$$\beta = \frac{(r-1)(h-1)s}{q} \tag{57}$$

whenever

$$h < \frac{q}{s(r-1)} + 1.$$

Note that the inequality simply ensures that $\beta > 0$.

Now, we relate the construction of DeVore with that in [32].

Theorem 22 *The matrix A constructed in [15] is a special case of the graph in Theorem 21 with $s = 1$, and any value for h. Therefore, a bipartite graph with the biadjacency matrix of [15] is a $(h, 1-\beta)$-expander with*

$$\beta = \frac{(r-1)(h-1)}{q} \tag{58}$$

whenever

$$h < \frac{q}{r-1} + 1.$$

Proof Suppose that $s = 1$ and that h is any integer. In this case, each polynomial f with coefficients in $\mathbb{F}_q$ of degree $r-1$ or less gets mapped into the pair $(y, f(y))$ as y ranges over $\mathbb{F}_q$. This is precisely what was called the "graph" of the polynomial f in [15].

Part IV: Numerical Experiments

16 Statistical Recovery

The preceding two parts of the chapter were devoted to two methods for *guaranteed* recovery of all sparse vectors. For such methods, $m = O(k \ln(n/k))$ measurements suffice to generate (with high probability) a matrix that satisfies the restricted isometry property (RIP) and thus can be used along with basis pursuit to recover *all* sufficiently sparse vectors. If the measurement matrix is chosen as the biadjacency matrix of a bipartite graph, then $m = k\sqrt{n}$ measurements suffice. Moreover, in practice, deterministic methods often require fewer measurements. Further details can be found in the Appendix.

If the requirement is relaxed from *guaranteed* recovery of *all* sufficiently sparse vectors to recovery of *most* sufficiently sparse vectors or *statistical* recovery, then there is a parallel body of research showing that the number of measurements m can be reduced quite substantially. In fact, $m = O(k)$ measurements suffice. In this subsection, we highlight just a few of the many papers in this area of research. To streamline the presentation, the papers are not always cited in chronological order.

In [42], the underlying assumption is that the unknown vector x is generated according to a known probability distribution p_X, which can in fact be used by the decoder. Three different dimensions of the probability distribution p_X are introduced, namely, the Rényi information dimension, the MMSE dimension, and the Minkowski dimension. The encoder is permitted to be nonlinear, in contrast to earlier cases where the encoding consisted of multiplication by a measurement matrix. The decoder is also permitted to be nonlinear but is assumed to be Lipschitz continuous. The optimal performance in this setting is analyzed. A central result in this paper states that asymptotically as the vector dimension n and the number of measurements m both approach infinity, statistical recovery is possible *if and only if*

$$m \geq n\bar{d}(p_X) + o(n),$$

where $\bar{d}(p_X)$ denotes the Rényi information dimension of p_X. Since the Rényi information dimension is comparable to the ratio k/n, the above result states that $O(k)$ measurements are sufficient. However, no procedure is given to construct an encoder–decoder pair.

In a series of papers [11, 12, 43], Donoho and various co-workers studied "phase transitions" in the performance of various recovery algorithms. A readable survey of these results is given in [44]. The unknown n-vector is assumed to be k-sparse, and the measurement vector $y \in \mathbb{R}^m$ equals Ax, where A consists of samples of normal random variables, scaled by the normalization factor $a/\sqrt{m}$. Two quantities are relevant here, namely, the "undersampling rate" $\delta = m/n$ and the sparsity $\rho = k/m$. In all of these papers, the aim is to show that for each algorithm there exists a sharp threshold $\rho_\theta(\delta)$ such that if $\rho > \rho_\theta(\delta)$, then the unknown vector is recovered with probability approaching one, whereas if $\rho < \rho_\theta(\delta)$, then the algorithm *fails* with probability approaching one.

Specifically, in [43], an algorithm known as "approximate message passing" (AMP) is analyzed. AMP is a simple thresholding type of algorithm that is much faster than minimizing the ℓ_1-norm. Specifically, suppose $\phi : \mathbb{R} \to \mathbb{R}$ is a smooth "threshold" function, and extend it to a map from $\mathbb{R}^n$ to $\mathbb{R}^n$ by applying it componentwise. The AMP algorithm begins with an initial guess $x^0 = 0$, and then one sets

$$x^{t+1} = \phi(A^\top w^t + x^t),$$

$$w^t = y - Ax^t + \frac{1}{\delta} w^{t-1} (\phi'(A^\top w^{t-1} + x^{t-1})),$$

where ϕ' denotes the derivative of ϕ. It is clear that AMP is much faster than ℓ_1-norm minimization. Despite this, it is shown in [43] that the phase transition behavior of AMP is comparable to that of ℓ_1-norm minimization. In [45], the AMP algorithm is modified to incorporate the results of [42], and phase transition results are derived. In this paper, the authors also introduce the idea of "spatial coupling" introduced in [46].

Finally, in [13], the authors study a very general class of algorithms. Suppose as before that $y \in \mathbb{R}^m$ equals Ax, where A consists of samples of normal random variables, scaled by the normalization factor $a/\sqrt{m}$. The decoding algorithm is

$$\hat{x} = \operatorname*{argmin}_z f(z) \text{ s.t. } y = Az,$$

where the "regularizer" $f(\cdot)$ is a convex function satisfying some technical conditions. So, this theory applies to ℓ_1-norm minimization. In this paper, a central role is played by the "descent cone" of f at a point x, which is defined as

$$\mathscr{D}(f, x) := \bigcup_{\tau > 0} \{h \in \mathbb{R}^n : f(x + \tau h) \leq f(x)\}.$$

It is clear that $\mathscr{D}(f, x)$ is indeed a cone. Next, for each cone, a quantity called the "statistical dimension," denoted by δ, is defined; see [13, Section 2.2] for a precise definition. With all these items in place, a central result is established; see [13, Theorem II].

Theorem 23 *Define $a(\varepsilon) := \sqrt{8 \log(4/\varepsilon)}$. With all other symbols as above, if*

$$m \leq \delta(\mathscr{D}(f, x)) - a(\varepsilon)\sqrt{n},$$

then the decoding algorithm fails with probability $\geq 1 - \varepsilon$. If

$$m \geq \delta(\mathscr{D}(f, x)) + a(\varepsilon)\sqrt{n},$$

then the decoding algorithm succeeds with probability $\geq 1 - \varepsilon$.

17 Phase Transitions

Phase transition refers to an abrupt change in the qualitative behavior of the solution to a problem as the parameters are changed. In the case of compressed sensing, let us define two quantities: $\theta := m/n$, which is known as the undersampling ratio and $\phi := k/m$, which is known as the oversampling ratio.[4] Suppose we choose integers $n, m < n$, together with a matrix A, and use basis pursuit as the decoder. If a k-sparse vector is chosen at random, we can ask: What is the probability that $(A, \Delta_{\mathrm{BP}})$ recovers the vector?

This question is answered in [6, 47] using techniques from combinatorial geometry, specifically polytope theory. Suppose P is a convex polytope, that is, the convex hull of a finite number of points in $\mathbb{R}^n$, and A as a $m \times n$ matrix. Polytopes have vertices, edges, and k-dimensional faces called *facets*. Let $f_k(P)$ denote the number of facets of dimension k. In particular, we can define various polytopes corresponding to k-sparse vectors in $\mathbb{R}^n$, which is called a "cross polytope" [47]. The image of P under A, denoted by AP, is also a polytope, and for each k, we have that $f_k(AP) \leq f_k(P)$. Moreover, it is shown in [47] that, if x is drawn at random from the cross polytope P, then the probability of recovering x via basis pursuit equals the ratio $f_k(AP)/f_k(P)$. Thus, the question becomes of analyzing the behavior of this ratio for specific polytopes P and specific matrices A.

In [10], it is proved that if A consists of samples of a normal (Gaussian) random variable, then as $n \to \infty$ this recovery probability (i.e., the face count ratio) exhibits a sharp change when ϕ is increased for a fixed θ. It is claimed that this behavior is observed even with moderate values of n such as $n = 1024$. In this paper, the authors make a distinction between two types of recovery, namely, uniform and nonuniform. In uniform recovery, basis pursuit is expected to recover *all* k-sparse vectors, with high probability (with respect to the randomly generated Gaussian matrix). In nonuniform recovery, there is also a uniform probability measure on the set of k-sparse vectors in $\mathbb{R}^n$, and basis pursuit is expected to recover a k-sparse vector with high probability (both with respect to the randomly generated k-sparse vector and the randomly generated Gaussian vector). Clearly, nonuniform recovery holds whenever uniform recovery holds, but the converse need not be true. In the present chapter, the focus is on uniform recovery.

Donoho and Tanner in [11] and [48] define the strong threshold ϕ_s and weak threshold ϕ_w as the threshold for uniform and nonuniform recovery, respectively. Unfortunately, there is no closed form expression for ϕ values either in the weak or the strong case. However, in [48], Theorems 1.4 and 1.5 suggest complicated formulas for these ϕ functions that work in the *asymptotic* case when $n \to \infty$ (or $\delta \to 0$). The complicated closed-form formulas for $\delta \to 0$ can be approximated as follows:

$$\phi_s(\theta) \approx \left| \frac{1}{2e \log(\sqrt{\pi}\theta)} \right|, \theta \to 0,$$

[4] This terminology is introduced in [6] with m/n denoted by δ and k/m denoted by ρ. Since these symbols are now used to denote different quantities in the compressed sensing literature, we use θ and ϕ instead.

$$\phi_w(\theta) \approx \left| \frac{1}{2\log(\theta)} \right|, \theta \to 0.$$

It can be seen that, as $\theta \to 0$, $\phi_w(\theta) \approx e\phi_s(\theta)$. This means that when n is very large and θ is very small (that is, very few measurements compared to the dimension of the unknown vector), the threshold in k for recovering *the vast majority* of k-sparse vectors is roughly e times the threshold for recovering *all* k-sparse vectors.

All of the results mentioned above are for Gaussian measurement matrices. They are rigorous and draw upon very deep results about Gaussian random variables. However, there is interest to see whether similar phase transition behavior is observed with other types of measurement matrices. It is shown in [14, 47, 49] that a large class of random and deterministic measurement matrices display the same phase transition boundary as Gaussian matrices. Specifically, in [14] the authors study various deterministic constructions for measurement matrices, such as spikes and sines, spikes and noiselets, Delsarte–Goethals frames, Grassmannian frames, Paley frames, and chirp matrices. They conclude that all of these matrices display the same phase transition boundary as with Gaussian matrices. However, as mentioned by the authors of [14], all of the deterministic matrices they study satisfy only the "statistical restricted isometry property (STRIP)" and not necessarily the RIP/RNSP. Thus, with the class of deterministic matrices studied in [14], there is no *guaranteed* recovery via basis pursuit. The main motivation for this part of the present chapter is to fill this gap, by studying phase transitions in basis pursuit with a class of deterministic matrices that are *guaranteed* to achieve sparse recovery, namely, the DeVore class and the array matrix class.

18 Numerical Experiments

In this section, we carry out two different numerical experiments to illustrate the use of binary matrices in compressed sensing. In each experiment, we compare the array code matrix proposed here with the DeVore construction of [15] and a random Gaussian matrix. In the first experiment, the objective is to compare both classes of binary matrices with random Gaussian matrices, while the objective in the second experiment is to compare the phase transition boundaries for all three classes of matrices. In each case, we generate 100 random k-sparse vectors and use the CVX package under MATLAB to perform ℓ_1-norm minimization.

18.1 Guaranteed Recovery

In this experiment, we fix the vector dimension n, and vary the sparsity count k. Specifically, the dimension n is chosen as $n = 149^2 = 22,201$, and two different sparsity counts k are chosen, namely, $k = 14$ and $k = 69$. For each of the array

Table 1 Comparison of DeVore, array code, and random Gaussian matrices for $n = 149^2 = 22,201$ and $k = 14, 69$

	Array matrix		DeVore matrix		Gaussian matrix	
k	m_A	T in sec.	m_D	T in sec.	m_G	T in sec.
14	2,235	29.014	841	15.94	11,683	259,100
69	10,430	248.5	19,321	1795	44,345	692,260

code matrices, the DeVore matrix, and a random Gaussian matrix, the number of measurements m is chosen so as to *guarantee* robust k-sparse recovery using basis pursuit. In the case of the random Gaussian matrix, the failure probability ξ is chosen as 10^{-9}, and the number of samples m is chosen in accordance with Theorem 2, specifically (7). Because the number of measurements m in each case is chosen to be large enough to guarantee recovery, the only items of interest are (i) value of m for the same n, k with different methods of generating A and (ii) the CPU time associated with basis pursuit in each case.

When $n = 149^2$ and $k = 14$, with the array code matrix we choose $q = \sqrt{n} = 149$ and $d_l = k + 1 = 15$, which leads to $m = d_l\sqrt{n} = 2,235$ measurements. With DeVore's construction, we choose q to be the next largest prime after $2k$, namely, $q = 29$ and $m = 29^2 = 841$. Because $k < \sqrt{n}/4$, the DeVore construction requires fewer measurements than the array code matrix, as expected. When $k = 69$, with the array code matrix, we choose $d_l = k + 1 = 70$ and $m = d_l\sqrt{n} = 10,430$ measurements. In contrast, with the DeVore construction, we choose q to be the next largest prime after $2k$, namely, 139, which leads to $m = q^2 = 19,321$. Because $k > \sqrt{n}/4$, the DeVore construction requires more measurements than the array code matrix, as expected. For the random Gaussian matrix, (7) gives $m = 11,683$ when $n = 149^2$, $k = 14$ and $m = 44,345$, that is, *more than* n, when $k = 69$. Therefore, there was no point in running the Gaussian method with $k = 69$.

The results are shown in Table 1. From this table, it can be seen that both classes of binary matrices (DeVore and array code) require significantly less CPU time compared to random Gaussian matrices. As shown in Example 3, the DeVore matrix is to be preferred when $k < \sqrt{n}/4$ while the array code matrix is to be preferred when $k > \sqrt{n}/4$. But in either case, both classes of matrices are preferable to random Gaussian matrices.

18.2 Phase Transition Study

In this subsection, we compare the phase transition behavior of the basis pursuit formulation with both classes of binary matrices (DeVore and array code), and random Gaussian matrices. The dimension of the vector n is chosen to be 1024, to match the previous literature on the topic. The phase transition boundary for the Gaussian case is computed using the software provided by Prof. David Donoho. For the DeVore

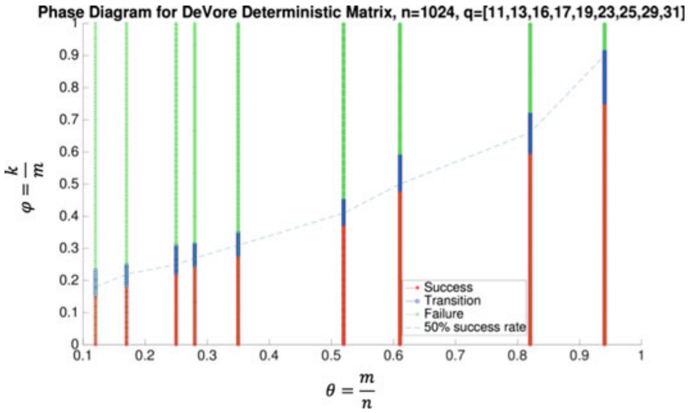


Fig. 1 Phase transition diagram with success, transition, and failure regions for $n = 1024$ using DeVore measurement matrix

class of binary matrices, we again chose $n = 1024 = 2^{10}$ which is an even power of a prime number. For the array code class, we chose $m = 961 = 31^2$, which is the nearest square of a prime number to 1024. For each class of binary matrix, there are only certain values of m that are permissible. For the DeVore class, m equals the square of a prime number q such that $m = q^2 < n$. Thus, the permissible choices for q are

$$\{11, 13, 16, 17, 19, 23, 25, 29, 31\}.$$

In the case of array matrices $n = 31^2 = q^2$, and the permissible values of m are lq as l ranges from 1 to $q - 1 = 30$ (Fig. 1).

Our first objective is to compare the phase transition width for binary versus Gaussian matrices. The phase transition width is defined as the interval of values of ϕ for which the recovery rate is 5 and 95%. The specific questions studied are given as follows:

1. Is the phase transition width the same for all three types of matrices?
2. As n is varied, does the phase transition width vary as $C/\sqrt{n}$ for some constant C that is *independent* of the method used?
3. What is the CPU time with each type of binary matrix?
4. Is the 95% recovery value of ϕ for a given θ the same for all three types of matrices?

The results are presented in Table 2. It can be seen that the transition widths are almost the same for the three methods, which suggests that phase transition is a universal property and is independent of the measurement matrix. The CPU time needed to run basis pursuit is also indicated in Table 2. From this, it is clear that binary measurement matrices provide a far more time-efficient recovery procedure, especially when n is large.

Table 2 Comparison of transition widths w, 50% success rate width ϕ_{50}, average width $\bar{w}$, and elapsed time T for $n = 1024$ using Binary DeVore matrix and Gaussian measurement matrix (subscript b and g, respectively)

θ	w_b	w_g	ϕ_{50_b}	ϕ_{50_g}	T_b in sec.	T_g in sec.
0.12	0.083	0.074	0.18	0.2	70	182
0.17	0.071	0.071	0.22	0.22	106	416
0.25	0.09	0.078	0.25	0.27	168	1435
0.28	0.073	0.059	0.27	0.28	222	1484
0.35	0.072	0.066	0.31	0.32	316	5038
0.52	0.08	0.07	0.41	0.39	636	8695
0.61	0.11	0.09	0.5	0.46	1695	12810
0.82	0.12	0.1	0.66	0.63	1744	13453
0.94	0.17	0.15	0.9	0.77	2261	15827
$\bar{w}$	0.097	0.084	–			

Next, we computed the width of the phase transition region for each method, using the formula $C_1/\sqrt{n}$ for some constant C_1 as claimed in [14]. We tried three different n values, ($n = 256, 512, 1024$), using DeVore's binary measurement matrices of dimensions $q^2 \times n$. As it is discussed earlier in this chapter, q must satisfy $\sqrt[3]{n} \leq q < \sqrt{n}$. For a fixed m and n, the phase transition zone is found by varying k in $[1, m]$. In order to find the constant C_1 for each n, we consider the average weight $\bar{w}$ and $C_1 = \bar{w} \times \sqrt{n}$. This gives us three different but yet close values of C_1. By setting $C_1 = 2.7$ (which is the average of all C_1 values), we can assume that for binary DeVore's measurement matrix, the phase transition width is of the form $\frac{2.7}{\sqrt{n}}$. In Table 3, Δ shows how far off this expression is for each n. Since Δ is around 0.01 for each n, we can claim that $C_1 = 2.7$ is a reasonable choice and for DeVore's binary measurement matrices, the phase transition width follows the same formula as in Gaussian matrices.

We repeated the same experiment for the array code matrix with $n = 961 = q^2$. In this case, $k < d_l - 1$, where d_l lies in the range 2 to q. Hence, k was varied in the range 1 to m with recovery guaranteed for $k < d_l - 1$. The phase diagram is shown in Fig. 2. Table 4 shows the phase transition width and the CPU time using the array code matrices. A comparison of Tables 2 and 4 shows that for almost a similar n value ($n = 1024$ vs. $n = 961$), array matrices are much faster in recovery than DeVore and Gaussian matrices while the phase transition width is nearly identical in all cases.

Our final observation is presented in Fig. 3. It shows that the experimental phase transition boundary for the array and DeVore binary deterministic matrices for 95% recovery perfectly matches the asymptotic (theoretical) curve shown in [43, Figure 1] and reproduced here.[5]

[5] We thank Prof. David Donoho for providing the software to reproduce the curve.

Table 3 Phase transition widths w, 50% success rate width ϕ_{50}, average width $\bar{w}$, and the constant C_1 for three different dimensions, $n = 256, 512, 1024$ using DeVore's binary measurement matrix

n	θ	w	ϕ_{50}	n	θ	w	ϕ_{50}	n	θ	w	ϕ_{50}
256	0.19	0.16	0.2	512	0.16	0.11	0.2	1024	0.12	0.083	0.18
	0.25	0.16	0.22		0.24	0.09	0.24		0.17	0.071	0.22
	0.32	0.14	0.31		0.33	0.095	0.3		0.25	0.09	0.25
	0.47	0.16	0.36		0.5	0.11	0.4		0.28	0.073	0.27
	0.66	0.17	0.47		0.57	0.11	0.43		0.35	0.072	0.31
	–	–	–		0.71	0.15	0.52		0.52	0.08	0.41
	–	–	–		–	–	–		0.61	0.11	0.5
	–	–	–		–	–	–		0.82	0.12	0.66
	–	–	–		–	–	–		0.94	0.17	0.9
$\bar{w}$	0.16				0.11				0.097		
C_1	2.56				2.53				3.104		
Δ	0.01				0.009				0.011		

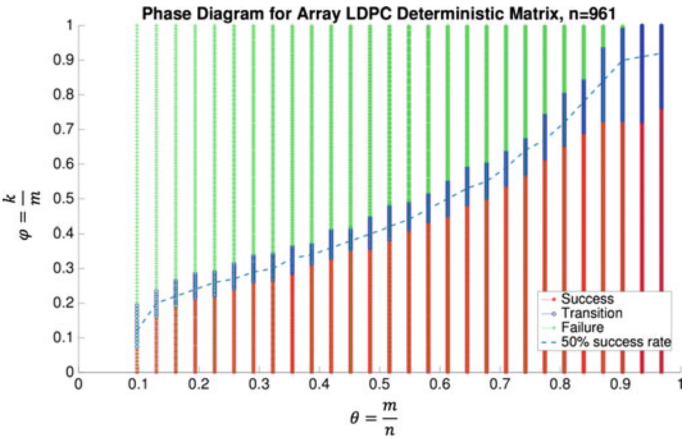


Fig. 2 Phase transition diagram with success, transition, and failure regions for $n = 961$ using array LDPC parity-check matrix

19 Computational Results for the Noniterative Algorithm

Theorems 20 and 22 show that the measurement matrix construction proposed in [15] falls within the ambit of both the restricted isometry property as well as expander graphs. Hence, this matrix can be used together with ℓ_1-norm minimization, the expander graph algorithm of Xu–Hassibi as well as our proposed algorithm. In this section, we compare the performance of all three algorithms using the DeVore construction. Note, however, that the number of rows of the matrix (or equivalently, the

Table 4 Phase transition widths w, 50% success rate width ϕ_{50}, and average width $\bar{w}$ for $n = 961$ using array LDPC parity-check matrix

θ	m	w	ϕ_{50}	T in sec.
0.1935	186	0.08	0.24	0.9423
0.2258	217	0.08	0.24	0.9351
0.2581	248	0.08	0.27	0.8931
0.2903	279	0.08	0.29	0.8732
0.3548	341	0.07	0.33	0.8458
0.5161	496	0.1	0.42	0.6909
0.6129	589	0.1	0.5	0.5946
0.8387	806	0.16	0.78	0.1818
0.9355	899	0.28	0.91	0.0385
$\bar{w}$	0.1144			

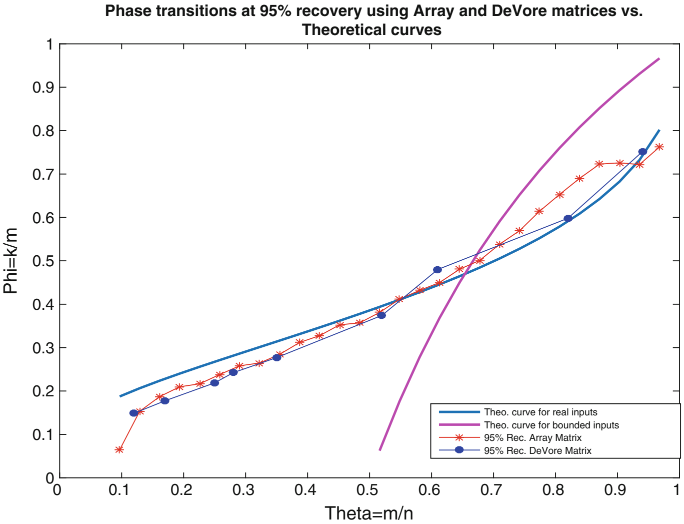


Fig. 3 Theoretical curves for real and bounded inputs and 95% recovery curve using array LDPC parity-check matrix and DeVore matrix

number of measurements) will vary from one method to another. This is discussed in this section.

We begin by comparing the number of measurements required by ℓ_1-norm minimization, expander graphs, and our method. In ℓ_1-norm minimization, as shown in Theorem 20, the matrix A, after column normalization dividing each column by $\sqrt{q}$, satisfies the RIP with constant $\delta_k = (k-1)(r-1)/q$. Combined with Theorem 1, we conclude that ℓ_1-norm minimization with the DeVore construction achieves robust k-sparse recovery, whenever

$$\frac{(\lceil tk \rceil - 1)(r-1)}{q} < \sqrt{\frac{t-1}{t}}. \tag{59}$$

To maximize the value of k for which the above inequality holds, we set r to its minimum permissible value, which is $r = 3$. Also, we replace $\lceil tk \rceil - 1$ by its upper bound tk, which leads to

$$\frac{2tk}{q} < \sqrt{\frac{t-1}{t}}, \text{ or } \frac{2k}{q} < \sqrt{\frac{t-1}{t^3}}.$$

Elementary calculus shows that the right side is maximized when $t = 1.5$. So, the RIP constant of the measurement matrix must satisfy

$$\delta_{tk} < \sqrt{(t-1)/t} = 1/\sqrt{3} \approx 0.577.$$

Let us choose a value of 0.5 for δ_{tk} to give some "cushion." Substituting the values $t = 1.5$, $r = 2$ in (59) and ignoring the rounding operations finally leads to the condition

$$\frac{3k}{q} < 0.5, \text{ or } q > 6k. \tag{60}$$

For expander graphs, we can calculate the expansion factor $1 - \beta$ from Theorem 21. This gives

$$\beta = \frac{(r-1)(h-1)s}{q}.$$

Since we wish the expansion factor $1 - \beta$ to be as close to one as possible, or equivalently, β to be as small as possible, we choose s to be its minimum value, namely, $s = 1$. Now, we substitute $r = 3$, $h = 2k$ (following [40]), and set $1 - \varepsilon \geq 3/4$, or equivalently $\varepsilon \leq 1/4$. This leads to

$$\frac{2(2k-1)}{q} \leq 1/4, \text{ or } q \geq 8(2k-1).$$

Finally, for the new algorithm, it has already been shown that $q \geq 2(r-1)k = 4k$. Therefore, the required number of measurements for each of the three algorithms are as shown in Table 5. Note that since the matrix A has q^3 columns, we must also have that $n \leq q^3$.

Next, we present a numerical example to compare the three methods. We chose $n = 20,000$ to be the dimension of the unknown vector x. Since all three methods produce a measurement matrix with $m = q^2$ rows, we must have $q < 141 \approx \sqrt{20000}$, because otherwise the number of measurements would exceed the dimension of the vector! Since the expander graph method requires the most measurements, the sparsity count k must satisfy $8(2k-1) < 141$, which gives $k \leq 9$. However, if we try to recover k-sparse vectors with $k = 9$ using the expander graph method, the number

Table 5 Number of measurements for various approaches

Method	ℓ_1-norm Min.	Expander graphs	New Alg.
Bound: $q \geq$	$6k$	$8(2k-1)$	$4k$
Bound: $m \geq$	$36k^2$	$64(2k-1)^2$	$16k^2$
q with $k = 6$	37	89	29
m with $k = 6$	1,369	7,921	841

of measurements m would be essentially equal to the dimension of the vector n. Hence, we chose value of $k = 6$. With this choice, the values of q and the number of measurements are shown in Table 5. Note that q must be chosen as a prime number.

Having chosen the values of n and k, we generated 100 different k-sparse n-dimensional vectors, with both the support set of size k and the nonzero values of x generated at random.[6] As expected, both the expander graph method and the new algorithm recovered the unknown vector x *exactly* in all 100 cases. The ℓ_1-norm minimization method recovers x with very small error. However, there was a substantial variation in the average time over the 100 runs. Our algorithm took an average of 0.0951 seconds, or about 95 ms, ℓ_1-norm minimization took 21.09 s, and the expander graph algorithm took 76.75 s. Thus, our algorithm was about 200 times faster than ℓ_1-norm minimization and about 800 times faster than the expander graph algorithm.

As a final example, we introduced measurement noise into the output. As per Theorem 17, if $y = Ax + \eta$ where $\|\eta\|_0 \leq M$, then it is still possible to recover x exactly by increasing the prime number q. (Note that it is also possible to retain the same value of q by reducing the sparsity count k so that $k + M$ is the same as before.) Note that the only thing that matters here is the number of nonzero components of the noise η and not their magnitudes. One would expect that if the norm of the noise gets larger and larger our algorithm would continue to recover the unknown sparse vector exactly, while ℓ_1-norm minimization would not be able to. In other words, our algorithm is tolerant to “shot” noise, whereas ℓ_1-norm minimization is not. The computational results bear this out. We choose $n = 20,000$ and $k = 6$ as before, and $M = 6$, so that we perturb the true measurement Ax in six locations. Specifically, we chose $\eta = \alpha v$ where each component of v is normally distributed, and then increased the scale factor α. Each experiment was repeated with 100 randomly generated sparse vectors and shot noise. The results are shown in Table 6.

20 Discussion

In this chapter, we have built upon a previously proven sufficient condition for *stable* k-sparse recovery, and showed that it actually guarantees *robust* k-sparse recovery, that is, basis pursuit achieves k-sparse recovery even in the presence of measurement

[6] MATLAB codes are available from the authors.

Table 6 Performance of new algorithm and ℓ_1-norm minimization with additive shot noise

	New Algorithm			ℓ_1-norm minimization		
Alpha	Err.	Time	Rec.	Err.	Time	Rec.
10^{-5}	0	0.1335	100	3.2887e-06	26.8822	0
10^{-4}	0	0.1325	100	3.2975e-05	26.6398	0
10^{-3}	0	0.1336	100	3.3641e-04	28.1876	0
10^{-2}	0	0.1357	100	0.0033	23.1727	0
10^{-1}	0	0.1571	100	0.033	28.9145	0
10	0	0.1409	100	1.3742	26.6362	0
20	0	0.1494	100	1.3967	26.5336	0

noise. We then derived a *universal lower bound* on the number of measurements in order for binary matrix to satisfy this sufficient condition. Ideally, we would like to prove a universal *necessary* condition along the following lines: If a left-regular binary measurement matrix A achieves robust k-sparse recovery of order k, then $d_l \geq \phi(k)$ where $\phi(\cdot)$ is some function that is waiting to be discovered. In such a case, the bounds in Theorem 11 would truly be universal. At present, there are no known universal necessary conditions for binary measurement matrices, other than Theorem 3 which is applicable to *all* matrices, not just binary matrices.

Note that, as shown in [18, Problem 13.6], a binary matrix does not satisfy the RIP of order k with constant δ unless

$$m \geq \min \left\{ \frac{1-\delta}{1+\delta} n, \left(\frac{1-\delta}{1+\delta} \right)^2 k^2 \right\}.$$

This negative result has often been used to suggest that binary matrices are not suitable for compressed sensing. However, RIP is only a *sufficient* condition for robust sparse recovery, and as shown here, it is possible to provide far weaker sufficient conditions for robust sparse recovery in terms of the RNSP, when the measurement matrix is binary.

Moreover, it is possible to compare the sample complexities implied by (7) for random Gaussian matrices with those corresponding to the DeVore class and the array code class, to see that when $n < 10^5$ and $k < \sqrt{n}$, in fact binary matrices require fewer measurements, as shown in Table 7.

One might argue that the bound in (7) is only a *sufficient* condition for the number of measurements, and that in actual examples, far fewer measurements suffice. This is precisely the motivation behind studying the phase transition of basis pursuit with binary matrices. As shown in Sect. 18.2, in fact, there is no difference between the phase transition behavior of random Gaussian matrices and binary matrices. This observation reinforces earlier observations in [14]. In other words, the fraction of randomly generated k-sparse vectors that can be recovered using m measurements is the same whether one uses Gaussian matrices or binary matrices. Given that basis pursuit can be implemented much more efficiently with binary measurement matrices

Table 7 Comparison of the number of measurements for the DeVore binary matrix, the array code binary matrix, and the random Gaussian matrix. Note that $m_D = q_D^2$ and $m_A = (k+1)q_A$. The quantity m_G is computed according to (7)

n	k	q_D	m_D	q_A	m_A	m_G
900	5	11	121	31	186	4,467
	10	23	529		341	6,682
	15	31	961		496	8,982
	20	41	1,681		651	10,863
10^4	20	47	2,209	101	2,121	14,436
	40	83	6,889		4,141	25,430
	60	127	16,129		6,161	35,600
	80	163	26,569		8,181	45,232
10^5	50	101	10,201	317	16,167	39,165
	100	211	44,521		32,017	71,878
	150	307	94,249		47,867	102,604
	200	401	160,801		63,717	132,030

than with random Gaussian matrices, and both classes of matrices exhibit similar phase transition properties, there appears to be a very strong case for preferring binary measurement matrices over random Gaussian matrices, notwithstanding the "order optimality" of the latter class.

There is one final point that we wish to make. Theorem 12 suggests that in order to use binary matrices for compressed sensing, it is better to use graphs with *small* girth, in fact, of girth 6. This runs counter to the intuition in LDPC decoding, where one wishes to design binary matrices with *large* girth. Indeed, in [50], the authors build on an earlier paper [51] and develop a message-passing type of decoder that achieves order optimality using a binary matrix. The binary matrices that are used in [50] all have *large* girth $\Omega(\log n)$ which is the theoretical upper bound. This discrepancy needs to be explored. At present, all that we can say is that the model for compressed sensing used in [50] is different from the one used here and in most of the compressed sensing literature. Specifically (to paraphrase a little bit), in [50] in the unknown vector, each component is binary, and the probability that the component equals one is k/n. Thus, the *expected value* of nonzero bits is k, but it could be larger or smaller. Accordingly, the actual sparsity count is a random number that could exceed k. The recovery results proved in [50] are also probabilistic in nature. It is worth further study to determine whether this difference is sufficient to explain why, in the present case, graphs of low girth are to be preferred.

Acknowledgements The contents of this chapter report various results from the doctoral thesis of the second author, carried out under the supervision of the first author. The authors thank Prof. David Doonoho and Mr. Hatef Monajemi of Stanford University for their helpful suggestions on phase transitions, and for providing code to enable us to reproduce their computational results. They also thank Prof. Phanindra Jampana of IIT Hyderabad for helpful discussions on the construction of Euler squares.

Appendix

In this appendix, we compare the number of measurements used by probabilistic as well as deterministic methods to guarantee that the corresponding measurement matrix A satisfies the restricted isometry property (RIP), as stated in Theorem 1. Note that the number of measurements is computed from the best available sufficient condition. In principle, it is possible that matrices with fewer rows might also satisfy the RIP. But there would not be any theoretical justification for using such matrices.

In probabilistic methods, the number of measurements m is $O(k \log(n/k))$. However, in reality, the O symbol hides a huge constant. It is possible to replace the O symbol by carefully collating the relevant theorems in [18]. This leads to the following explicit bounds.

Theorem 24 *Suppose X is a random variable with zero mean, unit variance, and suppose in addition that there exists a constant c such that*[7]

$$E[\exp(\theta X)] \leq \exp(c\theta^2), \ \forall \theta \in \mathbb{R}. \tag{61}$$

Define

$$\gamma = 2, \zeta = 1/(4c), \alpha = \gamma e^{-\zeta} + e^{\zeta}, \beta = \zeta, \tag{62}$$

$$\tilde{c} := \frac{\beta^2}{2(2\alpha + \beta)}. \tag{63}$$

Suppose an integer k and real numbers $\delta, \xi \in (0, 1)$ are specified, and that $A = (1/\sqrt{m})\Phi$, where $\Phi \in \mathbb{R}^{m \times n}$ consists of independent samples of X. Then, A satisfies the RIP of order k with constant δ with probability $\geq 1 - \xi$ provided

$$m \geq \frac{1}{\tilde{c}\delta^2} \left(\frac{4}{3} k \ln \frac{en}{k} + \frac{14k}{3} + \frac{4}{3} \ln \frac{2}{\xi} \right). \tag{64}$$

In (64), the number of measurements m is indeed $O(k \log(n/k))$. However, for realistic values of n and k, the number of measurements n would be comparable to, or even to exceed, n, which would render "compressed" sensing meaningless.[8] For "pure" Gaussian variables, it is possible to find improved bounds for m (see Theorem 2 which is based on [18, Theorem 9.27]. Also, for binary random variables where X equals ±1 with equal probability, another set of bounds is available [52]. While all of these bounds are $O(k \log(n/k))$, in practical situations the bounds are not useful.

[7] Such a random variable is said to be **sub-Gaussian**. A normal random variable satisfies (61) with $c = 1/2$.

[8] In many papers on compressed sensing, especially those using Gaussian measurement matrices, the number of measurements m is *not* chosen in accordance with any theory, but simply picked out of the air.

Table 8 Best available bounds for the number of measurements for various choices of n and k using both probabilistic and deterministic constructions. For probabilistic constructions, the failure probability is $\xi = 10^{-9}$. m_G, m_{SG}, m_A denote, respectively, the bounds on the number of measurements using a normal Gaussian, a sub-Gaussian with $c = 1/2$, and a bipolar random variable and the bound of Achlioptas. For deterministic methods, m_D denotes the number of measurements using DeVore's construction, while m_C denotes the number of measurements using chirp matrices

n	k	m_G	m_{SG}	m_A	m_D	m_C
10^4	5	5,333	28,973	3,492	841	197
10^4	6	5,785	31,780	3,830	1,369	257
10^4	7	6,674	37,308	4,496	1,681	401
10^4	8	7,111	40,035	4,825	2,209	487
10^4	9	7,972	45,424	5,474	2,809	677
10^4	10	8,396	48,089	5,796	3,481	787
10^5	10	10,025	57,260	6,901	3,481	787
10^5	12	11,620	66,988	8,073	5,041	1,163
10^5	14	13,190	76,582	9,229	6,889	1,601
10^5	16	14,739	86,061	10,372	9,409	2,129
10^5	18	16,268	95,441	11,502	11,449	2,707
10^5	20	17,781	104,733	12,622	16,129	3,371
10^6	5	7,009	38,756	4,671	10,201	1,009
10^6	10	11,639	66,431	8,006	10,201	1,009
10^6	15	16,730	96,976	11,687	10,201	1,949
10^6	20	21,069	123,076	14,832	16,129	3,371
10^6	25	25,931	152,373	18,363	22,201	5,477
10^6	30	30,116	177,635	21,407	32,041	7,753
10^6	50	47,527	283,042	34,110	94,249	21,911
10^6	60	55,993	334,440	40,304	128,881	31,687
10^6	70	64,335	385,171	46,417	175,561	43,271
10^6	80	72,573	435,331	52,462	229,441	56,659
10^6	90	80,718	484,992	58,447	292,681	71,837
10^6	100	88,781	534,210	64,378	358,801	88,807

This suggests that it is worthwhile to study *deterministic* methods for generating measurement matrices that satisfy the RIP. There are very few such methods. Indeed, the authors are aware of only three methods. The paper [15] uses a finite field method to construct a binary matrix, and this method is used in the present chapter. The paper [53] gives a procedure for choosing rows from a unitary Fourier matrix such that the resulting matrix satisfies the RIP. This method leads to the same values for the number of measurements m as that in [15]. Constructing partial Fourier matrices is an important part of reconstructing time-domain sparse signals from a limited number of frequency measurements (or vice versa). Therefore, the results of [53] can be used in this situation. In both of these methods, m equals q^2 where q is appropriately chosen prime number. Finally, in [54] a method is given based on chirp matrices. In this case, m equals a prime number q. Note that the partial Fourier matrix and the chirp matrix are complex, whereas the method in [15] leads to a binary matrix. In all three methods, $m = O(n^{1/2})$, which grows faster than $O(k \log(n/k))$. However, the constant under this O symbol is quite small. Therefore, for realistic values of k and n, the bounds for m from these methods are much smaller than those derived using probabilistic methods.

Table 8 gives the values of m for various values of n and k. Also, while the chirp matrix has fewer measurements than the binary matrix, ℓ_1-norm minimization with the binary matrix runs much faster than with the chirp matrix, due to the sparsity of the binary matrix. In view of these numbers, in the present chapter, we used DeVore's construction as the benchmark for the recovery of sparse vectors.

References

1. S. S. Chen, D. L. Donoho, and M. A. Saunders, "Atomic decomposition by basis pursuit," *SIAM Journal on Scientific Computing*, vol. 20, no. 1, pp. 33–61, 1999.
2. S. S. Chen, D. L. Donoho, and M. A. Saunders, "Atomic decomposition by basis pursuit," *SIAM Review*, vol. 41, no. 1, pp. 129–159, 2001.
3. E. J. Candès and T. Tao, "Decoding by linear programming," *IEEE Transactions on Information Theory*, vol. 51(12), pp. 4203–4215, December 2005.
4. E. J. Candès, J. Romberg, and T. Tao, "Stable signal recovery from incomplete and inaccurate measurements," *Communications in Pure and Applied Mathematics*, vol. 59(8), pp. 1207–1223, August 2006.
5. D. L. Donoho, "Compressed sensing," *IEEE Transactions on Information Theory*, vol. 52(4), pp. 1289–1306, April 2006.
6. D. L. Donoho, "For most large underdetermined systems of linear equations, the minimal ℓ_1-norm solution is also the sparsest solution," *Communications in Pure and Applied Mathematics*, vol. 59(6), pp. 797–829, 2006.
7. E. Candès, "The restricted isometry property and its implications for compresed sensing," *Comptes rendus de l'Académie des Sciences, Série I*, vol. 346, pp. 589–592, 2008.
8. A. Cohen, W. Dahmen, and R. DeVore, "Compressed sensing and best k-term approximation," *Journal of the American Mathematical Society*, vol. 22(1), pp. 211–231, January 2009.
9. K. D. Ba, P. Indyk, E. Price, and D. P. Woodruff, "Lower bounds for sparse recovery," in *Proceedings of the ACM-SIAM Symposium on Discrete Algorithms (SODA)*, January 2010, pp. 1190–1197.
10. D. L. Donoho and J. Tanner, "Neighborliness of randomly projected simplices in high dimensions," *Proceedings of the National Academy of Sciences*, vol. 102, pp. 9452–9457, July 2005.
11. D. L. Donoho, "High-dimensional centrally symmetric polytopes with neighborliness proportional to dimension," *Discrete and Computational Geometry*, vol. 35, no. 4, pp. 617–652, May 2006.
12. D. L. Donoho and J. Tanner, "Counting faces of randomly projected polytopes when the projection radically lowers dimension," *Journal of the American Mathematical Society*, vol. 22, no. 1, pp. 1–53, January 2009.
13. D. Amelunxen, M. Lotz, M. B. McCoy, and J. A. Tropp, "Living on the edge: phase transitions in convex programs with random data," *Information and Inference*, vol. 3, pp. 224–294, 2014.
14. H. Monajemi, S. Jafarpour, M. Gavish, and D. Donoho, "Deterministic matrices matching the compressed sensing phase transitions of gaussian random matrices," *Proceedings of the National Academy of Sciences of the United States of America*, vol. 110, no. 4, pp. 1181–1186, 2013.

15. R. DeVore, "Deterministic construction of compressed sensing matrices," *Journal of Complexity*, vol. 23, pp. 918–925, 2007.
16. T. Cai and A. Zhang, "Sparse representation of a polytope and recovery of sparse signals and low-rank matrices," *IEEE Transactions on Information Theory*, vol. 60(1), pp. 122–132, 2014.
17. R. Zhang and S. Li, "A proof of conjecture on restricted isometry property constants $\delta_{tk}(0<t<\frac{4}{3})$," *IEEE Transactions on Information Theory*, vol. 64, no. 3, pp. 1699–1705, March 2018.
18. S. Foucart and H. Rauhut, *A Mathematical Introduction to Compressive Sensing*. Springer-Verlag, 2013.
19. A. S. Bandeira, E. Dobriban, D. G. Mixon, and W. F. Sawin, "Certifying the restricted isometry property is hard," *IEEE Transactions on Information Theory*, vol. 59, no. 6, pp. 3448–3450, June 2013.
20. W. Xu and B. Hassibi, "Compressed sensing over the Grassmann manifold: A unified analytical framework," in *Proceedings of the 46th Allerton Conference*, 2008, pp. 562–567.
21. S. Foucart, "Stability and robustness of ℓ_1-minimizations with Weibull matrices and redundant dictionaries," *Linear Algebra and Its Applications*, vol. 441, pp. 4–21, 2014.
22. S. Ranjan and M. Vidyasagar, "Tight performance bounds for compressed sensing with conventional and group sparsity," arXiv:1606.05889v2, 2018.
23. S. Li, F. Gao, G. Ge, and S. Zhang, "Deterministic construction of compressed sensing matrices via algebraic curves," *IEEE Transactions on Information Theory*, vol. 58, no. 8, pp. 5035–5041, August 2012.
24. S. D. Howard, A. R. Calderbank, and S. J. Searle, "A fast reconstruction algorithm for deterministic compressive sensing using second order reedmuller codes," in *Proceedings of the 42nd IEEE Annual Conference on Information Sciences and Systems*, 2008, pp. 11–15.
25. R. R. Naidu, P. Jampana, and C. S. Sastry, "Deterministic compressed sensing matrices: Construction via euler squares and applications," *IEEE Transactions on Signal Processing*, vol. 64, no. 14, pp. 3566–3575, July 2016.
26. H. F. MacNeish, "Euler squares," *Annals of Mathematics*, vol. 23, no. 3, pp. 221–227, March 1922.
27. Y. Erlich, A. Gordon, M. Brand, G. J. Hannon, and P. P. Mitra, "Compressed genotyping," *IEEE Transactions on Information Theory*, vol. 56, no. 2, pp. 706–723, 2010.
28. R. Berinde, A. C. Gilbert, P. Indyk, H. Karloff, and M. J. Strauss, "Combining geometry and combinatorics: a unified approach to sparse signal recovery," in *Proceedings of the Forty-Sixth Annual Allerton Conference*, 2008, pp. 798–805.
29. P. Indyk and M. Ružić, "Near-optimal sparse recovery in the ℓ_1-norm," in *Proceedings of the 49th Annual IEEE Symposium on the Foundations of Computer Science (FoCS)*, 2008, pp. 199–207.
30. A. Gilbert and P. Indyk, "Sparse recovery using sparse matrices," *IEEE Proceedings*, vol. 98, no. 6, pp. 937–947, June 2010.
31. M. Lotfi and M. Vidyasagar, "A fast noniterative algorithm for compressive sensing using binary measurement matrices," *IEEE Transactions on Signal Processing*, vol. 67, pp. 4079–4089, August 2019.
32. V. Guruswami, C. Umans, and S. Vadhan, "Unbalanced expanders and randomness extractors from ParvareshVardy codes," *Journal of the ACM*, vol. 56, no. 4, pp. 20:1–20:34, 2009.
33. A. G. Dimakis, R. Smarandache, and P. O. Vontobel, "LDPC codes for compressed sensing," *IEEE Transactions on Information Theory*, vol. 58, no. 5, pp. 3093–3114, May 2012.
34. X.-J. Liu and S.-T. Xia, "Reconstruction guarantee analysis of binary measurement matrices based on girth," in *Proceedings of the International Symposium on Information Theory*, 2013, pp. 474–478.
35. M. Lotfi and M. Vidyasagar, "Compressed sensing using binary matrices of nearly optimal dimensions," arXiv:1808.03001, 2018.
36. S. Hoory, "The size of bipartite graphs with a given girth," *Journal of Combinatorial Theory, Series B*, vol. 86, pp. 215–220, 2002.

37. K. Yang and T. Helleseth, "On the minimum distance of array codes as LDPC codes," *IEEE Transactions on Information Theory*, vol. 49, no. 12, pp. 3268–3271, December 2003.
38. J. L. Fan, "Array codes as ldpc codes," in *Proceedings of 2nd International Symposium on turbo Codes*, 2000, pp. 543–546.
39. S. Sarvotham, D. Baron, and R. G. Baraniuk, "Sudocodes – fast measurement and reconstruction of sparse signals," in *Proceedings of the International Symposium on Information Theory*, 2006, pp. 2804–2808.
40. W. Xu and B. Hassibi, "Efficient compressive sensing with deterministic guarantees using expander graphs," in *Proceedings of IEEE Information Theory Workshop, Lake Tahoe*, 2007.
41. S. Jafarpour, W. Xu, B. Hassibi, and R. Calderbank, "Efficient compressed sensing using optimized expander graphs," *IEEE Transactions on Information Theory*, vol. 55, no. 9, pp. 4299–4308, 2009.
42. Y. Wu and S. Verdú, "Optimal phase transitions in compressed sensing," *IEEE Transactions on Information Theory*, vol. 58, no. 10, pp. 6241–6263, October 2012.
43. D. L. Donoho, A. Maleki, and A. Montanari, "Message-passing algorithms for compressed sensing," *Proceedings of the National Academy of Sciences*, vol. 106, no. 45, pp. 18 914–18 919, 2009.
44. D. L. Donoho and J. Tanner, "Precise undersampling theorems," *Proceedings of the IEEE*, vol. 98, no. 6, pp. 913–924, June 2010.
45. D. L. Donoho, A. Javanmard, and A. Montanari, "Information-theoretically optimal compressed sensing via spatial coupling and approximate message passing," *IEEE Transactions on Information Theory*, vol. 59, no. 11, pp. 7434–7464, November 2013.
46. F. Krzakala, M. Mzard, F. Sausset, Y. Sun, and L. Zdeborová, "Probabilistic reconstruction in compressed sensing: algorithms, phase diagrams, and threshold achieving matrices," *Journal of Statistical Mechanics: Theory and Experiment*, vol. 12, 2012.
47. D. Donoho and J. Tanner, "Observed universality of phase transitions in high-dimensional geometry, with implications for modern data analysis and signal processing," *Philosophical Transactions of The Royal Society, Part A: Mathematical, Physical and Engineering Sciences*, vol. 367, no. 1906, pp. 4273–4293, November 2009.
48. D. L. Donoho and J. Tanner, "Counting the faces of randomly-projected hypercubes and orthants, with applications," *Discrete and Computational Geometry*, vol. 43, no. 3, pp. 522–541, April 2010.
49. M. Bayati, M. Lelarge, and A. Montanari, "Universality in polytope phase transitions and message passing algorithms," arXiv:1207.7321v2, 2015.
50. A. Khajehnejad, A. S. Tehrani, A. G. Dimakis, and B. Hassibi, "Explicit matrices for sparse approximation," in *Proceedings of the International Symposium on Information Theory*, 2011, pp. 469–473.
51. S. Arora, C. Daskalakis, and D. Steurer, "Message-passing algorithms and improved lp decoding," in *Proceedings of the 41st Annual ACM Symposium on the Theory of Computing*, 2009, p. 3–12.
52. D. Achlioptas, "Database-friendly random projections: Johnson-Lindenstrauss with binary coins," *Journal of Computer and System Sciences*, vol. 66, pp. 671–687, 2003.
53. G. Xu and Z. Xu, "Compressed sensing matrices from Fourier matrices," *IEEE Transactions on Information Theory*, vol. 61(1), pp. 469–478, January 2015.
54. L. Applebaum, S. D. Howard, S. Searle, and R. Calderbank, "Chirp sensing codes: Deterministic compressed sensing measurements for fast recovery," *Applied and Computational Harmonic Analysis*, vol. 26, pp. 283–290, 2009.

Stochastic Optimization for Energy Storage Allocation in Smart Grids in the Presence of Uncertainty

Martina Bucciarelli, Simone Paoletti and Antonio Vicino

Abstract A key subject in the study of smart grids is to accommodate uncertainty in various contexts, including planning and operation of electricity grids in the presence of distributed generation from renewable energy sources, stochastic demand patterns, and varying network configurations. The impact of uncertainty on the solution of different problems formulated in the optimal power flow framework calls for stochastic programming paradigms in the form of two- or multi-stage problems, or optimization programs with chance constraints. In this chapter, we focus on the problem of optimally siting and sizing the energy storage systems in a distribution network. These devices are recognized as good candidates to tackle different issues, such as voltage/frequency regulation, minimal curtailment of renewable generation, peak shaving, or others. For the sizing problem, a scenario-based approach is adopted to cope with uncertain demand and generation profiles at the different buses of the network. A novel scenario reduction technique is presented to make the resulting stochastic optimization problem computationally tractable. A heuristic strategy based on network voltage sensitivity analysis is adopted to deal with the combinatorial nature of the energy storage siting problem. The overall procedure is tested on a IEEE benchmark network, highlighting good performance on a realistic case study.

M. Bucciarelli (✉) · S. Paoletti · A. Vicino
Dipartimento di Ingegneria dell'Informazione e Scienze Matematiche,
Università di Siena, via Roma 56, 53100 Siena, Italy
e-mail: bucciarelli@diism.unisi.it

S. Paoletti
e-mail: paoletti@diism.unisi.it

A. Vicino
e-mail: vicino@diism.unisi.it

© Springer Nature Switzerland AG 2018

T. Başar (ed.), *Uncertainty in Complex Networked Systems*,
Systems & Control: Foundations & Applications,
https://doi.org/10.1007/978-3-030-04630-9_9

1 Introduction

In spite of its contribution to reducing carbon dioxide production, the widespread and ever-increasing penetration of solar photovoltaic and wind power generation plants in electricity grids is posing several challenging problems to the operation of existing distribution electricity networks. Indeed, since the main feature of non-programmable renewable energy sources is intermittency, their introduction has consistently increased the coupling of grid planning and operation with weather and climate conditions. These issues have been strengthened by the dramatic changes of demand behavior at critical nodes of the grids, as a consequence of the introduction of new sources of uncertainty deriving from, e.g., demand response programs, charging of electric vehicles, and installation of heat pumps [20]. Appropriate planning and operation of modern electricity grids require to understand and model all these uncertainties, characterized by high dimensionality, strong interdependencies, and complex dynamics. For example, classical approaches based on generating probabilistic forecasts by exploiting predictive marginal probability distribution functions are not appropriate in a context where both spatial and temporal correlations are an essential feature of demand patterns and distributed generation (DG).

The main contribution of the present chapter consists of showing how an accurate modeling of uncertainties on demand and DG, taking into account both time and space interdependencies, allows one to solve robustly the problem of allocating energy storage systems (ESSs) in a distribution electricity grid with the aim, e.g., of supporting voltage control. It is indeed well recognized [1, 9, 22, 35] that the effect of different types of uncertainties on grid operation can be mitigated by the deployment of ESSs, which can act either as loads or as generators to compensate the local excess or lack of energy in the grid. While there exists a wide literature on ESS sizing in a deterministic context, where generation and demand curves are assumed to be known, tackling the ESS allocation problem by suitably accommodating uncertainty remains a challenging problem [2, 4, 13, 26, 27].

As mentioned above, ESS deployment allows for improving grid performance, reliability, flexibility, and security. In this chapter, we focus on voltage regulation in distribution networks, which, according to most regulatory frameworks, is a crucial aspect of the quality of service to end customers [11, 34]. For instance, excess of power from DG may determine overvoltages in the distribution grid, leading to a reverse power flow upstream the transformer, with a strong impact on system operation and protection management. In these situations, charging of the ESSs installed in the grid may help bring voltage magnitudes back within the limits. As an additional benefit, in this way, it is possible to avoid DG curtailment, which represents the action most widely used by distribution system operators to limit the impact of excessive production from DG. Use of ESSs may also help smooth the fast occurring voltage variations arising as one of the tangible effects of sudden changes in energy production and/or demand patterns.

The problem of the optimal ESS allocation must be tackled at the planning stage. The decision problem consists of defining the number of storage devices to be deployed, their locations (siting), and sizes (sizing). A literature review of ESS allocation techniques, classified according to both the ESS application and the methodology used to find a solution, can be found in the survey paper [36]. Optimal ESS siting and sizing are often carried out simultaneously, either through a cost–benefit analysis [29] or formulating a single optimization problem, e.g., in the form of a mixed integer nonlinear program [26], or a bi-level model [3]. In other cases, the two problems are dealt with separately in order to alleviate the computational burden. This is the approach taken in this chapter.

We formulate the ESS sizing problem in an optimal power flow (OPF) framework, where an appropriate cost function including storage installation and operation costs is optimized, subject to storage dynamics, power flow, and network constraints (see, e.g., [19, 21]). Demand and DG profiles at the different buses of the network are considered as realizations of stochastic processes, whose characteristics are estimated from available historical data. A scenario-based approach is taken for describing the statistics of uncertain variables, by generating a rich set of scenarios reflecting the spatial and temporal correlations of the processes involved. Then, the ESS sizing problem is formulated as a two-stage stochastic program [30], where the first-stage problem aims at minimizing a linear combination of storage installation and expected operation costs, and the second-stage problem is a standard OPF, whose solution provides the optimal storage control policy for given realizations of demand and DG [8]. The two-stage problem is approximated with a single-stage, multi-scenario OPF, which, in view of the richness of the scenario set, turns out to be typically intractable. To cope with this challenging computational complexity, a scenario reduction technique is devised, which relies on a metric exploiting the structure of the single-stage, multi-scenario problem [8].

With reference to the ESS siting problem, which is combinatorial in the number of buses of the network and the number of ESSs to be deployed, a heuristic procedure is adopted, which exploits the network voltage sensitivity matrix [15].

The overall ESS siting and sizing procedure is illustrated using the topology of an IEEE 37-bus test network with wind power generation, showing the effectiveness of the procedure in the presence of a very large scenario set.

The chapter is organized as follows. Section 2 presents the network model and constraints, and states the considered ESS allocation problem. This problem is addressed in Sect. 3 in a framework with uncertainty on demand and generation. In that section, the ESS siting and sizing procedures of [8, 15] are summarized by highlighting how uncertainty is dealt with in both of them. Numerical results obtained by applying the ESS allocation procedure to an IEEE 37-bus distribution network are illustrated in Sect. 4. Finally, conclusions are drawn in Sect. 5.

2 The ESS Allocation Problem

The decision problem considered in this chapter consists of defining the number of ESSs to be deployed in a distribution network, their locations (siting) and sizes (sizing) [35]. In order to formulate the problem mathematically, we first introduce the bus injection model of a distribution network, and the problem constraints related to ESS and network operation.

In the following, $\mathrm{Re}(z)$, $\mathrm{Im}(z)$, $|z|$, and z^* denote the real part, imaginary part, modulus, and complex conjugate of the complex number z, respectively. For a real number x, $\left[x\right]^+ = \max\{x, 0\}$ and $\left[x\right]^- = \max\{0, -x\}$. Moreover, $u(t)$ is the value of the variable u at time $t\Delta T$, where $t = 0, 1, 2, \dots$ is the discrete time index and ΔT is the time step.

2.1 Bus Injection Model

The bus injection model is one of the standard models used for power flow (PF) analysis and optimization [21]. The model involves nodal variables such as voltages and current/power injections. Let a distribution network be described by a graph $(\mathscr{N}, \mathscr{E})$, where $\mathscr{N} = \{1, 2, \dots, n\}$ is the set of nodes (*buses*) and $\mathscr{E}$ is the set of edges (*lines*). According to the classical π-model for the n-bus system [24], the admittance to ground at bus i is denoted by y_{ii}, while $y_{ij} = y_{ji}$ is the line admittance between buses i and j. If $(i, j) \notin \mathscr{E}$, $y_{ij} = 0$. At time t, the complex voltage at bus i is denoted by $V_i(t)$, while the net active and reactive power injections into bus i are denoted by $P_i(t)$ and $Q_i(t)$, respectively. These quantities are related by the power balance equations

$$P_i(t) = \mathrm{Re}\Big\{V_i(t) \sum_{j \in \mathscr{N}} V_j^*(t) Y_{ij}^*\Big\} \tag{1a}$$

$$Q_i(t) = \mathrm{Im}\Big\{V_i(t) \sum_{j \in \mathscr{N}} V_j^*(t) Y_{ij}^*\Big\}, \tag{1b}$$

where

$$Y_{ij} = \begin{cases} y_{ii} + \sum_{h \neq i} y_{ih} & \text{if } i = j \\ -y_{ij} & \text{otherwise.} \end{cases} \tag{2}$$

Without loss of generality, bus 1 represents the interconnection with an external grid, modeled as a *slack* bus with fixed complex voltage $V_1(t)$. The other buses in the set $\mathscr{L} = \{2, \dots, n\}$ are modeled as *load* buses, for which the known quantities are the net active and reactive power injections $P_i(t)$ and $Q_i(t)$. At time t and for given $V_1(t)$, $P_2(t)$, $Q_2(t)$, ..., $P_n(t)$, $Q_n(t)$, the PF problem consists of solving the system of nonlinear equations (1), $i \in \mathscr{N}$, with respect to the unknowns $P_1(t)$, $Q_1(t)$, $V_2(t)$,

…, $V_n(t)$. There are several methods for carrying out this task, the most popular being the Newton–Raphson method (see, e.g., [24, Chap. 3]).

2.2 Constraints

It is assumed that m ESSs are installed in the network. The subset of buses equipped with ESSs is denoted by $\mathscr{S} = \{s_1, \ldots, s_m\} \subseteq \mathscr{L}$. The energy level in the storage at bus s and time t is denoted by $e_s(t)$, satisfying the physical constraint

$$0 \leq e_s(t) \leq E_s, \tag{3}$$

where E_s is the ESS size. Let $r_s(t)$ be the active power exchanged by the ESS at bus s and time t. If power is injected into the ESS, we set $r_s(t) > 0$, whereas $r_s(t) < 0$ when power is extracted from the ESS. With the same convention about the sign, the reactive power exchanged by the ESS at bus s and time t is denoted by $b_s(t)$. Limits on the apparent power exchanged by the ESS are modeled through a polygonal approximation of the feasible region for the pair $\big(r_s(t), b_s(t)\big)$, described by

$$\varGamma_s\, r_s(t) + \varUpsilon_s\, b_s(t) \leq \varXi_s E_s, \tag{4}$$

where $\varGamma_s$, $\varUpsilon_s$, and $\varXi_s$ are given column vectors [8, 26]. The right-hand side of (4) is a function of E_s to model the possible dependence of power ratings on energy ratings for a given storage technology [32]. The first-order difference equation

$$e_s(t) = e_s(t-1) + \eta_s^c \left[r_s(t)\right]^+ \varDelta T - \frac{1}{\eta_s^d} \left[r_s(t)\right]^- \varDelta T \tag{5}$$

models the dynamics of $e_s(t)$, where η_s^c and η_s^d are the charging and discharging efficiencies of the storage at bus s, and $e_s(0)$ is the known initial condition. Since, in this chapter, we consider time horizons of one day, we further impose the constraint that the storage energy level is equal at the beginning and at the end of the day, namely,

$$e_s(T) = e_s(0), \tag{6}$$

where T is the number of time steps per day. This is done in order to decouple ESS operation over different days.

Additional constraints are included to describe desired and/or physical limits on PF variables. In order to keep the voltage magnitude between bounds imposed by voltage quality requirements, the following constraints are enforced:

$$\underline{v}_i^2 \leq |V_i(t)|^2 \leq \overline{v}_i^2, \tag{7}$$

where $\underline{v}_i \leq \overline{v}_i$ are given positive constants. Moreover, physical properties of the lines impose limits on the apparent power transferred from bus $i \in \mathscr{N}$ to the rest of the network through line $(i, j) \in \mathscr{E}$. These limits are expressed by the constraints

$$\big|V_i(t)\big[V_i(t) - V_j(t)\big]^* y_{ij}^*\big| \leq \overline{S}_{ij}, \tag{8}$$

where $\overline{S}_{ij} = \overline{S}_{ji}$ are given upper bounds.

Summarizing, for a generic bus $i \in \mathscr{L}$, possibly having loads, generators, and ESS connected to it, the power balance equations (1) can be rewritten by highlighting all contributions of active and reactive power

$$\mathrm{Re}\Big(V_i(t) \sum_{j\in\mathscr{N}} V_j^*(t)Y_{ij}^*\Big) = P_i^G(t) - P_i^D(t) - r_i(t) \tag{9a}$$

$$\mathrm{Im}\Big(V_i(t) \sum_{j\in\mathscr{N}} V_j^*(t)Y_{ij}^*\Big) = Q_i^G(t) - Q_i^D(t) - b_i(t), \tag{9b}$$

where $P_i^G(t)$ and $P_i^D(t)$ denote the active power generated and demanded at bus i, and $Q_i^G(t)$ and $Q_i^D(t)$ have a similar meaning in terms of reactive power (all these quantities are assumed to be nonnegative). In (9), if no load, generator or ESS is connected to bus i, the corresponding contribution is set equal to zero.

2.3 Problem Formulation

The decision about the number, the locations and the sizes of the ESSs to be deployed in the network is to be made by minimizing a combination of installation and operation costs. We consider a cost structure of the following type:

$$C(m, \lambda, x) = \rho m + \varsigma\big[c(x) + \gamma\, g(\lambda, x)\big], \tag{10}$$

where m is the number of ESSs, $\lambda = (s_1, \ldots, s_m)$ is the vector of ESS locations, and $x = (E_{s_1}, \ldots, E_{s_m})$ is the vector of ESS sizes. The term $c(x)$ in (10) represents the total energy capacity corresponding to the vector x as

$$c(x) = \sum_{s\in\mathscr{S}} E_s, \tag{11}$$

while $g(\lambda, x)$ is a measure of the (expected) operational costs corresponding to ESSs placed at the locations defined by the vector λ, and sized according to the vector x (see Sect. 3.1 for a discussion on the structure of the cost $g(\lambda, x)$). It is assumed that $g(\lambda, x) = \infty$ when there exist operating conditions (defined by demand and generation profiles in the network) for which it is not possible to satisfy the constraints (3)–(9). The nonnegative coefficients ρ, ς, and γ are used in (10) to trade-off the different cost terms.

With the above definitions, the considered ESS allocation problem reads as follows.

Problem 1 For a given distribution network described by the graph $(\mathscr{N}, \mathscr{E})$ and the admittance matrix $Y = [Y_{i,j}]_{i,j\in\mathscr{N}}$, find the number m of ESSs to be deployed, the vector of ESS locations λ, and the vector of ESS sizes x, which minimize the cost $C(m, \lambda, x)$ in (10).

3 Dealing with Uncertainty in ESS Allocation

Solving Problem 1 directly is a very challenging task due to the combinatorial nature of the siting problem, and the fact that future realizations of demand and generation, needed to evaluate the operation costs in (10), are unknown at the planning stage. The latter aspect calls for problem formulations taking this uncertainty explicitly into account.

In order to tackle the above issues, we adopt an iterative two-step procedure similar to that proposed in [15], and shown in Algorithm 1. At each iteration, the procedure first solves the siting problem for a given number of ESSs, and then computes the optimal sizes for those ESSs.

Algorithm 1 Iterative procedure for ESS allocation

```
for $m = 1$ to $n - 1$ do
    $\lambda^{(m)} \leftarrow$ Solve siting problem for $m$ ESSs
    $x^{(m)} \leftarrow$ Solve sizing problem for ESSs placed according to $\lambda^{(m)}$
    $C(m, \lambda^{(m)}, x^{(m)}) \leftarrow$ Evaluate total cost
end for
$m^\star = \arg\min_m C(m, \lambda^{(m)}, x^{(m)})$
return $(m^\star, \lambda^{(m^*)}, x^{(m^*)})$
```

The two steps of the proposed ESS allocation procedure are described in the following sections. The siting step of Algorithm 1 is performed by applying the heuristic approach proposed in [15]. The approach copes with the combinatorial nature of the siting problem by exploiting clustering and voltage sensitivity analysis to identify a set of buses with greater effect of power injection on voltage variations at the other buses. In turn, the sizing step is carried out following the scenario-based approach proposed in [8], which makes it possible to accommodate uncertainty on demand and generation described in the form of a very large set of daily profiles.

3.1 ESS Siting

The approach to ESS siting proposed in [15] is based on clustering and voltage sensitivity analysis. It applies to radial networks. First, a complete graph is built over the network using the entries of the voltage sensitivity matrix to weight the arcs. This complete graph is then partitioned into a given number of clusters. Finally, a bus is selected within each cluster with the aim of maximizing voltage controllability in the cluster. The approach is summarized in the following steps, where q is the predefined number of clusters.

1. *Network clustering.* First, the set $\mathscr{L}$ is partitioned into q disjoint subsets by running a clustering algorithm on the complete graph built over $\mathscr{L}$. The weight associated with the edge (i, j) is equal to the sensitivity of voltage magnitude at bus j to power injection at bus i, defined as

$$\Psi_{i,j} = \partial |V_j| / \partial P_i. \tag{12}$$

   Among several methods for graph clustering available in the literature (see, e.g., [25]), the one adopted in [15] is based on [17]. It searches for q subgraphs which form a partition of the original graph, while minimizing the sum of the weights associated with the removed edges. In our application, this amounts to constructing the partition of the complete graph which minimizes the sum of the voltage sensitivities associated with the removed connections. The outcome of the clustering algorithm is a partition $\{\mathscr{L}_h\}_{h=1}^q$ of the node set $\mathscr{L}$, from which subnetworks $\mathscr{G}_h = (\mathscr{L}_h, \mathscr{E}_h)$ are reconstructed by defining $\mathscr{E}_h = \{(i, j) \in \mathscr{E} : i \in \mathscr{L}_h, j \in \mathscr{L}_h\}$.
2. *Candidate buses.* For each subnetwork $\mathscr{G}_h$, the critical buses are identified as the buses with generation and the buses that are leaves of the original network. The corresponding set is denoted by $\mathscr{C}_h$. Then, the set Ω_h of candidate buses is formed with all the buses along the paths connecting any pair of critical buses of $\mathscr{G}_h$.
3. *Bus selection.* The ESS for subnetwork $\mathscr{G}_h$ is placed at bus s_h defined as

$$s_h = \arg \max_{i \in \Omega_h} \min_{j \in \mathscr{L}_h \setminus \{i\}} \Psi_{i,j}. \tag{13}$$

   This choice aims at maximizing the effect of power injections at one bus on voltage variations at the other buses of the subnetwork.

In the following, the above three steps are referred to as CSA algorithm. In practice, if a subnetwork $\mathscr{G}_h$ resulting from step 1 does not contain any bus affected by voltage problems, it can be reasonably left without ESS, thus avoiding to deploy an ESS which would turn out to be unnecessary. This implies that the number m of ESS locations resulting from running the CSA algorithm with the number of clusters equal to q, satisfies in general $m \leq q$.

In practical applications, the voltage sensitivity value $\Psi_{i,j}$ in (12) is estimated numerically by evaluating the voltage variation at bus j consequent to a unit power injection at bus i, which amounts to solving a PF problem. While it is true that the absolute values of (12) do depend on the particular demand and generation profiles, their relative values (which in fact determine the outcome of the CSA algorithm) are quite invariant, being mostly determined by network topology and line admittances [5, 33]. As a consequence, the outcome of the CSA algorithm is expected to be robust to uncertainty on demand and generation profiles, and the values in (12) can be computed once, e.g., considering average demand and generation profiles at each bus.

3.2 ESS Sizing

In the sizing step of Algorithm 1, the number and the locations of the ESSs to be deployed in the network are fixed. Hence, for given m and λ, and in view of Problem 1, the ESS sizing problem reads as the minimization of the cost $C(m, \lambda, x)$ in (10) with respect to the vector of ESS sizes x only, corresponding to solve

$$\min_{x \in \mathscr{X}} c(x) + \gamma \; g(\lambda, x), \tag{14}$$

where the set $\mathscr{X}$ includes the constraints that all ESS sizes must be nonnegative. Taking into account that, at the planning stage, future realizations of demand and generation are unknown, the above problem is cast in a setting which explicitly accounts for uncertainty. The two-stage stochastic optimization framework of [30] is suitable to this aim (see Appendix 1 for the general form of a two-stage stochastic program).

In order to formulate (14) as a stochastic optimization problem, we model the daily demand and generation profiles as stochastic quantities, and introduce the random vector $\mathbf{p}$ with elements $P_i^D(t)$, $Q_i^D(t)$, $P_i^G(t)$, and $Q_i^G(t)$ for all $i \in \mathscr{L}$ and $t \in \mathscr{T}$, where $\mathscr{T} = \{1, \ldots, T\}$. Then, we let

$$g(\lambda, x) := \mathbb{E}_{\mathbf{p}}[F(x, \mathbf{p})], \tag{15}$$

where, for a given realization p of $\mathbf{p}$, $F(x, p)$ is given by

$$F(x, p) := \min_{y \in \mathscr{Y}(x,p)} f(x, y, p), \tag{16}$$

and $\mathbb{E}_{\mathbf{p}}[\cdot]$ denotes expectation with respect to the probability distribution of the random variable $\mathbf{p}$. In (16), y denotes the vector containing the values $V_i(t)$, $r_s(t)$, and $b_s(t)$ for all $i \in \mathscr{L}$, $s \in \mathscr{S}$, and $t \in \mathscr{T}$, and $\mathscr{Y}(x, p)$ is the feasible solution set for y, defined as

$$\mathscr{Y}(x, p) = \{y : (3) - (9), i \in \mathscr{L}, s \in \mathscr{S}, (i, j) \in \mathscr{E}, t \in \mathscr{T}\}. \tag{17}$$

Notice that $\mathscr{S}$ is the (fixed) set of ESS locations corresponding to the vector λ. The cost function $f(x, y, p)$ in (16) represents daily operation costs. Hence, problem (16) aims at finding the ESS control policy $\{r_s(t), b_s(t)\}_{t=1}^{T}$ which minimizes daily operation costs under the demand and generation profiles defined by p, while satisfying constraints (3)–(6) on ESS dynamics, constraints (7) and (8) on voltage and apparent power, and power balance equations (9). In turn, $g(\lambda, x)$ in (15) represents the expected daily operation costs, assuming that the ESSs with sizes defined by x and placed at the locations defined by λ are optimally operated under all possible realizations of demand and generation.

The daily operation costs $f(x, y, p)$ can be the linear combination of several terms. Some examples are as follows:

- The average total line losses per time step, i.e.,

$$f_1(x, y, p) = \sum_{t \in \mathscr{T}} \sum_{i \in \mathscr{N}} P_i(t) \Delta T / T; \tag{18}$$

- The average energy exchanged by the ESSs per time step,

$$f_2(x, y, p) = \sum_{t \in \mathscr{T}} \sum_{s \in \mathscr{S}} |r_s(t)| \Delta T / T, \tag{19}$$

  which is a measure of the overall battery usage;
- The average energy exchanged at the slack bus per time step,

$$f_3(x, y, p) = \sum_{t \in \mathscr{T}} |P_1(t)| \Delta T / T, \tag{20}$$

  which is a measure of by how much the considered distribution network impacts the external grid.

With the choices (15) and (16), the ESS sizing problem (14) reads as the following two-stage stochastic program:

$$\min_{x \in \mathscr{X}} c(x) + \gamma \, \mathbb{E}_{\mathbf{p}}[F(x, \mathbf{p})] \tag{21a}$$

$$F(x, p) := \min_{y \in \mathscr{Y}(x,p)} f(x, y, p). \tag{21b}$$

As is standard in two-stage stochastic programming, when the set $\mathscr{Y}(x, p)$ is empty, we set $F(x, p) = \infty$. Hence, the fact that the cost in (21a) is finite for a given solution x requires that, by operating the ESSs with sizes defined by x, network constraints can be satisfied under all possible realizations of demand and generation.

Solving optimization problems like (21), where uncertainty on input data is modeled by continuous, multivariate random variables, is a very hard task in general.

A typical approach adopted in the literature to tackle these problems consists of discrete approximations based on scenarios [30]. In the considered problem, a scenario is a realization p_d of the random vector $\mathbf{p}$, i.e., a particular instance of daily demand and generation profiles at network buses. Assume that a set of scenarios $\mathscr{P} = \{p_1, \dots, p_D\}$ is sampled from the probability distribution of $\mathbf{p}$,[1] and define the set of indices $\mathscr{D} = \{1, \dots, D\}$. Let $\{\pi_d^{\mathscr{P}}\}_{d \in \mathscr{D}}$ be the approximating probability mass function of $\mathscr{P}$ with respect to the probability distribution of $\mathbf{p}$. Moreover, for each scenario p_d, define a vector of unknowns y_d with the same meaning as y in (16). Then, the discrete approximation of the two-stage problem (21) reads as the following (deterministic) single-stage problem:

$$\begin{aligned} J^* = \min_{x, \{y_d\}_{d \in \mathscr{D}}} \; & c(x) + \gamma \sum_{d \in \mathscr{D}} \pi_d^{\mathscr{P}} f(x, y_d, p_d) \\ \text{s.t.} \quad & x \in \mathscr{X}, \;\; y_d \in \mathscr{Y}(x, p_d), \;\; d \in \mathscr{D}. \end{aligned} \tag{22}$$

This approximation is justified by the fact that if $\mathbf{p}$ were actually discrete valued with support $\mathscr{P}$, then (21a) and (22) would be equivalent thanks to the interchangeability principle [30]. In the following, we will denote by x^* and y_d^* the values of the unknowns x and y_d at the optimum of (22), while E_s^* is the size of the ESS at bus s corresponding to the solution x^*.

In spite of the transformation into a deterministic problem, (22) is still hard to solve for a number of reasons. First, the problem is an AC OPF, which brings the intrinsic difficulties of this class of problems (e.g., nonconvexity). Second, time-coupling constraints determined by ESS dynamics and scenario-coupling constraints represented by ESS sizes (common to all scenarios) further increase the complexity of the problem. Finally, and most importantly, a good discrete approximation requires a very large number of scenarios. This may make problem (22) practically intractable even resorting to state-of-the-art relaxation techniques for OPF problems [19, 21], due to the huge number of variables involved, asking for prohibitive memory requirements. In other words, even if computation time might be not so critical at the planning stage, the real difficulty with problem (22) would be the practical impossibility to find a solution. In order to keep the problem size affordable, scenario-based approaches are often coupled in the literature with techniques to downsize the scenario set. To this aim, clustering algorithms such as K-means and centroid-linkage clustering are adopted [4, 26]. Another option is to apply scenario reduction techniques based on the notion of probability distance [12]. However, as shown in [6], all these techniques may fail in preserving the useful information contained in the original scenario set.

[1] In practical applications, scenarios can be obtained from historical data, or generated via suitable scenario generation techniques, such as simulation of identified models [10] and methods based on the use of copula functions [28].

3.2.1 Computing Bounds and a Feasible Solution

For the reasons described above, a decomposition approach is proposed in [8] to tackle problem (22). The approach is based on solving a problem similar to (22) for each scenario p_d as

$$\begin{aligned} \min_{x,y_d} \quad & c(x) + \gamma\, f(x, y_d, p_d) \\ \text{s.t.} \quad & x \in \mathscr{X},\ \ y_d \in \mathscr{Y}(x, p_d). \end{aligned} \tag{23}$$

This amounts to solving D multi-period OPF problems, for which solution strategies based on semidefinite programming (SDP) convex relaxations can be adopted, see, e.g., [19]. Notice that solving (23) is typically affordable. Moreover, the solution of the D single-scenario problems can be parallelized, thus reducing the overall computation time. The values of x and y_d at the optimum of problem (23) are denoted by $\widetilde{x}_d$ and $\widetilde{y}_d$, while $\widetilde{E}_{s,d}$ is the size of the ESS at bus s corresponding to the vector $\widetilde{x}_d$.

As shown in [8], the pairs $(\widetilde{x}_d, \widetilde{y}_d)$ can be used to compute a lower bound J_{LB} to the optimal cost J^* of (22) as

$$J_{\text{LB}} = \sum_{d \in \mathscr{D}} \pi_d^{\mathscr{P}} \left[c(\widetilde{x}_d) + \gamma\, f(\widetilde{x}_d, \widetilde{y}_d, p_d) \right]. \tag{24}$$

Lower bounds are useful to evaluate the quality of any feasible solution of (22). The computation of J_{LB} is straightforward, because the terms $c(\widetilde{x}_d) + \gamma\, f(\widetilde{x}_d, \widetilde{y}_d, p_d)$ on the right-hand side of (24) are the optimal costs of the problems (23) that were previously solved. A less conservative lower bound for the case $\gamma = 0$ is given by

$$J_{\text{LB}}^0 = \max_{d \in \mathscr{D}} c(\widetilde{x}_d). \tag{25}$$

A feasible solution of (22), and consequently an upper bound to J^*, can also be recovered from the solutions of the single-scenario problems (23), by constructing the following vector of ESS sizes:

$$\widetilde{x} = \max_{d \in \mathscr{D}} \ \widetilde{x}_d, \tag{26}$$

where the max is to be intended component-wise. It is shown in [8] that the solution $(\widetilde{x}, \{\widetilde{y}_d\}_{d \in \mathscr{D}})$ is feasible for (22), and an upper bound to the optimal cost of (22) is given by

$$J_{\text{UB}} = c(\tilde{x}) + \gamma \sum_{d \in \mathscr{D}} \pi_d^{\mathscr{P}}\ f(\tilde{x}, \tilde{y}_d, p_d). \tag{27}$$

If the cost function $f(x, y, p)$ is a linear combination of (18)–(20), it can be shown that $f(\tilde{x}, \tilde{y}_d, p_d) = f(\widetilde{x}_d, \widetilde{y}_d, p_d)$, where $f(\widetilde{x}_d, \widetilde{y}_d, p_d)$ is known, being the daily operation costs at the optimum of (23). Hence, the computation of the upper bound does not introduce any additional burden into the procedure. In the following, we denote by $\widetilde{E}_s$ the size of the ESS at bus s corresponding to the vector $\widetilde{x}$.

3.2.2 Optimal Solution via Scenario Reduction

When the objective of ESS sizing is only to minimize the total installed ESS capacity, and therefore $\gamma = 0$ in (22), an iterative procedure is proposed in [8] for solving problem (22) at the optimum. The procedure is based on a metric for scenario reduction which exploits the structure of the multi-scenario problem, and allows one to find, provided it exists, a downsized scenario set which can replace the original set in (22), while preserving the overall optimality of the solution. In the following, we say that a vector of ESS sizes $x \in \mathcal{X}$ is feasible for scenario p_d if and only if the set $\mathcal{Y}(x, p_d)$ is nonempty, i.e., for the given scenario p_d, it is possible to control the ESSs over $\mathcal{T}$ so that all ESS and network operation constraints are satisfied. A vector $x \in \mathcal{X}$ is feasible for problem (22) if and only if it is feasible for all scenarios.

The procedure is initialized by solving problem (23) with $\gamma = 0$ for each scenario $p_d \in \mathcal{P}$. This makes it possible to order the scenarios according to the cost $c(\widetilde{x}_d)$. We denote by $\widetilde{\mathcal{D}} = \{d_1, \dots, d_D\}$, a permutation of the set $\mathcal{D}$ such that $c(\widetilde{x}_{d_\ell}) \geq c(\widetilde{x}_{d_\kappa})$ if $\ell < \kappa$. Since $c(x^*) \geq c(\widetilde{x}_d)$ for all $d \in \mathcal{D}$, intuitively, the ordering defined by $\widetilde{\mathcal{D}}$ provides an indication of the scenarios that are most critical to determine the optimal solution x^*. This idea is exploited in Algorithm 2, which summarizes the procedure for solving problem (22) at the optimum [8].

Algorithm 2 Scenario reduction

```
1: Set $\ell_0 = 0$ and $\kappa = 1$
2: Select $\ell_\kappa$ such that $\ell_{\kappa-1} < \ell_\kappa \leq D$
3: $x_\kappa^* \leftarrow$ Solve problem (28) with $\mathcal{D}_\kappa = \{d_1, \dots, d_{\ell_\kappa}\}$
4: Check feasibility of $x_\kappa^*$ for all scenarios $p_d$, $d \in \mathcal{D} \setminus \mathcal{D}_\kappa$
5: if (feasibility == true) then
6:     return $x^* = x_\kappa^*$
7: else
8:     $\kappa \leftarrow \kappa + 1$
9:     goto 2
10: end if
```

At each iteration κ, the set of indices $\mathcal{D}_\kappa$ represents the ℓ_κ scenarios p_d with the largest total ESS capacities $c(\widetilde{x}_d)$. For this set of ℓ_κ scenarios, the following problem is solved:

$$\begin{aligned} \min_{x, \{y_d\}_{d \in \mathcal{D}_\kappa}} \quad & c(x) \\ \text{s.t.} \quad & x \in \mathcal{X},\ y_d \in \mathcal{Y}(x, p_d),\ d \in \mathcal{D}_\kappa . \end{aligned} \tag{28}$$

Notice that (28) coincides with (22) for $\gamma = 0$, considering a downsized scenario set in place of $\mathscr{P}$. The optimal value of x in (28) is denoted by x^*_κ. It is proven in [8] that if the vector x^*_κ is feasible for all scenarios p_d such that $d \in \mathscr{D} \setminus \mathscr{D}_\kappa$, then it is optimal for (22). Otherwise, the algorithm is iterated by increasing the cardinality ℓ_κ of the downsized scenario set. A rule of thumb to start the ℓ_κ-sequence is to set $\ell_1 = 1$. Since the vector of ESS sizes $x^*_1 = \widetilde{x}_{d_1}$ is already available for scenario p_{d_1}, one needs just to check the feasibility of this solution for all other scenarios. If the feasibility test fails, one can choose ℓ_2 at the knee of the curve $c(\widetilde{x}_{d_\ell})$, $\ell = 1, \dots, D$, compatibly with the size of the largest instance of problem (22) which can be solved.

Remark 1 Verifying the feasibility of a given vector of ESS sizes x for scenario p_d amounts to check whether the set $\mathscr{Y}(x, p_d)$ is nonempty. This problem can be formulated as an OPF in the unknown y_d with feasible solution set $\mathscr{Y}(x, p_d)$, regardless of the considered objective function. This test can often be avoided by exploiting the vector $\widetilde{x}_d$. Indeed, if the condition $x \geq \widetilde{x}_d$ is satisfied, where the inequality has to be intended component-wise, x is feasible for scenario p_d. If $x \geq \widetilde{x}_d$, x is said to dominate $\widetilde{x}_d$.

4 Numerical Results

The procedure for ESS allocation described in Sect. 3 is demonstrated using the topology of the IEEE 37-bus test network [18], which is shown in Fig. 1. The lengths of the lines correspond to the original test network, while line admittances are typical of low-voltage feeders. The network hosts 36 loads and 4 wind power generators, whose installed power is, respectively, 13 kW at bus 5, 11 kW at bus 11, and 10 kW at buses 17 and 35.

As far as the sizing problem is concerned, the bound $\overline{S}_{ij}$ in (8) is set to 50 kVA for all the lines, while 10% tolerance around the nominal voltage is allowed at each bus, i.e., $\underline{v}_i = 0.9$ pu and $\overline{v}_i = 1.1$ pu in (7). Vectors Γ_s, Υ_s, and Ξ_s in (4) are defined as

$$\Gamma_s = \begin{bmatrix} 1 \\ -1 \\ 0 \\ 0 \end{bmatrix}, \quad \Upsilon_s = \begin{bmatrix} 0 \\ 0 \\ 1 \\ -1 \end{bmatrix}, \quad \Xi_s = \frac{\rho_s}{\sqrt{2}} \begin{bmatrix} 1 \\ 1 \\ 1 \\ 1 \end{bmatrix}, \tag{29}$$

corresponding to the inner approximation of the constraint $r_s(t)^2 + b_s(t)^2 \leq \left(\rho_s E_s\right)^2$ on the apparent power exchanged by the ESS at time t, with a square with sides parallel to the coordinate axes. It is assumed that the choice of the ESSs is within a family (e.g., lithium-ion based) characterized by a coefficient $\rho_s = 1.5$ kVA/kWh in (29). Since the analysis presented in this section does not depend on the charging and discharging efficiencies of storage devices, η_s^c and η_s^d are set to 1 in (5). Consequently, the dynamics (5) of $e_s(t)$ simplifies as

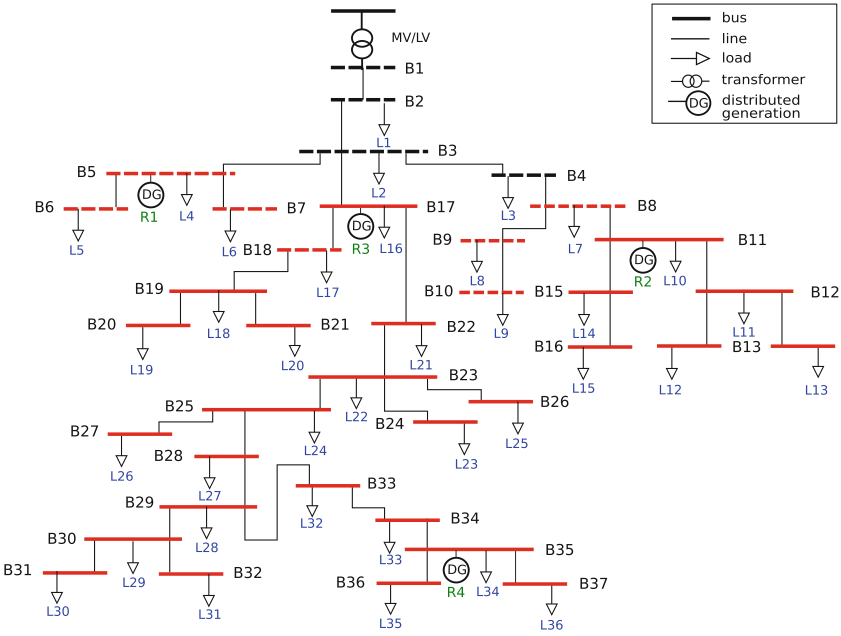


Fig. 1 IEEE 37-bus test network. Red bars represent the candidate buses, while dashed bars denote buses not affected by voltage problems

$$e_s(t) = e_s(t-1) + r_s(t)\Delta T, \tag{30}$$

where the initial energy level $e_s(0)$ is set to zero.

OPF problems are tackled via the SDP convex relaxation of [19]. SDP programs are implemented through the CVX modeling package for convex optimization [16], and solved via the SeDuMi solver [31] on an Intel Xeon 2.4 GHz CPU with 32 GB RAM. As long as it can be solved, the computation time of the SDP relaxation of the multi-scenario problem (22) grows almost quadratically with the cardinality of the set $\mathscr{D}$ [7]. A single-scenario problem is solved in about 2 min. For all examples discussed in this section, a feasible solution to the original problem (22), (23), or (28) is reconstructed from the optimal solution of the corresponding relaxed program by solving a series of PFs.

The next sections illustrate different applications of the procedures for ESS siting and sizing described in this chapter. Section 4.2 contains the results of the ESS siting and sizing procedures for two ESSs to be deployed. In Sect. 4.3, the whole ESS allocation procedure is presented, providing the optimal number, locations, and sizes of the ESSs for the considered network. In order to deal with uncertainty in the considered decision problem, Sect. 4.1 shows how a large number of scenarios of demand and DG is generated.

4.1 Scenario Generation

In order to apply the scenario-based approach described in Sect. 3.2 and test, in particular, the scenario reduction procedure of Sect. 3.2.2, a large number of scenarios of demand and DG is generated with the copula-based method described in Appendix 2. To this purpose, a scenario of demand and DG is considered as a set of daily profiles, one for each load and each generator, sampled with time step $\Delta T = 60$ min.

Concerning load profiles, it is assumed that there is no correlation between loads at different buses. For each load, hourly sampled real data of demand are used to estimate the empirical marginal distributions for each hour of the day and the correlation matrix. Then, $D = 1000$ daily demand profiles are independently generated for each of the 36 loads hosted in the network using the copula-based method. It is stressed that, for a single load, the generated scenarios take into account temporal correlations.

For wind power generation, a common approach in the literature is to focus on wind speed profiles, which are then converted into power (or energy) through the power curve of the wind turbine [23]. This is the approach taken in this section. Moreover, different from the demand scenarios, it is assumed that the four wind turbines in the network operate under strongly correlated wind conditions. Consequently, hourly sampled historical wind speed data at a single location are used to estimate the empirical marginal distributions for each hour of the day and the correlation matrix. Then, D daily wind speed profiles are generated using the copula-based method, and finally these profiles are converted into D wind energy generation profiles for each of the four wind turbines using the corresponding power curve. We recall that by plotting the hourly energy w generated by a wind turbine versus the hourly average wind speed v, the plotted points can be typically well approximated by a sigmoid function saturated below at 0 and above at the installed power $\overline{E}$ of the wind turbine. This leads to the model given below:

$$w = \begin{cases} \min\{\max\{0, E_\sigma(v)\}, \overline{E}\} & \text{if } v \leq v_{out} \\ 0 & \text{otherwise,} \end{cases} \tag{31}$$

where v_{out} is the cut-out wind speed of the wind turbine and $E_\sigma(v)$ is a suitable sigmoid function. There exist several mathematical expressions for sigmoid functions. The one considered in this work has the following form [14]:

$$E_\sigma(v) = b + (a - b)\left(1 + e^{(v - v_0)/c}\right)^d, \tag{32}$$

where $a > 0$, $b < 0$, $c < 0$, $d < 0$, and $v_0 > 0$ are the model parameters estimated from recorded measurements by solving a nonlinear least squares problem.

In the following, the generated scenario set is denoted by $\mathscr{P} = \{p_1, \ldots, p_{1000}\}$. In all solved problems, scenarios are considered equiprobable. With the choice made above for the voltage bounds in (7), the generated demand and DG profiles are such that, in the absence of ESSs, the network occasionally experiences over- and/or undervoltages.

Table 1 Results of the application of the CSA algorithm for $q = 4$

$Subnetwork$	$\mathscr{L}_h$	$\mathscr{C}_h$	Ω_h	s_h
$\mathscr{G}_1$	$\{2\}$	$\emptyset$	$\emptyset$	–
$\mathscr{G}_2$	$\{3, 4, 8, \ldots, 16\}$	$\{10, 11, 13, 14, 16\}$	$\{8, \ldots, 16\}$	11
$\mathscr{G}_3$	$\{5, 6, 7\}$	$\{5, 6, 7\}$	$\emptyset$	–
$\mathscr{G}_4$	$\{17, \ldots, 37\}$	$\{17, 20, 21, 24, 26, 27, 31, 32, 35, 36, 37\}$	$\{17, \ldots, 37\}$	22

4.2 Siting and Sizing for a Fixed Number of ESSs

This section illustrates the ESS siting and sizing procedures of Sects. 3.1 and 3.2. The number of ESSs to be deployed is fixed to $m = 2$.

For the considered 37-bus network, the voltage sensitivity values $\Psi_{i,j}$ defined in (12) are computed numerically. This is done by solving a PF problem for each pair (i, j), in order to evaluate the voltage variation at bus j consequent to a unit power injection at bus i. Since the relative values of the $\Psi_{i,j}$'s are mostly determined by the network topology and admittances of the lines (see the discussion at the end of Sect. 3.1), the $\Psi_{i,j}$'s are computed once, by considering the average demand and DG profiles over the scenario set $\mathscr{P}$. A pseudocolor plot of the values $\Psi_{i,j}$ is shown in Fig. 2. With this representation, it is possible to visualize the effect that a power injection at one bus has on voltage at another bus: the warmer the color of a cell of the pseudocolor plot, the stronger the coupling of the two buses corresponding to this cell. The values $\Psi_{i,j}$ are used in the CSA algorithm of Sect. 3.1. For a fixed number of clusters q, the CSA algorithm returns a suitable number of storage units $m \leq q$ to be installed, and the corresponding locations. The smallest number of clusters for which the CSA algorithm returns $m = 2$ is $q = 4$. Table 1 shows the composition of the four clusters, the corresponding sets of critical and candidate buses, and the locations of the two ESSs determined by the algorithm. Candidate buses are highlighted in red in Fig. 1, showing also the buses not affected by voltage problems in the absence of ESSs. This helps clarify that the set of candidate buses Ω_3 is empty (though in principle it should contain buses 5, 6, and 7) because the cluster $\mathscr{L}_3$ is formed only by buses not affected by voltage problems. As a consequence, the subnetwork $\mathscr{G}_3$ is left without ESS. On the other hand, the subnetwork $\mathscr{G}_1$ is left without ESS because it does not contain critical (and hence also candidate) buses. The CSA algorithm places two ESSs at buses 11 and 22, selected from the subnetworks $\mathscr{G}_2$ and $\mathscr{G}_4$ through the criterion defined by (13).

For given ESS locations returned by the CSA algorithm, the ESS sizing problem is tackled with $\gamma = 0$. Since out-of-memory issues hinder solving problem (22) directly for the whole scenario set $\mathscr{P}$ composed of 1000 scenarios, Algorithm 2 is applied to compute the optimal solution. At initialization, the single-scenario problem (23) is solved for each scenario. The corresponding vectors of optimal ESS sizes $\widetilde{x}_d = \big(\widetilde{E}_{11,d}, \widetilde{E}_{22,d}\big), d = 1, \ldots, 1000$ are shown in the scatter plot of Fig. 3. Then, the scenarios are sorted according to decreasing values of $c(\widetilde{x}_d)$, returning the ordered set

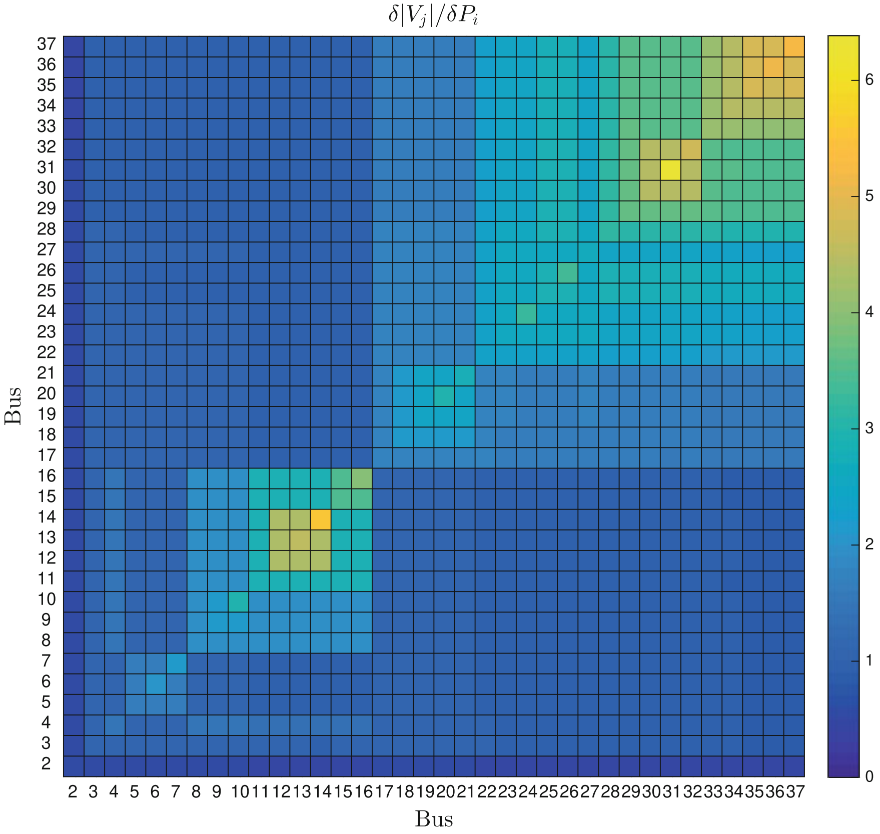


Fig. 2 Graphical representation of the voltage sensitivity matrix for the IEEE 37-bus test network

$\widetilde{\mathscr{D}} = \{d_1, d_2, d_3, \ldots, d_{1000}\}$. In the first iteration of Algorithm 2, $\mathscr{D}_1 = \{d_1\}$ is considered. Hence, $x_1^* = \widetilde{x}_{d_1}$, and one has to check whether the solution x_1^* is feasible for all the scenarios which correspond to solutions $\widetilde{x}_d$ not dominated by x_1^*, represented by black circles in Fig. 3 (red circles represent solutions $\widetilde{x}_d$ dominated by x_1^*). The feasibility check fails for ten scenarios. From the second to the ninth iteration of the algorithm ($\mathscr{D}_2 = \{d_1, d_2\}, \ldots, \mathscr{D}_9 = \{d_1, \ldots, d_9\}$), problem (28) is solved, returning in all iterations the same solutions $x_2^* = x_3^* = \cdots = x_9^*$. The feasibility check of these solutions fails for nine scenarios. From the 10th to the 27th iteration of the algorithm ($\mathscr{D}_{10} = \{d_1, \ldots, d_{10}\}, \ldots, \mathscr{D}_{27} = \{d_1, \ldots, d_{27}\}$), problem (28) is solved, returning in all iterations the same solutions $x_{10}^* = x_{11}^* = \cdots = x_{27}^*$, though different from the previous ones. The feasibility check of the new solutions fails for eight scenarios. In the 28th iteration, scenario $p_{d_{28}}$ is added. According to the ordered set $\widetilde{\mathscr{D}}$, this is the first scenario which requires a significant value of the ESS capacity installed at bus 11. Problem (28) is solved with $\mathscr{D}_{28} = \{d_1, \ldots, d_{28}\}$, returning the solution x_{28}^*, which is feasible for all scenarios except $p_{d_{33}}$. This solution is the same as those

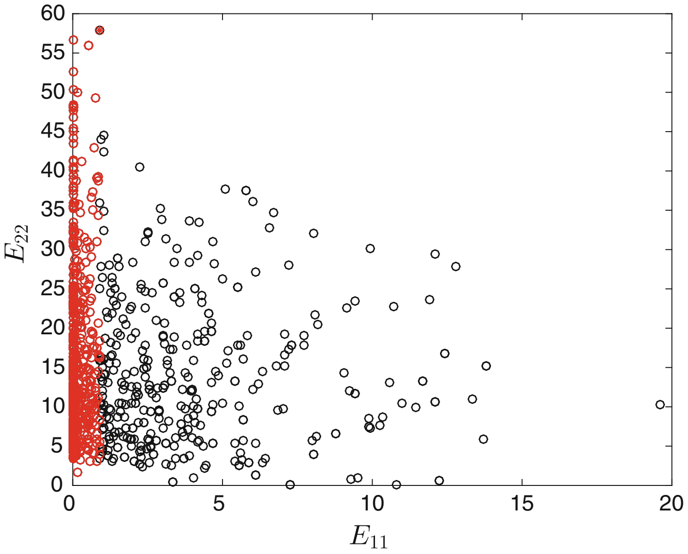


Fig. 3 The circles represent the solutions $\widetilde{x}_d$ of the single-scenario problem (23) for each of the considered 1000 scenarios. The red star represents $x_1^* = \widetilde{x}_{d_1}$, while red circles correspond to solutions $\widetilde{x}_d$ dominated by x_1^*

Table 2 Summary of the iterations of Algorithm 2. ESS sizes are expressed in kWh

κ	$E^*_{11,\kappa}$	$E^*_{22,\kappa}$	$c(x^*_\kappa)$
1	0.89	57.95	58.84
$2, \ldots, 9$	0.89	58.22	59.11
$10, \ldots, 27$	1.07	58.78	59.85
$28, \ldots, 32$	12.02	52.65	64.67
33	12.75	52.12	64.87

found solving problem (28) at iterations $\kappa = 29, \ldots, 32$ of the algorithm. The final step of the algorithm is reached by solving problem (28) with $\mathscr{D}_{33} = \{d_1, \ldots, d_{33}\}$. This iteration returns the solution x^*_{33}, which is feasible for all the scenarios. Hence, Algorithm 2 terminates with the optimal solution $x^* = x^*_{33}$ for problem (22). The 33 iterations of the algorithm are summarized in Table 2, where $E^*_{s,\kappa}$ is the ESS size at bus s corresponding to the solution x^*_κ. Notice that the algorithm generates a sequence of lower bounds $c(x^*_\kappa)$ converging to the optimal cost $c(x^*)$. A graphical representation of the iterations of Algorithm 2 is provided in Fig. 4.

Remark 2 In the proposed application of Algorithm 2, only one scenario is added at each iteration. This is done for illustrative purposes. In practice, one may speed up the convergence of the algorithm by adding scenarios according to suitable *ad hoc* rules. For instance, since the feasibility check fails for ten scenarios in the first

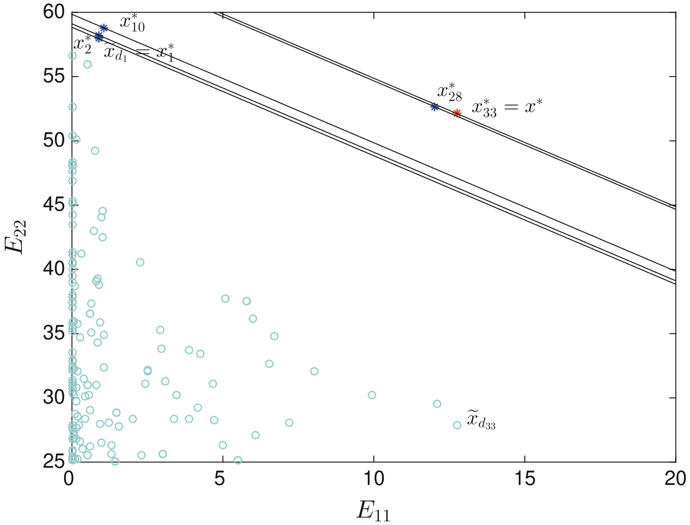


Fig. 4 Graphical representation of the ESS sizing procedure of Sect. 3.2. The solution $\widetilde{x}_{d_{33}}$ refers to scenario $p_{d_{33}}$. The oblique lines represent level curves of the cost function $c(x)$

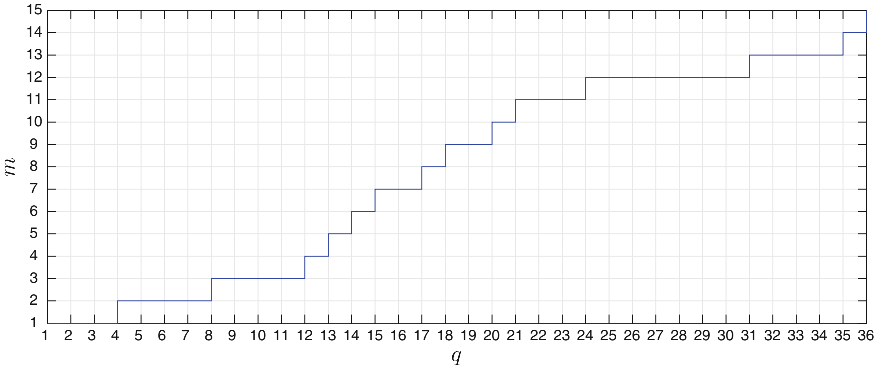


Fig. 5 Number of ESSs m returned by the CSA algorithm as a function of the number of clusters q for the IEEE 37-bus test network

iteration of the algorithm, one could argue that those scenarios are critical, and select ℓ_2 in Algorithm 2 so as to include them in the scenario set $\{p_{d_1}, \ldots, p_{d_{\ell_2}}\}$ considered in the second iteration.

4.3 *Optimal ESS Allocation*

In this section, the whole ESS allocation procedure is presented, providing the optimal number, locations, and sizes of the ESSs for the considered test network.

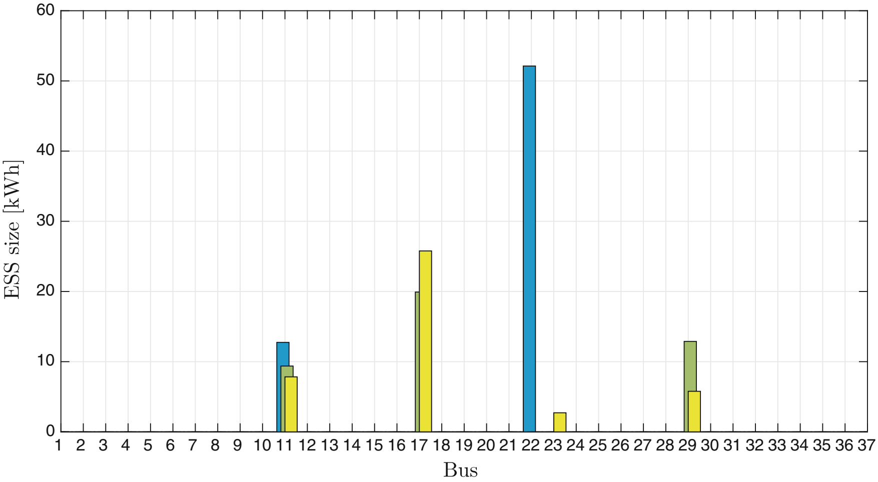


Fig. 6 Sizes and locations of the ESSs allocated for $m = 2$ (blue), $m = 3$ (green), and $m = 4$ (yellow)

Following Algorithm 1, the CSA algorithm is repeated for all possible values of q, ranging from 1 to 36. Figure 5 shows the number of ESSs m as a function of the number of clusters q. Notice that m is typically strictly less than q, and in the case $q = 36$ (each bus forming an individual cluster) m corresponds to the number of critical buses affected by voltage problems, i.e., $m = 15$. For a given value of q, the CSA algorithm returns not only a suitable number m of ESSs but also the vector of their locations $\lambda^{(m)}$. Optimal sizing of the ESSs placed according to $\lambda^{(m)}$ is tackled by solving problem (22) with $\gamma = 0$. This is done by applying Algorithm 2, because problem (22) cannot be solved directly for 1000 scenarios due to out-of-memory issues, as already discussed in Sect. 4.2. The corresponding vector of optimal ESS sizes is denoted by $x^{(m)}$, while $J^{(m)}$ denotes the optimal cost of problem (22), which, with the choice $\gamma = 0$, coincides with $c(x^{(m)})$. It is stressed that $x^{(m)}$ and $J^{(m)}$ can be computed only for $m \geq 2$. Indeed, for $m = 1$, the single-scenario problem (23) is infeasible for 73 scenarios, which implies that problem (22) is also infeasible. For $m = 2$, the iterations of Algorithm 2 were described in Sect. 4.2. For $m \geq 3$, Algorithm 2 always stops at the first iteration, since the solution $x_1^* = \widetilde{x}_{d_1}$ is feasible for the whole scenario set. Figure 6 shows the sizes and the locations of the ESSs allocated for $m = 2, 3, 4$.

For a given number m of ESSs, the total cost (10) is computed as

$$C(m, \lambda^{(m)}, x^{(m)}) = \rho m + \varsigma J^{(m)}, \tag{33}$$

where $\rho = 10$ k€ and $\varsigma = 575$ €/kWh. The total cost (33) as a function of m is shown in Fig. 7a. The minimum value of this cost is 54.26 k€, which corresponds to $m^* = 3$ ESSs deployed at buses 11, 17, and 29 (see Fig. 6).

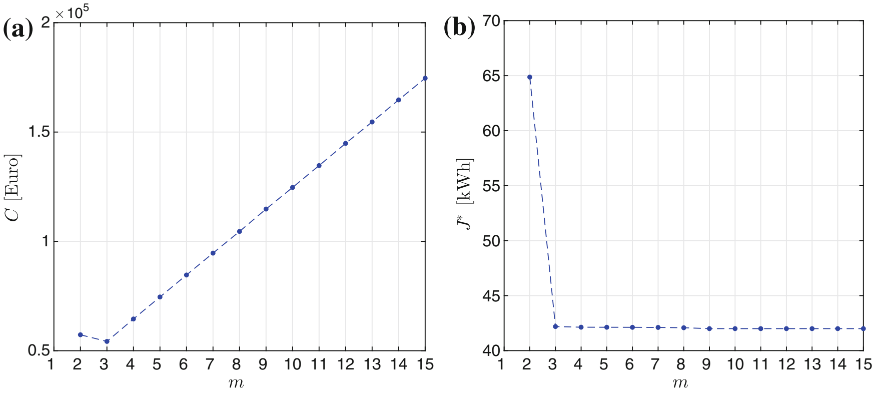


Fig. 7 Results of the proposed ESS allocation procedure for the IEEE 37-bus test network **a**: Total cost in (33) as a function of m, **b**: Term $J^{(m)}$ in (33) as a function of m. With the choice $\gamma = 0$, $J^{(m)}$ coincides with $c(x^{(m)})$

Figure 7b shows the term $J^{(m)}$ of (33) as a function of m. It can be observed that, when increasing the number of ESSs, this term tends to a constant value. This explains why the total cost in Fig. 7a grows linearly with m for $m \geq 4$.

Remark 3 In the proposed application of Algorithm 1, where $\gamma = 0$, the term $J^{(m)}$ in (33) can be computed thanks to Algorithm 2. When $\gamma > 0$, Algorithm 2 is not applicable, but it is possible to derive lower and upper bounds to the term $J^{(m)}$, as described in Sect. 3.2.1. These can be translated into bounds to the total cost in (33), which in turn may be used to select a suitable value of m, following a reasoning similar to that in Fig. 7a.

5 Conclusions and Future Work

In this chapter, we addressed the allocation problem of energy storage systems for voltage support in a distribution electricity network. As far as the sizing problem is concerned, a stochastic programming approach based on scenarios was adopted to accommodate uncertainty on future demand and generation profiles. Since the resulting stochastic optimization problem becomes rapidly intractable as the number of scenarios grows, an iterative procedure based on a scenario reduction technique was presented. This technique, exploiting the structure of the problem to be solved, generates a sequence of approximating optimization problems, whose solutions converge to the optimal one. With reference to the siting problem, which is combinatorial in the number of buses of the network and the number of energy storage systems to be deployed, a heuristic procedure was adopted, which exploits the voltage sensitivity matrix of the grid.

The overall procedure was tested on the topology of the IEEE 37-bus test network. The obtained numerical results show a good performance of the iterative sizing algorithm, with the solutions of the approximating problems converging rapidly to the optimal solution of the multi-scenario problem. This confirms the validity of the scenario reduction technique adopted, which extracts sequentially the most critical scenarios for the problem at hand from the rich set describing the statistics of the uncertain variables involved.

Ongoing work is focused on extending the proposed scenario-based technique to tackle grid operation problems in the presence of uncertainties modeled by probabilistic constraints.

Appendix 1

A two-stage stochastic program is an optimization problem written in the form

$$\min_{x \in \mathscr{X}} \mathbb{E}_{\mathbf{p}}[W(x, \mathbf{p})] \tag{34a}$$

$$W(x, p) := \min_{y \in \mathscr{Y}(x,p)} w(x, y, p), \tag{34b}$$

where x is the vector of decision variables of the first-stage problem (34a), $\mathscr{X}$ is the feasible solution set for x, and $\mathbb{E}_{\mathbf{p}}[W(x, \mathbf{p})]$ is the expectation (taken with respect to the probability distribution of the random variable $\mathbf{p}$) of the optimal value $W(x, \mathbf{p})$ of the second-stage problem (34b). In (34b), $\mathscr{Y}(x, p)$ is the feasible solution set for the second-stage decision variables y, and $w(x, y, p)$ is the cost function. Notice that for a given instance x of the first-stage variables and a given realization p of the random vector $\mathbf{p}$, (34b) is a deterministic problem.

Appendix 2

This appendix briefly describes the copula-based method to generate samples of correlated variables. Consider a multivariate random variable $Z = \{Z_1, \dots, Z_r\}$, and let the marginal cumulative distribution functions $F_{Z_i}(\cdot)$ and the correlation matrix R_Z be known. Assuming a Gaussian copula, the method to generate samples of Z works as follows:

1: Generate a sample $g = (g_1, \dots, g_r)$ from a multivariate normal distribution with zero mean and covariance matrix equal to R_Z.

2: Transform each entry g_i of g through the standard normal cumulative distribution function $\phi(\cdot)$

$$u_i = \phi(g_i).$$

3: Apply to each entry u_i, the inverse of $F_{Z_i}(\cdot)$

$$z_i = F_{Z_i}^{-1}(u_i).$$

The samples $z = (z_1, \dots, z_r)$ generated through the above procedure are statistically characterized by the right marginal cumulative distribution functions and correlation matrix, while the joint cumulative distribution functions of the samples actually depend on the choice of the copula function.

References

1. Aneke M, Wang M (2016) Energy storage technologies and real life applications - a state of the art review. Applied Energy 179:350–377
2. Awad ASA, EL-Fouly THM, Salama MMA (2017) Optimal ESS allocation for benefit maximization in distribution networks. IEEE Transactions on Smart Grid 8(4):1668–1678
3. Babacan O, Torre W, Kleissl J (2017) Siting and sizing of distributed energy storage to mitigate voltage impact by solar PV in distribution systems. Solar Energy 146:199–208
4. Baker K, Hug G, Li X (2017) Energy storage sizing taking into account forecast uncertainties and receding horizon operation. IEEE Transactions on Sustainable Energy 8(1):331–340
5. Brenna M, De Berardinis E, Delli Carpini L, Foiadelli F, Paulon P, Petroni P, Sapienza G, Scrosati G, Zaninelli D (2013) Automatic distributed voltage control algorithm in smart grids applications. IEEE Transactions on Smart Grid 4(2):877–885
6. Bucciarelli M, Giannitrapani A, Paoletti S, Vicino A, Zarrilli D (2016) Energy storage sizing for voltage control in LV networks under uncertainty on PV generation. In: Proc. of 2nd Int. Forum on Research and Technologies for Society and Industry, Bologna, Italy, pp 1–6
7. Bucciarelli M, Giannitrapani A, Paoletti S, Vicino A, Zarrilli D (2017) Sizing of energy storage systems considering uncertainty on demand and generation. IFAC-PapersOnLine 50(1):8861–8866
8. Bucciarelli M, Paoletti S, Vicino A (2018) Optimal sizing of energy storage systems under uncertain demand and generation. Applied Energy 225:611–621
9. Chu S, Majumdar A (2012) Opportunities and challenges for a sustainable energy future. Nature 488(7411):294–303
10. Conejo AJ, Carrión M, Morales JM (2010) Decision Making Under Uncertainty in Electricity Markets. Int. Series in Operations Research & Management Science, Springer, New York, NY
11. Dall'Anese E, Baker K, Summers T (2017) Chance-constrained AC optimal power flow for distribution systems with renewables. IEEE Transactions on Power Systems 32(5):3427–3438
12. Dupačová J, Gröwe-Kuska N, Römisch W (2003) Scenario reduction in stochastic programming. Mathematical Programming 95(3):493–511
13. Ghofrani M, Arabali A, Etezadi-Amoli M, Fadali MS (2013) A framework for optimal placement of energy storage units within a power system with high wind penetration. IEEE Transactions on Sustainable Energy 4(2):434–442
14. Giannitrapani A, Paoletti S, Vicino A, Zarrilli D (2016) Bidding wind energy exploiting wind speed forecasts. IEEE Transactions on Power Systems 31(4):2647–2656
15. Giannitrapani A, Paoletti S, Vicino A, Zarrilli D (2017) Optimal allocation of energy storage systems for voltage control in LV distribution networks. IEEE Transactions on Smart Grid 8(6):2859–2870
16. Grant M, Boyd S (2017) CVX: Matlab software for disciplined convex programming, version 2.1. [Online]. Available: http://cvxr.com/cvx (accessed Jun. 22, 2018)
17. Hespanha JP (2004) An efficient MATLAB algorithm for graph partitioning. [Online]. Available: http://www.ece.ucsb.edu/~hespanha/published/tr-ell-gp.pdf (accessed Jun. 22, 2018

18. Kersting W (1991) Radial distribution test feeders. IEEE Transactions on Power Systems 6(3):975–985
19. Lavaei J, Low SH (2012) Zero duality gap in optimal power flow problem. IEEE Transactions on Power Systems 27(1):92–107
20. Losi A, Mancarella P, Vicino A (2015) Integration of Demand Response into the Electricity Chain: Challenges, Opportunities and Smart Grid Solutions. Electrical Engineering Series, Iste-Wiley
21. Low S (2014) Convex relaxation of optimal power flow – Part I: Formulations and equivalence. IEEE Transactions on Control of Network Systems 1(1):15–27
22. Luo X, Wang J, Dooner M, Clarke J (2015) Overview of current development in electrical energy storage technologies and the application potential in power system operation. Applied Energy 137:511–536
23. Ma L, Luan S, Jiang C, Liu H, Zhang Y (2009) A review on the forecasting of wind speed and generated power. Renewable and Sustainable Energy Reviews 13(4):915–920
24. Momoh JA (2009) Electric Power System Applications of Optimization, 2nd edn. CRC Press, Boca Raton, FL
25. Nascimento MCV, de Carvalho ACPLF (2011) Spectral methods for graph clustering – A survey. European Journal of Operational Research 211(2):221–231
26. Nick M, Cherkaoui R, Paolone M (2014) Optimal allocation of dispersed energy storage systems in active distribution networks for energy balance and grid support. IEEE Transactions on Power Systems 29(5):2300–2310
27. Nick M, Cherkaoui R, Paolone M (2015) Optimal siting and sizing of distributed energy storage systems via alternating direction method of multipliers. International Journal of Electrical Power & Energy Systems 72:33–39
28. Papaefthymiou G, Kurowicka D (2009) Using copulas for modeling stochastic dependence in power system uncertainty analysis. IEEE Transactions on Power Systems 24(1):40–49
29. Sardi J, Mithulananthan N, Gallagher M, Hung DQ (2017) Multiple community energy storage planning in distribution networks using a cost-benefit analysis. Applied Energy 190:453–463
30. Shapiro A, Dentcheva D, Ruszczyński A (2014) Lectures on Stochastic Programming: Modeling and Theory, 2nd edn. MOS-SIAM Series on Optimization, Society for Industrial and Applied Mathematics, Philadelphia, PA, USA
31. Sturm J (1999) Using SeDuMi 1.02, A Matlab toolbox for optimization over symmetric cones. Optimization Methods & Software 11(1-4):625–653
32. Thrampoulidis C, Bose S, Hassibi B (2016) Optimal placement of distributed energy storage in power networks. IEEE Transactions on Automatic Control 61(2):416–429
33. Wang L, Liang DH, Crossland AF, Taylor PC, Jones D, Wade NS (2015) Coordination of multiple energy storage units in a low-voltage distribution network. IEEE Transactions on Smart Grid 6(6):2906–2918
34. Wang P, Liang DH, Yi J, Lyons PF, Davison PJ, Taylor PC (2014) Integrating electrical energy storage into coordinated voltage control schemes for distribution networks. IEEE Transactions on Smart Grid 5(2):1018–1032
35. Zhao H, Wua Q, Huc S, Xu H, Rasmussen CN (2015) Review of energy storage system for wind power integration support. Applied Energy 137:545–553
36. Zidar M, Georgilakis PS, Hatziargyriou ND, Capuder T, Škrlec D (2016) Review of energy storage allocation in power distribution networks: applications, methods and future research. IET Generation, Transmission & Distribution 10(3):645–652

A Data-Driven Basis Function Approach in Nonparametric Nonlinear System Identification

Er-Wei Bai and Changming Cheng

In memory of Dr Roberto Tempo.

Abstract In this chapter, a data-driven orthogonal basis function approach is proposed for nonparametric FIR nonlinear system identification. The basis functions are not fixed a priori and match the structure of the unknown system automatically. This eliminates the problem of blindly choosing the basis functions without a priori structural information. Further, based on the proposed basis functions, approaches are proposed for model order determination and regressor selection along with their theoretical justifications. Both random inputs and deterministic inputs are considered.

1 Introduction

System identification is often the first and critical step in system analysis, design, simulation, and control. In the literature, there exist a huge number of papers as well as various well-developed algorithms for linear system identification [11, 18, 28]. Despite a long history and practical demands, nonlinear system identification is far from mature both in theory and in practice [15, 21, 27, 29]. Because the structure of nonlinear systems is so rich, it is not expected that a single method could be effectively applied to all nonlinear systems. Instead, various identification methods have to be developed for different classes of nonlinear systems and for different intended purposes.

E.-W. Bai (✉) · C. Cheng
Department of Electrical and Computer Engineering, University of Iowa,
Iowa City, IA 52242, USA
e-mail: er-wei-bai@uiowa.edu

© Springer Nature Switzerland AG 2018

T. Başar (ed.), *Uncertainty in Complex Networked Systems*,
Systems & Control: Foundations & Applications,
https://doi.org/10.1007/978-3-030-04630-9_10

Roughly speaking, nonlinear system identification can be divided into two categories depending on available a priori information on the structure of the unknown system. If the structure of the unknown system is available a priori, the identification problem is reduced to a parameter estimation problem, essentially a nonlinear minimization problem. Issues are how to find a minimum and if the obtained minimum is a global minimum. The other category is that no a priori information is available on the structure. This is a much harder problem. Traditional ways to approach this problem are the Volterra and Wiener series representations [25]. They are elegant in theory but applications are often limited. For the Volterra series, its application is limited to very low-order kernels because the number of unknown parameters to be estimated increases exponentially. Further, identification has to be repeated every time when an additional kernel is deemed necessary and is added. For the Wiener series, the input is usually assumed to be Gaussian. For both the Volterra series and the Wiener series, the basic idea is a multivariable polynomial approximation of the unknown system and thus, a very high-order model is needed to be able to approximate the true but unknown nonlinear system. This makes them practically intractable unless the unknown system is close to a polynomial of low order. To overcome this problem, a fixed basis function approach developed for linear systems [23, 31] has been investigated and applied for nonlinear system identification with some success [10, 16, 30]. Typical basis functions are Fourier series, polynomials, and some orthogonal functions. In particular, orthogonal functions are very attractive because no previously obtained terms have to be reestimated when an additional term is added. Only the added term needs to be estimated. Clearly, success of the orthogonal basis function approach relies on the fact that a nonparametric nonlinear identification problem is reduced to a parametric parameter estimation problem and moreover, estimations of each term are separable in some sense. On the other hand, however, its advantage is also its weakness. Performance of an orthogonal basis function approach, like any basis function approach, depends on whether the chosen basis functions resemble the structure of the unknown nonlinear system. Without enough a priori information on the structure, a fixed basis function approach often requires a large number of terms to be able to reasonably approximate the true but unknown nonlinear system which has a considerable negative effect on the identification performance. Some ideas, e.g., tunable basis functions, are proposed in the literature including wavelets, neural network, fuzzy, etc [14, 33, 34]. Even with these tunable basis functions, adequate a priori information on the structure is still needed so that the tunable basis functions are rich enough to capture the unknown system. There is an additional difficulty with such tunable basis function approaches, i.e., minimization could be trapped in a local minimum.

In this work, we propose a data-driven basis function approach to nonlinear system identification. The basis functions are not fixed but are data generated as a part of identification. The basis functions are chosen as a result of identification and automatically match the structure of the unknown nonlinear system. This eliminates the problem of blindly guessing basis functions without knowing the structure of the unknown nonlinear system. Further, the chosen basis functions are orthogonal and when it is determined that an additional term is needed, all the previously calculated terms remain unchanged and only the added term has to be identified. This is

particularly useful since the order and the structure of the nonlinear system are unknown and have to be determined as a part of identification.

The main contribution is a framework that uses the data-driven orthogonal basis functions for nonparametric nonlinear system identification. The chosen orthogonal functions always match the system even when the system is unknown and very little a priori information on the structure of the unknown system is assumed. This is different from the existing literature where a fixed basis function is used for system identification. The work is motivated by [2, 26] though the driving force is completely different. In addition, approaches are proposed for model order determination and regressor selection. The first one is the combined residual analysis and modified Box–Pierce hypothesis test approach. It is known in the literature that the popular Box–Pierce test extensively used in linear identification [18, 29] is in general invalid for nonlinear identification and a modified Box–Pierce test is proposed in this work in the context of nonlinear system identification. The second approach is the relative and cumulative contribution approach. The approach utilizes the orthogonal properties of the basis functions and is simple and effective. To present the material without interruption, all the proofs are provided in Appendix.

2 System and Orthogonal Basis Functions

Consider a general nonparametric nonlinear finite impulse response (FIR) system

$$\begin{aligned} y[k] &= f(u[k-1], u[k-2], ..., u[k-n]) + v[k] \\ &= \bar{c} + \sum_{j=1}^{n} \bar{f}_j(u[k-j]) + \sum_{1\le j_1<j_2\le n} \bar{f}_{j_1 j_2}(u[k-j_1], u[k-j_2]) + ... \\ &+ \sum_{1\le j_1<j_2<...<j_m\le n} \bar{f}_{j_1 j_2 ... j_m}(u[k-j_1], u[k-j_2], ..., u[k-j_m]) \\ &+ v(k),\ \ k = 1, 2, ..., N \end{aligned}$$

where $y[k]$ and $u[k]$ are output and input measurements. It is assumed that

1. The input $u[k]$ is an independent and identically distributed (iid) random sequence in a (unknown) open interval $I \in R$ with a (unknown) probability density function $\psi(\cdot)$. The noise $v[k]$ is a sequence of iid random variables with zero mean and bounded variance.
2. The exact time lag is unknown and only the upper bound n is available.
3. The functions $\bar{f}_{j_1 j_2 ... j_l}$'s, $l \le n$, referred to as l-factor terms, are unknown and describe interactions of variables $u[k-j_1], u[k-j_2], ..., u[k-j_l]$. No structural prior information on $\bar{f}_{j_1 j_2 ... j_l}$'s is assumed.

To convey the idea clearly without tedious and unilluminating detailed technical derivations, we will focus on the system with upto 2-factor interactive terms in this work.

$$\begin{aligned} y[k] &= f(u[k-1], u[k-2], ..., u[k-n]) + v[k] \\ &= \bar{c} + \sum_{j=1}^{n} \bar{f}_j(u[k-j]) + \sum_{1 \le j_1 < j_2 \le n} \bar{f}_{j_1 j_2}(u[k-j_1], u[k-j_2]) + v[k] \end{aligned} \quad (1)$$

All the results of this work can be trivially but cumbersomely extended to a general system with arbitrary interactive terms. Obviously, for a system upto 2-factor interactive terms, there are totally $M + 1 = 1 + n + n(n-1)/2$ terms in the system, one constant term, n 1-factor terms $\bar{f}_j$'s and $\frac{n(n-1)}{2}$ 2-factor terms $\bar{f}_{j_1 j_2}$'s.

What we are concerned are:

- How to determine orthogonal basis functions $\phi_i(\cdot)$'s, $i = 0, 1, ..., M$, based on the given data set $\{y[k], u[k]\}_1^N$ that represents the unknown system (1)?
- How to identify these basis functions?
- Once the basis functions $\phi_i(\cdot)$'s are determined, it does not mean that all $M + 1$ terms are needed. In most practical cases, only the terms $i = 0, 1, ..., p < M + 1$ are needed. How to find the order p?
- Even the order p is found, the system could be sparse in the sense that not all terms $i = 0, 1, 2, ..., p$ are present and many terms are actually zero. How to identify those terms so they can be removed?

In the following derivation, we denote the expectation operator by $\mathbf{E}$ and conditional expectation operators for given $u[k-j_1] = x_{j_1}$, and/or $u[k-j_2] = x_{j_2}$ by, respectively,

$$\mathbf{E}(y[k] \mid u[k-j_1] = x_{j_1}),$$

$$\mathbf{E}(y[k] \mid u[k-j_1] = x_{j_1}, u[k-j_2] = x_{j_2}),$$

$$\mathbf{E}(f_{j_1 j_2}(u[k-j_1], u[k-j_2]) \mid u[k-j_1] = x_{j_1}),$$

$$\mathbf{E}(f_{j_1 j_2}(u[k-j_1], u[k-j_2]) \mid u[k-j_2] = x_{j_2}).$$

For every x_{j_1} and $x_{j_2} \in I$, define the normalized functions $f_{j_1 j_2}$'s and f_j's in (2).

$$\begin{aligned} f_{j_1 j_2}(x_{j_1}, x_{j_2}) = &\ \bar{f}_{j_1 j_2}(x_{j_1}, x_{j_2}) - \mathbf{E}(\bar{f}_{j_1 j_2}(u[k-j_1], u[k-j_2]) \mid u[k-j_2] = x_{j_2}) \\ &- \mathbf{E}(\bar{f}_{j_1 j_2}(u[k-j_1], u[k-j_2]) \mid u[k-j_1] = x_{j_1}) \\ &+ \underbrace{\mathbf{E}\{\bar{f}_{j_1 j_2}(u[k-j_1], u[k-j_2])\}}_{c_{j_1 j_2}}, \quad 1 \le j_1 < j_2 \le n \end{aligned}$$

$$f_1(x_1) = \bar{f}_1(x_1) + \sum_{i=2}^{n} \mathbf{E}(\bar{f}_{1i}(u[k-1], u[k-i]) \mid u[k-1] = x_1)$$

$$-\underbrace{\mathbf{E}\{\bar{f}_1(u[k-1]) + \sum_{i=2}^{n}\mathbf{E}(\bar{f}_{1i}(u[k-1],u[k-i]) \mid u[k-1]=x_1)\}}_{c_1}$$

$$\begin{aligned}
f_j(x_j) &= \bar{f}_j(x_j) + \sum_{i=j+1}^{n}\mathbf{E}(\bar{f}_{ji}(u[k-j],u[k-i]) \mid u[k-j]=x_j) \\
&+ \sum_{i=1}^{j-1}\mathbf{E}(\bar{f}_{ij}(u[k-i],u[k-j]) \mid u[k-j]=x_j) \\
&- \underbrace{\mathbf{E}\{\bar{f}_j(u[k-j]) + \sum_{i=j+1}^{n}\mathbf{E}(\bar{f}_{ji}(u[k-j],u[k-i]) \mid u[k-j]=x_j)}_{c_j^1} \\
&+ \underbrace{\sum_{i=1}^{j-1}\mathbf{E}(\bar{f}_{ij}(u[k-i],u[k-j]) \mid u[k-j]=x_j)\}}_{c_j^2},\ j=2,...,n-1
\end{aligned}$$

$$\begin{aligned}
f_n(x_n) &= \bar{f}_n(x_n) + \sum_{i=1}^{n-1}\mathbf{E}(\bar{f}_{in}(u[k-i],u[k-n]) \mid u[k-n]=x_n) \\
&- \underbrace{\mathbf{E}\{\bar{f}_n(u[k-n]) + \sum_{i=1}^{n-1}\mathbf{E}(\bar{f}_{in}(u[k-i],u[k-n]) \mid u[k-n]=x_n)\}}_{c_n}
\end{aligned}$$

$$c = \bar{c} - \sum_{1\le j_1<j_2\le n} c_{j_1 j_2} + \sum_{j=1}^{n} c_j,\ \ with\ \ c_j = c_j^1 + c_j^2. \tag{2}$$

Then, the system (1) can be rewritten as

$$\begin{aligned}
y[k] = c + \sum_{j=1}^{n} f_j(u[k-j]) + \sum_{1\le j_1<j_2\le n} f_{j_1 j_2}(u[k-j_1],u[k-j_2]) \\
+ v[k],\ k=1,2,\dots,N
\end{aligned} \tag{3}$$

We are now in a position to define data dependent orthogonal basis functions ϕ_i, $i=0,...,M$.

$$\phi_0 = c \Longrightarrow \phi_0 \quad \phi_j(x_j) = f_j(x_j),\ j=1,...,n \Longrightarrow \phi_1,...,\phi_n,$$
$$\phi_{\frac{2n}{2}-1+j}(x_1,x_j) = f_{1j}(x_1,x_j),\ j=2,...,n \quad \Longrightarrow \phi_{n+1},...,\phi_{2n-1},$$

$$\begin{aligned}
&\phi_{\frac{2n-1}{2}2-2+j}(x_2, x_j) = f_{2j}(x_2, x_j),\ \ j = 3, ..., n \quad \Longrightarrow \phi_{2n}, ..., \phi_{3n-3},\\
&\phi_{\frac{2n-2}{2}3-3+j}(x_3, x_j) = f_{3j}(x_3, x_j),\ \ j = 4, ..., n \quad \Longrightarrow \phi_{3n-2}, ..., \phi_{4n-6},\\
&\quad \ldots\\
&\phi_{\frac{2n-(n-3)}{2}(n-2)-(n-2)+j}(x_{n-2}, x_j) = f_{(n-2)j}(x_{n-2}, x_j), \quad \Longrightarrow \phi_{\frac{n^2+n}{2}-2}, \phi_{\frac{n^2+n}{2}-1},\\
&\qquad\qquad\qquad\qquad j = n-1, n\\
&\phi_{\frac{2n-(n-2)}{2}(n-1)-(n-1)+j}(x_{n-1}, x_j) = f_{(n-1)j}(x_{n-1}, x_j), \quad \Longrightarrow \phi_{\frac{n^2+n}{2}},\ j = n
\end{aligned}$$

When the meaning is clear from the context, we interchangeably use

$$\begin{aligned}
&\phi_j[k] = \phi_j(u[k-j]),\ \ j = 1, ..., n\\
&\phi_j[k] = \phi_j(u[k-1], u[k-j+n-1]), j = n+1, ..., 2n-1\\
&\phi_j[k] = \phi_j(u[k-2], u[k-j+2n-3]), j = 2n, ..., 3n-3\\
&\qquad \ldots\\
&\phi_j[k] = \phi_j(u[k-n+2], u[k-j+M-n-1]), j = M-2, M-1\\
&\phi_j[k] = \phi_j(u[k-n+1], u[k-n]), j = M = n(n+1)/2.
\end{aligned}$$

Clearly, ϕ_0 denotes the constant term, $\phi_j(x_j)$'s, $j = 1, ..., n$, represent the 1-factor terms and $\phi_i(x_{j_1}, x_{j_2})$'s, $i = n+1, ..., M$, are 2-factor terms. The following theorem is the main result of this section.

Theorem 1 *Consider the system (1). Then we have:*

1. *The system (1) can be represented by the data-driven basis functions ϕ_i's,*

$$y[k] = \sum_{i=0}^{M} \phi_i[k] + v[k] \tag{4}$$

where $M = n + n(n-1)/2 = n(n+1)/2$.

2. *The data-driven basis functions ϕ_i's are orthogonal. i.e., for all $1 \leq j \leq M$ and $0 \leq j_1 < j_2 \leq M$,*

$$\mathbf{E}\phi_j[k] = 0, \quad \mathbf{E}\phi_{j_1}[k]\phi_{j_2}[k] = 0.$$

3. *The unknown ϕ_j's are the expectations or conditional expectations of the output,*

$$\begin{aligned}
\phi_0 &= \mathbf{E}\{y[k]\},\\
\phi_j(x_j) &= \mathbf{E}\{y[k] \mid u[k-j] = x_j\} - \phi_0,\ \ j = 1, ..., n,\\
\phi_{\frac{2n}{2}-1+j}(x_1, x_j) &= \mathbf{E}\{y[k] \mid u[k-1] = x_1, u[k-j] = x_j\}\\
&\quad - \phi_1(x_1) - \phi_j(x_j) - \phi_0,\ \ j = 2, ..., n\\
\phi_{\frac{2n-1}{2}2-2+j}(x_2, x_j) &= \mathbf{E}\{y[k] \mid u[k-2] = x_2, u[k-j] = x_j, \}\\
&\quad - \phi_2(x_2) - \phi_j(x_j) - \phi_0,\ \ j = 3, ..., n\\
&\quad \ldots
\end{aligned}$$

$$\begin{aligned}\phi_{\frac{2n-(n-3)}{2}(n-2)-(n-2)+j}(x_{n-2}, x_j) &= \mathbf{E}\{y[k] \mid u[k-n+2] = x_{n-2}, u[k-j] = x_j\} \\ &\quad - \phi_{n-2}(x_{n-2}) - \phi_j(x_j) - \phi_0,\ \ j = n-1, n \\ \phi_{\frac{2n-(n-2)}{2}(n-1)-(n-1)+j}(x_{n-1}, x_j) &= \mathbf{E}\{y[k] \mid u[k-n+1] = x_{n-1}, u[k-j] = x_j\} \\ &\quad - \phi_{n-1}(x_{n-1}) - \phi_j(x_j) - \phi_0,\ \ j = n \qquad (5)\end{aligned}$$

From the theorem, we see that not only the system (1) can be represented by the data-driven basis functions ϕ_i's as in (4) but also these basis functions are orthogonal and can be estimated separately. If the estimate $\widehat{y} = \sum_{i=0}^{p} \phi_i[k]$ is deemed to be not sufficient enough and an additional term $\phi_{p+1}[k]$ is needed, then only the additional term $\phi_{p+1}[k]$ has to be identified and added to the model. No previously obtained terms $\phi_i,\ i = 0, 1..., p$ have to be reestimated.

3 Identification Under Random Inputs

Though the basis functions ϕ_i's are determined, they depend on the unknown system and have to be identified from the given data set. From Theorem 1, these unknown ϕ_i's are the expectations or conditional expectations of the output. Now the question is how to calculate these expectation values by empirical averages based on the available input–output measurement data set $\{y[k], u[k]\}_1^N$. In this work, we adopt a fairly simple yet efficient kernel approach which was developed in our previous works [5, 6]. To this end, let x_j be any point in the interval I in which the input $u[\cdot]$ lies, define

$$\varphi_j(x_j, k) = |u[k-j] - x_j|.$$

Let $\delta > \min \varphi_j(x_j, k)$ be any positive constant. Let

$$M_j(x_j) = \{m_j(1), m_j(2), ..., m_j(l_j)\}$$

be a set that contains integers $m_j(i)$'s such that $m_j(i) \in M_j(x_j) \Leftrightarrow \delta > \varphi_j(x_j, m_j(i))$. $l_j(x_j)$ is the number of elements in $M_j(x_j)$ that is the same as the number of $\varphi_j(x_j, k)$'s that are smaller than δ. Define, for each j and x_j,

$$w_j(x_j, k) = \begin{cases} \frac{\delta - \varphi_j(x_j,k)}{l_j\delta - \sum_{i=1}^{l_j} \varphi_j(x_j, m_j(i))} & k \in M_j(x_j) \\ 0 & k \notin M_j(x_j) \end{cases}.$$

Obviously for all k, j and x_j, $w_j(x_j, k) \geq 0$ and $\sum_{k=1}^{N} w_j(x_j, k) = \sum_{i=1}^{l_j} w_j(x_j, m_j(i)) = 1$. Similarly, for any pair $0 \leq j_1 < j_2 \leq n$ and $(x_{j_1}, x_{j_2}) \in I^2$, define

$$\varphi_{j_1 j_2}(x_{j_1}, x_{j_2}, k) = \|(u[k-j_1], u[k-j_2]) - (x_{j_1}, x_{j_2})\|_2.$$

If $\delta > \min \varphi_{j_1 j_2}(x_{j_1}, x_{j_2}, k)$, let $M_{j_1 j_2}(x_{j_1}, x_{j_2}) = \{m_{j_1 j_2}(1), m_{j_1 j_2}(2), ..., m_{j_1 j_2}(l_{j_1 j_2})\}$ be a set such that $k \in M_{j_1 j_2}(x_{j_1}, x_{j_2}) \Leftrightarrow \delta > \varphi_{j_1 j_2}(x_{j_1}, x_{j_2}, k)$. Define

$$w_{j_1 j_2}(x_{j_1}, x_{j_2}, k) = \begin{cases} \frac{\delta - \varphi_{j_1 j_2}(x_{j_1}, x_{j_2}, k)}{l_{j_1 j_2}\delta - \sum_{i=1}^{l_{j_1 j_2}} \varphi_j(x_{j_1}, x_{j_2}, m_{j_1 j_2}(i))} & k \in M_{j_1 j_2}(x_{j_1}, x_{j_2}) \\ 0 & k \notin M_{j_1 j_2}(x_{j_1}, x_{j_2}) \end{cases}.$$

Notice that the same properties hold

$$w_{j_1 j_2}(x_{j_1}, x_{j_2}, k) \geq 0, \;\; \sum_{k=1}^{N} w_{j_1 j_2}(x_{j_1}, x_{j_2}, k) = \sum_{i=1}^{l_{j_1 j_2}} w_{j_1 j_2}(x_{j_1}, x_{j_2}, m_{j_1 j_2}(i)) = 1.$$

Now, for a given pair $(x_{j_1}, x_{j_2}) \in I^2$, we define the estimates $\widehat{\phi_i}$, $i = 0, 1, ..., M$,

$$\begin{aligned}
\widehat{\phi}_0 &= \frac{1}{N}\sum_{k=1}^{N} y[k], \\
\widehat{\phi}_j(x_j) &= \sum_{k=1}^{N} w_j(x_j, k) y[k] - \widehat{\phi}_0, \;\; j = 1, ..., n, \\
\widehat{\phi}_{\frac{2n}{2}-1+j}(x_1, x_j) &= \sum_{k=1}^{N} w_{1j}(x_1, x_j, k) y[k] - \widehat{\phi}_1(x_1) - \widehat{\phi}_j(x_j) - \widehat{\phi}_0, \;\; j = 2, ..., n \\
\widehat{\phi}_{\frac{2n-1}{2}2-2+j}(x_2, x_j) &= \sum_{k=1}^{N} w_{2j}(x_2, x_j, k) y[k], -\widehat{\phi}_2(x_2) - \widehat{\phi}_j(x_j) - \widehat{\phi}_0, \;\; j = 3, ..., n \\
&\dots \\
\widehat{\phi}_{\frac{2n-(n-3)}{2}(n-2)-(n-2)+j}(x_{n-2}, x_j) &= \sum_{k=1}^{N} w_{n-2,j}(x_{n-2}, x_j, k) y[k] \\
&\quad - \widehat{\phi}_{n-2}(x_{n-2}) - \widehat{\phi}_j(x_j) - \widehat{\phi}_0, \;\; j = n-1, n \\
\widehat{\phi}_{\frac{2n-(n-2)}{2}(n-1)-(n-1)+j}(x_{n-1}, x_j) &= \sum_{k=1}^{N} w_{n-1,j}(x_{n-1}, x_j, k) y[k] \\
&\quad - \widehat{\phi}_{n-1}(x_{n-1}) - \widehat{\phi}_j(x_j) - \widehat{\phi}_0, \;\; j = n \qquad (6)
\end{aligned}$$

Theorem 2 *Consider the system (4) and the estimates above. For any $x_{j_1}, x_{j_2} \in I$, assume*

- *The unknown basis functions ϕ_i's are differentiable with the Lipschitz constant L for $x_{j_1}, x_{j_2} \in I$.*
- *Let $\psi(\cdot)$ be the (unknown) probability density function of the input $u[\cdot]$ and $\psi(\cdot)$ is nonzero at x_{j_1}, x_{j_2}, i.e.,*

$$\psi(x_{j_1}) > 0,\ \psi(x_{j_2}) > 0.$$

- $\delta \to 0$ *and* $\delta^2 N \to \infty$ *as* $N \to \infty$.

Then, as $N \to \infty$*, we have in probability that*

$$\widehat{\phi}_0 \to \phi_0,$$

$$\widehat{\phi}_j(x_j) \to \phi_j(x_j),\ j = 1, 2, ..., n$$

$$\widehat{\phi}_j(x_{j_1}, x_{j_2}) \to \phi_j(x_{j_1}, x_{j_2}),\ 1 \le j_1 < j_2 \le n,\ j = n+1, ..., M$$

Moreover asymptotically, $|\widehat{\phi}_j(x_j) - \phi_j(x_j)|^2 \sim O(\delta + \frac{1}{\delta N}),\ j = 1, 2, ..., n$ *and* $|\widehat{\phi}_j(x_{j_1}, x_{j_2}) - \phi_j(x_{j_1}, x_{j_2})|^2 \sim O(\delta + \frac{1}{\delta^2 N}),\ 1 \le j_1 < j_2 \le n,\ j = n+1, ..., M.$

4 Order Determination

How many terms should be included in the model or equivalently how to determine the order p of the estimate $\widehat{f} = \sum_{i=0}^{p} \widehat{\phi}_i[k]$ is an important and difficult part of identification. This amounts to if the chosen order is sufficient to represent the unknown nonlinear system or an additional term or terms should be added to the estimate. A related issue is the regressor selection. Even if the order is accurately obtained, some terms ϕ_i's are irrelevant to the output and should not be included in the estimate. How to find and remove those terms are also important. These two issues are closely related. We propose two approaches towards these two issues.

4.1 Combined Residual Analysis and Statistical Test

The idea of the statistical test is fairly simple. Suppose the order p is sufficient so that the estimate $\sum_{i=0}^{p} \phi_i[k]$ represents the true but unknown system f well. Then, the residual

$$r[k] = y[k] - \sum_{i=0}^{p} \phi_i[k] \approx v[k]$$

is almost white. In other words, if the residual is white, nothing more can be squeezed out from the data and thus the order p is sufficient. Let

$$\mu = \mathbf{E}r[k],\ \gamma[j] = \mathbf{E}(r[k] - \mu)(r[k-j] - \mu),\ \rho[j] = \gamma[j]/\gamma[0]$$

denote the mean, the lag-j autocovariance and the lag-j correlation coefficient of $r[k]$, respectively. If the residual $r[k]$ is white, it follows that

$$\gamma[1] = \gamma[2] = ... = 0, \quad \rho[1] = \rho[2] = ... = 0$$

In particular, for the system (4), $r[k] = y[k] - \sum \phi_i[k]$ is a function of $u[k-1], u[k-2], ..., u[k-n]$ and $r[k-n] = y[k-n] - \sum \phi_i[k-n]$ is a function of $u[k-n-1], u[k-n-2], ..., u[k-2n]$. They are automatically independent. Thus, what we have to to check is if

$$\rho[1] = \rho[2] = ... = \rho[n-1] = 0$$

The most effective test in the literature for checking if $\rho[1] = \rho[2] = ... = \rho[n-1] = 0$ are the Box–Pierce test [9] and its variants which have been widely accepted and applied for linear system identification [18, 29]. It states as follows: for large N,

$$N \sum_{j=1}^{n-1} \rho[j]^2 = N(\rho[1], ..., \rho[n-1]) \begin{pmatrix} \rho[1] \\ \vdots \\ \rho[n-1] \end{pmatrix} \tag{7}$$

follows a chi-square distribution with (n-1) degree of freedom if $r[k]$ is white. This provides a framework for statistical hypothesis tests. Let

$$H_0 : \text{the residual } r[k] \text{ is white.}$$

Then, the null hypothesis H_0 can be tested based on $N \sum_{j=1}^{n-1} \rho[j]^2$ and the $\chi^2(n-1)$ distribution. If H_0 is accepted, $r[k]$ is considered to be white and the order p is accepted. To test the hypothesis, we calculate $N \sum_{j=1}^{n-1} \rho[j]^2$ based on the residual. Let the threshold d be taken from the $\chi^2(n-1)$ distribution with α being the level of significance, i.e., the probability to reject H_0 though H_0 is true. The hypothesis H_0 is accepted if $N \sum_{j=1}^{n-1} \rho[j]^2 \le d$ and is rejected if $N \sum_{j=1}^{n-1} \rho[j]^2 > d$ and we conclude that the order p is not high enough.

There are two problems however. The first is that what we really test is not if the residual $r[k]$ is white or not but if $r[i]$ and $r[j]$ are uncorrelated or not. The Box–Pierce test (7) works well for this purpose in linear identification but may not work for nonlinear identification. If the residual $r[k]$ exhibits some nonlinear dependence which is usually the case in nonlinear identification because no actual ϕ_i's are available and only their estimates $\widehat{\phi_i}$'s are known. This unavoidably adds some nonlinear dependence on the residual. In such a case, the Box–Pierce test does not work well. In fact, the Box–Pierce test could be invalid and provide some misleading conclusions [32]. Therefore, a modified Box–Pierce test is needed in the presence of nonlinear dependence of $r[k]$. The second problem is that even the null hypothesis H_0 is accepted, it does not necessarily mean that $r[k]$ is white. Since the null hypothesis only tests if H_0 should be accepted given H_0 is true. There is no way of knowing the probability

$$Prob\{\, accept\ H_0 : \ H_0\, is\ false\}$$

This is referred to as the second type of error and is hard to answer. Thus, there must an additional and independent test to ensure reasonably that H_0 is not false. We deal with these two problems separately.

Modified Box–Pierce test: Let $r[k] = y[k] - \sum_{i=0}^{p} \widehat{\phi}_i[k]$ be the residual. Denote the sampled mean, the lag-j autocovariance, the lag-j correlation coefficient by respectively

$$\widehat{\mu} = \frac{1}{N}\sum_{k=1}^{N} r[k],\ \widehat{\gamma}[j] = \frac{1}{N-j}\sum_{k=j+1}^{N}(r[k]-\widehat{\mu})(r[k-j]-\widehat{\mu}),\ \widehat{\rho}[j] = \widehat{\gamma}[j]/\widehat{\gamma}[0]$$

It was shown in [19] that for large N,

$$N(\widehat{\rho}[1], ..., \widehat{\rho}[n-1])V^{-1}\begin{pmatrix}\widehat{\rho}[1]\\ \vdots \\ \widehat{\rho}[n-1]\end{pmatrix} \tag{8}$$

follows a chi-square distribution with (n-1) degree of freedom when H_0 is true, where

$$V = C/\gamma[0]^2 = \begin{pmatrix} c_{11} & \cdots & c_{1,n-1} \\ \vdots & \ddots & \vdots \\ c_{n-1,1} & \cdots & c_{n-1,n-1}\end{pmatrix}/\gamma[0]^2$$

$$c_{ij} = \sum_{q=-\infty}^{\infty} \mathbf{E}(r[k]-\mu)(r[k-i]-\mu)(r[k+q]-\mu)(r[k+q-j]-\mu)$$

$$i, j = 1, ..., n-1$$

with μ being the mean value of $r[k]$. The difference is that the identity matrix is used in the Box–Pierce test (7) while in the modified Box–Pierce test (8), the actual autocovariance matrix V is used. The modified Box–Pierce test is reliable for large N even the residual $r[k]$ exhibits nonlinear dependence. For our application, however, the actual autocovariance matrix V is unknown and has to be estimated. To this end, let

$$W[k] = \begin{pmatrix} (r[k]-\widehat{\mu})(r[k-1]-\widehat{\mu}) \\ (r[k]-\widehat{\mu})(r[k-2]-\widehat{\mu}) \\ \vdots \\ (r[k]-\widehat{\mu})(r[k-n+1]-\widehat{\mu})\end{pmatrix}$$

and $K(x)$ be the triangle kernel function

$$K(x) = \begin{cases} 1-|x|, & |x| \le 1 \\ 0, & |x| > 1 \end{cases}$$

Now, define the estimate $\widehat{V}$ of V by $\widehat{C}/\widehat{\gamma}[0]^2$ with

$$\widehat{C} = \sum_{q=-l}^{l} K(\frac{q}{l})\frac{1}{N-n+1-|q|}\sum_{k} W[k]W[k-q]'$$
$$= \sum_{q=-l}^{0} K(\frac{q}{l})\frac{1}{N-n+1+q}\sum_{k=n}^{N+q} W[k]W[k-q]'$$
$$+ \sum_{q=1}^{l} K(\frac{q}{l})\frac{1}{N-n+1-q}\sum_{k=n+q}^{N} W[k)W[k-q]'$$

where l is the bandwidth of the kernel $K(\cdot)$. Note all the variables $\widehat{\mu}$, $\widehat{\rho}[j]$, $W[k]$ and $\widehat{\gamma}[j]$ are computable. Now, we show that the modified Box–Pierce test is still valid if the actual autocovariance matrix V is replaced by its estimate as discussed above,

Theorem 3 *Consider the residual $r[k]$ and the corresponding $\widehat{\mu}$, $\widehat{\gamma}[j]$, $\widehat{\rho}[j]$ and $\widehat{V} = \widehat{C}/\widehat{\gamma}[0]^2$. Then,*

$$Q_{n-1} = N(\widehat{\rho}[1], ..., \widehat{\rho}[n-1])\widehat{V}^{-1}\begin{pmatrix}\widehat{\rho}[1]\\ \vdots \\ \widehat{\rho}[n-1]\end{pmatrix} \tag{9}$$

converges, in distribution as $N \to \infty$, to a chi-square distribution with (n-1) degree of freedom if the residual $r[k]$ is white, provided that

$$l \to \infty,\ l/N \to 0,\ as\ N \to \infty$$

Residual analysis: As discussed above, the hypothesis test is effective only it is reasonably sure that H_0 is not false. A very simple but a common sense way is to check the magnitude of the residual. There are two purposes. If the estimate represents the system well or the order is adequate, the residual should be small. On the other hand, we do not want to over-fit the system. In this regard, the parsimony principle applies. Let $r_p[k] = y[k] - \sum_{i=0}^{p} \widehat{\phi_i}[k]$ be the residual where the subscript p indicates the order of the estimate. Define the average error

$$e[p] = \frac{1}{N}\sum_{k=1}^{N} r_p[k]^2$$

Obviously, the average error $e[p]$ is a monotonically decreasing function of the order p as depicted in the top diagram of Fig. 4. Initially, $e[p]$ decreases as the order increases because the model picks up relevant terms ϕ_i's of the unknown system. However, even when the correct order has been reached, the value $e[p]$ still decreases because additionally added terms try to model noise. The improved "fit" is harmful since it models noise but not the system. However, the decrease from the over-fit is less significant than the decrease when the relevant terms are picked up by the estimate. Therefore, what we are looking for is where the curve $e[p]$ is small and flattened, known as the "knee" in Fig. 4.

We are now in a position to state the combined residual analysis and hypothesis test approach for order determination.

Step 1: Carry out identification by estimating $\widehat{\phi_i}$ as described in the previous sections.
Step 2: Calculate the residual $r_p[k]$ for each p and plot the average error $e[p]$ vs p as shown in the top diagram of Fig. 4.
Step 3: Find the knee in the curve where the average error $e[p]$ is small and flattened. Determine the corresponding order p for the hypothesis test.
Step 4: Calculate Q_{n-1} as in (9) and carry out the modified Box–Pierce test. Let the threshold d be taken from the $\chi^2(n-1)$ distribution with α being the level of significance usually 0.03–0.05, i.e., the probability to reject H_0 though H_0 is true. The hypothesis H_0 is accepted if $Q_{n-1} \leq d$ and we conclude that the order p is sufficient. The hypothesis H_0 is rejected if $Q_{n-1} > d$ and we conclude that the order p is not high enough and an additional term or terms should be included in the estimate. Then, the test is repeated with $p \to p+1$.

4.2 Relative and Cumulative Contribution Approach

In order determination, what we are interested in is not if a particular term $\phi_i[k]$ contributes or not, but whether the contribution is significant or not. Identification is always a balance between model accuracy and model parsimony. The data-driven orthogonal approach discussed in the previous sections allows us to decompose the total contribution into a sum of individual contributions, referred to as the relative contribution in this work, and provides a reliable way for the order determination and regressor selection. To this end, we propose a relative contribution approach for order determination and regressor selection that exploits the orthogonal properties of the basis functions. Consider the system (4). It is easily verified from the orthogonal properties of $\phi_i[k]$'s that

$$\mathbf{E}y[k]^2 = \mathbf{E}\{\sum_{i=0}^{M} \phi_i[k] + v[k]\}^2 = \sum_{i=0}^{M} \mathbf{E}\phi_j[k]^2 + \mathbf{E}v[k]^2$$

We now define the relative contribution $R_c[j]$ as

$$R_c[j] = \frac{\mathbf{E}\phi_j[k]^2}{\mathbf{E}y[k]^2}, \ j = 0, ..., M$$

Since the square term is proportional to energy, the meaning of the relative contribution $R_c[p]$ is the percent of energy in the p's term to the overall output energy.

Obviously, if the p's term is insignificant, the relative contribution $R_c[p]$ should be small and not be a part of the estimate.

A closely related concept is the cumulative contribution $C_c[p]$

$$C_c[p] = \sum_{j=0}^{p} R_c(j) = \sum_{j=0}^{p} \frac{\mathbf{E}\phi_j[k]^2}{\mathbf{E}y[k]^2},\ p = 0, ..., M$$

which measures the contribution of first $p + 1$ terms relative to the overall output. Obviously, if the order p is correct, the cumulative contribution $C_c[p]$ should be close to unit and is flattened in the curve $C_c[p]$ vs p. It is important to point out that because of noise contribution term $\mathbf{E}v[k]^2$, the cumulative contribution can never reach 100%. To test the order based on the cumulative contribution, an estimate of the relative contribution of the unknown noise has to be done. This makes the method based on the cumulative contribution less efficient compared to the relative contribution approach.

In reality, ϕ_i' are unavailable and only their estimates $\widehat{\phi}_i$'s are available. However, because of their convergence properties, $\widehat{\phi}_i \to \phi_i$ as $N \to \infty$, we may define the estimates of $R_c[p]$ and $C_c[j]$ by

$$\widehat{R}_c[j] = \frac{\frac{1}{N}\sum_{k=1}^{N} \widehat{\phi}_j[k]^2}{\frac{1}{N}\sum_{k=1}^{N} y[k]^2}$$

and

$$\widehat{C}_c[p] = \sum_{j=0}^{p} \frac{\frac{1}{N}\sum_{k=1}^{N} \widehat{\phi}_j[k]^2}{\frac{1}{N}\sum_{k=1}^{N} y[k]^2}.$$

The substitution is reliable for large N because of the convergence property.

To test whether the pth term should be included, we compute $\widehat{R}_c[p]$ and choose a threshold d_1, for example $d_1 = 0.03$ or 3%. If $\widehat{R}_c[p] \geq d_1$, the pth term is included. Otherwise the term is discarded. This not only provides the order of the system but also determines exactly which term should be included in the model.

5 Deterministic Inputs and Galois Sequence

Generally, there are two ways to estimate the structure of the system. The first one is full scale system identification. The idea is to identify the system including each $\bar{f}_i$ and $\bar{f}_{ij}$ and then enumerate all possible models for different combinations of $\bar{f}_i$ and $\bar{f}_{ij}$ as well as n. Some performance measures are calculated and the model that achieves the best performance is chosen. Then, the corresponding n is the estimate of time lag and the surviving terms of $\bar{f}_i$ and $\bar{f}_{ij}$ are retained in the system. All other $\bar{f}_i$'s and $\bar{f}_{ij}$'s are considered to be negligible. The method does not distinguish between model structural estimation and full scale system identification. Note that the system is nonparametric and nonlinear. Hence, identification is usually computationally expensive and the optimization algorithm could be stuck in a local minimum. It is certainly advantageous if the structure of the system can be estimated before a full scale system identification is performed. To this end, we propose two different methods.

5.1 Visual Inspection Method

Recall that in structural estimation, we are interested not in full scale system identification, but rather in finding a simple and reliable way to estimate the structure, in particular to determine the terms $\bar{f}_i$ and $\bar{f}_{ij}$ which contribute significantly. In this section, we assume that the input is at our disposal (which admittedly may be restrictive in some applications). Under such an assumption, the first problem is to find an input sequence that is simple and has the ability to separate the contributions of $\bar{f}_i$ and $\bar{f}_{ij}$,

$$U_{2^3} = \begin{pmatrix} u(1) & u(0) & u(-1) \\ u(2) & u(1) & u(0) \\ u(3) & u(2) & u(1) \\ u(4) & u(3) & u(2) \\ u(5) & u(4) & u(3) \\ u(6) & u(5) & u(4) \\ u(7) & u(6) & u(5) \\ u(8) & u(7) & u(6) \\ u(9) & u(8) & u(7) \end{pmatrix} = \begin{pmatrix} a_1 & a_1 & a_1 \\ a_2 & a_1 & a_1 \\ a_2 & a_2 & a_1 \\ a_2 & a_2 & a_2 \\ a_1 & a_2 & a_2 \\ a_2 & a_1 & a_2 \\ a_1 & a_2 & a_1 \\ a_1 & a_1 & a_2 \\ a_1 & a_1 & a_1 \end{pmatrix}. \tag{10}$$

To this end, let l be a prime number that indicates the number of levels of input, i.e., $u[k] = \{a_1, a_2, ..., a_l\}$, usually $|a_i| \neq |a_j|$ to avoid ambiguity for quadratic nonlinearities. To excite the system to the maximum extent, the input sequence should contain all possible combinations of n-tuple $(a_{i_1}, a_{i_2}, \dots, a_{i_n}), a_{i_j} = a_1, \dots, , a_l$. The minimum length of such a generating sequence is $n + l^n - 1$. The Galois sequence is such a sequence which has been investigated in [13, 20] for worst-case identification. Galois sequence has many desirable properties. It is a periodic pseudorandom sequence with period l^n [20] and can be easily generated [13]. More importantly, within one period, it produces each n-tuple combination exactly once [20]. Note that the Galois sequence defined here is slightly different from the traditional one [13] as we need all the n-tuples to be included. This small difference can be easily taken care of and in fact this definition is exactly the same as in [20]. An example of $G(l^n)$ for $n = 3$ and $l = 2$ is given in (10). To average out the effect of noise, the input sequence is repeated L times, i.e.,

$$U_{Ll^n} = \left.\begin{pmatrix} U_{l^n} \\ U_{l^n} \\ \vdots \\ U_{l^n} \end{pmatrix}\right\} L\text{times}. \tag{11}$$

Before performing structural estimation, it is interesting to observe that the representation (1) of the system is actually not unique. For instance, $\bar{f}_1 \to \bar{f}_1 + c$ and $\bar{f}_2 \to \bar{f}_2 - c$ for any constant c would not change the input–output relationship which implies that the structure of the system, as represented in (1), is not identifiable. To overcome this problem, we normalize the system to make the averages of $\bar{f}_i$ and $\bar{f}_{ij}$ with respect to the input equal to zero. Let

$$g_{j,ij}(u[k-j]) = \frac{1}{l}\sum_{m=1}^{l} \bar{f}_{ij}(a_m, u[k-j]),$$

$$g_{i,ij}(u[k-i]) = \frac{1}{l}\sum_{m=1}^{l} \bar{f}_{ij}(u[k-i], a_m)$$

be the partial average of $\bar{f}_{ij}$ with respect to the first and second variables respectively and

$$\check{c}_{ij} = \frac{1}{l^2}\sum_{m_1=1}^{l}\sum_{m_2=1}^{l} \bar{f}_{ij}(a_{m_1}, a_{m_2})$$

be the total average. Define

$$\check{f}_{ij}(u[k-i], u[k-j]) = \bar{f}_{ij}(u[k-i], u[k-j]) - g_{j,ij}(u[k-j]) - g_{i,ij}(u[k-i]) + \check{c}_{ij}. \tag{12}$$

Obviously, the average of this new function is zero,

$$\sum_{m=1}^{l} \check{f}_{ij}(a_m, u[k-j]) = \sum_{m=1}^{l} \check{f}_{ij}(u[k-i], a_m) = 0. \tag{13}$$

To make the average of $\bar{f}_i$ equal to zero, let, for each $1 \le i \le n$,

$$\check{f}_1(u[k-1]) = \bar{f}_1(u[k-1]) + \sum_{i=2}^{n} g_{1,1i}(u[k-1]) - \underbrace{\frac{1}{l}\sum_{m=1}^{l}[\bar{f}_1(a_m) + \sum_{i=2}^{n} g_{1,1i}(a_m)]}_{\check{c}_1},$$

$$\check{f}_{n-1}(u[k-n+1]) = \bar{f}_{n-1}(u[k-n+1]) + \sum_{i=1}^{n-2} g_{(n-1),i(n-1)}(u[k-n+1])$$

$$+g_{(n-1),(n-1)n}(u[k-n+1]) - \underbrace{\frac{1}{l}\sum_{m=1}^{l}[\bar{f}_{n-1}(a_m) + \sum_{i=1}^{n-2} g_{(n-1),i(n-1)}(a_m) + g_{(n-1),(n-1)n}(a_m)]}_{\check{c}_{n-1}},$$

$$\check{f}_n(u[k-n]) = \bar{f}_n(u[k-n]) + \sum_{i=1}^{n-1} g_{n,in}(u[k-n]) - \underbrace{\frac{1}{l}\sum_{m=1}^{l}[\bar{f}_n(a_m) + \sum_{i=1}^{n-1} g_{n,in}(a_m)]}_{\check{c}_n}. \quad (14)$$

Since,

$$\sum_{m=1}^{l} \check{f}_i(a_m) = 0, \ \forall i \quad (15)$$

by taking $\check{c} = \bar{c} - \sum_{1\le i<j\le n} \check{c}_{ij} + \sum_{i=1}^{n} \check{c}_i$, it follows that the system (1) can be rewritten as

$$y[k] = \check{c} + \sum_{i=1}^{n} \check{f}_i(u[k-i]) + \sum_{1\le i<j\le n} \check{f}_{ij}(u[k-i], u[k-j]) + v[k], \ k = 1, 2, \ldots, Ll^n. \quad (16)$$

This makes the representation unique. For each $1 \le i < j \le n$, $m_i, m_j = 1, \ldots, l$ and $s = 1, 2, \ldots, L$, define the partial averages of the output,

$$Z^{ij}_{m_i m_j s} = \frac{1}{l^{n-2}} \sum_{\substack{t=1 \\ u[k-i]=a_{m_i}, u[k-j]=a_{m_j}}}^{l^n} y[(s-1)l^n + k]$$

$$\begin{aligned} Z^{ij}_{m_i m_j \cdot} &= \tfrac{1}{L}\textstyle\sum_{k=1}^{L} Z^{ij}_{m_i m_j s} \\ Z^{ij}_{m_i \cdot\cdot} &= \tfrac{1}{l}\textstyle\sum_{m_j=1}^{l} Z^{ij}_{m_i m_j \cdot} \\ Z^{ij}_{\cdot m_j \cdot} &= \tfrac{1}{l}\textstyle\sum_{m_i=1}^{l} Z^{ij}_{m_i m_j \cdot} \\ Z^{ij}_{\cdots} &= \tfrac{1}{l}\textstyle\sum_{m_i=1}^{l} Z^{ij}_{m_i \cdot\cdot} = \tfrac{1}{l}\textstyle\sum_{m_j=1}^{l} Z^{ij}_{\cdot m_j \cdot} \end{aligned} \quad (17)$$

The subscript “dot” indicates that average has been taken with respect to this variable, e.g., $Z^{ij}_{m_i m_j \cdot}$ is the average of $Z^{ij}_{m_i m_j s}$ with respect to the last variable s.

To provide a physical interpretation of the above variables, let us focus on the system (16) with $n = 3$, $l = 2$ and the Galois sequence $GF(2^3)$ as in (10) and (11). Within one period, it is clear that for any fixed column of U_{2^3}, half of the entries have values at a_1 and the other half are at a_2. Further, it is straightforward using (13) and (15) to show that for $i = 1$ and $j = 2$,

$$Z^{12}_{11s} = \check{c} + \check{f}_1(a_1) + \check{f}_2(a_1) + \check{f}_{12}(a_1, a_1) + (v[(s-1)2^3 + 1] + v[(s-1)2^3 + 8])/2,$$
$$Z^{12}_{12s} = \check{c} + \check{f}_1(a_1) + \check{f}_2(a_2) + \check{f}_{12}(a_1, a_2) + (v[(s-1)2^3 + 5] + v[(s-1)2^3 + 7])/2,$$

$$Z^{12}_{21s} = \check{c} + \check{f}_1(a_2) + \check{f}_2(a_1) + \check{f}_{12}(a_2, a_1) + (v[(s-1)2^3 + 2] + v[(s-1)2^3 + 6])/2,$$
$$Z^{12}_{22s} = \check{c} + \check{f}_1(a_2) + \check{f}_2(a_2) + \check{f}_{12}(a_2, a_2) + (v[(s-1)2^3 + 3] + v[(s-1)2^3 + 4])/2.$$

Moreover,

$$Z^{12}_{11\cdot} = \check{c} + \check{f}_1(a_1) + \check{f}_2(a_1) + \check{f}_{12}(a_1, a_1) + \frac{1}{L}\sum_{s=1}^{L}(v[(s-1)2^3 + 1] + v[(s-1)2^3 + 8])/2,$$

$$Z^{12}_{12\cdot} = \check{c} + \check{f}_1(a_1) + \check{f}_2(a_2) + \check{f}_{12}(a_1, a_2) + \frac{1}{L}\sum_{s=1}^{L}(v[(s-1)2^3 + 2] + v[(s-1)2^3 + 7])/2,$$

$$\begin{aligned} Z^{12}_{1\cdot\cdot} &= \check{c} + \check{f}_1(a_1) + \frac{1}{2L}\sum_{s=1}^{L}\{(v[(s-1)2^3 + 2] + v[(s-1)2^3 + 7])/2 \\ &\quad + (v[(s-1)2^3 + 1] + v[(s-1)2^3 + 8])/2\}, \\ Z^{12}_{\cdot\cdot\cdot} &= \check{c} + \frac{1}{4L}\sum_{t=1}^{L2^3} v[k]. \end{aligned}$$

Clearly, an estimate $\check{c}$ is obtained by $Z^{12}_{\cdot\cdot\cdot}$ and an estimate $\check{f}_1(a_1)$ is obtained by $Z^{12}_{1\cdot\cdot} - Z^{12}_{\cdot\cdot\cdot}$. The results can be trivially but cumbersomely extended to the system (16) with any $n \geq 2$, $l \geq 2$ and i, j as summarized in the following theorem.

Theorem 4 *Consider the system (16) for any $n \geq 2$, $l \geq 2$ with the Galois input as in (11) and the variables defined in (17). Then, for any $1 \leq i < j \leq n$ and $m_i, m_j = 1, \ldots, l$, we have*

$$Z^{ij}_{m_i m_j s} = \check{c} + \check{f}_i(a_{m_i}) + \check{f}_j(a_{m_j}) + \check{f}_{ij}(a_{m_i}, a_{m_j}) + \varepsilon^{ij}_{m_i m_j s}$$

where $\varepsilon^{ij}_{m_i m_j s}$'s are iid with zero mean and variance σ^2/l^{n-2} and

$$Z^{ij}_{m_i m_j \cdot} = \check{c} + \check{f}_i(a_{m_i}) + \check{f}_j(a_{m_j}) + \check{f}_{ij}(a_{m_i}, a_{m_j}) + \frac{1}{L}\sum_{s=1}^{L}\varepsilon^{ij}_{m_i m_j s},$$

$$Z^{ij}_{m_i \cdot\cdot} = \check{c} + \check{f}_i(a_{m_i}) + \frac{1}{lL}\sum_{m_j=1}^{l}\sum_{s=1}^{L}\varepsilon^{ij}_{m_i m_j s},$$

$$Z^{ij}_{\cdot m_j \cdot} = \check{c} + \check{f}_j(a_{m_j}) + \frac{1}{lL}\sum_{m_i=1}^{l}\sum_{s=1}^{L}\varepsilon^{ij}_{m_i m_j s},$$

$$Z^{ij}_{\cdot\cdot\cdot} = \check{c} + \frac{1}{ll^{n-2}L}\sum_{k=1}^{Ll^n} v[k].$$

Therefore, for a large L, very good estimates of $\check{c}$, $\check{f}_i$, and $\check{f}_{ij}$ are available from $Z^{ij}_{m_i m_j \cdot}$, $Z^{ij}_{m_i \cdot\cdot}$, $Z^{ij}_{\cdot m_j \cdot}$, and $Z^{ij}_{\cdots}$ that are computable from the input–output measurements. The implication of the above result is that the graph of $\check{f}_i(a_{m_i})$ $(\check{f}_j(a_{m_j}))$ versus a_{m_i} (a_{m_j}) is obtained by the graph of its estimate

$$\tilde{f}_i(a_{m_i}) = Z^{ij}_{m_i \cdot\cdot} - Z^{ij}_{\cdots} \ \text{ vs } \ a_{m_i} \ \text{ or}$$
$$\tilde{f}_j(a_{m_j}) = Z^{ij}_{\cdot m_j \cdot} - Z^{ij}_{\cdots} \ \text{ vs } \ a_{m_j}$$

and the graph of $\check{f}_{ij}(a_{m_i}, a_{m_j})$ versus (a_{m_i}, a_{m_j}) is obtained by $\tilde{f}_{ij}(a_{m_i}, a_{m_j}) = (Z^{ij}_{m_i m_j \cdot} - Z^{ij}_{m_i \cdot\cdot} - Z^{ij}_{\cdot m_j \cdot} + Z^{ij}_{\cdots})$ and

$$\tilde{f}_{ij}(a_{m_i}, a_{m_j}) \ \text{ vs } \ (a_{m_i}, a_{m_j}).$$

Accordingly, the contribution of $\check{f}_i(a_{m_i})$ and $\check{f}_{ij}(a_{m_i}, a_{m_j})$ can be visually inspected by the graphs of $\tilde{f}_i(a_{m_i})$ and $\tilde{f}_{ij}(a_{m_i}, a_{m_j})$. We make two comments here.

- Structural estimation is similar to model validation in identification. One can never validate a model unless all possible inputs have been applied. This is clearly impossible in practice. In structural estimation, one can only say that the contribution of $\check{f}_i(a_{m_i})$ or $\check{f}_{ij}(a_{m_i}, a_{m_j})$ is negligible with respect to the applied input. Therefore, the values $a_1, \ldots, a_l$ are important and have to be chosen judiciously.
- In general, increasing the level l excites the system at more points and this is quite useful for nonlinear system identification. However, there is a balance between the number of levels l and the complexity of the implementation. For $l = 2$ or any binary input, the minimum length of the sequence to cover all possible n-tuple combinations is 2^n and for an l level input, the minimum length becomes l^n. Thus, the complexity increases quickly as l gets larger.
- In general, a visual inspection works only for 2-factor terms.

5.2 Analysis of Variance (ANOVA)

The visual inspection approach discussed above is intuitive, efficient but Ad Hoc. If an estimate $\tilde{f}_i$ is nonzero but small, it is hard to determine if the term should be retained or discarded because of noise. To make the idea mathematically rigorous, in this section, we develop a statistical hypothesis test based on the well-known analysis of variance (ANOVA) and F distribution tests. To this end we make an assumption.

Assumption 5.1 The noise $v[\cdot]$ is iid Gaussian with zero mean and variance σ^2.

The Gaussian assumption is needed for the mathematical derivation. However, it has been well documented in the literature [17] that ANOVA is quite robust against violation of the Gaussian assumption. Consider the system (16), the input (11), and the variables (17). Let, for each $1 \leq i < j \leq n$,

$$\begin{array}{ll} SS^{ij}_T & = \sum_{m_i=1}^{l} \sum_{m_j=1}^{l} \sum_{s=1}^{L} (Z^{ij}_{m_i m_j s} - Z^{ij}_{\cdots})^2 \\ SS^{ij}_{m_i\cdot} & = \sum_{m_i=1}^{l} lL(Z^{ij}_{m_i\cdot\cdot} - Z^{ij}_{\cdots})^2 \\ SS^{ij}_{\cdot m_j} & = \sum_{m_j=1}^{l} lL(Z^{ij}_{\cdot m_j\cdot} - Z^{ij}_{\cdots})^2 \\ SS^{ij}_{\cdot\cdot} & = \sum_{m_i=1}^{l} \sum_{m_j=1}^{l} L(Z^{ij}_{m_i m_j\cdot} - Z^{ij}_{\cdot m_j\cdot} - Z^{ij}_{m_i\cdot\cdot} + Z^{ij}_{\cdots})^2 \\ SS^{ij}_E & = \sum_{m_i=1}^{l} \sum_{m_j=1}^{l} \sum_{s=1}^{L} (Z^{ij}_{m_i m_j s} - Z^{ij}_{m_i m_j\cdot})^2. \end{array} \tag{18}$$

The following theorem can be shown by some algebraic manipulations and the Cochran Theorem [24].

Theorem 5 *Consider the variables defined in (18). Then,*

- $SS^{ij}_T = SS^{ij}_{m_i\cdot} + SS^{ij}_{\cdot m_j} + SS^{ij}_{\cdot\cdot} + SS^{ij}_E$.
- $SS^{ij}_{m_i\cdot}$, $SS^{ij}_{\cdot m_j}$, $SS^{ij}_{\cdot\cdot}$, *and* SS^{ij}_E *are statistically independent.*
- $\frac{l^{n-2}}{\sigma^2} SS^{ij}_E \sim \chi^2(l^2(L-1))$ *is* χ^2 *distributed with* $l^2(L-1)$ *degrees of freedom.*
- *If* $\check{f}_{ij}(a_{m_i}, a_{m_j}) = 0$ *for all* $m_i, m_j = 1, \ldots, l$, *then*

$$\frac{l^{n-2}}{\sigma^2} SS^{ij}_{\cdot\cdot} \sim \chi^2((l-1)^2).$$

- *If* $\check{f}_i(a_{m_i}) = 0$ *for all* $m_i = 1, \ldots, l$, *then*

$$\frac{l^{n-2}}{\sigma^2} SS^{ij}_{m_i\cdot} \sim \chi^2(l-1).$$

- *If* $\check{f}_j(a_{m_j}) = 0$ *for all* $m_j = 1, \ldots, l$, *then*

$$\frac{l^{n-2}}{\sigma^2} SS^{ij}_{\cdot m_j} \sim \chi^2(l-1).$$

This theorem sets the foundation for the test of three null hypotheses,

$$\begin{array}{ll} H_{0ij}: & \check{f}_{ij}(a_{m_i}, a_{m_j}) = 0, \ \forall a_{m_i}, a_{m_j} = 1, \ldots, l, \\ H_{0i\cdot}: & \check{f}_i(a_{m_i}) = 0, \ \forall a_{m_i} = 1, \ldots, l, \\ H_{0\cdot j}: & \check{f}_j(a_{m_j}) = 0, \ \forall a_{m_j} = 1, \ldots, l, \end{array}$$

by the F-test because if H_{0ij} is true then

$$T^{ij} = \frac{SS^{ij}_{\cdot\cdot}/(l-1)^2}{SS^{ij}_E/(l^2(L-1))} \sim F((l-1)^2, l^2(L-1)), \text{ for all } 1 \leq i < j \leq n,$$

is F-distributed with $(l-1)^2$ and $l^2(L-1)$ degrees of freedom. Similarly, if $H_{0i\cdot}$ is true,

$$T^1 = \frac{SS^{12}_{m_i\cdot}/(l-1)}{SS^{12}_E/(l^2(L-1))} \sim F(l-1, l^2(L-1))$$

and if $H_{0\cdot j}$ is true, $\forall j = 2, \dots, n$,

$$T^j = \frac{SS^{1j}_{\cdot m_j}/(l-1)}{SS^{1j}_E/(l^2(L-1))} \sim F(l-1, l^2(L-1)).$$

The null hypothesis H_{0ij} is rejected if $T^{ij} > F_\alpha((l-1)^2, l^2(L-1))$ where α denotes the level of significance, usually in the range $0.01 - 0.1$. The tests for $H_{0i\cdot}$ and $H_{0\cdot j}$ are similar. The results from the hypothesis tests are used to determine which f_i or f_{ij} should be retained with a certain confidence in probability.

6 Full Scale Identification

For full scale system identification, using the Galois sequence is not appropriate because the Galois sequence only excites the system at a finite points. We assume in this section that the input $u[k]$ is an iid random sequence in a (unknown) open interval $I \in R$ with a (unknown) probability density function $\psi(\cdot)$. Then, the results of [3] can be used. Similar to the structural estimation case, the system (1) needs to be normalized for identification purposes. Let $\mathbf{E}$ be the expectation operator. Define the partial averages,

$$\begin{aligned}
c_{ij} &= \mathbf{E}\{\bar{f}_{ij}(u[k-i], u[k-j])\},\\
c_1 &= \mathbf{E}\{\bar{f}_1(u[k-1]) + \sum_{j=2}^{n} \mathbf{E}(\bar{f}_{1j}(u[k-1], u[k-j]) \mid u[k-1] = x_1)\},\\
c_i^1 &= \mathbf{E}\{\bar{f}_i(u[k-i]) + \sum_{j=i+1}^{n} \mathbf{E}(\bar{f}_{ij}(u[k-i], u[k-j]) \mid u[k-i] = x_i)\},\\
c_i^2 &= \sum_{j=1}^{i-1} \mathbf{E}(\bar{f}_{ji}(u[k-j], u[k-i]) \mid u[k-i] = x_i),\\
c_n &= \mathbf{E}\{\bar{f}_n(u[j-n]) + \sum_{j=1}^{n-1} \mathbf{E}(\bar{f}_{jn}(u[k-j], u[k-n]) \mid u[k-n] = x_n)\}.
\end{aligned}$$

Now, for every x_i and $x_j \in I$, define

$$\begin{aligned}
f_{ij}(x_i, x_j) = \bar{f}_{ij}(x_i, x_j) &- \mathbf{E}(\bar{f}_{ij}(u[k-i], u[k-j]) \mid u[k-j] = x_j)\\
&- \mathbf{E}(\bar{f}_{ij}(u[k-i], u[k-j]) \mid u[k-i] = x_i) + c_{ij}, \quad 1 \le i < j \le n,
\end{aligned}$$

$$\begin{aligned}
f_1(x_1) &= \bar{f}_1(x_1) + \sum_{j=2}^{n} \mathbf{E}(\bar{f}_{1j}(u[k-1], u[k-j]) \mid u[k-1] = x_1) - c_1, \\
f_i(x_i) &= \bar{f}_i(x_i) + \sum_{j=i+1}^{n} \mathbf{E}(\bar{f}_{ij}(u[k-i], u[k-j]) \mid u[k-i] = x_i) \\
&\quad + \sum_{j=1}^{i-1} \mathbf{E}(\bar{f}_{ji}(u[k-j], u[k-i]) \mid u[k-i] = x_i) - c_i^1 - c_i^2, \ \ i = 2, 3, \ldots, n-1, \\
f_n(x_n) &= \bar{f}_n(x_n) + \sum_{i=1}^{n-1} \mathbf{E}(\bar{f}_{in}(u[k-i], u[k-n]) \mid u[k-n] = x_n) - c_n.
\end{aligned} \tag{19}$$

Next, with $c = \bar{c} - \sum_{1\le i<j\le n} c_{ij} + \sum_{i=1}^{n} c_i$, $c_i = c_i^1 + c_i^2$, the system (1) can be written as

$$y[k] = c + \sum_{i=1}^{n} f_i(u[k-i]) + \sum_{1\le i<j\le n} f_{ij}(u[k-i], u[k-j]) + v[k], \ \ k = 1, 2, \ldots, N \tag{20}$$

with

$$\begin{aligned}
\mathbf{E} f_i(u[k-i]) &= \mathbf{E}(f_{ij}(u[k-i], u[k-j]) \mid u[k-i] = x_i) \\
&= \mathbf{E}(f_{ij}(u[k-i], u[k-j]) \mid u[k-j] = x_j) = 0.
\end{aligned}$$

The problem is how to identify f_i and f_{ij}. Observe that these variables are conditional expectations and thus can be calculated by empirical data easily, for instance using the kernel estimation method [3]. To this end, we define the kernel functions. A continuous, bounded and radially symmetric function $K(\cdot)$ is said to be a kernel function if

$$K(z) = \begin{cases} > 0, & z \in [-1, 1] \\ 0, & z \notin [-1, 1] \end{cases} \quad \text{and} \quad \int_{-1}^{1} K(z)dz = 1. \tag{21}$$

Now, the estimates of c, f_i and f_{ij} can be defined for each $x_i, x_j \in I$ in which the input $u[\cdot]$ lies,

$$\hat{c} = \frac{1}{N}\sum_{k=1}^{N} y[k] \tag{22}$$

$$\hat{f}_i(x_i) = \frac{\sum_{k=1}^{N} K(\frac{x_i - u[k-i]}{\delta}) y[k]}{\sum_{k=1}^{N} K(\frac{x_i - u[k-i]}{\delta})} - \hat{c}, \ \ i = 1, \ldots, n$$

$$\hat{f}_{ij}(x_i, x_j) = \frac{\sum_{k=1}^{N} K(\frac{\|(x_i, x_j) - (u[k-i], u[k-j])\|}{\delta}) y[k]}{\sum_{k=1}^{N} K(\frac{\|(x_i, x_j) - (u[k-i], u[k-j])\|}{\delta})} - \hat{f}_i(x_i) - \hat{f}_j(x_j) - \hat{c}, \ \ 1 \le i < j \le n$$

where $\delta > 0$ is the bandwidth. The following result, which is a standard exercise, follows from [3].

Theorem 6 *Consider the system (3) with differentiable f_i and f_{ij}, and any kernel function defined above. Then, for any $x_i, x_j \in I$, provided that the input density function is positive at x_i, x_j, i.e., $\psi(x_i), \psi(x_j) > 0$ and $\delta \rightarrow 0$, $\delta^2 N \rightarrow \infty$ as $N \rightarrow \infty$, we have*

$$\hat{c} \rightarrow c$$

$$\hat{f}_i(x_i) \rightarrow f_i(x_i)$$

$$\hat{f}_{ij}(x_i, x_j) \rightarrow f_{ij}(x_i, x_j)$$

in probability as $N \rightarrow \infty$.

7 Comparisons with Existing Methods

A new representation for a class of nonlinear nonparametric system has been proposed in (16). Further, structural estimation and full scale identification have been discussed in the previous section. Naturally, two questions arise. The first one is what are the advantages of the representation (16) as compared to some existing methods, in particular the fixed basis approach and the Volterra series? Second, even if one accepts the representation (16), why use the structural estimation and system identification techniques discussed in the previous section as compared to the traditional approach of identifying $f(u[k-1], \ldots, u[k-n])$ directly? We address these two issues in this section.

7.1 Relation with the Volterra Series

If the system (16) is smooth with an upper bound n on the time lag, its Volterra series is given by

$$y[k] = h_0 + \sum_{l=1}^{n} \sum_{i_1=0}^{\infty} \sum_{i_2=i_1}^{\infty} \cdots \sum_{i_l=i_{l-1}}^{\infty} h_l(i_1, \ldots, i_l) \cdot u[k-i_1]u[k-i_2] \ldots u[k-i_l] + v[k].$$

Two of the major advantages of the Volterra series are (1) it is in a closed form and (2) it is parametric. In other words, any smooth nonlinear nonparametric system can always be written in the above form. Further, identification becomes a linear estimation of the coefficients h_l's. However, the Volterra series also has some disadvantages. In this work, we are mainly interested in verifying if the Volterra series is a good candidate for the system of short term memory and low degree of interaction as in (1) or (3). To this end, we need to understand the differences between a system of low degree of interaction and a system of low order in the classical sense. Traditionally, a system is said to be of low order if it can be written as or at least can be well approximated by a low-order multidimensional polynomial. For instance, a system is said to be first order if it is linear

$$y[k] = f(u[k-1], \dots, u[k-n]) = c + \sum_{i=1}^{n} \alpha_i u[k-i]$$

or to be of second order if

$$y[k] = c + \sum_{i=1}^{n} \alpha_i u[k-i] + \sum_{1 \le j_1 \le j_2 \le n} \gamma_{j_1 j_2} u[k-j_1]u[k-j_2].$$

Clearly, in both cases, the system is of 1-factor or 2-factor terms. In general, a system of low order in the traditional sense implies low degree of interaction. The other way around is however incorrect. For example, $e^{u[k-1]}$ is an 1-factor term that is not necessarily of low order depending on the input magnitude. Also, $(u[k-1]u[k-2])^{10}$ is a 2-factor term which may not be approximated well by a second-order polynomial. Therefore, nonlinear systems of low order in the traditional sense are low degree interaction systems but the reverse implication is not necessarily true. Now, we consider a Volterra series approach. A second-order Volterra series is a model that contains all the first- and second-order kernels $u[k-i]$'s and $u[k-j_1]u[k-j_2]$'s. This model is a 2-factor interaction system. However, a 2-factor system $y[k] = e^{u[k-1]} + (u[k-1]u[k-2])^{10}$ is definitely not represented well by a low-order Volterra series.

In summary, if a nonlinear system of short-term memory and low degree of interaction resembles the structure of a low-order multidimensional polynomial, the Volterra series is a good candidate. If the system is far away from a polynomial or the order of the polynomial is high, the Volterra series is not a good candidate simply because too many terms are needed to approximate the given system. In such a case, i.e., the unknown system is of low degree of interaction but not necessarily a low-order polynomial, the proposed representation is a vital choice. This observation is not surprising because the Volterra series is an extension of Taylor polynomial expansion of an analytic function. The advantages of the proposed representation for systems of short memory and low degree of interaction will be further illustrated in the simulation section.

7.2 Basis Function Approach

Without structural information, a fixed basis function approach is often used in nonlinear system identification. Typical basis functions are Fourier series, polynomials, and some orthogonal versions. Obviously, the success of a basis function approach relies on how much a priori information is available on the unknown structure. If the chosen basis functions resemble the structure of the unknown nonlinear system, only a few terms are needed to represent the unknown system. In this case, identification is likely to be successful. Otherwise, a fixed basis function approach requires a large number of terms which has a considerable negative effect on the identification step. The advantage of the proposed representation is that, if a nonlinear system has short-term memory and low degree of interaction which fits (3), then no additional structural information is required. In other words, there is no need to choose any basis functions and whether a chosen basis function resembles the unknown structure is no longer an issue.

7.3 Traditional One Shoot Kernel Approach

Once the representation of (1) or (3) is accepted, the second question is why to use the identification method proposed in the previous section and why not to identify the nonlinear function $f(u[k-1], \dots, u[k-n])$ directly, which is a traditional approach. The difference is that the identification method proposed in this work decomposes a potentially high-dimensional nonlinear identification problem into a number of one- or two-dimensional problems. Since the method proposed in the work is kernel based, we compare it with the one shoot kernel based identification method.

First, for the one shoot kernel estimation of $f(u[k-1], \dots, u[k-n])$ under iid inputs, the asymptotic convergence rate [12] is $O(N^{-\frac{\alpha}{2\alpha+n}})$, where N is the total number of data points and α depends on the choices of the kernel functions and the bandwidth. For the method proposed in the work, because identification is one or two dimensional, the asymptotic convergence rate is $O(N^{-\frac{\alpha}{2\alpha+n}}|_{n=2}) = O(N^{-\frac{\alpha}{2\alpha+2}})$ [12]. Thus, asymptotically, there is an advantage to use the proposed method.

Next, we consider the case that N is large but fixed. For nonlinear system identification, the curse of dimensionality is always a concern even for a modest n. We use similar arguments and examples as in [2] to illustrate the situation. Let $u[\cdot]$ be uniformly distributed in $I = [-1, 1]$. Suppose one wants to estimate $f(x_1, x_2, \dots, x_n)$ at a point $(x_1, x_2, \dots, x_n) \in I^n$. Since any nonparametric identification scheme, including the kernel approach, is in some form of local smoother or weighted average based on the measurement data in the neighborhood of $(x_1, x_2, \dots, x_n)$, there must be enough data in the neighborhood to average out the effects of noise and the uncertainty due to lack of structural information. For simplicity, suppose the neighborhood is a hyper-box with the side length 0.1. Then, the volume of I^n is 2^n and the volume of the neighborhood is 0.1^n. This implies that the probability that a measurement data $(u[k-1], u[k-2], \dots, u[k-n])$ is in the neighborhood of $(x_1, x_2, \dots, x_n)$ is $(1/20)^n$ that goes to zero exponentially as n gets large. For a large N, there are likely $N \cdot (1/20)^n$ measurements in the neighborhood. Unless N is huge, there is not enough data in a neighborhood for identification purpose even for a modest n. For the proposed method, however, the maximum dimension is two. The curse of dimensionality is not a problem. For instance, let $n = 8$. Then, the problem becomes identification of 8 1-factor terms $f_j(u[k-j])$, $j = 1, 2, \ldots, 8$, and 28 2-factor terms $f_{j_1 j_2}(u[k-j_1], u[k-j_2])$. Though the number of identification steps increases, the complexity of identification is reduced drastically. Because of decoupling, the probability of an $u[k-j]$ in the neighborhood of x_j for one-dimensional identification is 0.05 and the probability of $(u[k-j_1], u[k-j_2])$ in the neighborhood of (x_{j_1}, x_{j_2}) is 0.0025. Suppose that the total number of data points is $N = 10^5$. This implies that likely there are 5000 or 250 measurements in the neighborhood for identification of 1-factor or 2-factor terms, respectively. Recall that if the eight-dimensional $f(x_1, \ldots, x_8)$ is identified directly, the probability that a data vector is in the neighborhood of $(x_1, \ldots, x_8)$ is $(1/20)^8$. With $N = 10^5$, the probability that there is one measurement in a neighborhood is $(1/2)^8 \cdot 10^{-3} = \frac{1}{2^8 10^3}$ that makes identification nearly impossible. Clearly, the performance of identification of the 1-factor or 2-factor term can be substantially improved for the same N, compared to the identification of a eight-dimensional problem f. This effectively combats the curse of dimensionality.

8 Numerical Simulation

We now provide numerical simulation examples. We separate the discussions about random inputs and Galois sequence inputs.

8.1 *Random Inputs*

Example 1 Consider a nonlinear system

$$y[k] = f(u[k-1], u[k-2], u[k-3], u[k-4], u[k-5]) + v[k]$$

$$= \underbrace{1.25/3}_{\phi_0=c} + \underbrace{u[k-1]}_{\phi_1=f_1} + \underbrace{10 \cdot u[k-2]^3}_{\phi_2=f_2} + \underbrace{5 \cdot u[k-3]^2 - 1.25/3}_{\phi_3=f_3}$$

$$+ \underbrace{0}_{\phi_4=f_4} + \underbrace{0}_{\phi_5=f_5} + + \underbrace{5 \cdot u[k-1] * u[k-2]}_{\phi_6=f_{12}} + \underbrace{0}_{\phi_7=f_{13}} +$$

$$\underbrace{0}_{\phi_8=f_{14}} + \underbrace{0}_{\phi_9=f_{15}} + \underbrace{0.5 \cdot sin(2\pi(u[k-2] + u[k-3]])}_{\phi_{10}=f_{23}}$$

$$+ \underbrace{0}_{\phi_{11}=f_{24}} + + \underbrace{0}_{\phi_{12}=f_{25}} + + \underbrace{0}_{\phi_{13}=f_{34}} + + \underbrace{0}_{\phi_{14}=f_{35}} + + \underbrace{0}_{\phi_{15}=f_{45}} + v[k] \tag{23}$$

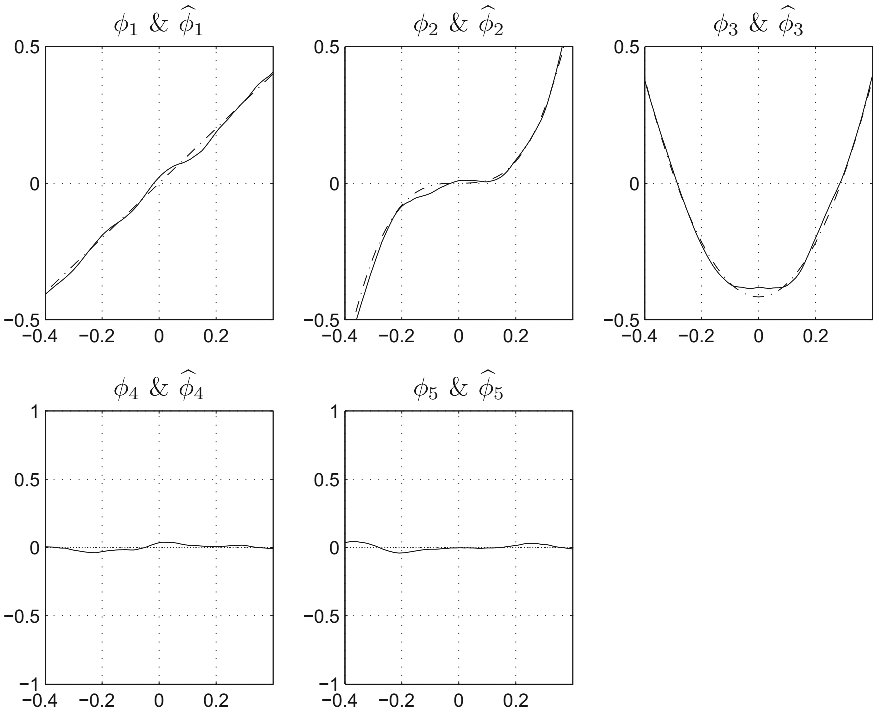
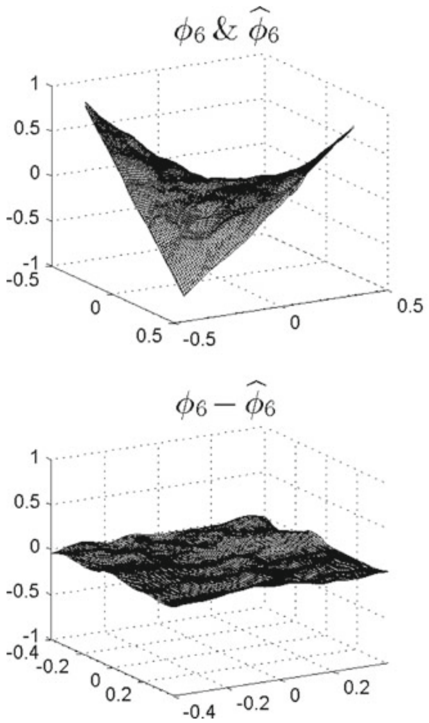


Fig. 1 $\phi_j[k] = f_j(u[k-j])$'s (solid) and their estimates $\widehat{\phi}_j[k]$ (dashdot), $j = 1, 2, 3, 4, 5$

No prior structural information on f is available. The time lag of the system is unknown and only an upper bound of $n = 5$ is assumed. For simulation, $N = 20,000$ and $\delta = 0.1$. The input $u[\cdot]$ is independent and uniformly distributed in $[-0.5, 0.5]$, and the noise $v[\cdot]$ is iid Gaussian with $SNR = 20\,\text{dB}$.

Figure 1 shows the actual but unknown $\phi_j[k]$(solid), $j = 1, ..., 5$ and their estimates $\widehat{\phi}_j[k]$ (dashdot), $j = 1, ..., 5$, respectively. The top diagrams of Fig. 2 show $\phi_6[k]$, $\phi_{10}[k]$ superimposed with their estimates $\widehat{\phi}_6[k]$, $\widehat{\phi}_{10}[k]$. The estimation errors of $\phi_6[k] - \widehat{\phi}_6[k]$ and $\phi_{10}[k] - \widehat{\phi}_{10}[k]$ are in the bottom diagrams. The estimates $\widehat{\phi}_j[k]$'s, $j = 7, 8, 9, 11, 12, 13, 14$ and 15 are in Fig. 3. It can be seen that all the estimates fit the actual but unknown functions well.

To determine the order of the estimation model, we calculate the residual and plot the average error as a function of the estimation order p as in the top diagram of Fig. 4. Obviously, there is a drastic reduction in the average error for the order $p = 10$ and there is a little change for $p > 10$. Thus, we take $p = 10$ and test if the order $p = 10$ is acceptable by the modified Box–Pierce test (9). When $p = 10$, $Q_{n-1} = Q_4 = 5.6434$. Let the level of significance be 0.05. This corresponds to, from the $\chi^2(n-1) = \chi^2(4)$ distribution table, the threshold $d = 9.4877$. Since $Q_4 = 5.6434 < d = 9.4877$. The order $p = 10$ is accepted which is in fact the actual but unknown order. The order determination can also be carried by the relative contribution $R_c[p]$ shown



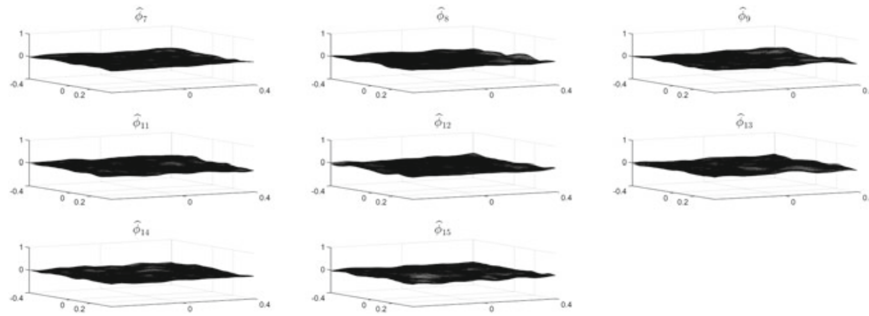
ϕ_6 & $\widehat{\phi_6}$

ϕ_{10} & $\widehat{\phi_{10}}$

$\phi_6 - \widehat{\phi_6}$

$\phi_{10} - \widehat{\phi_{10}}$

Fig. 2 $\phi_6[k]$, $\widehat{\phi_6}[k]$ and $\phi_{10}[k]$, $\widehat{\phi_{10}}[k]$



$\widehat{\phi}_7$ $\widehat{\phi}_8$ $\widehat{\phi}_9$

$\widehat{\phi}_{11}$ $\widehat{\phi}_{12}$ $\widehat{\phi}_{13}$

$\widehat{\phi}_{14}$ $\widehat{\phi}_{15}$

Fig. 3 $\widehat{\phi}_j[k]$, $j = 7, 8, 9, 11, 12, 13, 14$ and 15

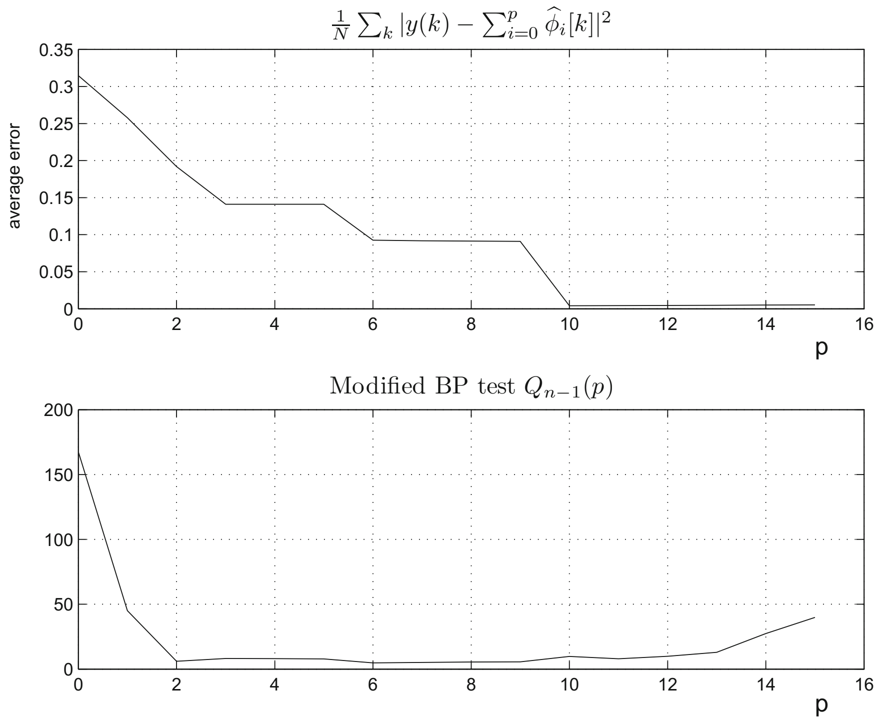
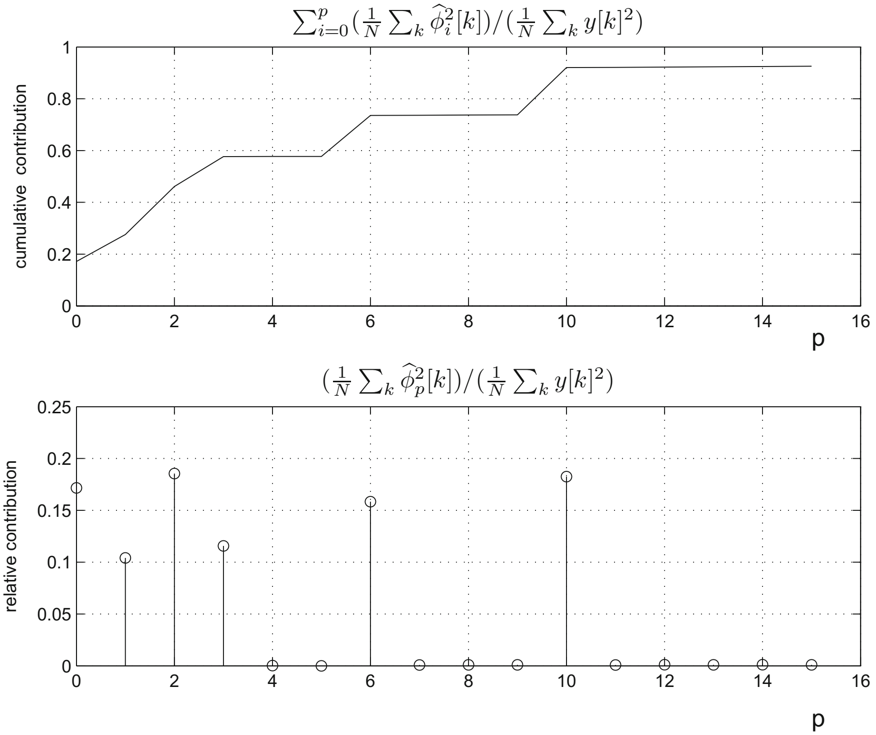


Fig. 4 Average error versus the estimation order

Table 1 Relative contributions for N = 20000, 10000, 5000 and $d_1 = 0.03$, respectively

N	20000	$\geq d_1$	10000	$\geq d_1$	5000	$\geq d_1$
$\widehat{R}_c[0]$	0.1717	√	0.1819	√	0.1793	√
$\widehat{R}_c[1]$	0.1041	√	0.0802	√	0.0792	√
$\widehat{R}_c[2]$	0.1854	√	0.1480	√	0.1504	√
$\widehat{R}_c[3]$	0.1156	√	0.1001	√	0.1152	√
$\widehat{R}_c[4]$	0.0002		0.0003		0.0005	
$\widehat{R}_c[5]$	0.0000		0.0000		0.0000	
$\widehat{R}_c[6]$	0.1584	√	0.1464	√	0.1574	√
$\widehat{R}_c[7]$	0.0008		0.0021		0.0035	
$\widehat{R}_c[8]$	0.0009		0.0028		0.0045	
$\widehat{R}_c[9]$	0.0009		0.0029		0.0036	
$\widehat{R}_c[10]$	0.1826	√	0.1199	√	0.1176	√
$\widehat{R}_c[11]$	0.0007		0.0019		0.0039	
$\widehat{R}_c[12]$	0.0010		0.0026		0.0050	
$\widehat{R}_c[13]$	0.0009		0.0019		0.0090	
$\widehat{R}_c[14]$	0.0011		0.0026		0.0040	
$\widehat{R}_c[15]$	0.0010		0.0023		0.0050	


$\sum_{i=0}^{p}(\frac{1}{N}\sum_k \widehat{\phi}_i^2[k])/(\frac{1}{N}\sum_k y[k]^2)$

$(\frac{1}{N}\sum_k \widehat{\phi}_p^2[k])/(\frac{1}{N}\sum_k y[k]^2)$

Fig. 5 Cumulative and relative contributions

in Table 1 as well as in the bottom diagram of Fig. 5. The cumulative contribution $C_c[p]$ is shown in the top diagram of Fig. 5. To determine which term $\widehat{\phi}_j$ should be included in the estimate, let the threshold $d_1 = 0.03$. If $\widehat{R}_c[j] \geq d_1$, we include the corresponding term $\widehat{\phi}_j$ in the model. Otherwise the contribution of the corresponding term is deemed to be insignificant and omitted in the model. Clearly, from Table 1, only the terms $\widehat{\phi}_0$, $\widehat{\phi}_1$, $\widehat{\phi}_2$, $\widehat{\phi}_3$, $\widehat{\phi}_6$ and $\widehat{\phi}_{10}$ contribute significantly and should be included in the model. Simply put, the system time lag is determined to be $n = 3$, though the upper bound is assumed to be 5. Further, it is determined that the system contains only 6 terms, $\phi_0 = c$, $\phi_1 = f_1$, $\phi_2 = f_2$, $\phi_3 = f_3$, $\phi_6 = f_{12}$, and $\phi_{10} = f_{23}$ and all other terms are zero. The conclusion is consistent with the true but unknown system.

Finally, to validate the obtained estimate $\widehat{f} = \sum_{i=0,1,2,3,6,10} \widehat{\phi}_i[k]$, a fresh input

$$u[k] = 0.5\sin(k/10) \cdot \cos(k/20), \;\; k = 1..., 150$$

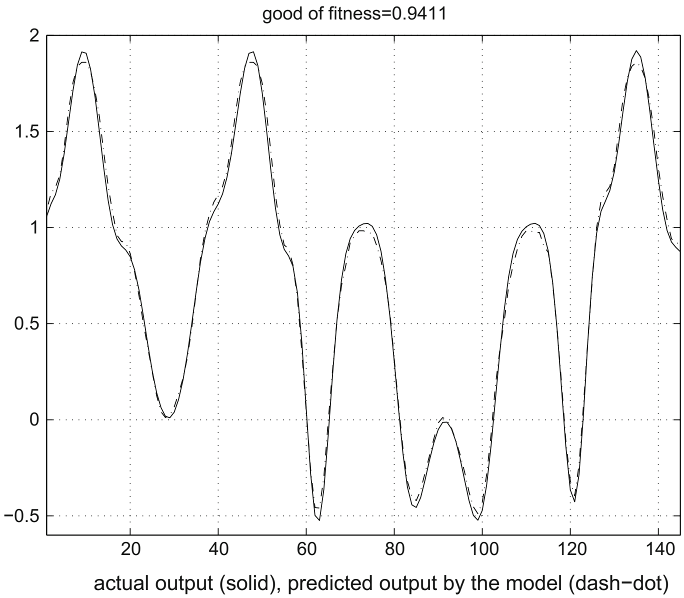


Fig. 6 Actual output (solid) and predicted output (dash-dot) for a fresh input

is generated which is completely different from the white noise input that was used for identification. A standard goodness-of-fit criterion

$$(1 - \sqrt{\frac{\sum_k (y[k] - \widehat{y}[k])^2}{\sum_k (y[k] - \frac{1}{N}\sum_k y[k])^2}}) \times 100\% \qquad (24)$$

is calculated. Based on the fresh input, the output $y[k]$ of the actual but unknown nonlinear system (23) is generated as well as the predicted output $\widehat{y}[k]$ based on the estimate

$$\begin{aligned} \widehat{y}[k] &= \widehat{f}(u[k-1], [u-2], u[k-3], u[k-4], u[k-5]) \\ &= \widehat{\phi}_0 + \widehat{\phi}_1[k] + \widehat{\phi}_2[k] + \widehat{\phi}_3[k] + \widehat{\phi}_6[k] + \widehat{\phi}_{10}[k]. \end{aligned}$$

Figure 6 shows the actual output $y[k]$ (solid) and the predicted output $\widehat{y}[k]$ (dash-dot) with the goodness-of-fit 0.9411, an almost perfect fit. This validates the effectiveness of the identification method proposed in the work along with its order determination and regressor selection.

8.2 Galois Sequence Inputs

In this subsection, we discuss two numerical examples that shed lights on the efficiency of the proposed representation and identification method using Galois sequence inputs in the context of existing methods.

Example 2

$$
\begin{aligned}
w[k] &= u[k] - 0.3u[k]^3 \\
x[k] &= 0.3x[k-1] - 0.02x[k-2] + 0.5w[k-1] + 0.4w[k-2] \\
y[k] &= x[k] + 0.4x[k]^2 + v[k]
\end{aligned}
$$

The noise $v[k]$ is an iid zero mean and unit variance Gaussian random variable multiplied by 0.2. The actual nonlinear system is IIR and therefore there are no exact f_i and f_{ij}. We represent the system by (3) assuming that the maximum time lag $n \leq 8$. Note determination of the order of an unknown nonlinear system is an interesting and open problem which is out of scope of the work. Here we just assume that the upper bound $n = 8$ is available (admittedly it could be restrictive in some applications).

First, structural estimation is carried out by using a binary Galois sequence $GF(2^8)$ with $n = 8, l = 2$ and $L = 11$ and $a_1 = 1$, $a_2 = 0$. ANOVA was used to calculate T^{ij} and T^i as shown in Table 2 that are the averages of 50 Monte Carlo simulations.

For the hypothesis tests, we choose $\alpha = 0.1$. From the F distribution, we have $F_{0.1}(1, 40) = 2.84$. By the F-tests, we have T^1, T^2, T^3, T^4, T^{12}, T^{13}, $T^{23} > 2.84$, and all other T^i, $T^{ij} < 2.84$ as can be seen in Table 1. Thus, we reject the hypotheses that f_1, f_2, f_3, f_4, f_{12}, f_{13}, and f_{23} are negligible and assume that all other terms are zero. Second, these non-negligible terms are identified with iid input uniformly in $[-1.5, 1.5]$, a triangle kernel [3] with $\delta = 0.4$ and the total number of data points $N = 5000$. Further, their estimates are used to construct the model

Table 2 Calculated T^i and T^{ij} for polynomial input nonlinearity

i	1	2	3	4	5	6	7	8
T^i	3986	4617	371.5	23.3	2.5	1	1.2	1.1

T^{ij}		j						
		2	3	4	5	6	7	8
i	1.0	56	49	1.2	1.1	0.9	0.6	1.1
	2		5.8	1.5	1,.0	0.9	0.7	1.0
	3			1.2	1.3	0.7	1.2	0.9
	4				1.3	1.0	1.0	0.9
	5					1.0	0.9	0.8
	6						0.9	0.7
	7							0.8

Table 3 Goodness-of-fits for the polynomial input nonlinearity

	Proposed method	Fourth-order Volterra	Second-order fixed basis	Traditional one shoot
Gof	0.9470	0.9563	0.8121	0.6762

$$\begin{aligned}\hat{y}[k] &= \hat{c} + \hat{f}_1(u[k-1]) + \hat{f}_2(u[k-2]) + \hat{f}_3(u[k-3]) + \hat{f}_4(u[k-4]) \\ &\quad + \hat{f}_{12}(u[k-1], u[k-2]) + \hat{f}_{13}(u[k-1], u[k-3]) + \hat{f}_{23}(u[k-2], u[k-3]).\end{aligned}$$

To validate the model, the input is generated

$$u[k] = 1.5\sin(k/10)\cos(k/20),\ \ k = 1, \ldots, 160$$

as well as the corresponding actual outputs $y[k]$ and predicted outputs $\hat{y}[k]$'s.

Figures 7, 8, 9, and 10 show $y[k]$, $\hat{y}[k]$'s predicted by the proposed method, the Volterra series of fourth order, a fixed basis of polynomial upto the second order and the one shoot method respectively as well as their gof's. Since the actual nonlinearity is a polynomial, the proposed method, the Volterra series, and the fixed basis of polynomial all perform satisfactory, significantly better than the one shoot method as expected. An overview of the performances is given in Table 3.

Example 3

$$\begin{aligned}w[k] &= u[k] - 0.3u[k]^3 e^{1.4u[k]} \\ x[k] &= 0.3x[k-1] - 0.02x[k-2] + 0.5w[k-1] + 0.4w[k-2] \\ y[k] &= x[k] + 0.4x[k]^2 + v[k].\end{aligned}$$

The only difference between Examples 2 and 3 is that the input nonlinearity now contains an exponential term. All other simulation conditions remain the same. T^i and T^{ij} for Example 3 are given in Table 4 for a binary test input $GF(l^n)$ with $n = 8, l = 2$ and $L = 11$.

With $\alpha = 0.1$ and by the F-test as shown in Table 4, only the terms f_1, f_2, f_3, f_4, f_5, f_{12}, f_{13}, f_{14}, f_{23}, and f_{24} are not negligible and thus the model is given by

$$\begin{aligned}\hat{y}[k] &= \hat{c} + \hat{f}_1(u[k-1])\hat{f}_2(u[k-2]) + \hat{f}_3(u[k-3]) + \hat{f}_4(u[k-4]) + \hat{f}_5(u[k-5]) \\ &\quad + \hat{f}_{12}(u[k-1], u[k-2]) + \hat{f}_{13}(u[k-1], u[k-3]) + \hat{f}_{14}(u[k-1], u[k-4]) \\ &\quad + \hat{f}_{23}(u[k-2], u[k-3]) + \hat{f}_{24}(u[k-2], u[k-4]).\end{aligned}$$

Under the same validation input, the corresponding $y[k]$ and predicted $\hat{y}[k]$ by various methods are shown in Figs. 11, 12, 13 and 14. The corresponding gof's are given in Table 5.

The results of the second-, third-, fourth-, fifth-, and sixth-order Volterra series are also shown in Table 5 and Fig. 12, exhibiting a considerable performance dete-

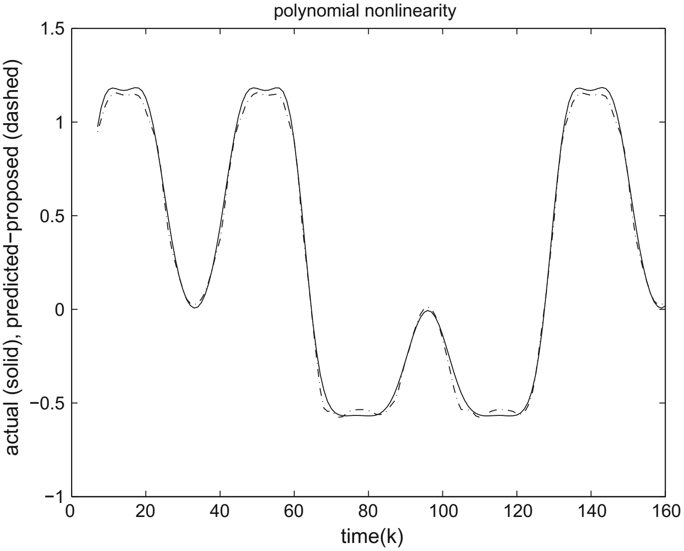


Fig. 7 Actual $y[k]$ and predicted $\hat{y}[k]$ by the proposed method with gof $= 0.9470$ (polynomial nonlinearity)

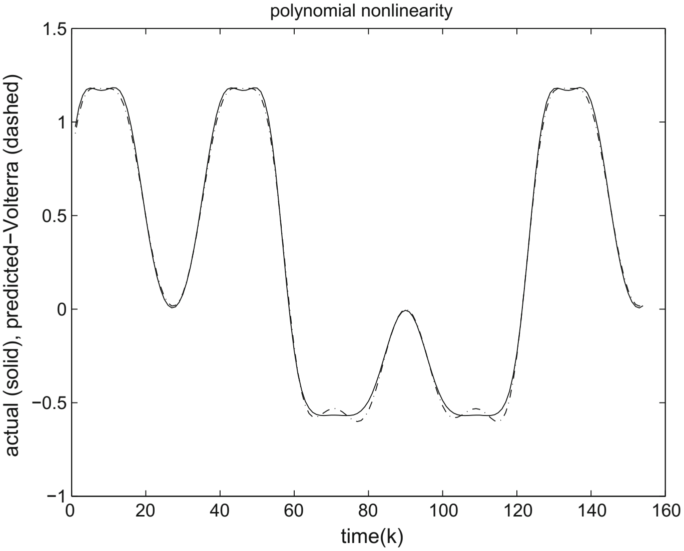


Fig. 8 Actual $y[k]$ and predicted $\hat{y}[k]$ by an fourth-order Volterra with gof $= 0.9563$ (polynomial nonlinearity)

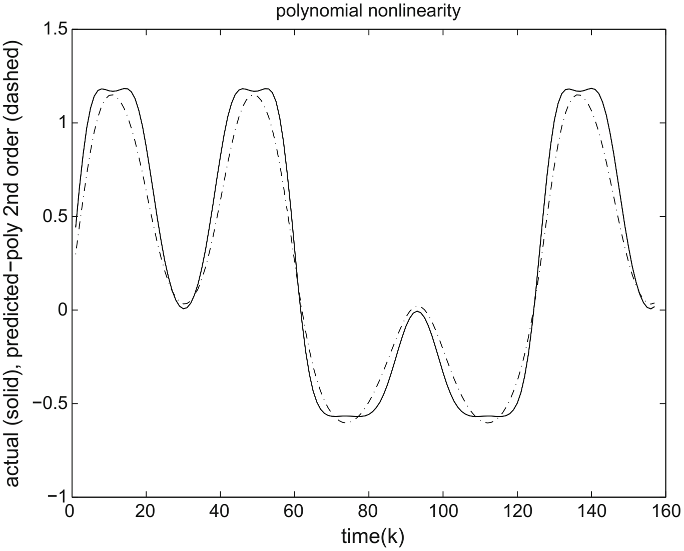


Fig. 9 Actual $y[k]$ and predicted $\hat{y}[k]$ by a second polynomial with gof = 0.8121 (polynomial nonlinearity)

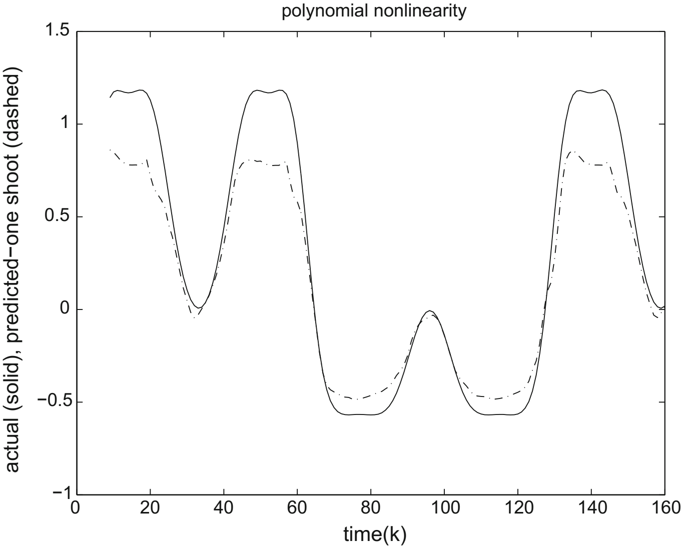


Fig. 10 Actual $y[k]$ and predicted $\hat{y}[k]$ by one shoot method with gof = 0.6762 (polynomial nonlinearity)

Table 4 T^i and T^{ij} for exponential nonlinearity

i	1	2	3	4	5	6	7	8
T^i	25784	30338	2336	123	8	1	1	1

T^{ij}		j						
		2	3	4	5	6	7	8
i	1	846	65	4	1	1	1	1
	2		78	5	1	1	1	1
	3			1	1	1	1	1
	4				1	1	1	1
	5					1	1	1
	6						1	1
	7							1

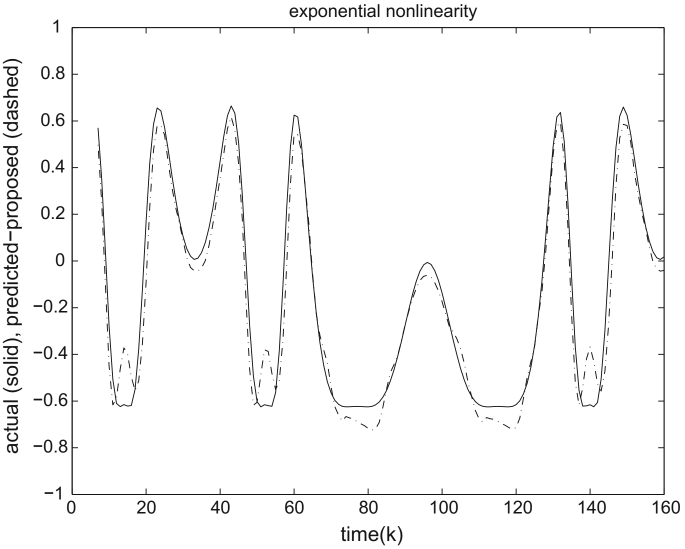


Fig. 11 Actual $y[k]$ and predicted $\hat{y}[k]$ by the proposed method (exponential nonlinearity)

rioration. This is because a low-order polynomial approximation in $u[\cdot]$ like the Volterra series is inefficient to model an exponential function. This demonstrates the advantage of the proposed representation along with structural estimation and system identification for nonlinear nonparametric system of short-term memory and low degree of interaction. It is interesting to note that a higher order Volterra does not necessarily imply a better identification result because variance error also increases as the order gets high. The gofs of the fixed basis function for the second- and third-order polynomials are 0.2299 and 0.1659, respectively. Figure 13 demonstrates the

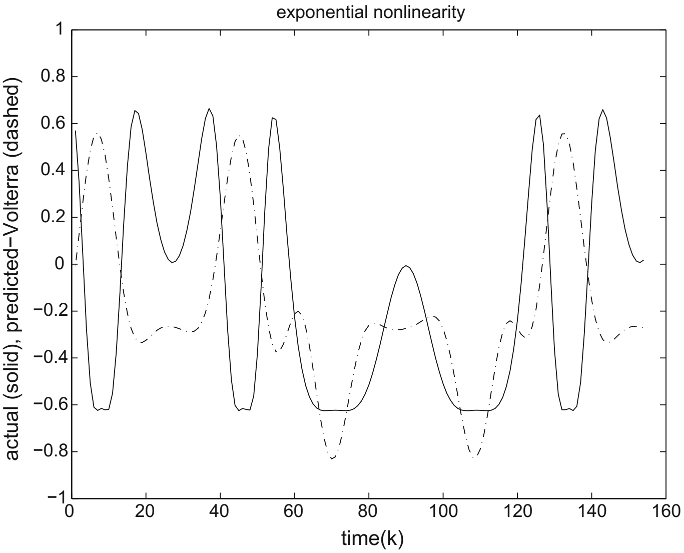


Fig. 12 Actual $y[k]$ and predicted $\hat{y}[k]$ by a third-order Volterra (exponential nonlinearity)

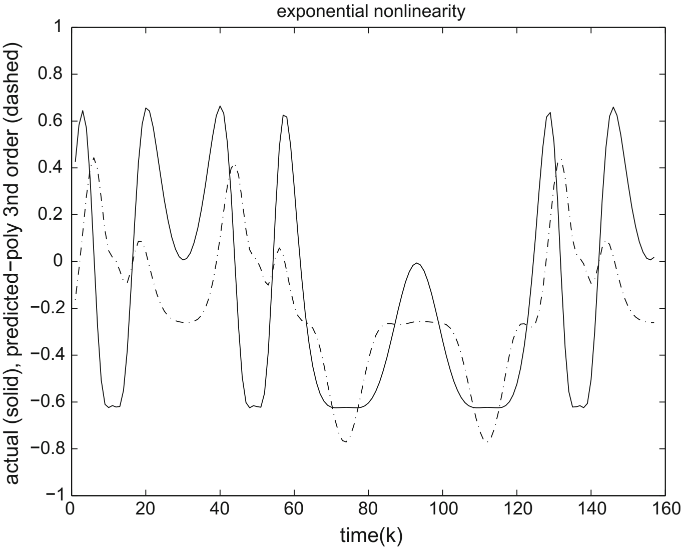


Fig. 13 Actual $y[k]$ and predicted $\hat{y}[k]$ by a third polynomial (exponential nonlinearity)

corresponding $y[k]$ and $\hat{y}[k]$ for the fixed basis function approach of third order. Again, the performance of a fixed basis function approach depends on if the chosen functions resemble the unknown structure or not. The result of the one shoot kernel is shown in Fig. 14 with gof $= 0.1679$, a poor performance. The reason is that for

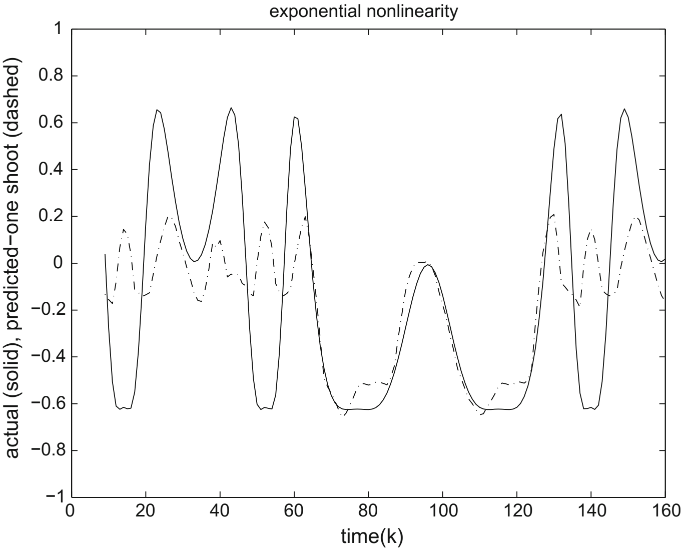


Fig. 14 Actual $y[k]$ and predicted $\hat{y}[k]$ by one shoot method (exponential nonlinearity)

Table 5 Goodness-of-fits for the exponential input nonlinearity

	proposed method	2nd order fixed basis	3rd order fixed basis	traditional one shoot
gof	0.6855	0.2299	0.1679	0.2722

Volterra (order)	2nd	3rd	4th	5th	6th
gof	-0.3437	-0.7652	-0.6194	-8.6657	-7.5490

a higher dimension $n = 8$, the bandwidth δ has to be large or there is no data in the neighborhood that consequently increases the bias. In the simulation, bandwidth was carefully adjusted to find the best gof which is reported here. It is clear, for Example 3 which is of short-term memory and low-order interaction, the proposed method outperforms any other method.

9 Discussion

In this section, we provide discussions and try to shed some lights on the proposed method.

- Orthogonalization and marginal influences: The essential step of the work is an orthogonalization procedure that allows us to write the output as a summation of marginal influences of the input variables. Then, these marginal influences are estimated by empirical averages weighted by a kernel function. This is related to the additive or generalized additive systems investigated in the statistics literature [12], especially discussed in a recent publication [26].

- FIR and iid assumptions: The orthogonalization is achieved in the work by assuming iid inputs and FIR structure of the unknown nonlinear system. The iid assumption removes statistical correlations between input variables and makes orthogonalization easier. The iid condition is however not critical as long as the correlations between $u[k-i]$'s and $u[k-j]$'s are available so they can be canceled out in the orthogonalization procedure. On the other hand, the FIR assumption on the nonlinear system is critical. Without this assumption, the output $y[k]$ is a function of the previous outputs $y[k-i]$'s as well as the input $u[k-j]$'s which are correlated. The exact correlation between $y[k-i]$ and $u[k-j]$ relies on the system to be identified. This makes cancelation of the correlations between the output variables and between the output and input variables very difficult. We are working along this direction and some preliminary results have been reported in [4].
- Kernel estimator and the choice of the bandwidth: The kernel estimator (6) is a smooth version of a conditional mean. The unknown function is estimated by the empirical mean of the measurements in the neighborhood of the point to be estimated. The size of the neighborhood, referred to as the bandwidth δ, controls the number of measurements to be used. The idea is to represent the unknown nonlinearities locally. All measurements outside the neighborhood $\varphi(k) > \delta$, are not used to construct the estimates. The choice of δ balances the trade-off between the bias and the variance. A large δ implies a large bandwidth interval and accordingly more data is used that results in a small variance. On the other hand, because more data points area used even with those not in a close vicinity, the approximation error gets large, which gives rise to a large bias term. A small δ produces just the opposite, a large variance and a small bias. Hence, increasing δ tends to reduce the variance but at the same time increases the bias. The best choice is to balance the bias and the variance. There is a huge literature on this topic and some guidelines are available in [12, 22, 26] for the choice of the bandwidth δ. For instance, the optimal bandwidth can be derived by minimizing the mean square error if the analytical expression exists. Alternatively, a data-driven bandwidth can be derived by using the leaving-one-out criterion. For details, see [12] and the references within.
- Recursive algorithms: The kernel estimator proposed in the work can be calculated recursively when the new data become available. First, let $\widehat{\phi}_0^{N+1}$ and $\widehat{\phi}_0^N$ be the estimates of ϕ_0 at $N+1$ and N, respectively, where the superscripts $N+1$ and N emphasize on the dependence of the data upto $N+1$ and N, respectively. It is easily verified that

$$\widehat{\phi}_0^{N+1} = \frac{N}{N+1}\widehat{\phi}_0^N + \frac{1}{N+1} \cdot y[N+1].$$

To calculate $\widehat{\phi}_j^{N+1}(x_j)$ from $\widehat{\phi}_j^N(x_j)$, $j = 1, 2, ..., n$, recursively, consider

1. Collect new data $y[N+1], u[N]$ and calculate $\varphi_j(x_j, N+1) = |u[N+1-j] - x_j|$.
2. If $\delta \le \varphi_j(x_j, N+1)$, then

$$w_j^{N+1}(x_j, k) = \begin{cases} w_j^N(x_j, k), & k = 1, 2, ..., N \\ 0, & k = N+1 \end{cases}$$

3. $\widehat{\phi}_j^{N+1}(x_j) = \widehat{\phi}_j^N(x_j)$. Reset $N+1 \Rightarrow N$ and go back to step 1.
4. If $\delta > \varphi_j(x_j, N+1)$, let

$$\lambda(N+1) = \frac{l_j\delta - \sum_{i=1}^{l_j} \varphi_j(x_j, m_j(i))}{l_j\delta - \sum_{i=1}^{l_j} \varphi_j(x_j, m_j(i)) + \delta - \varphi_j(x_j, N+1)}$$

and define

$$w_j^{N+1}(x_j, k) = \begin{cases} w_j^N(x_j, k) \cdot \lambda(N+1), \ k \in M_j = \{m_j(1), ..., m_j(l_j)\} \\ \frac{\delta - \varphi_j(x_j, N+1)}{(l_j+1)\delta - \sum_{i=1}^{l_j} \varphi_j(x_j, m_j(i)) - \varphi_j(x_j, N+1)}, \ k = N+1 \\ 0, \ k \notin \{N+1, m_j(1), ..., m_j(l_j)\} \end{cases}$$

Identify $N+1 = m_j(l_j+1)$.
5. $\widehat{\phi}_j^{N+1}(x_j) = \widehat{\phi}_j^N(x_j) \cdot \lambda(N+1) + w_j^{N+1}(x_j, N+1)y(N+1)$. Reset $l_j + 1 \Rightarrow l_j$, $N+1 \Rightarrow N$ and go back to step 1.

Other $\widehat{\phi}_j$, $j > n$, can be similarly calculated recursively.

- Higher factor interactive term systems and computational complexity: This work focuses on the system upto 2-factor interactive terms. All the results can be extended to higher factor interactive term systems. We summarize the procedures for a 3-factor term system.

  Step 1: Consider the system (1). Define $f_{j_1 j_2 j_3}$ which is the normalized $\bar{f}_{j_1 j_2 j_3}$ so that the average is zero with respect to any x_i and (x_i, x_j).
  Step 2: Redefine $\bar{f}_{j_1 j_2}$ by adding the original $\bar{f}_{j_1 j_2}$ to all the 2-factor terms with index $j_1 j_2$ resulting from the normalization of $\bar{f}_{j_1 j_2 j_3}$. Normalize $\bar{f}_{j_1 j_2}$ to have $f_{j_1 j_2}$.
  Step 3: Redefine $\bar{f}_j$ by adding the original $\bar{f}_j$ to all the 1-factor terms with the index j resulting from the previous steps. Normalize $\bar{f}_j$ to have f_j. Also, adjust the constant term c.

  Then, the orthogonal functions ϕ_j's and their estimates $\widehat{\phi}_j$'s can be similarly defined. The estimates enjoy the same convergence properties as in the 2-factor term case.
  In theory, the procedure can be extended to any factor term system. However, the number of terms increases exponentially and so is the computational complexity. Practically, the method proposed in the work is more efficient for a low-order factor term system, say 2-factor or 3-factor interactive term systems with a modest time lag n.

- Curse of dimensionality: A common feature of most nonlinear identification methods in the literature is to find directly the nonlinearity f representing the input–output relationship of the system. This amounts to solving a high-dimensional nonlinear identification problem directly and is usually difficult if n is not small. One of the main problems is the curse of dimensionality in nonparametric identification. To illustrate the situation, let $u[\cdot]$ be uniformly distributed in $I = [-0.5, 0.5]$. Suppose one wants to estimate $f(x_1, x_2, ..., x_n)$ at a point $(x_1, x_2, ..., x_n) \in I^n$. Since any nonparametric identification scheme is in some form of local smoother or weighted average based on the measurement data in the neighborhood of $(x_1, x_2, ..., x_n)$, there must be enough data in the neighborhood to average out the effects of noise and the uncertainty due to lack of structural information. For simplicity, suppose the neighborhood is a hyper-box with the side length 0.1. Then, the volume of I^n is $1^n = 1$ and the volume of the neighborhood is 0.1^n. This implies the probability that a measurement data $(u[k-1], u[k-2], ..., u[k-n])$ is in the neighborhood of $(x_1, x_2, ..., x_n)$ is $0.1^n/1 = 0.1^n$ that goes to zero exponentially as the order or dimension n gets larger. Let N be the number of total data measurements. For a large N, it is likely there are $N \cdot 0.1^n$ measurements in the neighborhood. Unless N is huge, there is not enough data in a neighborhood for identification purpose even for a modest n.
  Now, consider the proposed method for a low-order factor term system, say for an 2-factor term system. The aim of the method is not to estimate the high-dimensional f directly but to estimate the unknown interactive terms f_j and $f_{j_1 j_2}$ or the orthonormal functions ϕ_j's. Moreover, identification of each interactive term is decoupled with each other. This is very beneficial. For instance, let $n = 5$. Then, the problem becomes identification of five 1-dimensional 1-factor terms $f_j(u[k-j])$, $j = 1, 2..., 5$, and ten 2-dimensional 2-factor terms $f_{j_1 j_2}(u[k-j_1], u[k-j_2])$, $1 \le j_1 < j_2 \le 5$. Though the number of identifications is increased, the complexity of identification is reduced drastically. Because of decoupling, the probability of an $u[k-j]$ in the neighborhood of x_j for one-dimensional identification is $0.1/1 = 0.1$ and the probability of $(u[k-j_1], u[k-j_2])$ in the neighborhood of (x_{j_1}, x_{j_2}) is $0.1^2/1 = 0.1^2$. Suppose the total number of data points is $N = 10^4$. This implies that it is likely there are 10^3 or 10^2 measurements in the neighborhood for identification of 1-factor or 2-factor terms, respectively. Recall that if the five-dimensional $f(x_1, x_2, x_3, x_4, x_5)$ is identified directly, the probability that a data vector is in the neighborhood of $(x_1, x_2, x_3, x_4, x_5)$ is 0.1^5. With $N = 10^4$, the probability that there is one measurement in a neighborhood is 0.1. That makes that identification is nearly impossible in the presence of noise. Clearly, the performance of identification of the 1-factor or 2-factor term can be substantially improved for the same N, compared to the identification of a five-dimensional problem f. This effectively combats the curse of dimensionality. In a sense, the approach proposed here is to replace a difficult high-dimensional problem by a number of less-difficult and manageable low-dimensional problems.
- Combined residual analysis and statistical test: A version of the Box–Pierce test is developed in the context of nonlinear system identification. The reason behind this choice is that traditional Box–Pierce tests do not work well if there is a nonlinear

Table 6 Goodness-of-fit as a function of N and δ

	$N = 20000$	$N = 10000$	$N = 5000$
$\delta = 0.12$	0.9280	0.9186	0.9204
$\delta = 0.1$	0.9411	0.9376	0.9062
$\delta = 0.08$	0.9457	0.9174	0.8994

dependence and could give misleading conclusions [32]. The modified Box–Pierce test overcomes this problem. Moreover, any Box–Pierce test assumes that the null hypothesis is true and then tests based on a measured data set if the null hypothesis should be accepted with a given probability. It alone can never answer the question of the second type error as discussed in the work. The contribution of the work is to deal with this problem by combining the Box–Pierce test with residual analysis. This reasonably guarantees that the null hypothesis is true before the Box–Pierce test.

In the Box–Pierce test and the residual analysis, the choices of the level of significance and other parameters are always tricky and subjective. Whether the level of significance 0.01 or 0.03 is enough is tightly connected to the intended purpose of the model. If prediction is the intended purpose, the identified model should be validated on a fresh data to verify if the identified model fulfills the intended purpose. It may take several iterations to have some good design parameters for a particular application.

- Finite data performance: The proposed method is convergent. The convergence rate is $O(\frac{1}{\sqrt{\delta^2 N}})$ for a system upto 2-factor interactive terms and is $O(\frac{1}{\sqrt{\delta^l N}})$ for a system upto to l-factor interactive terms. Like most of nonlinear identification algorithms, the analysis of finite data performance of the proposed method is very hard to carried analytically. We provide numerical simulations to demonstrate the finite data performance in terms of robustness of the choices with respect to the data length N, the bandwidth δ, and the order determination. To see the effect of data length N on the order determination, the same example (23) was simulated under the same simulation conditions for $N = 20000$, 10000, and 5000 respectively. The results are in Table 1 and fairly consistent even N experiences a large variation from 5000 to 20000. To test the effects of the data length N and the bandwidth δ on the obtained model, we use the goodness-of-fit (24) as an indicator. Table 6 shows goodness-of-fit for various N and δ. Again, the identified model, in terms of prediction error, is robust with respect to variations of design parameters N and δ.

10 Concluding Remarks

In this work, a data-driven orthogonal basis function approach is proposed for nonlinear system identification. The main advantage is that it eliminates the guessing works when there is a little priori information on the structure of the unknown system. Further the data driven basis functions are orthogonal and thus enjoy many preferable properties. We are working on extending the results presented in the work to IIR nonlinear systems.

In addition, methods are proposed for order determination and regressor selection. These topics are generally very hard for nonlinear system identification. The methods proposed have potential to be applicable to many nonlinear system identification schemes and we felt they deserve more studies.

Finally, two structure identification methods under deterministic inputs are proposed to estimate the structure of the system before a full scale system identification is performed. They can efficiently simplify the procedure of system identification.

Appendix

Proof of Theorem 1: The first part is directly from the definition of ϕ_i's. Also from the definition, it is easily verified that $\mathbf{E}\phi_j[k] = 0$ for $j = 1, ..., n$. $\mathbf{E}\phi_j[k] = 0$, $j = n+1, ..., M$ follows from $\mathbf{E} f_{j_1 j_2}(u[k-j_1], u[k-j_2]) = 0$. We now show $\mathbf{E}\phi_{j_1}[k]\phi_{j_2}[k] = 0$. For $0 \le j_1 < j_2 \le n$, $\mathbf{E}\phi_{j_1}[k]\phi_{j_2}[k] = \mathbf{E}\phi_{j_1}[k]\mathbf{E}\phi_{j_2}[k] = 0$ because of independence of $u[k-j_1]$ and $u[k-j_2]$. The proofs for other j_1 and j_2 follow from the same arguments as

$$\begin{aligned}\mathbf{E}\phi_1[k]\phi_{n+1}[k] &= \mathbf{E}\phi_1(u[k-1])\phi_{n+1}(u[k-1], u[k-2]) \\ &= \mathbf{E}\{\phi_1(u[k-1])\mathbf{E}\{\phi_{n+1}(u[k-1], u[k-2]) \mid u[k-1]\}\} = 0.\end{aligned}$$

To show the third part, observe

$$y[k] = c + \sum_{j=1}^{n} f_j(u[k-j]) + \sum_{1 \le j_1 < j_2 \le n} f_{j_1 j_2}(u[k-j_1], u[k-j_2]) + v[k],$$

$$\begin{aligned}\mathbf{E}y[k] &= c = \phi_0 \\ \mathbf{E}\{y[k]|u[k-j] = x_j\} &= c + f_j(x_j) = \phi_0 + \phi_j(x_j),\ \ j = 1, ..., n \\ \mathbf{E}\{y[k] \mid u[k-j_1] &= x_{j_1}, u[k-j_2] = x_{j_2}\} \\ &= c + f_{j_1}(x_{j_1}) + f_{j_2}(x_{j_2}) + f_{j_1 j_2}(x_{j_1}, x_{j_2}) \\ &= \phi_0 + \phi_{j_1}(x_{j_1}) + \phi_{j_2}(x_{j_2}) + f_{j_1 j_2}(x_{j_1}, x_{j_2}),\ \ 1 \le j_1 < j_2 \le n\end{aligned}$$

Then, the conclusion follows from the definition of ϕ_j's.

Proof of Theorem 2: The first part is from Theorem 1 and the law of large numbers,

$$\widehat{\phi_0} = \frac{1}{N}\sum y[k] \rightarrow \mathbf{E}y[k] = \phi_0.$$

For the second part, from the assumptions $\delta \to 0, \delta N \to \infty$ as $N \to \infty$, the number of samples $u[k-j]$'s in the interval,

$$\varphi_j(x_j, k) = |u[k-j] - x_j| \leq \delta$$

converges to $2\psi(x_j)\delta N \to \infty$, where the probability density function of the input at x_j, $\psi(x_j)$, is assumed to be positive, or the number of elements $l_j \to 2\psi(x_j)\delta N \to \infty$. Now,

$$|\widehat{\phi}_j(x_j) - \phi_j(x_j)| = |\sum_{k=1}^{N} w_j(x_j, k)y[k] - \phi_j(x_j) - \widehat{\phi}_0|$$

$$= |\sum_{k=1}^{N} w_j(x_j, k)(\phi_0 - \widehat{\phi}_0) + \sum_{k=1}^{N} w_j(x_j, k)(\phi_j(u[k-j]) - \phi_j(x_j))$$

$$+ \sum_{i=1, i\neq j}^{n} \sum_{k=1}^{N} w_j(x_j, k)\phi_i(u[k-i]) + \sum_{j=n+1}^{M} \sum_{k=1}^{N} w_j(x_j, k)\phi_j[k] + \sum_{k=1}^{N} w_j(x_j, k)v[k]|$$

$$= |\sum_{l=1}^{l_j} w_j(x_j, m_j(l))(\phi_0 - \widehat{\phi}_0) + \sum_{l=1}^{l_j} w_j(x_j, m_j(l))(\phi_j(u[m_j(l) - j]) - \phi_j(x_j))$$

$$+ \sum_{i=1, i\neq j}^{n} \sum_{l=1}^{l_j} w_j(x_j, m_j(l))\phi_i(u[m_j(l) - i]) + \sum_{j=n+1}^{M} \sum_{l=1}^{l_j} w_j(x_j, m_j(l))\phi_j[m_j(l)] +$$

$$\sum_{l=1}^{l_j} w_j(x_j, m_j(l))v[m_j(l)]| \leq |\sum_{l=1}^{l_j} w_j(x_j, m_j(l))(\phi_0 - \widehat{\phi}_0)|$$

$$+ |\sum_{l=1}^{l_j} |w_j(x_j, m_j(l))(\phi_j(u[m_j(l) - j]) - \phi_j(x_j))|$$

$$+ |\sum_{i=1, i\neq j}^{n} \sum_{l=1}^{l_j} w_j(x_j, m_j(l))\phi_i(u[m_j(l) - i])|$$

$$+ |\sum_{j=n+1}^{M} \sum_{l=1}^{l_j} w_j(x_j, m_j(l))\phi_j[m_j(l)]| + |\sum_{l=1}^{l_j} w_j(x_j, m_j(l))v[m_j(l)]|$$

With L being the Lipschitz constant and from the orthogonal properties of ϕ_j, $l_j \to \infty$, $w_j(x_j, m_j(l)) \geq 0$ and $\sum_{l=1}^{l_j} w_j(x_j, m_j(l)) = 1$,

$$\sum_{l=1}^{l_j} w_j(x_j, m_j(l))(\phi_0 - \widehat{\phi}_0) = \phi_0 - \widehat{\phi}_0,$$

$$|\sum_{l=1}^{l_j} |w_j(x_j, m_j(l))(\phi_j(u[m_j(l) - j]) - \phi_j(x_j))| \le \delta L,$$

$$|\sum_{i=1, i\neq j}^{n} \sum_{l=1}^{l_j} w_j(x_j, m_j(l))\phi_i(u[m_j(l) - i])|^2$$

$$\rightarrow |\sum_{i=1, i\neq j}^{n} \mathbf{E}\{\phi_i(u[k-i]) \mid u[k-j] = x_j\}|^2 + O(\frac{1}{\delta N}),$$

$$|\sum_{j=n+1}^{M} \sum_{l=1}^{l_j} w_j(x_j, m_j(l))\phi_j[m_j(l)]|^2$$

$$\rightarrow |\sum_{j=n+1}^{M} \mathbf{E}\{\phi_j[k] \mid u[k-j] = x_j\}|^2 + O(\frac{1}{\delta N}),$$

$$|\sum_{l=1}^{l_j} w_j(x_j, m_j(l))v[m_j(l)]|^2 \rightarrow |\mathbf{E}v[k]|^2 + O(\frac{1}{N}).$$

Therefore,

$$|\widehat{\phi}_j(x_j) - \phi_j(x_j)| \rightarrow |\phi_0 - \widehat{\phi}_0| + \delta L + O(\frac{1}{\sqrt{\delta N}}) \rightarrow 0, \ j = 1, ..., n$$

This completes the proof of the second part. The proofs of the third part are similar. The only difference is that the convergence rate is $O(\frac{1}{\sqrt{\delta^2 N}})$ as $N \rightarrow \infty$.
Proof of Theorem 3: It is easily verified that

$$\int_{\infty}^{\infty} |K(x)|dx < \infty, \ \int_{-\infty}^{\infty} |\int_{-\infty}^{\infty} K(x)e^{-j\omega x}dx|d\omega < \infty.$$

The rest part of the proof follows directly from Lemma 2 of [19].

References

1. Akcay, H. and P. Heuberger, (2001), “A frequency domain iterative identification algorithm using general orthonormal basis functions”, *Automatica*, **37**, pp. 663–674.
2. Bai, E.W. (2005), “Identification of additive nonlinear systems”, *Automatica*, **41**, pp. 1247–1253.
3. Bai, E.W. (2008), “Non-Parametric Nonlinear System Identification: a Data-Driven Orthogonal Basis Function Approach”, *IEEE Trans on Automatic Control*, **53**, pp. 2615–2626.

4. Bai, E.W. and K.S. Chan, (2008), “Identification of additive nonlinear systems and its application in generalized Hammerstein models”, *Automatica*, **44**, pp. 430–436.
5. Bai, E.W. and Y. Liu, (2007), “Recursive direct weight optimization in nonlinear system identification: A minimal probability approach”, *IEEE Trans on Automatic Control*, **52**, pp. 1218–1231.
6. Bai, E.W. R. Tempo and Y. Liu, (2007), “Identification of nonlinear systems without prior structural information”, *IEEE Trans on Automatic Control*, **52**, pp. 442–453.
7. Bai, E.W. (2008), “Non-Parametric Nonlinear System Identificationation: a Data-Driven Orthogonal Basis Function Approach”, *IEEE Trans on Automatic Control*, **53**, pp. 2615–2626.
8. Bai, E.W. and R. Tempo, (2010), “Representation and Identification of Nonparametric Nonparametriconlinear Systems of Short Term Memory and Low Degree of Interactions”, *Automatica*, **46**, pp. 1595–1603.
9. Box, G.E.P. and D. Pierce, (1970), “Distribution of residual autocorrelations in autoregressive-integrated moving average time series models”, *J of the American Statistical Association*, **65**, pp. 1509–1526.
10. Chen, S, SA Billings and W Luo (1989), “Orthogonal least squares methods and their application to non-linear system identification”, *Int. J. Control*, **50**, pp. 1873–1896.
11. Cerone, V. and D. Regruto, (2007), “Bounding the parameters of linear systems with input backlash”, *IEEE Trans on Automatic Control*, **52**, pp. 531–536.
12. Fan, J and I. Gijbels (1996) Local polynomial modelling and its applications, Chapman &Hall/CRC.
13. Godfrey, G. (1993), PERTURBATION SIGNAL FOR SYSTEM IDENTIFICATION, Prentice-Hall, New York.
14. Harris, CJ, X Hong and Q Gan (2002), Adaptive modeling, estimation and fusion from data: a neurofuzzy approach, Springer Verlag.
15. Juditsky, A. H. Hjalmarsson, A. Benveniste, B. Delyon, L. Ljung, J. Sjoberg and Q. Zhang (1995), “Nonlinear block-box models in system identification: Mathematical foundations,” *Automatica*, **31**, pp. 1725–1750.
16. Li, K. J. Peng and G. Irwin, (2005), “A fast nonlinear model identification method”, *IEEE Trans on Automatic Control*, **50**, pp. 1211–1216.
17. Lind, I. and L. Ljung (2005), “Regression selection with the analysis of variance method”, *Automatica*, **41**, pp. 693–700.
18. Ljung, L. (1999), SYSTEM IDENTIFICATION: THEORY FOR THE USER, 2nd Ed. Prentice-Hall, Upper Saddle River.
19. Lobato, I.N. J. Nankervis and N. Savin, (2002), “Testing for zero autocorrelation in the presence of statistical dependence”, *Econometric Theory*, **18**, pp. 730–743.
20. Makila, P.M. (1991), “Robust identification and Galois sequence”, *Int. J. of Control*, **54**, pp. 1189–1200.
21. Milanese, M. and C. Novara, (2004), “Set membership identification of nonlinear systems”, *Automatica*, **40**, pp. 957–975.
22. Nadaraya, E. (1989), NONPARAMETRIC ESTIMATION OF PROBABILITY DENSITIES AND REGRESSION CURVES, Kluwer Academic Pub. Dordrecht, The Netherlands.
23. Ninness, B. H. Hjalmarsson and F. Gustafsson, (1999), “On the Fundamental Role of Orthonormal Bases in System Identification”, *IEEE Transactions on Automatic Control*, **44**, No. 7, pp. 1384–1407.
24. Papoulis, P. and A. Pillai (2002), PROBABILITY, RANDOM VARIABLES AND STOCHASTIC PROCESSES (4th Ed), McGraw Hill, Boston.
25. Rugh, W. (1981), NONLINEAR SYSTEM THEORY, John Hopkins University Press, London.
26. Sperlich, S. D. Tjostheim and L. Yang (2002), “Non-parametric estimation and testing of interaction in additive models”, *Econometric Theory*, **18**, pp. 197–251.
27. Sjoberg, J. Q. Zhang, L. Ljung, A. Benveniste, B. Delyon, P-Y. Glorennec, H. Hjalmarsson and A. Juditsky (1995), “Nonlinear black-box modeling in system identification: a unified overview”, *Automatica*, **31**, pp. 1691–1724.

28. Soderstrom, T. and P. Stoica (1989), SYSTEM IDENTIFICATION Prentice-Hall, New York, NY.
29. Soderstrom, T. P. Van den Hof, B. Wahlberg and S. Weiland (Eds) (2005), "Special issue on data-based modeling and system identification", *Automatica*, **41**, No. 3, pp. 357–562.
30. Westwick, D.T. and K.R. Lutchen, (2000), "Fast, robust identification of nonlinear physiological systems using an implicit basis function", *Annals of Biomedical Engineering*, **28**, pp. 1116–1125.
31. Van den Hof, P.M.J. P.S.C. Heuberger and J. Bokor, (1995) "System identification with generalized orthonormal basis functions", *Automatica*, **31**, pp. 1821–1834.
32. Velasco, C. and I Lobato, (2004), "A simple and general test for white noise", *Econometric Society 2004 Latin American Meetings*, number 112, Santiago, Chile.
33. Zhang, Q. (1997), "Using wavelet network in nonparametric estimation", *IEEE Trans. on Neural Networks*, **8**, pp. 227–236.
34. Zhu, Q.M. and S.A. Billings, (1996) "Fast orthogonal identification of nonlinear stochastic models and radial basis function neural networks", *Int J. of Control*, **64**, pp. 871–886.

Part III
Distributed Systems and Algorithms

Perspectives on Network Systems and Mathematical Sociology

Francesco Bullo and Noah E. Friedkin

Abstract This chapter reviews selected topics in network systems and mathematical sociology. We first review classic results on Perron–Frobenius and algebraic graph theory. We then focus on mathematical sociology and describe models of opinion dynamics in social influence systems, including the classic French–Harary–DeGroot and the Friedkin–Johnsen models. Based on recent controlled experiments, we present recent empirical results on opinion dynamics along single issues and sequences of issues. Finally, motivated by these empirical results, we describe some mathematical models for the evolution of social power and influence systems via the reflected appraisal mechanism.

1 Introduction

Recent years have witnessed the emergence of a discipline of study focused on modeling, analyzing, and designing dynamic phenomena over networks. We refer to such systems as network systems; they are also equivalently referred to as multi-agent or distributed systems. This emerging discipline, rooted in graph theory, control theory, and matrix analysis, is increasingly relevant because of its broad set of application domains. Network systems appear naturally in (i) social networks and mathematical sociology, (ii) electric, mechanical, and physical networks, and (iii) animal behavior, population dynamics, and ecosystems. Network systems are designed in the context of networked control systems, robotic networks, power grids, parallel and scientific computation, and transmission and traffic networks, to name a few.

F. Bullo (✉)
Department of Mechanical Engineering and Center for Control, Dynamical-Systems
and Computation, 2325 Engineering Bldg II, University of California at Santa Barbara,
Santa Barbara, CA 93106-5070, USA
e-mail: bullo@engineering.ucsb.edu

N. E. Friedkin
Department of Sociology and Center for Control, Dynamical-Systems and Computation,
University of California at Santa Barbara, Santa Barbara, CA 93106-5070, USA
e-mail: friedkin@soc.ucsb.edu

© Springer Nature Switzerland AG 2018

T. Başar (ed.), *Uncertainty in Complex Networked Systems*,
Systems & Control: Foundations & Applications,
https://doi.org/10.1007/978-3-030-04630-9_11

Within this broad context, the disciplines of social networks and mathematical sociology have themselves received growing attention. Building on a classic history of work starting in the 50s, the study of influence systems and opinion dynamics has become a modern topic of interest to social scientists, engineers, computer scientists, and physicists. The scientific trend toward quantitate analysis in the social sciences is motivated by the availability of insightful datasets and sharper statistical and mathematical analysis tools.

A recent outstanding survey on social networks is given in [43]. Recent excellent treatments of network systems and their applications are given in the recent books [2, 3, 7, 8, 14, 17, 36, 45] and recent related surveys include [9, 27, 35, 40, 46]. The books and articles [6, 10, 16, 31, 38, 39] are instead excellent references on network science.

Against this background, this chapter is a review document intended for scientists interested in network systems and cooperative control as well as social networks and mathematical sociology. This chapter has a dual focus. First, we review classic results in the theory of linear network systems and place them in an algebraic framework based on Perron–Frobenius and algebraic graph theory. For example, we characterize the set of non-negative matrices in terms of irreducibility and primitivity. Second, we focus on mathematical sociology and describe models of opinion dynamics in social influence systems, including the classic French–Harary–DeGroot and the Friedkin–Johnsen models. Motivated by recent empirical evidence on opinion dynamics along single issues and sequences of issues, we then describe some mathematical models for the evolution of social power and influence systems via the reflected appraisal mechanism.

Paper Organization and Related Literature

Sections 2 and 3 review Perron Frobenius and algebraic graph theory. Classic references on this material include [26, 30, 47]. This content may be regarded as a highly-abbreviated version of the first part of the recent textbook [7].

Section 4 describes models of opinion dynamics. This classic field initiated with the seminal papers by [1, 15, 19, 29]. The classic discrete-time linear averaging model is well known as the DeGroot model, but a more accurate historic name would be the French–Harary–DeGroot model since modeling concepts were contained in [19] and analysis results in [29]. It is worth remarking how the 15 years before DeGroot the mathematical analysis in [29] was rather sophisticated already and included the concept of average consensus. The second model we review is the Friedkin–Johnsen model, which is an elaboration of the French–Harary–DeGroot. Documented in [22, 23], this model is still based upon linear averaging but it includes also an attachment to initial opinions. Recent results on variations of this model are given in [18, 25, 41, 42, 44].

Section 5 reviews the empirical findings on influence system evolution in small deliberative groups that are documented and analyzed in [21, 24]. The human subject experiments focused on both the opinion formation process on a single issue as well as on the influence network evolution that takes place along a sequence of opinion dynamic issues. Via multilevel linear regression analysis, we provide statistical evidence that the observed human subjects behavior is consistent with (1) the Friedkin–Johnsen model for single-issue opinion formation and (2) a reflected appraisal mechanism for the network evolution along issues. We remark that the papers [21, 24] report a rich collection of opinion dynamics phenomena and issue sequence effects on influence network structure, only some of which are reviewed here.

Section 6 reviews the mathematical model of social power and influence network evolution proposed by [34]. The key idea is to combine the French–Harary–DeGroot model of opinion dynamics with the Friedkin formalization of the reflected appraisal mechanism. Recent results on this model and its variations include the following. [33] completes the analysis in [34] by treating the case of reducible interaction matrices. For single-time scale models, [12] proposes a continuous-time distributed model and [32] proposes a dynamical flow model of interpersonal appraisals. Only preliminary results in [37] are known at this time for the case of stubborn individuals. Reference [51] obtains results on exponential convergence and the setting of time-varying interaction networks. [11] treats the case of switching and stochastic interaction matrices.

2 Perron–Frobenius Theory

Here we review the widely established Perron–Frobenius theory for non-negative matrices. We start by classifying non-negative matrices in terms of their zero/nonzero pattern and of the asymptotic behavior of their powers.

Definition 1 (*Irreducible and primitive matrices*) A square $n \times n$ non-negative matrix A, for $n \geq 2$, is

(i) *irreducible* if $\sum_{k=0}^{n-1} A^k$ is positive,
(ii) *primitive* if there exists $k \in \mathbb{N}$ such that A^k is positive.

A matrix that is not irreducible is said to be *reducible*.

In equivalent words, the matrix A is irreducible if, for any pair of indices (i, j) there exists an exponent $k = k(i, j) \leq (n-1)$ such that $(A^k)_{ij} > 0$. It is not hard to show that, if a non-negative matrix is primitive, then it is also irreducible (Fig. 1).

We now state the main results in Perron–Frobenius theory and characterize the properties of the spectral radius of a non-negative matrix.

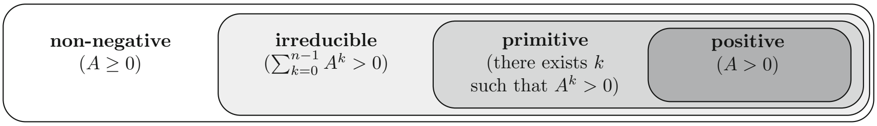


Fig. 1 The set of non-negative square matrices and its increasingly smaller subsets of irreducible, primitive and positive matrices

Theorem 1 (Perron–Frobenius Theorem) *Consider a square $n \times n$ non-negative matrix A, for $n \geq 2$. If A is irreducible, then*

(i) there exists a simple positive eigenvalue λ satisfying $\lambda \geq |\mu| \geq 0$ for all other eigenvalues μ,
(ii) the right and left eigenvectors v_{right} and v_{left} of λ are unique and positive, up to rescaling.

If additionally A is primitive, then

(iii) the eigenvalue λ satisfies $\lambda > |\mu|$ for all other eigenvalues μ.

The real non-negative eigenvalue λ is the spectral radius $\rho(A)$ of A and it is usually referred to as the *dominant or Perron eigenvalue* of A. The right and left eigenvectors v_{right} and v_{left} (unique up to rescaling and selected non-negative) of the dominant eigenvalue λ are called the *right and left dominant eigenvectors*, respectively.

Finally, the Perron–Frobenius Theorem for primitive matrices has immediate consequences for the asymptotic behavior of the discrete-time dynamical system $x(k+1) = Ax(k)$, that is, for the powers A^k as $k \to \infty$.

Proposition 1 (Powers of primitive matrices) *Consider a square $n \times n$ non-negative matrix A, for $n \geq 2$. Let λ be the dominant eigenvalue and let v_{right} and v_{left} be the right and left dominant eigenvectors of A normalized so that they are both positive and satisfy $v_{right}^{\top} v_{left} = 1$. Then*

$$\lim_{k \to \infty} \frac{A^k}{\lambda^k} = v_{right} v_{left}^{\top}.$$

3 Algebraic Graph Theory

In this section, we review some basic and prototypical results that involve correspondences between graphs and adjacency matrices. We let G denote a weighted digraph and A its weighted adjacency matrix or, equivalently, we let A be a non-negative matrix and we let G be its *associated weighted digraph* (i.e., the digraph with nodes $\{1, \ldots, n\}$ and with weighted adjacency matrix A).

We start with some basic definitions about a directed graph G. A node i is *globally reachable* if, for every other node j, there exists a directed walk-in G from node j to node i. A directed graph is *strongly connected* if each node is globally reachable. A *subgraph* of G is a subset of nodes and edges of G. A subgraph H is a *strongly connected component* of G if H is strongly connected and any other subgraph of G containing H is not strongly connected. A directed graph G is *aperiodic* if there exists no integer that divides the length of each cycle of G.

We will also need the notion of condensation of a digraph. Given a directed graph G, the *condensation digraph* of G is formed by contracting each strongly connected component into a single node and letting an arc exist from one component to another if and only if at least one arc exists from a member of one component to a member of the other in G. The condensation digraph is acyclic and, therefore, contains at least one sink.

The first result we present relate the powers of the adjacency matrix with directed walks on the graph.

Lemma 1 *Let G be an unweighted digraph with unweighted adjacency matrix $A_{0,1} \in \{0,1\}^{n\times n}$. For all $i, j \in \{1,\dots,n\}$ and $k \in \mathbb{N}$, the (i, j) entry of $A_{0,1}^k$ equals the number of directed walks of length k (including walks with self-loops) from node i to node j.*

Moreover, if G is a weighted digraph with weighted adjacency matrix A, then the (i, j) entry of A^k is positive if and only if there exists a directed walk of length k (including walks with self-loops) from node i to node j.

Theorem 2 (Connectivity properties of the digraph and positive powers of the adjacency matrix) *Let G be a weighted digraph with $n \geq 2$ nodes and weighted adjacency matrix A. The following statements are equivalent:*

(i) A is irreducible, that is, $\sum_{k=0}^{n-1} A^k > 0$;
(ii) there exists no permutation matrix P such that $P^\top AP$ is block triangular;
(iii) G is strongly connected;
(iv) for all partitions $\{\mathscr{I}, \mathscr{J}\}$ of the index set $\{1,\dots,n\}$, there exists $i \in \mathscr{I}$ and $j \in \mathscr{J}$ such that $\{i, j\}$ is an edge in G.

Let us remark that, instead of the order in which we presented matters here, most references define an irreducible matrix through property (ii) or, possibly, through property (iv).

Theorem 3 (Strongly connected and aperiodic digraph and primitive adjacency matrix) *Let G be a weighted digraph with weighted adjacency matrix A. Then G is strongly connected and aperiodic if and only if A is primitive.*

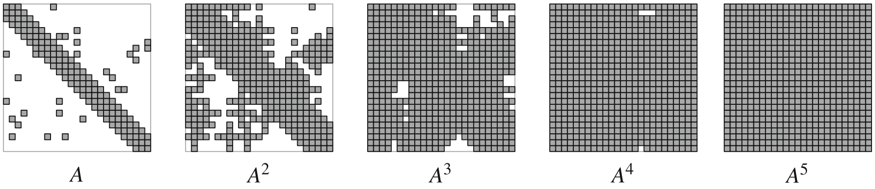


Fig. 2 These five images depict increasing powers of a non-negative matrix $A \in \mathbb{R}^{25\times 25}$. The digraph associated to A is strongly connected and has self-loops at each node so that, by Theorem 3, there exists k (in this case $k = 5$) such that $A^k > 0$

4 Mathematical Models for the Evolution of Opinions

This section reviews some classic models for opinion dynamics. We focus on basic linear and affine models, whose relevance is established empirically (Fig. 2).

We start by presenting some convergence results for systems of the form

$$x(k+1) = Ax(k), \qquad \text{where } A \text{ is row-stochastic.} \tag{1}$$

Recall that the non-negative square matrix A is said to be row-stochastic if all its row-sums are equal to one, that is, if $A\mathbb{1}_n = \mathbb{1}_n$. Therefore, the right eigenvector of the eigenvalue 1 can be selected as $\mathbb{1}_n$.

The discrete-time averaging model (1) is well known as the DeGroot model, but a more accurate historic name would be the French–Harary–DeGroot model, as discussed in the introduction. The matrix A describes an *interpersonal influence network.*

Theorem 4 (Consensus for row-stochastic matrices with a globally-reachable aperiodic strongly connected component) *Let A be a row-stochastic matrix and let G be its associated digraph. The following statements are equivalent:*

(A1) the eigenvalue 1 is simple, $\rho(A) = 1$, and all other eigenvalues have magnitude strictly smaller than 1,
(A2) A is semi-convergent (i.e., $\lim_{k\to\infty} A^k$ exists and is finite) and $\lim_{k\to\infty} A^k = \mathbb{1}_n v_{left}^\top$, for some $v_{left} \in \mathbb{R}^n$, $v_{left} \geq 0$, and $\mathbb{1}_n^\top v_{left} = 1$,
(A3) the digraph associated to A contains a globally reachable node and the subgraph of globally reachable nodes is aperiodic.

If any, and therefore all, of the previous conditions are satisfied, then the matrix A is said to be indecomposable *and the following properties hold:*

(i) $v_{left} \geq 0$ is the left dominant eigenvector of A and $(v_{left})_i > 0$ if and only if node i is globally reachable;
(ii) the solution to the averaging model $x(k+1) = Ax(k)$ in Eq. (1) satisfies

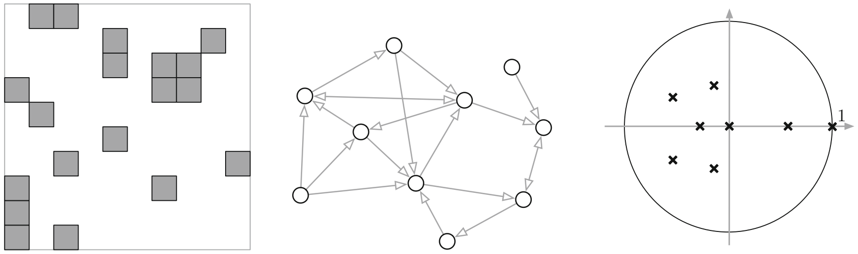
$$\lim_{k\to\infty} x(k) = \big(v_{left}^\top x(0)\big)\mathbb{1}_n;$$

In this case we say that the dynamical system achieves consensus*;*
(iii) if additionally A is doubly-stochastic, then $v_{left} = \frac{1}{n}\mathbb{1}_n$ (because $A^\top \mathbb{1}_n = \mathbb{1}_n$ and $\frac{1}{n}\mathbb{1}_n^\top \mathbb{1}_n = 1$) so that

$$\lim_{k\to\infty} x(k) = \frac{\mathbb{1}_n^\top x(0)}{n}\mathbb{1}_n = \text{average}\big(x(0)\big)\mathbb{1}_n.$$

In this case we say that the dynamical system achieves average consensus.

The limiting vector is, therefore, a weighted average of the initial conditions. The relative weights of the initial conditions are the convex combination coefficients



(a) A rows-stochastic matrix; in each row, nonzero entries are equal and sum to 1.

(b) The corresponding digraph has an aperiodic subgraph of globally reachable nodes.

(c) The spectrum of the adjacency matrix includes a dominant eigenvalue.

Fig. 3 An example indecomposible row-stochastic matrix, its associated digraph consistent with Theorem 4(A2), and its spectrum consistent with Theorem 4(A1)

$(v_{\text{left}})_1, \dots, (v_{\text{left}})_n$. In a social influence network, the coefficient $(v_{\text{left}})_i$ is regarded as the "social influence" of agent i.

In Fig. 3 we show a non-negative matrix that is indecomposable, together with its directed graph and its spectrum.

The implication (A3) $\Longrightarrow$ (ii) amounts to a result in which the structure of the network determines its function, i.e., the asymptotic behavior of the averaging system.

Next, we consider the general case of digraphs that do not contain globally reachable nodes, that is, digraphs whose condensation digraph has multiple sinks. In what follows, we say that a node is *connected* with a sink of a digraph, if there exists a directed walk from the node to any node in the sink.

Theorem 5 (Convergence for row-stochastic matrices with multiple aperiodic sinks) *Let A be a row-stochastic matrix, let G be its associated digraph, and let $M \geq 2$ be the number of sinks in the condensation digraph $C(G)$. If each of the M sinks is aperiodic, then*

(i) the semi-simple eigenvalue $\rho(A) = 1$ has multiplicity equal M and is strictly larger than the magnitude of all other eigenvalues, hence A is semi-convergent,

(ii) there exist M left eigenvectors of A, denoted by $v^m_{left} \in \mathbb{R}^n$, for $m \in \{1, \dots, M\}$, with the properties that: $v^m_{left} \geq 0$, $1_n^\top v^m_{left} = 1$ and $(v^m_{left})_i$ is positive if and only if node i belongs to the m-th sink,

(iii) the solution to the averaging model $x(k+1) = Ax(k)$ with initial condition $x(0)$ satisfies

$$\lim_{k\to\infty} x_i(k) = \begin{cases} (v^m_{left})^\top x(0), & \textit{if node } i \textit{ belongs to the } m\textit{-th sink,} \\ (v^m_{left})^\top x(0), & \textit{if node } i \textit{ is connected only with the } m\textit{-th sink,} \\ \displaystyle\sum_{m=1}^{M} z_{i,m}\big((v^m_{left})^\top x(0)\big), & \textit{if node } i \textit{ is connected to more than one sink,} \end{cases}$$

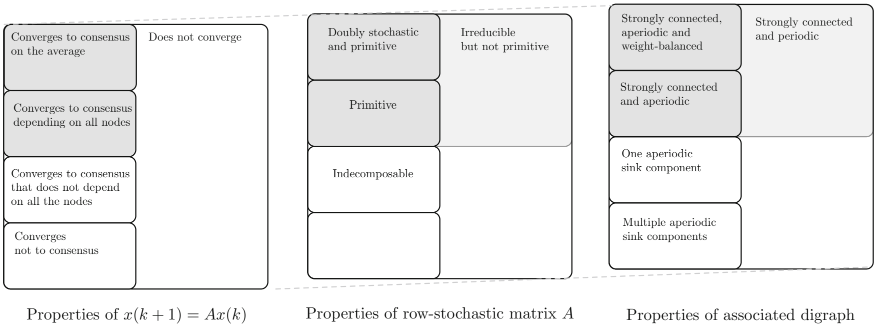

Converges to consensus on the average
Converges to consensus depending on all nodes
Converges to consensus that does not depend on all the nodes
Converges not to consensus
Does not converge
Doubly stochastic and primitive
Primitive
Indecomposable
Irreducible but not primitive
Strongly connected, aperiodic and weight-balanced
Strongly connected and aperiodic
One aperiodic sink component
Multiple aperiodic sink components
Strongly connected and periodic
Properties of $x(k+1) = Ax(k)$
Properties of row-stochastic matrix A
Properties of associated digraph

Fig. 4 Corresponding properties for the discrete-time averaging dynamical system $x(k+1) = Ax(k)$, the row-stochastic matrix A and the associated weighted digraph

where, for each node i connected to more than one sink, the coefficients $z_{i,m}$, $m \in \{1, \dots, S\}$, are convex combination coefficients and are strictly positive if and only if there exists a directed walk from node i to the sink m.

Note that convergence does not occur to consensus (not all components of the state are equal) and the final value of all nodes is independent of the initial values at nodes which are not in the sinks of the condensation digraph. We summarize the discussion in this section with a figure summarizing the asymptotic behavior of the French–Harary–DeGroot discrete-time averaging systems; see Fig. 4.

We next consider the opinion dynamics model by [22] which, for generic parameter values, features persistent disagreement and lack of consensus. As before we let A be a row-stochastic matrix whose associated digraph describes an interpersonal influence network. We assume that every individual is naturally given an *openness level* $\lambda_i \in [0, 1]$, $i \in \{1, \dots, n\}$, describing how open the individual is to interpersonal influence and, therefore, to changing her initial opinion about a subject. We then define $\Lambda = \text{diag}(\lambda_1, \dots, \lambda_n)$, where diag is the standard operator that maps an array to a diagonal matrix.

The *Friedkin–Johnsen model* of opinion dynamics is defined by

$$x(k+1) = \Lambda A x(k) + (I_n - \Lambda) x(0), \tag{2}$$

where, for individual i, $x_i(k)$ represents the current opinion and $x_i(0)$ represents the initial opinion or prejudice. The *Friedkin–Johnsen model* is again an averaging model with stubborn individuals in the sense that here every individual i exhibits an attachment $(1 - \lambda_i)$ to its initial opinion $x_i(0)$.

Theorem 6 (Persistent disagreement in the Friedkin–Johnsen model) *Consider a square $n \times n$ non-negative matrix A, for $n \geq 2$, and a diagonal matrix Λ with entries in* $[0, 1]$. *Assume that*

(A1) at least one individual has a strictly positive attachment to its initial opinion, that is, $\lambda_i < 1$ for at least one individual i; and

(A2) the interpersonal influence network contains directed walks from each individual with openness level equal to 1 *to an individual* j *with openness level* $\lambda_j < 1$.

Then the following statements hold:

(i) the matrix ΛA *is convergent, that is,* $\rho(\Lambda A) < 1$,
(ii) the total influence matrix $V = (I_n - \Lambda A)^{-1}(I_n - \Lambda)$ *is well defined and row-stochastic, and*
(iii) the limiting opinions satisfy $\lim_{k\to+\infty} x(k) = Vx(0)$.

We conclude with some remarks. As predicted in the model formulation, consensus is not achieved asymptotically because of the attachment to initial opinions. If Assumption (A1) is not satisfied and, therefore, $\Lambda = I_n$, then we recover the French–Harary–DeGroot opinion dynamics model.

Finally, it is worth noting that the original work [22], see also [23], make the additional assumption that $\lambda_i = 1 - a_{ii}$, for $i \in \{1, \dots, n\}$. This additional assumption is justified by sociological reasons and introduces coupling between the openness level and the interpersonal influence values. Other properties of this model are studied in [4, 24, 44].

5 Empirical Findings on the Evolution of Opinions and Influence Networks

We here review the empirical findings on influence system evolution in small deliberative groups that are documented and analyzed in [21, 24]. The human subject experiments focus on both the opinion formation process on a single issue as well as on the influence network evolution that takes place along a sequence of issues.

5.1 *The Friedkin–Johnsen Model on Judgmental Issues*

We collected data in experiments on 30 groups of 4 individuals assembled to discuss a sequence of 15 risk/reward choice-dilemma issues. Choice-dilemma issues are judgmental issues, in which no absolute truth exists. In risk/reward dilemmas individuals develop opinions about the minimum level of confidence (measured as a scalar value in the [0, 1] interval) required to accept a risky option with a high payoff over a less risky option with a low payoff. In other words, individuals are asked to answer questions of the following type:

> What is your minimum level of confidence (scored 0–100) required to accept a risky option with a high payoff rather than a less risky option with a low payoff?

Table 1 Prediction of an individual's final opinion on an issue. Opinions are scaled $0 - 100$. Notes: F-J stands for Friedkin–Johnsen. Standard errors are in parentheses; * $p \leq 0.05$ ** $p \leq 0.01$ *** $p \leq 0.001$; balanced random-intercept multilevel longitudinal design; maximum likelihood estimation with robust standard errors; $n = 1,800$

	(a)	(b)	(c)
F-J prediction		0.897*** (0.018)	1.157*** (0.032)
Initial opinion			−0.282*** (0.031)
Constant	58.975*** (1.550)	5.534 (1.176)	6.752*** (1.124)
Log likelihood	−8579.835	−7329.003	−7241.097

Questions are selected from a variety of domains, including medical, financial, and professional. The groups are assembled for face-to-face discussion and put under pressure to reach consensus via instructions similar to "reaching consensus is desirable, but not required." Each human subject recorded privately and chronologically on each issue:

(i) an initial opinion about the issue prior to the group discussion,
(ii) a final opinion after the group discussion (which lasted anywhere between 3–27 min), and
(iii) an allocation of "100 influence units" to the four components of the group. These influence units are described as follows: "these allocations represent your appraisal of the relative influence of each group member's opinion on your own final opinion."

The 15 issues were presented in random order and subjects were assigned to groups randomly to eliminate bias in group composition. We refer to [24] for details about the maximum likelihood multilevel random-intercept linear regression and its software implementation.

In summary, this regression analysis, presented in Table 1 below, confirms that the Friedkin–Johnsen model has predictive value for the final opinion achieved by a group discussing risk/reward choice issues.

5.2 *The Friedkin–Johnsen Model on Intellective and Multidimensional Issues*

We next briefly describe how opinion averaging models are predictive also in the setting of intellective and resource allocation issues. In other words, we extend our analysis from the risk/reward choice dilemmas to two other types of issues: analytical reliability problems with exact answers and multidimensional constrained resource allocation issues.

First, the empirical findings in [21] deal with analytical reliability problems based on Bayesian reasoning. These problems have an exact answer and, therefore are referred to as intellective issues. It is known that in such problems often, but not always, "truth wins" in the sense that the correct answer propagates from a correct individual to the others in the group. Here is an example drawn from the medical field.

> Two medical teams are working independently to achieve a cure for a disease. Team A succeeds if it can solve two scientific problems A_1 and A_2 with independent success probabilities $P[A_1] = 0.60$ and $P[A_2] = 0.45$. Team B succeeds if it can solve three scientific problems B_1, B_2, and B_3, with independent success probability $P[B_1] = 0.80$, $P[B_2] = 0.85$, $P[B_3] = 0.95$. What is your estimate of the probability that the disease will be cured?

While we refer the reader to [21] for the detailed findings, we summarize the work here by stating that the Friedkin–Johnsen model (i) has predictive value for the final opinion expressed by the group member and (ii) substantially clarifies how truth wins in groups engaged in sequences of intellective issues based on an evolving centrality of the truth in the groups.

Second, in forthcoming publications, we will report empirical findings on group decision-making on resource allocation distributions under conditions of uncertainty. Here is an example drawn from the political field.

> If you were a State Legislator, what would be your opinion on the percentage of state tax revenues that should be allocated to each the following categories: (i) Spending on Education, (ii) Spending on State Employee Wages, Health Care, and Pensions, (iii) Spending on State Physical Infrastructure Improvements, and (iv) All Other Categories (Welfare, Other Costs of Government, etc.)? These percentages must sum to 100%.

Preliminary results indicate how multidimensional opinions are constrained to evolve in certain polytopic spaces and how a single Friedkin–Johnsen model is predictive of the final group decision. The findings establish a natural meshing of automatic polytopic decision spaces, weighted averaging models, and group decision making on uncertain resource allocation problems. These findings provide a mechanistic explanation for the bounded-rationality phenomenon of satisficing, that is, the achievement of satisfactory consensus distribution as described by the Nobel award-winning work by [49].

5.3 A Reflected Appraisal Mechanism Explaining Influence Network Evolution

We next consider network evolution phenomena along sequences and, specifically, we postulate a mechanism for network evolution. As documented in [13, 20, 28], the reflected appraisal mechanism is a psychological process that affects the levels of closure-openness levels of individuals in response to an individual's perception of how others see and evaluate him or her. In this mechanism it is postulated that individuals react to their perception of their social influence, or social power, in the group decision making.

If an individual is perceived to have had a large role in influencing a group outcome, then that individual tends to elevate his or her own self-weight or, equivalently, his or her own closure level to interpersonal influence. Conversely, if an individual is perceived to have (or really does have) limited and diminishing influence on a group outcome, then the self-weight will tend to diminish.

It is a consequence of this postulated mechanism of reflected appraisal that individuals come to think of themselves in ways that are affected by what other individuals think of them. In other words, levels of stubbornness and closure-openness to interpersonal influence are ultimately social constructions and not personality characteristics.

To mathematize this group psychological mechanism, we start by describing loosely a simplified and crude model for it:

> Each individual dampens/elevates her self-weight according to her prior influence centrality in prior issues.
>
> Specifically, along the issue sequence $s = 1, 2, \dots,$ the self-weight of each individual at issue $s + 1$ is set equal to the relative control of that individual on the prior issue s.

Here, relative control over an issue outcome is tantamount to social power of an individual in the group. Here also note how we have simplified the mechanism (influence centrality) to assume that individuals are capable of perceiving from their peers their actual level of relative control.

With the notation introduced in Sect. 4 for the Friedkin–Johnsen model in Eq. (2), we define the following issue-dependent concepts:

$$\begin{aligned}
A(s) &= \text{influence matrix at issue } s,\\
a_{ii}(s) &= \text{self-weight (level of closure to influence) of individual } i \text{ at issue } s,\\
V(s) &= \text{total influence matrix at issue } s,\\
c_i(s) = V(s)^{\top}\mathbb{1}_n/n &= \text{social power of individual } i \text{ at issue } s,\\
\bar{c}_i(s) = \frac{1}{s}\sum_{t=1}^{s} c_i(t) &= \text{issue-averaged social power of individual } i \text{ up until issue } s.
\end{aligned}$$

We next perform a regression analysis of the empirical data collected in [24] to determine whether or not individuals' self-weights on issue $s + 1$ adjust along the issue sequence $s = 1, 2 \dots$ in correspondence with their social power at issue s or issue-averaged social power until issue s. As before we perform a maximum likelihood multilevel random-intercept linear regression and we refer to [24] for the corresponding technical details. The findings in Table 2 confirm that both social power and issue-averaged social power do indeed predict individuals' issue-specific self-weights on the following issues. The effect of social power $c_i(s)$ on self-weight $a_{ii}(s + 1)$ is constant along the issue sequence. Remarkably, instead, the effect of

Table 2 Prediction of an individual's level of closure to influence $a_{ii}(s+1)$ based on the individual's prior centrality $c_i(s)$ and time-averaged cumulative centrality $\bar{c}_i(s) = \frac{1}{s}\sum_{t=1}^{s} c_i(t)$. Standard errors are in parentheses. Notes: * $p \leq 0.05$ ** $p \leq 0.01$ *** $p \leq 0.001$; balanced random-intercept multilevel longitudinal design; maximum likelihood estimation with robust standard errors; $n = 1{,}680$

	(a)	(b)	(c)
$c_i(s)$		0.336*** (0.104)	
$\bar{c}_i(s)$			0.404** (0.159)
s		0.002 (0.004)	−0.018*** (0.005)
$s \times c_i(s)$		0.171 (0.012)	
$s \times \bar{c}_i(s)$			0.095*** (0.018)
Constant	0.643*** (0.016)	0.515*** (0.030)	0.498*** (0.039)
Log likelihood	−367.331	−327.051	−293.656

issue-averaged social power $\bar{c}_i(s)$ on self-weight $a_{ii}(s+1)$ increases along the issue sequence.

6 Mathematical Models for the Evolution of Influence Networks

Motivated by the empirical findings in the previous section we now propose a basic dynamical model for the evolution of self-weight, social power, and influence networks through the process of reflected appraisal. The key references for this section are [20] where a first model is proposed and [34] where a comprehensive modeling and analysis framework is developed.

6.1 Models of Reflected Appraisal = Dynamics of the Influence Network

We start by revisiting the French–Harary–DeGroot model in Eq. (1) parametrized by a single row-stochastic matrix A. We start with the fundamental observation that the entries of A do not all have the same interpretation. From an applied psychological viewpoint, the diagonal entries are self-weight values, that is, measures of self-appraisal, levels of closure to interpersonal influence and stubbornness. The off-

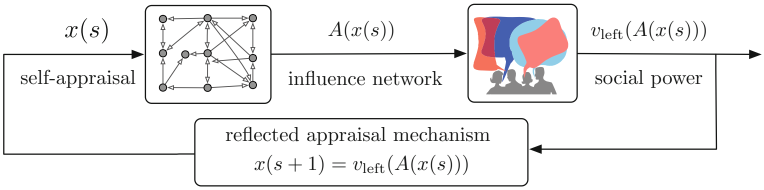

$x(s)$
self-appraisal
$A(x(s))$
influence network
$v_{\text{left}}(A(x(s)))$
social power
reflected appraisal mechanism
$x(s+1) = v_{\text{left}}(A(x(s)))$

Fig. 5 An illustration of the reflected appraisal mechanism as a feedback mechanism, leading to the definition of a closed-loop dynamical system. Here $A(x)$ is given as in Eq. (4)

diagonal terms are instead interpersonal accorded weights, that is, they represent what influence an individual is willing to accord to another. Under mild connectivity assumptions, it is possible to re-parametrize the matrix A in the following way. First, we define the self-weights

$$A_{ii} =: x_i \in [0, 1]. \tag{3}$$

Second, we assume the existence of a zero-diagonal row-stochastic matrix W, whose off-diagonal entries W_{ij} are *relative interpersonal accorded weights* satisfying the equality $A_{ij} =: (1 - x_i)W_{ij}$. In short, we can now write

$$A(x) = \text{diag}(x) + \text{diag}(\mathbb{1}_n - x)W. \tag{4}$$

Before proceeding, we define the left dominant eigenvector for W to be $w = (w_1, \ldots, w_n) = v_{\text{left}}(W)$. We recall that the right dominant eigenvector of W is $\mathbb{1}_n$ and that, under irreducibility assumptions, Theorem 1 implies that the left dominant eigenvector is positive and unique with the scaling $\mathbb{1}_n^\top w = 1$.

One can show that, after some manipulation and almost everywhere, the following equation relates the dominant eigenvector of $A(x)$ with that of W:

$$v_{\text{left}}(A(x)) = \Big(\frac{w_1}{1 - x_1}, \ldots, \frac{w_n}{1 - x_n}\Big) / \sum_{i=1}^{n} \frac{w_i}{1 - x_i}.$$

We are now ready to implement in simple, even crude, mathematical form the reflected appraisal mechanism described in the previous section: "along issues $s = 1, 2, \ldots,$ individual dampens/elevates self-weight according to prior influence centrality." We turn this into the following equation:

$$x(s+1) = v_{\text{left}}(A(x(s))), \tag{5}$$

that is, the self-weights are set equal to the relative control of the individuals on prior issues, i.e., their social power. Note that, after at most one iteration, the state of this system takes value in the simplex $\Delta_n = \{y \in \mathbb{R}^n \mid y \geq 0, \mathbb{1}_n^\top y = 1\}$. The definition of this dynamical system is illustrated in Fig. 5. We refer to the dynamical system (5) as to the DeGroot-Friedkin model, as introduced in [34] and motivated by the foundational works in [15, 20].

6.2 Equilibrium and Asymptotic Convergence Analysis

Now that we have defined a dynamical system for the evolution of self-weights and social power, we can investigate what long-term predictions are consistent with this model. It is of interest to characterize the existence and stability of equilibria, the role of network structure and parameters, and whether the influence system has a tendency toward the emergence of *autocracy* (social power concentrated in one individual) and *democracy* (social power equitably distributed among all individuals).

Theorem 7 (Equilibria and convergence) *Let W be the zero-diagonal row-stochastic matrix of relative interpersonal accorded weights and consider the resulting DeGroot-Friedkin model in Eq.* (5)*, for $n \geq 3$. Assume that W is irreducible, that w is its dominant left eigenvector, and that its associated digraph does not have star topology. Then*

(i) in the interior of the simplex there exists a unique fixed point $x^* = x^*(w_1, \dots, w_n)$*,*

(ii) from almost all initial conditions the following convergence result holds:

$$\lim_{s\to\infty} x(s) = \lim_{s\to\infty} v_{left}(A(x(s))) = x^*,$$

so that, in other words, individuals forget their initial conditions, and

(iii) the fixed point is characterized by a phenomenon of accumulation of social power and self-appraisal at the top in the following sense:

- *the fixed point x^* has same ordering of $(w_1, \dots, w_n)$, i.e., if $w_i \geq w_j$ then also $x_i^* \geq x_j^*$, and*
- *x^* is an extreme version of $(w_1, \dots, w_n)$ in the sense that there exists a social power threshold p such that, each individual i satisfies either $x_i^* < w_i < p$ or $p < w_i < x_i^*$.*

A special case of this result is the emergence of democracy for matrices W of relative interpersonal accorded weights that are doubly-stochastic. In this case, one can easily verify that the theorem above implies:

(i) the unique nontrivial fixed point is $\dfrac{1_n}{n}$, and

(ii) $\lim_{s\to\infty} x(s) = \lim_{s\to\infty} v_{\text{left}}(A(x(s))) = \dfrac{1_n}{n}$.

In other words, such networks are characterized by uniform social power and no power accumulation at the top. In simple words, one may say that the influence system is functioning as a democracy.

The other relevant special case is that of a star topology associated to W; this setting is not a direct consequence of Theorem 7 and required an ad-hoc analysis. In this case, the DeGroot-Friedkin dynamics leads to the emergence of autocracy in the following sense. If W has star topology with center j:

(i) there are no fixed points, other than the vertices of the simplex, and
(ii) $\lim_{s\to\infty} x(s) = \lim_{s\to\infty} v_{\text{left}}(A(x(s))) = e_j$,

where e_j is the jth vector of the canonical basis. In other words, individual j, the center node of the star topology, comes to be the autocrat of the influence system. In this case, the topology of the interpersonal accorded weights leads to extreme power accumulation, in the sense that the autocrat j has full power.

Naturally, we refer to the original paper for a much more detailed treatment and for the detailed proofs. It is worth, however, to review the method of proof for the statements in the main Theorem 7. We first establish the existence of the equilibrium point x^* via the Brouwer Fixed Point Theorem. Uniqueness is proved by contradiction through an elementary calculation. We next establish the following monotonicity property. Let $i_{\max}$ denote the individual with maximum $\frac{x_j(0)}{x_j^*}$, for simplicity let us here assume that it is unique. Then it turns out that $i_{\max}$ remains the index corresponding to the largest $\frac{x_j(s)}{x_j^*}$ for all subsequent issues s. (A similar result holds for $i_{\min}$.) In turn, this monotonicity allows us to prove convergence via a variation on classic "max-min" Lyapunov function:

$$V(x) = \max_j \left(\ln \frac{x_j}{x_j^*} \right) - \min_j \left(\ln \frac{x_j}{x_j^*} \right).$$

It is historically interesting to mention that, to the best of our knowledge, the earliest work introducing a max-min Lyapunov function is the work [50] on distributed optimization. This work is however related to the classic work by [5]. We also refer to [48] for a review of this history and for a study of consensus in noncommutative spaces.

7 Conclusions

This chapter has reviewed a large literature on the mathematics of network systems and its application to the study of dynamical models for the evolution of opinions and influence systems. We have presented both mathematical results and empirical findings.

Overall our recent works provide a new perspective on influence networks and social power, grounded in multiple human subject experiments and based on both multilevel regression and control theoretical analysis. We have designed, executed, and analyzed experiments on group discussions for judgmental and intellective issues. We have proposed, analyzed, and validated a novel dynamical model with feedback. In turn, this model provides a novel mechanism that may explain the phenomenon of power accumulation and emergence of autocracy in certain influence networks.

Ongoing and future research will focus on (1) studying the mathematical robustness of our findings to modeling assumptions, (2) studying and modeling the evolution of the matrix of interpersonal accorded weights, and (3) performing larger scale controlled experiments perhaps via online software. We will also endeavor to design and validate intervention strategies to influence group discussions.

Acknowledgements This chapter is humbly prepared in honor of our dear friend and collaborator Roberto Tempo. Both authors are deeply thankful for the opportunity to collaborate with him on the themes of this article during the last years of his career. His visits to Santa Barbara were full with intellectual excitement, unending discussions, friendship, and joy. His intelligence, humor, positive and friendly attitude, and wisdom are already sorely missed.
This material is based upon work supported by, or in part by, the U.S. ARL and the U.S. ARO under grant W911NF-15-1-0577, the AFOSR under grant FA9550-15-1-0138, and the DOE under grant XAT-6-62531-01. The first author thanks Florian Dörfler and Alessandro Giua for insightful conversations on the topics of this document.
The first part of this chapter is a highly abbreviated version of the first part of the textbook [7]. All images in this chapter are reproduced here with permission from [7] – they are licensed under the Creative Commons BY-SA 4.0 Attribution-ShareAlike International License. Tables 1 and 2 are reproduced here with permission from [24] – they are licensed under the Creative Commons SA 4.0 Attribution International License.

References

1. R. P. Abelson. Mathematical models of the distribution of attitudes under controversy. In N. Frederiksen and H. Gulliksen, editors, *Contributions to Mathematical Psychology*, volume 14, pages 142–160. Holt, Rinehart, & Winston, 1964. ISBN 0030430100.
2. M. Arcak, C. Meissen, and A. Packard. *Networks of Dissipative Systems: Compositional Certification of Stability, Performance, and Safety*. Springer, 2016. ISBN 978-3-319-29928-0. https://doi.org/10.1007/978-3-319-29928-0.
3. H. Bai, M. Arcak, and J. Wen. *Cooperative Control Design*. Springer, 2011. ISBN 1461429072.
4. D. Bindel, J. Kleinberg, and S. Oren. How bad is forming your own opinion? *Games and Economic Behavior*, 92: 248–265, 2015. https://doi.org/10.1016/j.geb.2014.06.004.
5. G. Birkhoff. Extensions of Jentzsch's theorem. *Transactions of the American Mathematical Society*, 85 (1): 219–227, 1957. https://doi.org/10.2307/1992971.
6. S. Boccaletti, V. Latora, Y. Moreno, M. Chavez, and D. U. Hwang. Complex networks: Structure and dynamics. *Physics Reports*, 424 (4-5): 175–308, 2006. https://doi.org/10.1016/j.physrep.2005.10.009.
7. F. Bullo. *Lectures on Network Systems*. CreateSpace, 1 edition, 2018. ISBN 978-1986425643. http://motion.me.ucsb.edu/book-lns. With contributions by J. Cortés, F. Dörfler, and S. Martínez.
8. F. Bullo, J. Cortés, and S. Martínez. *Distributed Control of Robotic Networks*. Princeton University Press, 2009. ISBN 978-0-691-14195-4. URL http://www.coordinationbook.info.
9. Y. Cao, W. Yu, W. Ren, and G. Chen. An overview of recent progress in the study of distributed multi-agent coordination. *IEEE Transactions on Industrial informatics*, 9 (1): 427–438, 2013. https://doi.org/10.1109/TII.2012.2219061.
10. C. Castellano, S. Fortunato, and V. Loreto. Statistical physics of social dynamics. *Reviews of Modern Physics*, 81 (2): 591–646, 2009. https://doi.org/10.1103/RevModPhys.81.591.
11. G. Chen, X. Duan, N. E. Friedkin, and F. Bullo. Social power dynamics over switching and stochastic influence networks. *IEEE Transactions on Automatic Control*, 2018. https://doi.org/10.1109/TAC.2018.2822182. To appear.
12. X. Chen, J. Liu, M.-A. Belabbas, Z. Xu, and T. Başar. Distributed evaluation and convergence of self-appraisals in social networks. *IEEE Transactions on Automatic Control*, 62 (1): 291–304, 2017. https://doi.org/10.1109/TAC.2016.2554280.

13. C. H. Cooley. *Human Nature and the Social Order*. Charles Scribner Sons, New York, 1902.
14. E. Cristiani, B. Piccoli, and A. Tosin. *Multiscale Modeling of Pedestrian Dynamics*. Springer, 2014. ISBN 978-3-319-06619-6.
15. M. H. DeGroot. Reaching a consensus. *Journal of the American Statistical Association*, 69 (345): 118–121, 1974. https://doi.org/10.1080/01621459.1974.10480137.
16. D. Easley and J. Kleinberg. *Networks, Crowds, and Markets: Reasoning About a Highly Connected World*. Cambridge University Press, 2010. ISBN 0521195330.
17. B. A. Francis and M. Maggiore. *Flocking and Rendezvous in Distributed Robotics*. Springer, 2016. ISBN 978-3-319-24727-4.
18. P. Frasca, H. Ishii, C. Ravazzi, and R. Tempo. Distributed randomized algorithms for opinion formation, centrality computation and power systems estimation: A tutorial overview. *European Jounal of Control*, 24: 2–13, 2015. https://doi.org/10.1016/j.ejcon.2015.04.002.
19. J. R. P. French Jr. A formal theory of social power. *Psychological Review*, 63 (3): 181–194, 1956. https://doi.org/10.1037/h0046123.
20. N. E. Friedkin. A formal theory of reflected appraisals in the evolution of power. *Administrative Science Quarterly*, 56 (4): 501–529, 2011. https://doi.org/10.1177/0001839212441349.
21. N. E. Friedkin and F. Bullo. How truth wins in opinion dynamics along issue sequences. *Proceedings of the National Academy of Sciences*, 114 (43): 11380–11385, 2017. https://doi.org/10.1073/pnas.1710603114.
22. N. E. Friedkin and E. C. Johnsen. Social influence networks and opinion change. In S. R. Thye, E. J. Lawler, M. W. Macy, and H. A. Walker, editors, *Advances in Group Processes*, volume 16, pages 1–29. Emerald Group Publishing Limited, 1999. ISBN 0762304529.
23. N. E. Friedkin and E. C. Johnsen. *Social Influence Network Theory: A Sociological Examination of Small Group Dynamics*. Cambridge University Press, 2011. ISBN 9781107002463.
24. N. E. Friedkin, P. Jia, and F. Bullo. A theory of the evolution of social power: Natural trajectories of interpersonal influence systems along issue sequences. *Sociological Science*, 3: 444–472, 2016a. https://doi.org/10.15195/v3.a20.
25. N. E. Friedkin, A. V. Proskurnikov, R. Tempo, and S. E. Parsegov. Network science on belief system dynamics under logic constraints. *Science*, 354 (6310): 321–326, 2016b. https://doi.org/10.1126/science.aag2624.
26. F. R. Gantmacher. *The Theory of Matrices*, volume 1 and 2. Chelsea, New York, 1959. ISBN 0-8218-1376-5 and 0-8218-2664-6. Translation of German edition by K. A. Hirsch.
27. F. Garin and L. Schenato. A survey on distributed estimation and control applications using linear consensus algorithms. In A. Bemporad, M. Heemels, and M. Johansson, editors, *Networked Control Systems*, LNCIS, pages 75–107. Springer, 2010. https://doi.org/10.1007/978-0-85729-033-5_3.
28. V. Gecas and M. L. Schwalbe. Beyond the looking-glass self: Social structure and efficacy-based self-esteem. *Social Psychology Quarterly*, 46 (2): 77–88, 1983. https://doi.org/10.2307/3033844.
29. F. Harary. A criterion for unanimity in French's theory of social power. In D. Cartwright, editor, *Studies in Social Power*, pages 168–182. University of Michigan, 1959. ISBN 0879442301. http://psycnet.apa.org/psycinfo/1960-06701-006.
30. R. A. Horn and C. R. Johnson. *Matrix Analysis*. Cambridge University Press, 1985. ISBN 0521386322.
31. M. O. Jackson. *Social and Economic Networks*. Princeton University Press, 2010. ISBN 0691148201.
32. P. Jia, N. E. Friedkin, and F. Bullo. Opinion dynamics and social power evolution: A single-timescale model. *IEEE Transactions on Control of Network Systems*, Dec. 2017a. Submitted.
33. P. Jia, N. E. Friedkin, and F. Bullo. Opinion dynamics and social power evolution over reducible influence networks. *SIAM Journal on Control and Optimization*, 55 (2): 1280–1301, 2017b. https://doi.org/10.1137/16M1065677.
34. P. Jia, A. MirTabatabaei, N. E. Friedkin, and F. Bullo. Opinion dynamics and the evolution of social power in influence networks. *SIAM Review*, 57 (3): 367–397, 2015. https://doi.org/10.1137/130913250.

35. S. Martínez, J. Cortés, and F. Bullo. Motion coordination with distributed information. *IEEE Control Systems*, 27 (4): 75–88, 2007. https://doi.org/10.1109/MCS.2007.384124.
36. M. Mesbahi and M. Egerstedt. *Graph Theoretic Methods in Multiagent Networks*. Princeton University Press, 2010. ISBN 9781400835355.
37. A. MirTabatabaei, P. Jia, N. E. Friedkin, and F. Bullo. On the reflected appraisals dynamics of influence networks with stubborn agents. In *American Control Conference*, pages 3978–3983, Portland, OR, USA, June 2014. https://doi.org/10.1109/ACC.2014.6859256.
38. M. E. J. Newman. The structure and function of complex networks. *SIAM Review*, 45 (2): 167–256, 2003. https://doi.org/10.1137/S003614450342480.
39. M. E. J. Newman. *Networks: An Introduction*. Oxford University Press, 2010. ISBN 0199206651.
40. K.-K. Oh, M.-C. Park, and H.-S. Ahn. A survey of multi-agent formation control. *Automatica*, 53: 424–440, 2015. https://doi.org/10.1016/j.automatica.2014.10.022.
41. S. E. Parsegov, A. V. Proskurnikov, R. Tempo, and N. E. Friedkin. A new model of opinion dynamics for social actors with multiple interdependent attitudes and prejudices. In *IEEE Conf. on Decision and Control*, pages 3475–3480, 2015.
42. S. E. Parsegov, A. V. Proskurnikov, R. Tempo, and N. E. Friedkin. Novel multidimensional models of opinion dynamics in social networks. *IEEE Transactions on Automatic Control*, 62 (5): 2270–2285, 2017. https://doi.org/10.1109/TAC.2016.2613905.
43. A. V. Proskurnikov and R. Tempo. A tutorial on modeling and analysis of dynamic social networks. Part I. *Annual Reviews in Control*, 43: 65–79, 2017. https://doi.org/10.1016/j.arcontrol.2017.03.002.
44. C. Ravazzi, P. Frasca, R. Tempo, and H. Ishii. Ergodic randomized algorithms and dynamics over networks. *IEEE Transactions on Control of Network Systems*, 2 (1): 78–87, 2015. https://doi.org/10.1109/TCNS.2014.2367571.
45. W. Ren and R. W. Beard. *Distributed Consensus in Multi-vehicle Cooperative Control*. Communications and Control Engineering. Springer, 2008. ISBN 978-1-84800-014-8.
46. W. Ren, R. W. Beard, and E. M. Atkins. Information consensus in multivehicle cooperative control. *IEEE Control Systems*, 27 (2): 71–82, 2007. https://doi.org/10.1109/MCS.2007.338264.
47. E. Seneta. *Non-negative Matrices and Markov Chains*. Springer, 2 edition, 1981. ISBN 0387297650.
48. R. Sepulchre, A. Sarlette, and P. Rouchon. Consensus in non-commutative spaces. In *IEEE Conf. on Decision and Control*, pages 6596–6601, Atlanta, USA, Dec. 2010. https://doi.org/10.1109/CDC.2010.5717072.
49. H. A. Simon. *Administrative Behavior. A Study of Decision-making Processes in Administrative Organization*. Free Press, 1947.
50. J. N. Tsitsiklis, D. P. Bertsekas, and M. Athans. Distributed asynchronous deterministic and stochastic gradient optimization algorithms. *IEEE Transactions on Automatic Control*, 31 (9): 803–812, 1986. https://doi.org/10.1109/TAC.1986.1104412.
51. M. Ye, J. Liu, B. D. O. Anderson, C. Yu, and T. Başar. Evolution of social power in social networks with dynamic topology. *IEEE Transactions on Automatic Control*, 2018. https://doi.org/10.1109/TAC.2018.2805261. To appear.

Distributed Randomized Algorithms for PageRank Computation: Recent Advances

Hideaki Ishii and Atsushi Suzuki

Dedicated to Roberto Tempo

Abstract PageRank is a well-known centrality measure for the web used in search engines, representing the importance of each web page. Its computation is very large scale as the rankings for all pages in the entire web are to be calculated at once, and this has prompted various studies on the algorithmic aspects of this problem. In this chapter, we first present a short overview on the recent studies on distributed algorithms that have been developed in the systems control area. These algorithms share the features that (i) each page computes its own PageRank value by interacting with pages connected over hyperlinks and (ii) gossip-type randomization is employed in the update schemes. Then, we introduce a new class of distributed algorithms for PageRank, which is based on a simple but novel interpretation. It is demonstrated via analysis and numerical simulations that these algorithms have significant advantages in their convergence performances in comparison with other existing techniques. The chapter ends with a brief summary of the works on randomization-based distributed algorithms, heavily influenced by the collaboration with Roberto Tempo, to whom this writing is dedicated.

This work was supported in part by the JST CREST Grant No. JPMJCR15K3 and by JSPS under Grant-in-Aid for Scientific Research Grant No. 15H04020.

H. Ishii (✉) · A. Suzuki
Department of Computer Science, Tokyo Institute of Technology, Yokohama 226-8502, Japan
e-mail: ishii@c.titech.ac.jp

A. Suzuki
e-mail: suzuki@sc.dis.titech.ac.jp

© Springer Nature Switzerland AG 2018
T. Başar (ed.), *Uncertainty in Complex Networked Systems*,
Systems & Control: Foundations & Applications,
https://doi.org/10.1007/978-3-030-04630-9_12

1 Introduction

For search engines at Google, one of the many measures used for ranking the web pages in search results is the so-called PageRank. For each web page, the PageRank value provides a measure of its importance or popularity, which is based on the network structure of the web in terms of the hyperlinks. A page is considered more important and popular if it receives more hyperlinks from other pages and especially those that are important themselves. When such a structure in the hyperlinks is present around a page, it suggests how easily users surfing the web might arrive there, even by chance. The notion of PageRank was proposed by the co-founders of Google, Brin and Page, in [7]. It has received a great deal of interest especially in the context of complex networks as it is an effective measure of centrality. General references on this topic include the monograph [33] and the overview papers [23, 27].

The problem of computing PageRank itself has been a subject of extensive studies over the years. Despite the simple nature of the problem, because of the problem size involving billions of pages in the web, its efficient computation has remained as a difficult task. For centralized computation, the simple power method has been the realistic option for this reason. Alternative methods have been studied based on Monte Carlo simulations of the underlying Markov chain (e.g., [1]) and distributed algorithms (e.g., [48]).

Recently, in the systems control community, PageRank has gained much attention from the viewpoint of distributed computation. In particular, in [26], it was pointed out that the PageRank problem shared several similarities with the multi-agent consensus problem (e.g., [8, 40]) and randomized distributed algorithms were developed. The approach there is to view the web as a network of pages capable of communicating with neighbors connected via hyperlinks. Then, in a distributed manner, each web page can act as an agent which computes its own PageRank value iteratively by exchanging data with other pages. To cope with the network size, the pages randomly determine when to initiate updates, which is sometimes called gossiping [6]. The method is guaranteed to converge in the mean-square sense. However, it involves the time averaging of the state values, resulting in the convergence rate of order $1/k$ with respect to the updating time k.

The focus of this chapter is the research activities on the topic of PageRank that have taken place since. The chapter consists of roughly three parts. In the first, we provide a brief overview on the subject of distributed computation of PageRank, starting with the work of [26]. More recently, studies focusing on convergence speeds have appeared. In particular, it has been found that convergence with exponential rate is possible. Notably, in [57], the PageRank problem is formulated as a least-squares problem and then a distributed gradient-descent algorithm is applied. This work also points out the difficulty in assuming the global parameter of the total number of web pages to be known by all pages, leading to alternative algorithms that enable the PageRank calculation without the knowledge. The work [14] employs another technique of matching pursuit algorithms for solving linear equations and provides a randomized version. On the other hand, in [32], a modified gradient-descent algorithm is constructed so that the states of all pages remain to have the total equal to one throughout its execution. We also refer to [43], which studied stochastic gradient algorithms for PageRank. In this first part, we will introduce and discuss these algorithms and their different features such as their convergence speeds and required loads for communication and computation.

In the second part, we propose a novel approach towards the PageRank computation from a slightly different perspective [50]. By making use of the property that PageRank involves a stochastic matrix representing the network structure of the pages, we reformulate the problem in a certain way, expressed as an infinite matrix series. This formulation leads us to a completely different set of algorithms tailored to the problem. Specifically, we propose algorithms for both synchronous and asynchronous settings in the communication among the linked pages. Their convergence properties are fully analyzed in the development. For the asynchronous case, we employ randomization-based gossiping, but the probability to be selected for updates need not be uniform. We show that they have desirable characteristics including exponential convergence and relatively low requirements for the communication among agents. Through numerical examples, we carry out a detailed comparison of the algorithms discussed in the chapter.

The novel aspects of the proposed algorithms can be summarized as follows. First, the reformulation idea is simple and its advantage may not be immediately clear. This is partly because additional states are introduced for the pages, which increase the computational burden. In fact, in the synchronous case, the convergence is not necessarily faster than the power method. Second, in the proposed randomized algorithms, the states are guaranteed to reach the true PageRank values from below in a monotonic fashion. Hence, despite the randomization, the responses of the states are smooth, which may explain the efficiency of the approach. Third, in the randomization, the pages to initiate updates can be chosen under arbitrary distributions. It should be noted that no change is necessary in the algorithm due to the chosen distribution. This leaves a certain degree of freedom in enhancing the convergence speed as discussed in the numerical example section. Furthermore, the pages communicate over only their outgoing hyperlinks and do not require the knowledge of the incoming ones as in some methods in the literature.

As the last part, which is the shortest, we discuss the different roles that randomization may play in networked systems problems and, in particular, multi-agent consensus problems. In addition to gossiping in communication, probabilistic techniques can be useful in enhancing distributed decision-making as well as cybersecurity levels for systems in hazardous environments where malicious attackers may take advantage to disrupt the execution of algorithms and control.

Finally, we should note that, in the area of systems control, studies on PageRank have grown in a spectrum of interesting directions; see [27] for more discussions. For distributed algorithms, the approach of [26] has been extended, for example, to incorporate aggregation of pages to realize more efficient computation in [29]. Stronger convergence properties with probability one are established with the help of stochastic approximation results in [58]. Moreover, in [12, 28, 36], different probability distributions are employed for the randomized updates in the pages, making them capable to function, e.g., even if the channels for the communication among pages are unreliable. Other works conducted studied on the problem of finding the ranges of PageRank values when a subset of the hyperlinks is uncertain in the sense whether they are actually present [25], optimization of PageRank for pages of interest by changing the link structure [13, 19], and a game theoretic analysis for enhancing PageRank through aggregation of pages [38].

This chapter is organized as follows: We first give an overview on the PageRank problem in Sect. 2. In Sect. 3, we introduce the recent works on distributed computation approaches for the PageRank. In Sect. 4, an alternative formulation for the problem is presented, which is then used for deriving two novel distributed algorithms based on randomized gossiping. Illustrative numerical examples are provided in Sect. 5. A more general discussion on the topic of randomization-based techniques in the context of multi-agent systems is provided in Sect. 6. The chapter is finally concluded in Sect. 7.

Notation: For vectors and matrices, inequalities are used to denote entry-wise inequalities: For $X, Y \in \mathbb{R}^{n\times m}$, $X \leq Y$ implies $x_{ij} \leq y_{ij}$ for all i, j; in particular, we say that the matrix X is nonnegative if $X \geq 0$ and positive if $X > 0$. A probability vector is a nonnegative vector $v \in \mathbb{R}^n$ such that $\sum_{i=1}^n v_i = 1$. A matrix $X \in \mathbb{R}^{n\times n}$ is said to be (column) stochastic if it is nonnegative and each column sum equals 1, i.e., $\sum_{i=1}^n x_{ij} = 1$ for each j. Let $\mathbf{1}_n \in \mathbb{R}^n$ be the vector whose entries are all 1 as $\mathbf{1}_n := [1 \cdots 1]^T$. For a vector x, we use $\|x\|$ to denote its the Euclidean norm. For a discrete set $\mathscr{D}$, its cardinality is given by $|\mathscr{D}|$.

2 The PageRank Problem

In this section, we introduce the basics of PageRank and its interpretations commonly employed for its computation [7, 27, 33].

The underlying idea for PageRank is to regard the entire web as a directed graph consisting of web pages with hyperlinks. By solely using the network structure there, PageRank provides a powerful measure of centrality, indicating how important or popular each web page is.

Let n be the total number of pages in the web; we assume $n \geq 2$ to avoid the trivial case. The web graph is given by $\mathscr{G} := (\mathscr{V}, \mathscr{E})$, where $\mathscr{V} := \{1, 2, \dots, n\}$ is the set of vertices representing the web pages, and $\mathscr{E}$ is the set of hyperlinks connecting the pages. Here, $(i, j) \in \mathscr{E}$ holds if and only if page i has a hyperlink to page j. In such a case, for page i, page j becomes its out-neighbor, whereas page i is the in-neighbor of page j.

The hyperlinks are not always mutual, so this graph is generally a directed graph. When a node does not have any outgoing link, it is referred to as a dangling node. Here, to simplify the discussion, we assume that all pages have at least one outgoing hyperlink. This is commonly done by slightly modifying the structure of the web, specifically by adding hyperlinks from such dangling nodes, which correspond to the use of back buttons; see, e.g., [33] for more details.

Next, we define the hyperlink matrix $A = \left(a_{ij}\right) \in \mathbb{R}^{n\times n}$ of this graph by

$$a_{ij} := \begin{cases} \frac{1}{n_j} & \text{if } \ i \in \mathscr{L}_j, \\ 0 & \text{otherwise}, \end{cases} \tag{1}$$

where $\mathscr{L}_i := \{j : (i, j) \in \mathscr{E}\}$ is the set of outgoing neighbors of page i and n_i is its cardinality, i.e., $n_i := |\mathscr{L}_i|$. By the assumption that all pages have one or more hyperlinks, this matrix A is (column) stochastic, that is, it is a nonnegative matrix where the sum of entries in each column is equal to 1.

For the web consisting of n pages, the PageRank vector $x^* \in [0, 1]^n$ is defined as

$$x^* = (1 - m)Ax^* + \frac{m}{n}\mathbf{1}_n, \ \ \mathbf{1}_n^T x^* = 1, \tag{2}$$

where the parameter is chosen as $m \in (0, 1)$. Note that x^* is a nonnegative vector, and the second equation above indicates that it is a probability vector. For m, it is common to use the value 0.15 as proposed by [7]; we follow this convention in this chapter.

The definition in (2) can be rewritten as

$$x^* = Mx^*, \ \ \mathbf{1}_n^T x^* = 1, \tag{3}$$

where the modified link matrix M is given by

$$M := (1 - m)A + \frac{m}{n}\mathbf{1}_n\mathbf{1}_n^T.$$

Since M is a convex combination of two stochastic matrices A and $(1/n)\mathbf{1}_n\mathbf{1}_n^T$, it is stochastic as well. It is now clear that x^* is the eigenvector of the link matrix M corresponding to the eigenvalue 1. Such an eigenvector x^* exists and is unique; this follows from Perron's theorem [24] because the stochastic matrix M has the property of being positive.

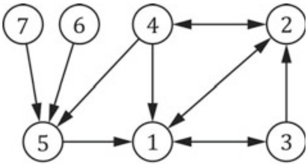
For its computation, the PageRank vector x^* can be obtained by solving the linear equation (2) or (3). The practical issue that requires serious attention is the size of the problem. Recall that the dimension of the PageRank vector is the same as the number of pages in the web. Hence, the computation must rely on algorithms that have simple structures.

A common approach, which is centralized, is to employ the power method. It is expressed by the iteration of the form

$$x(k + 1) = (1 - m)Ax(k) + \frac{m}{n}\mathbf{1}_n, \tag{4}$$

where $x(k) \in \mathbb{R}^n$ is the state whose initial value $x(0)$ can be taken as any probability vector. By Perron's theorem [24], it follows that $x(k) \to x^*$ as $k \to \infty$.

Fig. 1 An example graph with seven nodes



Another interesting interpretation of PageRank is that of the *random surfer* model. It follows from the expression in (3) that the PageRank vector x^* can be regarded as the stationary distribution of a Markov chain whose transition matrix is represented by the stochastic matrix M. We may imagine a person who surfs the web in a random manner: When he visits one page, with probability $1 - m$, he chooses one of the links with equal probability; otherwise, with probability m, he decides to jump to any of the pages in the web with equal probability, that is, $1/n$. Under this model, the PageRank of page i can be regarded as the probability that such a surfer visits there in the steady state. Clearly, the link structure of the web creates pages which are more likely to be visited by such an imaginary surfer.

We now present a simple example to illustrate the problem of PageRank.

Example 1 Consider the web consisting of seven pages depicted in Fig. 1. The hyperlink matrix A of this web is given by

$$A = \begin{bmatrix} 0 & \frac{1}{2} & \frac{1}{2} & \frac{1}{3} & 1 & 0 & 0 \\ \frac{1}{2} & 0 & \frac{1}{2} & \frac{1}{3} & 0 & 0 & 0 \\ \frac{1}{2} & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & \frac{1}{2} & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & \frac{1}{3} & 0 & 1 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}.$$

We can calculate the PageRank vector of this graph as

$$x^* = \left[0.316 \; 0.259 \; 0.156 \; 0.132 \; 0.0951 \; 0.0214 \; 0.0214\right]^T.$$

It is noted that the indices of the pages are set according to the order of their PageRanks. Pages 1 and 2 have, respectively, four and three incoming links, making their rankings high. Pages 6 and 7 have no incoming hyperlink and, as a result, take the lowest possible PageRank, which is equal to $m/n = 0.15/7 = 0.0214$. We should emphasize that the number of links is not the only factor that determines PageRank. Both pages 3 and 4 have only one incoming link, but take better rankings than page 5, which has three links. This is because the ranks also depend on the values of the pages from which the links originate. In this respect, pages 3 and 4 are more advantageous than page 5, whose links include those from pages 6 and 7, having minor impact on its importance.

3 Distributed Algorithms for PageRank

In this section, we discuss the recent studies on randomized distributed algorithms for the PageRank computation and their differences. Namely, we focus on the methods developed in Ishii and Tempo [26], You et al. [57], Dai and Freris [14], and Lagoa et al. [32].

The computation of PageRank may be costly if it is performed centrally because of the size of the problem determined by the number of pages in the entire web. Hence, distributed computation is one natural approach to resolve this issue. In the systems control community, this viewpoint is particularly motivated by the recent research on coordinated control of multi-agent systems (e.g., [8, 40]). In the setting of the web, the pages may act as agents interacting over the hyperlinks to compute their own PageRank values through an iterative algorithm. In practice, resources for computation and communication are available at the numerous web servers where the data regarding the pages connected via hyperlinks is available.

In what follows, we present several distributed algorithms for PageRank computation. We use the common notation for the value of page i at time k, which is expressed as $x_i(k)$. In view of the size of the system, one issue is how to coordinate the pages in terms of the timings for them to initiate updates and communication of their values. Here, we bring in randomization in the pages' decisions and employ the so-called *gossip*-type communication: At each time step, one of the pages is randomly chosen in an independently identically distributed (i.i.d.) manner. Denoting the index of the page chosen at time k by $\theta(k) \in \mathcal{V}$, we have

$$\mathrm{Prob}\big\{\theta(k) = i\big\} = \frac{1}{n} \text{ for } i \in \mathcal{V} \text{ and } k \in \mathbb{Z}_+. \tag{5}$$

All of the algorithms discussed here are equipped with this mechanism.

Among the distributed algorithms, there are differences in how the chosen page $\theta(k)$ interacts with its neighboring linked pages. Some approaches require that page $\theta(k)$ send its current value $x_{\theta(k)}(k)$ to its out-neighbors whereas other algorithms mandate communications with its in-neighbors. Another aspect that will become an important difference is the level of synchronization necessary among the pages in their clocks. In general, it is difficult to expect that a common clock exists, shared by all pages with perfect synchronization.

3.1 Towards Distributed Computations

First, we present the approach of [26], which introduced distributed algorithms from the viewpoint of coordinated control of multi-agent systems. The update law for this case is motivated by the centralized iterative one in (4). In comparison with other algorithms, a key feature is the reliance on the use of stochastic matrices.

The idea is to employ distributed link matrices A_i containing the ith column of the link matrix A given in (1), and the remaining columns are set so that A_i becomes a stochastic matrix. Here, we present the version from [27], which takes a slightly simpler form. More concretely, the distributed link matrices A_i, $i = 1, \dots, n$, are defined by

(i) The ith column of A_i is equal to the ith column of A.
(ii) The diagonal entries of the columns other than the ith one are equal to one.
(iii) The remaining entries are chosen to be zero.

It is clear that these matrices A_i are column stochastic by construction.

As a consequence, according to the probability distribution of the process $\theta(k)$, the average matrix $\overline{A} := \mathbb{E}[A_{\theta(k)}]$ takes a special form as

$$\overline{A} = \frac{1}{n}\sum_{i=1}^{n} A_i = \frac{2}{n}A + \left(1 - \frac{2}{n}\right) I. \tag{6}$$

Notice that this matrix $\overline{A}$ is a convex combination of two column stochastic matrices.

The distributed update law using the link matrices can be represented as follows:

$$x(k+1) = (1-\hat{m})A_{\theta(k)}x(k) + \frac{\hat{m}}{n}\mathbf{1}_n, \tag{7}$$

where the initial vector $x(0)$ is a probability vector and $\hat{m} \in (0, 1)$ is a parameter to be determined, corresponding to $m = 0.15$ in the centralized algorithm (4). This update law is accompanied by the time average process $y(k)$ of the states $x(0), \dots, x(k)$ given by

$$y(k) := \frac{1}{k+1}\sum_{\ell=0}^{k} x(\ell). \tag{8}$$

Observe that each page i can locally compute the average $y_i(k)$ of its own past states $x_i(\ell)$, $\ell = 0, \dots, k$. It is also noted that the state $x(k)$ and hence the average $y(k)$ are both probability vectors at all k.

We now discuss the convergence properties of the update scheme (7) and (8). The parameter $\hat{m}$ is to be set as

$$\hat{m} := \frac{2m}{n - m(n-2)}.$$

This choice allows us to represent the PageRank vector x^* based on the average matrix $\overline{A}$ in (6) as

$$x^* = \big(1-\hat{m}\big)\overline{A}x^* + \frac{\hat{m}}{n}\mathbf{1}.$$

Then, we can establish that the expected value $\mathbb{E}[x(k)]$ of the states converges to the PageRank vector x^*, that is, $\mathbb{E}[x(k)] \to x^*$ as $k \to \infty$. While it is not possible to show that the state vector $x(k)$ itself converges to the PageRank vector x^*, it follows that its time average $y(k)$ does so in the mean-square sense. More specifically, for any initial state $x(0)$ which is a probability vector, it holds that

$$\mathbb{E}\big[\|y(k)-x^*\|^2\big] \to 0 \;\text{ as} k \to \infty.$$

This kind of convergence is referred to as ergodicity of random processes and the time average plays an important role. The original state $x(k)$ in general demonstrates persisting oscillations without convergence. Furthermore, the scheme converges with probability one, as was shown in [58] using methods from stochastic approximation.

We discuss a few issues that may be of concern about the update scheme (7) and (8). They become relevant because of the involvement of the time averaging in $y(k)$. One is that the speed of convergence is somewhat limited. It can be shown to be linear and, more specifically, of the order $1/k$. Another is the necessity for the pages to be synchronized; this is needed for correctly computing the time average $y_i(k)$ by all pages as pointed out in [57]. Finally, in this algorithm, each page i must store and update two variables, namely, $x_i(k)$ and $y_i(k)$. Interesting extensions of this class of algorithms have been made in [22, 46], where applications to sensor localization in wireless networks, social dynamics, and state estimation in power systems can be found.

3.2 *Enhancement in Convergence Speed*

Next, we proceed to discuss the alternative approach of the work [57] for distributed computation of the PageRank vector. Their approach is based on the viewpoint of distributed optimization, which in turn allows us to adopt existing algorithms from the area and to demonstrate its exponential convergence.

The starting point is to rewrite the PageRank vector x^* in its definition as the solution to the linear equation given by

$$[I-(1-m)A]\,x^* = \frac{m}{n}\mathbf{1}_n. \tag{9}$$

This implies that the vector can be obtained through an unconstrained optimization given by

$$x^* = \arg\min_x \left\| [I-(1-m)A]\,x - \frac{m}{n}\mathbf{1}_n \right\|^2. \tag{10}$$

Under this formulation, the PageRank computation problem can be further reduced to a form having a distributed nature more explicitly. Let

$$H := I-(1-m)A \;\text{ and }\; g := \frac{m}{n}\mathbf{1}_n. \tag{11}$$

Denote by $\tilde{h}_i^T \in \mathbb{R}^n$ the ith row of the matrix H. The optimization problem (10) can be expressed as

$$x^* = \arg\min_x \sum_{i=1}^{n} \left(\tilde{h}_i^T x - g_i\right)^2. \tag{12}$$

For solving this optimization problem, the work [57] presents a distributed gradient-descent algorithm, which can be seen as an extension of the randomized Kaczmarz algorithm of [59]. An interesting feature is that the index $\theta(k)$ of the chosen page follows a Markov chain whose states correspond to the pages in the web; this is introduced to deal with the situation where the total number n of pages in the web is unknown.

We now present the simple case where $\theta(k)$ is an i.i.d. random process according to (5). This version requires the knowledge of the size n of the web. The update scheme can be given as follows:

$$\begin{aligned} x(k+1) &= x(k) - \frac{1}{2n} \cdot \frac{d}{dx} \left(g_{\theta(k)} - \tilde{h}_{\theta(k)}^T x\right)^2 \Big|_{x=x(k)} \\ &= x(k) + \frac{1}{n} \tilde{h}_{\theta(k)} \left(g_{\theta(k)} - \tilde{h}_{\theta(k)}^T x(k)\right), \end{aligned} \tag{13}$$

where the initial condition is set as $x(0) = 0$.

It is shown in [57] that the randomized algorithm in (13) has a guaranteed convergence rate and, in fact, it exponentially converges almost surely to the true PageRank vector x^*. This is an important characteristic, which is not attainable in the approach of [26] based on stochastic matrices and time averaging of the state. Another difference is that this algorithm involves only one variable per page for the PageRank computation.

On the other hand, it is important to note that the necessary communication load among the nodes may be high and requires the knowledge of the in-neighbors at each page. This can be confirmed since in the update rule (13), the row $\tilde{h}_{\theta(k)}$ of H corresponding to the chosen page $\theta(k)$ at time k appears twice. This indicates that (i) the values $x_j(k)$ of the in-neighbors $j \in \mathcal{N}_{\theta(k)}$ of page $\theta(k)$ must be collected for obtaining $\tilde{h}_{\theta(k)}^T x(k)$ and then (ii) the value $g_{\theta(k)} - \tilde{h}_{\theta(k)}^T x(k)$ is sent back to the same in-neighbors for the update of their own values $x_j(k)$.

3.3 Reduction in Communication Loads

The perspective of linear equations was the motivation also for the third approach for PageRank computation proposed in [14]. The distributed algorithm there employs the technique of matching pursuit algorithms from the area of signal processing (e.g., [39]). Matching pursuit is for approximating a signal with a finite number of functions (called atoms). As a consequence, this algorithm is guaranteed to possess exponential convergence as well.

In contrast to the approach of [57], which required the updating pages to communicate with their in-neighbors, the algorithm of [14] involves interactions only with the out-neighbors. Such neighbors are easily known to any page as they can be reached through its own hyperlinks.

To this end, we introduce the notations for the columns of the matrix H in (11). Let $h_i \in \mathbb{R}^n$ be the ith column of H. In this case, each page is equipped with two scalar variables denoted by $x_i(k)$ and $r_i(k)$, whose initial values are given by $x_i(0) = 0$ and $r_i(0) = m/n$. As in the algorithms discussed so far, let $\theta(k)$ be the page chosen at time k via the probability density in (5) in an i.i.d. fashion. Then, the two variables are updated as

$$x(k+1) = x(k) + \frac{h_{\theta(k)}^T r(k)}{\|h_{\theta(k)}\|^2} e_{\theta(k)}, \tag{14}$$

$$r(k+1) = r(k) - \frac{h_{\theta(k)}^T r(k)}{\|h_{\theta(k)}\|^2} h_{\theta(k)}, \tag{15}$$

where e_j is the jth column of the identity matrix I_n.

This scheme has the property that $Hx(k) + r(k)$ remains constant. This can be easily verified by multiplying H from the left of (14) and then adding it with (15), which yields

$$Hx(k+1) + r(k+1) = Hx(k) + r(k), \quad k \geq 0.$$

In particular, because the initial values have been chosen as $x_i(0) = 0$ and $r_i(0) = m/n$, this implies that

$$Hx(k) + r(k) = r(0) = g, \quad k \geq 0. \tag{16}$$

However, in general, in this scheme, the vector $x(k)$ is not consistent, meaning that $x(k)$ is not a probability vector. We can check this by multiplying $\mathbf{1}_n^T$ from the left of (16) and obtain

$$\mathbf{1}_n^T (Hx(k) + r(k)) = \mathbf{1}_n^T \big(mx(k) + r(k)\big) = \mathbf{1}_n^T g = m.$$

Thus, we have $\mathbf{1}_n^T x(k) = 1 - \mathbf{1}_n^T r(k)/m$.

It can be shown that this algorithm has exponential convergence in the mean-square sense, that is, it holds that $\mathbb{E}\big[\|x(k) - x^*\|^2\big] \to 0$ as $k \to \infty$. In view of (16), the convergence property can be attained by showing that $\mathbb{E}\big[\|r(k)\|^2\big]$ goes to zero exponentially fast.

A notable difference of this algorithm from that of [57] is the use of the columns $\{h_i\}$ of the matrix H instead of the rows $\{\tilde{h}_i\}$. In the networked system under consideration, the nonzero entries of the ith column h_i correspond to the out-neighbors of page i. In the update scheme (14) and (15) for page $\theta(k)$, first $h_{\theta(k)}^T r(k)$ must be computed, which requires the values $r_j(k)$ of the out-neighbor j be sent to page $\theta(k)$. Then, in (14), only $x_{\theta(k)}(k)$ is updated whereas in (15), the values $r_j(k)$ are updated for all out-neighbors j of page $\theta(k)$. This means that page $\theta(k)$ sends its own state value $r_{\theta(k)}(k)$ to all of its out-neighbors. Furthermore, it is clear that the norm $\|h_{\theta(k)}\|^2$ appearing in both (14) and (15) can be computed locally at each page in an offline manner before the execution of the algorithm.

3.4 Exponential Convergence with Consistency

All schemes that we have seen so far with exponential convergence do not have the property of consistency, that is, $x(k)$ is not a probability vector. This aspect is pointed out and then improved in the scheme introduced by [32]. Lack of consistency may be problematic in practice since the update schemes will terminate the updates in their states after a finite number of steps. Even at that point, there is no guarantee that the vector $x(k)$ is a stochastic vector. We skip the details of the update scheme; though it involves only two variables per page, the description of the algorithm tends to be complicated.

4 An Alternative Approach to PageRank

In this section, we present a new formulation of PageRank by transforming its original definition [50]. Then, novel distributed algorithms are developed where this formulation becomes the key. The idea itself is simple, but its advantage in the context of distributed computation of PageRank will become clear.

4.1 Reformulation of the PageRank Problem

The formula of PageRank in (2) can be transformed as

$$\begin{aligned} x^* = (1-m)Ax^* + \frac{m}{n}\mathbf{1}_n \quad &\Longleftrightarrow \quad x^* = [I-(1-m)A]^{-1}\,\frac{m}{n}\mathbf{1}_n \\ &\Longleftrightarrow \quad x^* = \sum_{t=0}^{\infty}[(1-m)A]^t\,\frac{m}{n}\mathbf{1}_n. \qquad (17) \end{aligned}$$

In the last transformation, the Neumann series is applied. Notice that $(1-m)A$ is a Schur stable matrix because the link matrix A is stochastic and thus has the spectral radius equal to 1.

The formula in (17) suggests that the PageRank computation can be carried out iteratively in several ways. It is immediate to write down an equation for the state $x(k) \in \mathbb{R}^n$ given by

$$x(k) = \sum_{t=0}^{k} [(1-m)A]^t \, \frac{m}{n} \mathbf{1}_n. \tag{18}$$

The power method in (4) is a compact way to realize this using only $x(k)$ as the state. There, we can express the state $x(k)$ as the solution to the linear system. With a slight difference in the time index, it follows that

$$x(k) = [(1-m)A]^k \, x(0) + \sum_{t=0}^{k-1} [(1-m)A]^t \, \frac{m}{n} \mathbf{1}_n.$$

The contribution of the initial value $x(0)$ in the first term on the right-hand side attenuates asymptotically, but it is effective in maintaining consistency in the state and, thus, it always holds that $\mathbf{1}_n^T x(k) = 1$ for all k.

Another approach to the expression in (18) is to use a redundant iteration by having an additional state, denoted by $z(k) \in \mathbb{R}^n$. Set the initial states as $x(0) = z(0) = (m/n)\mathbf{1}_n$. Then, the update scheme of the two states is given as follows:

$$\begin{aligned} x(k+1) &= x(k) + (1-m)Az(k), \\ z(k+1) &= (1-m)Az(k). \end{aligned} \tag{19}$$

Through this alternative algorithm, we can obtain the PageRank vector x^*. We formally state this along with other properties of this algorithm as a proposition in the following. Similar properties will appear in our development of distributed algorithms.

Proposition 1 *In the update scheme in (19), the states $x(k)$ and $z(k)$ satisfy the following:*

(i) $z(k) \to 0$ *as* $k \to \infty$.
(ii) $x(k) \le x(k+1) \le x^*$ *for* k.
(iii) $x(k) \to x^*$ *as* $k \to \infty$.

Proof (i) As the link matrix A is stochastic, its spectral radius equals 1, and thus $(1-m)A$ is a Schur stable matrix. This implies that $z(k)$ converges to zero.

(ii) Note that $z(k) \ge 0$ because A is stochastic and $z(0) > 0$. Furthermore, we have $x(0) > 0$. Thus, it is clear that $x(k)$ is nondecreasing as a function of k. The fact that it is upper bounded by x^* follows from (iii).

(iii) From (19), we can write $x(k)$ as

$$\begin{aligned} x(k) &= \sum_{t=1}^{k} z(t) + x(0) = \sum_{t=1}^{k} [(1-m)A]^t \, z(0) + x(0) \\ &= \sum_{t=0}^{k} [(1-m)A]^t \, \frac{m}{n} \mathbf{1}_n. \end{aligned} \tag{20}$$

This and (17) indicate that the state $x(k)$ converges to x^*. ■

We have a few remarks on the alternative approach introduced above in comparison with the power method in (4). First, the computation uses the second state $z(k)$ in addition to $x(k)$. As seen in (20), this state $z(k)$ is integrated over time to compute $x(k)$ in (18). Second, the initial values of $x(k)$ and $z(k)$ are fixed to $(m/n)\mathbf{1}_n$, and there is no freedom in these choices. Hence, each time the computation takes place through the update scheme (19), the algorithm cannot, for example, make use of the PageRank values computed in the past as initial guesses. This point may be a limitation of this approach. Also, the initial states are not probability vectors as in the power method. In fact, $x(k)$ becomes a probability vector only asymptotically when converging to x^*. Third, notice that n/m is the minimum PageRank value, which will be assigned to pages having no incoming links. For such pages, the states will not change during the updates.

Though we do not discuss in this chapter, there is a generalized PageRank definition which uses a probability vector $v \in \mathbb{R}^n$ instead of $(1/n)\mathbf{1}_n$, that is, $x^* = (1-m)Ax^* + mv$ (e.g., [33]). In such a case, the proposed algorithm can be easily modified by replacing the initial states with $x(0) = z(0) = mv$.

We now turn our attention to distributed algorithms. From the perspective of such algorithms, one interpretation of (19) can be given as follows:

1. At time 0, all pages start with the value m/n.
2. At time k, each page attenuates its current value by $1-m$ and then sends it to its linked pages after equally dividing it. At that time, page i computes the weighted sum of the values received from the neighbors having links to the page.

We finally present a distributed algorithm based on (19) with synchronous communication.

Algorithm 1 (*Synchronous distributed algorithm*) For each page i, set the initial values as $x_i(0) = z_i(0) = m/n$. At each time k, page i transmits its value $z_i(k)$ to its neighbors along its outgoing hyperlinks and then makes updates for its two states $x_i(k)$ and $z_i(k)$ as

$$\begin{aligned} x_i(k+1) &= x_i(k) + \sum_{j:\, i \in \mathscr{L}_j} \frac{1-m}{n_j} z_j(k), \\ z_i(k+1) &= \sum_{j:\, i \in \mathscr{L}_j} \frac{1-m}{n_j} z_j(k). \end{aligned}$$

Through simulations in Sect. 5, we will demonstrate that this synchronized algorithm may not be particularly fast, especially in comparison with the power method.

Moreover, due to the additional state $z(k)$, the algorithm requires more memory and computation. The advantage of the proposed reformulation however becomes evident in the asynchronous versions of this distributed algorithm, which will be presented in the next subsection.

4.2 Gossip-Type Distributed Algorithms

In this subsection, we extend the distributed algorithm discussed above so that the pages may interact with each other at different time instants. The algorithms are based on randomized gossip communication among the pages similarly to those presented in Sect. 3.

In the asynchronous update schemes, at each time k, one page $\theta(k) \in \mathscr{V}$ is randomly chosen, which transmits its current state value to the linked pages. We present two algorithms which differ in their probability distributions for selecting the updating pages. One uses the uniform distribution and the other is more general. In both cases, the distributions remain fixed throughout the execution of the algorithms; thus, the updating pages are chosen in an i.i.d. manner.

4.2.1 Algorithm Based on the Uniform Distribution

First, we consider the case where the selection of the updating pages follows the uniform distribution. The proposed distributed algorithm for this case is outlined below.

Algorithm 2 (*Distributed randomized algorithm*) For page $i \in \mathscr{V}$, set the initial values as $x_i(0) = z_i(0) = m/n$. At time k, the following steps are executed:

1. Select one page $\theta(k)$ based on the uniform distribution as in (5).
2. Page $\theta(k)$ transmits its value $z_{\theta(k)}(k)$ over its outgoing links.
3. Each page i updates its values $x_i(k)$ and $z_i(k)$ as

$$x_i(k+1) = \begin{cases} x_i(k) + \frac{1-m}{n_{\theta(k)}} z_{\theta(k)}(k) & \text{if } i \in \mathscr{L}_{\theta(k)}, \\ x_i(k) & \text{otherwise,} \end{cases}$$
$$z_i(k+1) = \begin{cases} 0 & \text{if } i = \theta(k), \\ z_i(k) + \frac{1-m}{n_{\theta(k)}} z_{\theta(k)}(k) & \text{if } i \in \mathscr{L}_{\theta(k)}, \\ z_i(k) & \text{otherwise.} \end{cases} \quad (21)$$

This distributed algorithm has a simple structure, which can be seen to be efficient from both computational and communication viewpoints. Each page keeps track of its states $x_i(k)$ and $z_i(k)$ and when it is randomly chosen as $\theta(k) = i$, it transmits one of its states, namely $z_i(k)$, to its neighboring pages along its outgoing hyperlinks.

Such hyperlinks are clearly known to the pages, and the necessary communication is limited with only one value at a time, without any data sent back from the linked pages. Other pages not linked by page $\theta(k)$ will simply keep their states unchanged. Since the time index k is irrelevant and not involved in the computation, there is no synchronization required in time among the pages.

The resemblance of this algorithm to Algorithm 1 is obvious. The two states $x_i(k)$ and $z_i(k)$ play similar roles in both algorithms. The differences are that in the asynchronous case, the updates are made with one neighbor at a time, and also both $x_i(k)$ and $z_i(k)$ are integrated over time. For $z_i(k)$, this was not the case in Algorithm 1. The two variables are updated differently when page i is the selected page $\theta(k)$ at time k: In such cases, its own $z_i(k)$ is set to zero. By contrast, in Algorithm 1, $z_i(k)$ is zero only in the case where page i has no incoming link.

We now rewrite this algorithm in the vector form. First, let $Q := (1-m)A$. Denote the ith columns of the $(n \times n)$-identity matrix I_n and Q, respectively, by e_i and q_i. Then, we define the matrices $Q_i, R_i \in \mathbb{R}^{n\times n}$ by

$$\begin{aligned} Q_i &:= \begin{bmatrix} e_1 \ e_2 \ \cdots \ e_{i-1} \ q_i \ e_{i+1} \ \cdots \ e_n \end{bmatrix}, \\ R_i &:= \begin{bmatrix} 0_n \ 0_n \ \cdots \ 0_n \ q_i \ 0_n \ \cdots \ 0_n \end{bmatrix}, \end{aligned}$$

where in both matrices, it is the ith column that is equal to q_i. Note that the matrices Q, Q_i, and R_i are all nonnegative matrices for $i \in \mathscr{V}$.

Let the initial states be $x(0) = z(0) = (m/n)\mathbf{1}_n$. The update schemes in (21) for the two states can be written in a compact form as

$$\begin{aligned} x(k+1) &= x(k) + R_{\theta(k)} z(k), \\ z(k+1) &= Q_{\theta(k)} z(k). \end{aligned} \tag{22}$$

We are now ready to present the main result for this distributed algorithm for PageRank computation. It shows that the true PageRank values can be obtained almost surely.

Theorem 1 *Under Algorithm 2, the PageRank vector x^* is computed with $x(k) \to x^*$ as $k \to \infty$ with probability one. In particular, the following two properties hold:*

(i) $x(k) \le x(k+1) \le x^*$ *holds for* $k \ge 0$.
(ii) $\mathbb{E}[x(k)] \to x^*$ *as* $k \to \infty$*, and the convergence speed is exponential.*

This theorem guarantees that the proposed gossip-based algorithm computes the true PageRank almost surely in a fully distributed fashion. In particular, similar to the synchronous case, the state vector $x(k)$ is a nondecreasing function of time k elementwise. Furthermore, its convergence to the PageRank vector is shown to be exponential in the mean, that is, the mean $\mathbb{E}[x(k)]$ approaches x^* exponentially fast. These two properties indicate that despite the use of randomization in the updates, there will not be any oscillation in the trajectories of the states. We will see that this is a unique feature among the other distributed algorithms.

In comparison with the algorithms presented in Sect. 3, our method is based on a simple reinterpretation of the definition of PageRank from the systems viewpoint, and it seems well suited for the PageRank computation in terms of convergence. We also note that similarly to [14], our algorithm does not require the pages to know the incoming links. Different from [14], communication in our scheme is directed in the sense that page i must transmit its value $z_i(k)$ to its outgoing neighbors, but need not receive their values. We will make further comparisons among the different schemes later in Sect. 4.2.3.

4.2.2 Generalization to Nonuniform Distributions

We next generalize the gossip-type distributed algorithm to the case where the pages will be chosen from distributions not limited to the uniform one. This extension is an interesting feature of the proposed approach and makes the algorithm more suitable for its use in a distributed environment. For example, depending on the computational and communication resources, the pages or the servers that carry out the PageRank computation may like to update at different frequencies [12]. Even in such situations, this algorithm is capable of computing the correct values with probability one.

Consider an i.i.d. random sequence $\{\theta(k)\}$ for the page selections. Let p_i be the probability of page i to be chosen at each time k. Assume that all p_i are strictly positive and $\sum_{i=1}^{n} p_i = 1$. The distributed algorithm for this nonuniform case is outlined below.

Algorithm 3 (*Generalized distributed randomized algorithm*) For page $i \in \mathcal{V}$, set the initial values as $x_i(0) = z_i(0) = m/n$. At time k, execute the following steps:

1. Select one page $\theta(k)$ based on the distribution p_i:

$$\text{Prob}\big\{\theta(k) = i\big\} = p_i \ \ \text{for } i \in \mathcal{V}. \tag{23}$$

2. Page $\theta(k)$ transmits its value $z_i(k)$ to pages over its outgoing links.
3. Each page i updates its values $x_i(k)$ and $z_i(k)$ as in (21) of Algorithm 2.

For this algorithm, we now state the main result.

Theorem 2 *Under Algorithm 3, the PageRank vector x^* is computed with $x(k) \to x^*$ as $k \to \infty$ with probability one. In particular, the following two properties hold:*

(i) $x(k) \leq x(k+1) \leq x^*$ *holds for* $k \geq 0$.
(ii) $\mathbb{E}\,[x(k)] \to x^*$ *as* $k \to \infty$*, and the convergence speed is exponential.*

This theorem can be established similarly to Theorem 1.

This gossip-type distributed algorithm can be carried out even if the probability distribution for the page selection is not uniform. Though other algorithms may be able to deal with nonuniform selection [12, 28, 36], in those cases, additional computations and adjustments are often required. In contrast, in our algorithm, no change is necessary and the update scheme performed by each page remains exactly the same. We have seen that the state values increase monotonically to reach the true PageRank. This might indicate that increasing the selection probabilities of pages with large values may lead to faster convergence. We will examine this idea in the context of a numerical example later.

Another idea for assigning the probabilities is to make them time varying. In particular, for pages having no hyperlink pointing to them, it is enough if they transmit their values to the neighbors once in the entire run of the algorithm. This can greatly reduce the amount of the overall communication required in the algorithm. As we have seen above, such pages are already given their PageRank values, equal to m/n, as their initial states. By examining the update scheme in (21), it is clear that once such a page i transmits the state $z_i(k)$ for the first time to its linked pages, this state $z_i(k)$ is set to zero and then will remain so for the rest of the time since it will not receive any data from others. The other state $x_i(k)$ will stay unchanged at its true PageRank value m/n.

4.2.3 Comparison of Different Methods

So far, we have introduced five different methods for the computation of PageRank by randomized distributed algorithms. We have seen that they have different features in terms of convergence speed, necessary computation and communication resources, and so on. In Table 1, we summarize the various aspects of these algorithms. The five algorithms are listed in the chronological order that they appeared in the literature.

The aspects that are shown here are the following:

(i) Data received from: Each time the page $\theta(k)$ is chosen at time k for initiating an update, it may use for updating its own state the data received from other pages. These pages are linked either by the incoming hyperlinks (in-neighbors) or the outgoing ones (out-neighbors).

(ii) Data sent to: The updating page $\theta(k)$ sends its own state, which will be used for the updates by the pages that receive it. Again, such pages may be the in-neighbors or out-neighbors, depending on the algorithms.

(iii) Time synchronization: In the distributed update schemes, the pages may require time synchronization among them. This is in fact needed only in the scheme of [26] for accurately computing the time average of the states.

(iv) Consistency: The state vector $x(k)$ is said to be consistent if it is a probability vector, i.e., $\sum_{i=1}^{n} x_i(k) = 1$ at all times k. As discussed in [57], this property may not be critical and may not be possible to achieve especially if the total number of pages in the network is unknown.

(v) Convergence speed: The method of [26] is not exponential in its convergence speed. This is because it uses the time average and thus becomes linear. Other algorithms all have exponential rates for their convergence.

(vi) Simulation result: We will see in the next section that the five algorithms exhibit different performances in numerical simulations. This point will be further discussed there.

Table 1 Comparison of randomized distributed algorithms

Method	Data received from	Data sent to	Time synch.	Consistent	Conv. speed	Simulation result
Ishii and Tempo [26]	None	Out-neighbors	Yes	Yes	Linear	Slow
You et al. [57]	In-neighbors	In-neighbors	No	No	Exponential	Medium
Dai and Freris [14]	Out-neighbors	Out-neighbors	No	No	Exponential	Fast
Lagoa et al. [32]	In-neighbors	In-neighbors	No	Yes	Exponential	Medium
Algorithm 2	None	Out-neighbors	No	No	Exponential	Very fast

5 Numerical Examples

To illustrate the performance of the distributed algorithms discussed so far in Sects. 3 and 4, we present results obtained through numerical simulations. We apply the different update schemes to two types of graphs and compare their properties including convergence speeds.

5.1 *Small Graph*

The first case that we consider is the simple graph with seven pages shown in Fig. 1 from Example 1.

5.1.1 Synchronized Algorithms

As an initial step, we examine the performance of the following two synchronous algorithms: The power method in (4) and Algorithm 1 from Sect. 4. These algorithms may be more suited for centralized implementation, but if proper synchronization can be introduced, distributed implementation should be possible as discussed in Sect. 4.

Their differences can be summarized as follows: (i) They have been derived from different viewpoints. The power method follows the original definition of (3) while Algorithm 1 is based on the interpretation expressed as the Neumann series (18) and has not been studied elsewhere. (ii) The numbers of variables per page are one for the power method and two for Algorithm 1. (iii) For the initial states, the power method can take any initial value as long as it is a probabilistic vector; in this simulation, we used uniform values, i.e., $(1/n)\mathbf{1}_n$. In the meantime, Algorithm 1 requires $x(0)$ to be fixed as $(m/n)\mathbf{1}_n$, which is also uniform, but not a probabilistic vector. On the other hand, these two algorithms share the property of being deterministic. Thus, the responses of pages 6 and 7 become exactly the same since both of them have no incoming link due to the structure of the graph.

The time response of the PageRank value for each page is shown in Fig. 2 for the two algorithms. We observe that the power method converges faster and, for most of the nodes, it takes less than 10 time steps. In the responses of the proposed algorithm, the convergence is slower and takes about 30 time steps. It is noticeable that they are nondecreasing with respect to time, a property shown in Proposition 1(i). Also, recall that for pages 6 and 7, in the proposed algorithm, the PageRank values of these pages are equal to the assigned initial values m/n. Hence, for these pages, the proposed algorithm requires no update.

5.1.2 Distributed Algorithms via Gossiping

Next, we discuss the simulation results for the gossip-based distributed algorithms using the simple network.

We make comparisons of the convergence performance of the five algorithms shown in Table 1. All five algorithms select one page at each time k based on the uniform distribution as shown in (5), and we applied the same sequence $\{\theta(k)\}$ to them for each run. As discussed earlier, in the two algorithms of [14, 57], the total number n of pages in the web may be unknown; here, we assume that n is known by all pages. Concerning the initial states, only our proposed algorithm requires that the pages take fixed values, equal to m/n. Other algorithms have some freedom in the choices. Here, however, we set them so that all pages are given the same initial values: For the algorithm of [14], it was set to 0, and in the remaining two algorithms, we took $1/n$.

The time responses of the calculated PageRank values of the pages are plotted in Fig. 3. We omit the result for page 7 as its behavior is similar to that of page 6. It is observed that the responses for most algorithms are oscillatory or noisy due to the randomization in the gossiping for updates and communication. On the other hand, there are certain levels of differences in the speeds of convergence among the algorithms. The responses of [26] appear to be the slowest and the most oscillatory with high peaks, possibly reflecting the fact of being the only non-exponential algorithm among the five.

In view of this, the proposed algorithm, namely Algorithm 2, is characteristic in that despite the randomization, the profile of the responses is smooth and again nondecreasing as in Fig. 2. This behavior is most visible in the plot for page 5. It is also clear that the proposed algorithm is the fastest in terms of convergence time for all pages in comparison with other algorithms. This is more evident in Fig. 4 where the total errors in the states from the true PageRank are displayed for all five algorithms in the logarithmic scale.

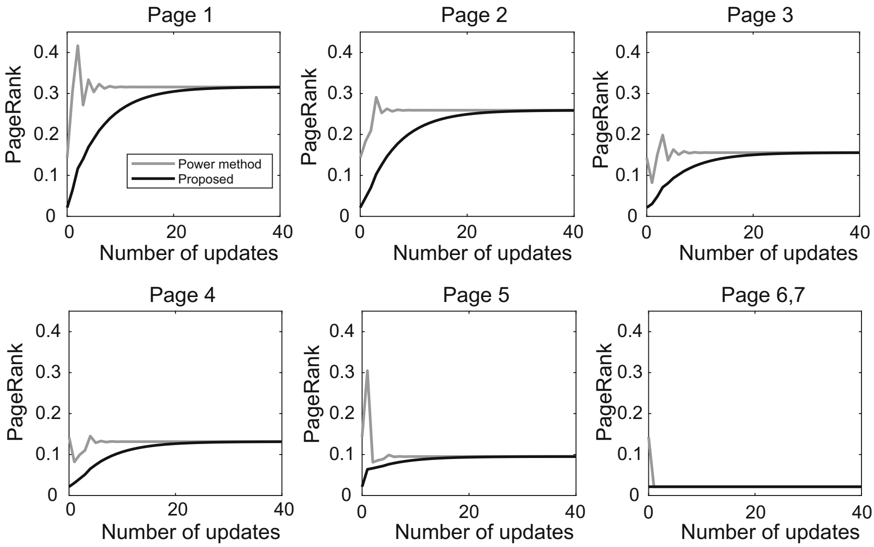


Fig. 2 Time responses of the synchronous algorithms for the small graph: The power method and the proposed Algorithm 1

5.1.3 Comparison of Distributions in Page Selection

In this part of the simulation, we illustrate how the convergence speed can be improved by employing Algorithm 3 with a nonuniform distribution for the random selection of $\theta(k)$. As discussed in Sect. 4.2.2, to improve convergence of the algorithm, it seems reasonable to increase the selection probability of pages which are expected to take larger PageRank values. We adjusted the probabilities so that pages having more incoming links are more likely to be selected, and each page's probability of selection is larger than 0. In particular, we assigned each page the probability proportional to its in-degree plus 1.

In Fig. 5, the time responses of the pages are shown for two algorithms, Algorithm 2 using the original uniform distribution in (5) and Algorithm 3 using this nonuniform distribution. As in Fig. 3, the responses of page 7 are omitted. We confirm that the nonuniform distribution is capable to further accelerate the convergence by a certain margin. It remains to be investigated what kind of distribution can in general be beneficial in improving the convergence rate.

5.2 *Large Graph*

We proceed to apply the five distributed algorithms to a larger web data. Specifically, we randomly generated a graph with 60 nodes. Figure 6 displays the network

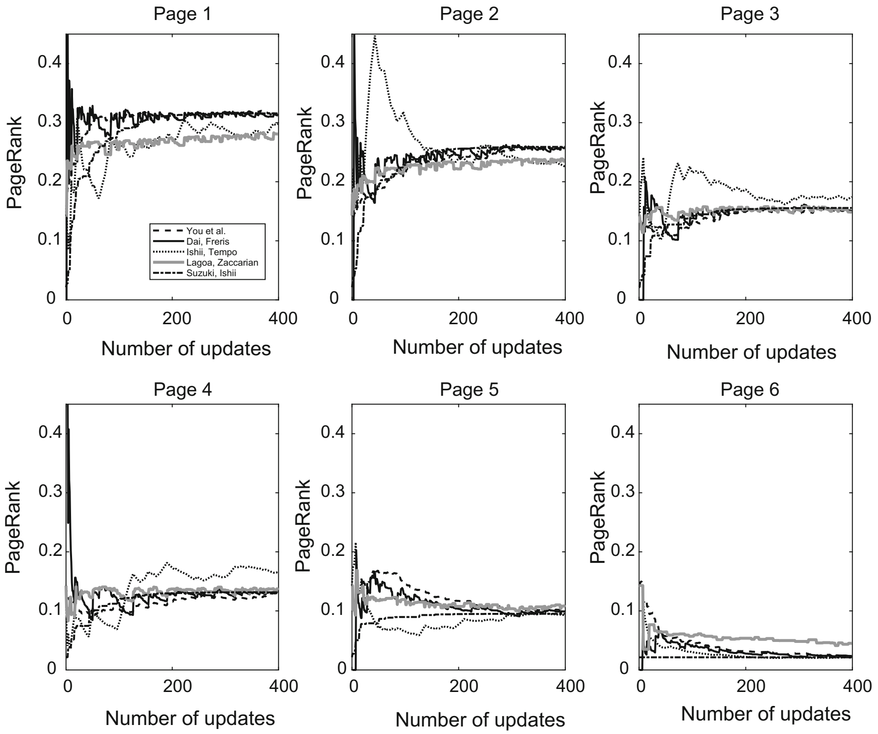


Fig. 3 Time responses of the asynchronous algorithms for the small graph: Ishii and Tempo [26], You et al. [57], Dai and Freris [14], Lagoa et al. [32], and the proposed Algorithm 2

structure where the dots indicate the nonzero entries of the hyperlink matrix A. The first eight pages are designed to be popular and receive hyperlinks from roughly one-third of the remaining nodes. In addition, each node has up to two hyperlinks to randomly selected nodes. In total, there are 223 hyperlinks and no dangling node in the network.

We applied the five randomized distributed algorithms with similar initial conditions. The responses of the sum of the errors are shown in Fig. 7. Here, we confirm that the performance of the proposed algorithm is the fastest and the error reduces exponentially. While the response of [26] is the slowest, the three methods of [14, 32, 57] exhibit exponential convergence. It is noted that we made simulations with other graphs of various sizes and observed similar results in general.

6 Discussion on Randomization in Multi-agent Systems

In this section, we would like to discuss, from a more general perspective, the roles that randomization plays in distributed algorithms and control for multi-agent sys-

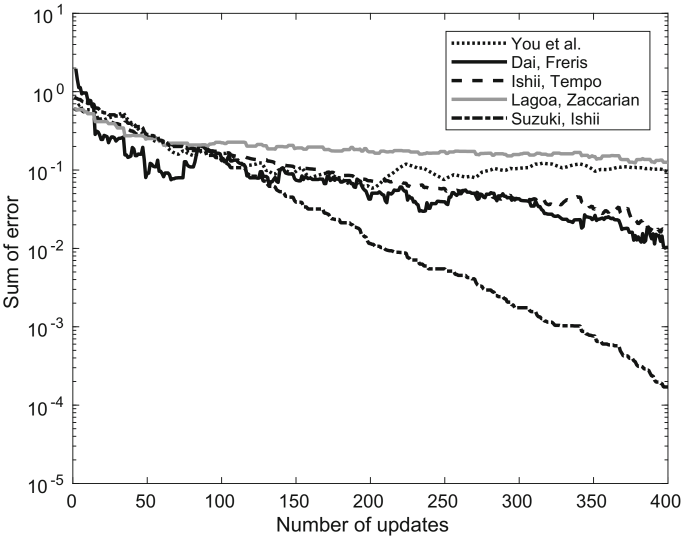


Fig. 4 Time responses of the errors in the asynchronous algorithms for the small graph: Ishii and Tempo [26], You et al. [57], Dai and Freris [14], Lagoa et al. [32], and the proposed Algorithm 2

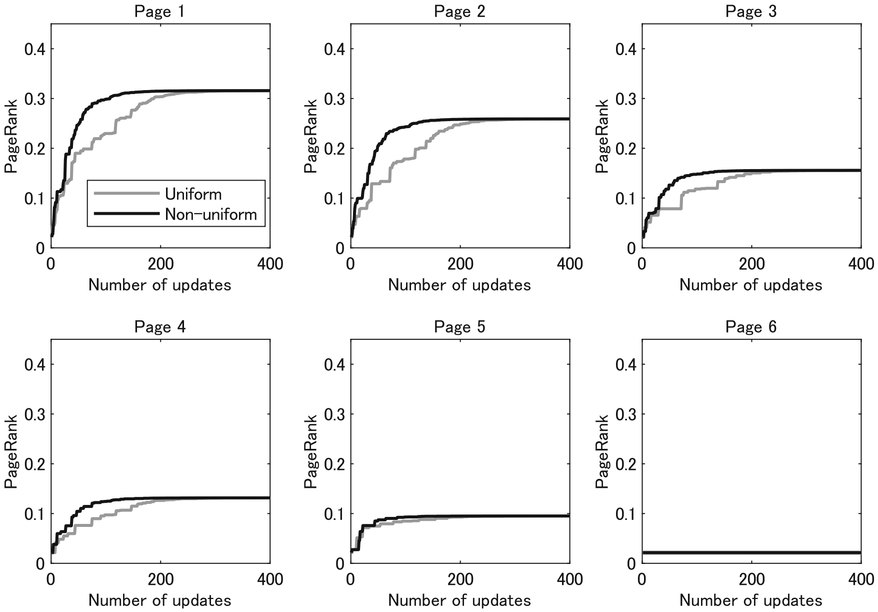


Fig. 5 Time responses in the asynchronous algorithms for the small graph: Algorithm 2 (uniform distribution) and Algorithm 3 (nonuniform distribution)

tems. This is in fact a broad subject as randomized techniques can now be found to be employed in many ways and we do not intend to be exhaustive. Our discussion hence will be limited to the recent research that we conducted and the works that are related.

Randomization techniques have received a significant level of attention within the community of systems control in the last two decades or so. In the early times, the motivation for employing such techniques originated from the need to address the issue of computational complexity arising in the context of uncertain and hybrid systems. For such systems, many control analysis and design problems are known to be computationally difficult to solve and can even be NP-hard (e.g., [4]). Application of probabilistic techniques to such problems has been found to be useful in developing computationally efficient algorithms. Recent developments can be found in the monograph [52]; see also the survey paper [53].

In large-scale network systems, randomized algorithms have been widely employed, but the distributed nature of such systems calls for the exploitation of randomization with a purpose different from that of relaxing computational complexity as discussed above. Here, we would like to highlight three essential roles in multi-agent systems that randomized techniques can play. Those are related to (i) communication, (ii) decision-makings through dithering, and (iii) cybersecurity. In the following, we briefly describe recent progress along these directions.

(i) In multi-agent systems, communication among the agents must be initiated by the individual agents since there is often no centralized entity that would command them to synchronize. Thus, as we have seen in this chapter, communicating at

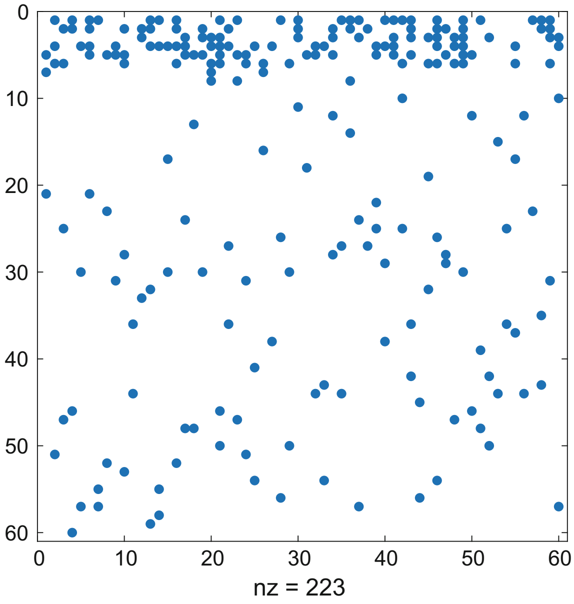
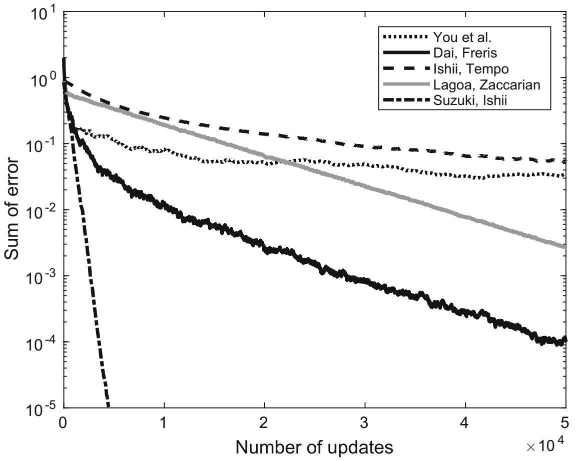


Fig. 6 Network structure of the large graph with 60 web pages

Fig. 7 Time responses of the errors in the asynchronous algorithms for the large graph: Ishii and Tempo [26], You et al. [57], Dai and Freris [14], Lagoa et al. [32], and the proposed Algorithm 2



randomly chosen time instants can be a useful option. It is also a realistic model in the case of wireless communication; if collisions occur due to simultaneous transmissions, retransmissions will be made after some waiting times, whose lengths are randomly chosen. Communication at random times is sometimes referred to as gossiping [6] and has been exploited in a number of works in multi-agent systems including [9–11, 15, 30, 34, 46, 51, 56].

(ii) In distributed algorithms, randomized algorithms can help the process of decision-makings by introducing a certain level of noise or perturbation in the system. In signal processing, dithering is a well-known probabilistic method in quantization [55]. It introduces random noise before the operation of quantizing a real-valued signal. In audio signals, for instance, dithering is commonly used for reducing unnatural sounds in quantized signals that can result from certain periodicity introduced through the analog-to-digital conversion.

This method has been found useful in the context of multi-agent systems as well. In particular, in the so-called quantized consensus problems, agents take integer values in their states. There, some update schemes employ randomized quantization so that, for example, the state may be rounded up or down randomly. Such a method has the effect of introducing perturbation in the consensus process so as to avoid the states being stuck before reaching consensus. For related studies, see, e.g., [2, 9, 10, 15, 21, 30].

(iii) In potentially hazardous environments where malicious attackers may exploit the vulnerabilities in systems and communication networks (e.g., [18, 44, 47]), randomization can be a viable method in raising the security level. For example, intruders interested in the data exchanged among agents may need more resources to attack or to eavesdrop on the communications when the times are chosen randomly. Such a stochastic scheme is proposed and analyzed in a multi-agent consensus problem in [31]; the agents' communication is disrupted by jamming attacks, but the energy for emitting jamming signals is constrained as in [49]. Making the transmission times unpredictable becomes the key to realize consensus even under a less stringent condition for the attackers.

On the other hand, in the literature of distributed algorithms in computer science, multi-agent consensus has been long studied. An important class of problems there includes fault-tolerant consensus for multi-agent systems in the presence of faulty agents or even those which are driven by malicious attackers. Such agents may not follow the a priori given update rules. The non-faulty, regular agents are equipped with a resilient version of the consensus algorithm, which determines the neighbors taking suspicious values and thus to be ignored in the updates. Such problems have been studied in, e.g., [3, 5, 54] in computer science, and more recently in, e.g., [16, 17, 35] in the systems control literature.

In computer science, probabilistic algorithms have been known to improve resilience. In distributed decision-making problems, various "impossibility results" have been derived, showing that deterministic approaches are insufficient for achieving the desired goal with certain scalability properties [20, 37, 45]. We would like to mention that recently, in [15], it was established that in asynchronous update schemes for resilient consensus, probabilistic gossip-based communication among agents can be superior to deterministic approaches in terms of the necessary network structures; this may be seen as a form of an impossibility result.

More generally, probabilistic techniques have been extensively studied in the area of algorithms in computer science; see, e.g., the monographs [41, 42] and the references therein. As discussed in [53], randomized algorithms can be classified into two categories: The Monte Carlo type and the Las Vegas type. Roughly speaking, algorithms of the Monte Carlo type may produce incorrect outputs with limited probabilities whereas the Las Vegas types are guaranteed to provide correct solutions with probability one. Many of the algorithms in the studies of uncertain and hybrid control systems belong to the Monte Carlo type. They often rely on random sampling in the uncertain sets, which are continuous sets. On the other hand, those discussed in this chapter are of the Las Vegas type. Indeed, in Theorems 1 and 2 of Sect. 4, we have seen that the convergence of the randomized algorithms is guaranteed almost surely.

7 Conclusion

In this chapter, we have introduced the problem of PageRank computation from the perspective of systems and control and then provided a short overview on the recent developments on distributed algorithms. We have also proposed a new class of distributed algorithms for the computation of PageRank using a new interpretation of its definition. Specifically, two types of distributed algorithms have been obtained: One is synchronous in that all agents update their state values at the same time, while in the other, randomization is used for determining the page that initiates an update at each time step. Regarding their convergence properties, it has been established that they are exponential. The relation of the proposed algorithms to those in the literature has been discussed as well. One characteristics of our approach making it suitable for distributed implementation is that it does not need to follow the uniform distribution. We have shown through simulations that our algorithms exhibit superior performance in both a simple web and a large-scale web. Finally, a general discussion from a broader perspective on the advantages that randomization may bring to distributed algorithms has been given.

In future research, we will further analyze the convergence speeds of the proposed algorithms and employ other schemes for page selections. We are also interested in studying other problems where our approach can be useful in developing distributed algorithms.

Acknowledgements The authors are sincerely grateful to Roberto Tempo for our long-term collaboration, which led us to the research on randomized algorithms for large-scale network systems presented in this chapter. He will be remembered as a wonderful collaborator, mentor, and friend.

References

1. K. Avrachenkov, N. Litvak, D. Nemirovsky, and N. Osipova. Monte Carlo methods in PageRank computation: When one iteration is sufficient. *SIAM J. Numer. Anal.* 45:890–904, 2007.
2. T. C. Aysal, M. J. Coates, and M. G. Rabbat. Distributed average consensus with dithered quantization. *IEEE Trans. Signal Processing*, 56:4905–4918, 2008.
3. M. H. Azadmanesh and R. M. Kiechafer. Asynchronous approximate agreement in partially connected networks. *Int. J. Parallel Distrib. Networks*, 5:26–34, 2002.
4. V. D. Blondel and J. N. Tsitsiklis. A survey of computational complexity results in systems and control. *Automatica*, 36:1249–1274, 2000.
5. Z. Bouzid, M. G. Potop-Butucaru, and S. Tixeuil. Optimal Byzantine-resilient convergence in uni-dimensional robot networks. *Theoretical Computer Science*, 411:3154–3168, 2010.
6. S. Boyd, A. Ghosh, B. Prabhakar, and D. Shah. Randomized gossip algorithms. *IEEE Trans. Inform. Theory*, 52:2508–2530, 2006.
7. S. Brin and L. Page. The anatomy of a large-scale hypertextual Web search engine. *Computer Networks & ISDN Systems*, 30:107–117, 1998.
8. F. Bullo. *Lectures on Network Systems*. CreateSpace, 2018.
9. K. Cai and H. Ishii. Quantized consensus and averaging on gossip digraphs. *IEEE Trans. Autom. Contr.* 56:2087–2100, 2011.
10. K. Cai and H. Ishii. Convergence time analysis of quantized gossip consensus on digraphs. *Automatica*, 48:2344–2351, 2012.
11. R. Carli, F. Fagnani, P. Frasca, and S. Zampieri. Gossip consensus algorithms via quantized communication. *Automatica*, 46:70–80, 2010.
12. T. Charalambous, C. Hadjicostis, M. Rabbat, and M. Johansson. Totally asynchronous distributed estimation of eigenvector centrality in digraphs with application to the PageRank problem. In *Proc.* 55*th IEEE Conf. on Decision and Control*, pages 25–30, 2016.
13. B. C. Csáji, R. M. Jungers, and V. D. Blondel. PageRank optimization by edge selection. *Discrete Applied Mathematics*, 169:73–87, 2014.
14. L. Dai and N. Freris. Fully distributed PageRank computation with exponential convergence. *arXiv:1705.09927*, 2017.
15. S. M. Dibaji, H. Ishii, and R. Tempo. Resilient randomized quantized consensus. *IEEE Trans. Autom. Contr.* 63:2508–2522, 2018.
16. S.M. Dibaji and H. Ishii. Resilient multi-agent consensus with asynchrony and delayed information. In *Proc.* 5*th IFAC Workshop on Distributed Estimation and Control in Networked Systems*, pages 28–33, 2015.

17. S.M. Dibaji and H. Ishii. Resilient consensus of second-order agent networks: Asynchronous update rules with delays. *Automatica*, 81:123–132, 2017.
18. H. Fawzi, P. Tabuada, and S. Diggavi. Secure estimation and control for cyber-physical systems under adversarial attacks. *IEEE Trans. Autom. Contr.* 59:1454–1467, 2014.
19. O. Fercoq, M. Akian, M. Bouhtou, and S. Gaubert. Ergodic control and polyhedral approaches to PageRank optimization. *IEEE Trans. Autom. Contr.* 58:134–148, 2013.
20. M. J. Fischer, N. A. Lynch, and M. S. Paterson. Impossibility of distributed consensus with one faulty process. *J. ACM*, 32:374–382, 1985.
21. P. Frasca, R. Carli, F. Fagnani, and S. Zampieri. Average consensus on networks with quantized communication. *Int. J. Robust & Nonlinear Control*, 19:1787–1816, 2009.
22. P. Frasca, H. Ishii, C. Ravazzi, and R. Tempo. Distributed randomized algorithms for opinion formation, centrality computation and power systems estimation: A tutorial overview. *European J. Control*, 24:2–13, 2009.
23. D. F. Gleich. PageRank beyond the Web. *SIAM Review*, 57(3):321–363, 2015.
24. R. A. Horn and C. R. Johnson. *Matrix Analysis*. Cambridge Univ. Press, 1985.
25. H. Ishii and R. Tempo. Computing the PageRank variation for fragile web data. *SICE J. Control, Measurement, and System Integration*, 2:1–9, 2009.
26. H. Ishii and R. Tempo. Distributed randomized algorithms for the PageRank computation. *IEEE Trans. Autom. Contr.* 55:1987–2002, 2010.
27. H. Ishii and R. Tempo. The PageRank problem, multi-agent consensus and web aggregation: A systems and control viewpoint. *IEEE Control Systems Magazine*, 34:34–53, 2014.
28. H. Ishii, R. Tempo, and E.-W. Bai. PageRank computation via a distributed randomized approach with lossy communication. *Syst. & Cont. Lett.* 61:1221–1228, 2012.
29. H. Ishii, R. Tempo, and E.-W. Bai. A web aggregation approach for distributed randomized PageRank algorithms. *IEEE Trans. Autom. Contr.* 57:2703–2717, 2012.
30. A. Kashyap, T. Başar, and R. Srikant. Quantized consensus. *Automatica*, 43:1192–1203, 2007.
31. K. Kikuchi, A. Cetinkaya, T. Hayakawa, and H. Ishii. Stochastic communication protocols for multi-agent consensus under jamming attacks. In *Proc. 56th IEEE Conf. on Decision and Control*, pages 1657–1662, 2017.
32. C. M. Lagoa, L. Zaccarian, and F. Dabbene. A distributed algorithm with consistency for PageRank-like linear algebraic systems. In *Proc. 20th IFAC World Congress*, pages 5339–5344, 2017.
33. A.N. Langville and C.D. Meyer. *Google's PageRank and Beyond: The Science of Search Engine Rankings*. Princeton University Press, 2006.
34. J. Lavaei and R. M. Murray. Quantized consensus by means of gossip algorithm. *IEEE Trans. Autom. Contr.* 57:19–32, 2012.
35. H. J. LeBlanc, H. Zhang, X. Koutsoukos, and S. Sundaram. Resilient asymptotic consensus in robust networks. *IEEE J. Selected Areas Comm.* 31:766–781, 2013.
36. J. Lei and H.-F. Chen. Distributed randomized PageRank algorithm based on stochastic approximation. *IEEE Trans. Autom. Contr.* 60:1641–1646, 2015.
37. N. A. Lynch. *Distributed Algorithms*. Morgan Kaufmann, San Francisco, CA, 1997.
38. J. M. Maestre, H. Ishii, and E. Algaba. Node aggregation for enhancing PageRank. *IEEE Access*, 5:19799–19811, 2017.
39. S. G. Mallat and Z. Zhang. Matching pursuits with time-frequency dictionaries. *IEEE Trans. Signal Processing*, 41:3397–3415, 1993.
40. M. Mesbahi and M. Egerstedt. *Graph Theoretic Methods in Multiagent Networks*. Princeton University Press, 2010.
41. M. Mitzenmacher and E. Upfal. *Probability and Computing: Randomized Algorithms and Probabilistic Analysis*. Cambridge University Press, 2005.
42. R. Motwani and P. Raghavan. *Randomized Algorithms*. Cambridge University Press, 1995.
43. A.V. Nazin and B.T. Polyak. Randomized algorithm to determine the eigenvector of a stochastic matrix with application to the PageRank problem. *Automation and Remote Control*, 72:342–352, 2011.

44. F. Pasqualetti, F. Dörfler, and F. Bullo. Attack detection and identification in cyber-physical systems. *IEEE Trans. Autom. Contr.* 58:2715–2729, 2013.
45. M. Rabin. Randomized Byzantine generals. In *Proc. 24th IEEE Symp. Foundations of Comp. Sci.* pages 403–409, 1983.
46. C. Ravazzi, P. Frasca, R. Tempo, and H. Ishii. Ergodic randomized algorithms and dynamics over networks. *IEEE Trans. Control of Network Syst.* 2:78–87, 2015.
47. H. Sandberg, S. Amin, and K.H. Johansson, guest editors. Special issue on cyberphysical security in networked control systems. *IEEE Control Systems Magazine*, 35(1), 2015.
48. A. Sarma, A. Molla, G. Pandurangan, and E. Upfal. Fast distributed PageRank computation. *Theoretical Computer Science*, 561:113–121, 2015.
49. D. Senejohnny, P. Tesi, and C. De Persis. A jamming-resilient algorithm for self-triggered network coordination. *IEEE Trans. Control of Network Syst.* 5:981–990, 2018.
50. A. Suzuki and H. Ishii. Distributed randomized algorithms for PageRank based on a novel interpretation. In *Proc. American Control Conf.* pp. 472–477, 2018.
51. A. Tahbaz-Salehi and A. Jadbabaie. A necessary and sufficient condition for consensus over random networks. *IEEE Trans. Autom. Contr.* 53:791–795, 2008.
52. R. Tempo, G. Calafiore, and F. Dabbene. *Randomized Algorithms for Analysis and Control of Uncertain Systems, with Applications,* Second Edition. Springer, London, 2013.
53. R. Tempo and H. Ishii. Monte Carlo and Las Vegas randomized algorithms for systems and control: An introduction. *European J. Control*, 13:189–203, 2007.
54. L. Tseng and N. H. Vaidya. Fault-tolerant consensus in directed graphs. In *Proc. ACM Symp. Principles of Distributed Comput.* pages 451–460, 2015.
55. R. A. Wannamaker, S. P. Lipshitz, J. Vanderkooy, and J. N. Wright. A theory of nonsubtractive dither. *IEEE Trans. Signal Processing*, 48:499–516, 2000.
56. K. You, Z. Li, and L. Xie. Consensus condition for linear multi-agent systems over randomly switching topologies. *Automatica*, 49:3125–3132, 2013.
57. K. You, R. Tempo, and L. Qiu. Distributed algorithms for computation of centrality measures in complex networks. *IEEE Trans. Autom. Contr.* 62:2080–2094, 2017.
58. W. Zhao, H.-F. Chen, and H. Fang. Convergence of distributed randomized PageRank algorithms. *IEEE Trans. Autom. Contr.* 50:1177–1181, 2013.
59. A. Zouzias and N. M. Freris. Randomized extended Kaczmarz for solving least squares. *SIAM J. Matrix Analysis and Applications*, 34:773–793, 2013.

Distributed Optimization in Multi-agent Networks Using One-bit of Relative State Information

Jiaqi Zhang and Keyou You

Abstract This chapter is concerned with the design of distributed discrete-time algorithms to cooperatively solve an additive cost optimization problem in multi-agent networks. The striking feature of our distributed algorithms lies in the use of only the sign of relative state information between neighbors, which substantially differentiates our algorithms from others in the existing literature. Moreover, the algorithm does not require the interaction matrix to be doubly-stochastic. We first interpret the proposed algorithms in terms of the penalty method in optimization theory and then perform non-asymptotic analysis to study convergence for static network graphs. Compared with the celebrated distributed subgradient algorithms, which however use the exact relative state information, the convergence speed is essentially not affected by the loss of information. We also extend our results to the cases of deterministically and randomly time-varying graphs. Finally, we validate the theoretical results by simulations.

1 Introduction

In recent years, distributed optimization problems in multi-agent systems have attracted increasing attention. Distributed optimization is concerned with that all agents to cooperatively minimize a sum of local objective functions over a graph. The key feature of such an optimization problem lies in that each agent only knows a local component of the objective function and thus must cooperate with its neighbors to compute the optimal value. The interaction between nodes is modeled by an algebraic graph. The motivating examples for distributed computation include the AUV

Parts of the results in this chapter have previously been appeared in [26–28].

J. Zhang · K. You (✉)
Department of Automation, and BNRist, Tsinghua University, Beijing 100084, China
e-mail: youky@tsinghua.edu.cn

J. Zhang
e-mail: zjq16@mails.tsinghua.edu.cn

© Springer Nature Switzerland AG 2018

T. Başar (ed.), *Uncertainty in Complex Networked Systems*,
Systems & Control: Foundations & Applications,
https://doi.org/10.1007/978-3-030-04630-9_13

formation control [24], large-scale machine learning [4, 17, 25], and the distributed quantile regression over sensor networks [21].

To solve the distributed optimization problem, the majority of the algorithms (see e.g., [11, 15, 16, 21] and the references therein) are generally comprised of two parts. One is to drive all agents to consensus, which is based on the well-known consensus algorithm [18]. The other one is to push the consensus value toward an optimal point by using the local (sub)gradient in each node. In this case, subgradient-based algorithms have been widely used. To achieve consensus of the multi-agent network, most of the existing methods require each agent to access the state values of its neighbors at each time, either exactly [15, 18] or in a quantized form [19, 23]. However, in some situations, an agent may only roughly know relative state measurements between its neighbors. For example, consider the case of several robots working in a plane, when each robot can only tell which quadrant its neighbor is lying by cheap sensors but not the neighbor's accurate relative position. Thus, the information accessible is restricted to be only one bit. Note that this is different from the quantized setting in [19], which studied the effects of exchanging a quantized rather than an exact state between neighbors. This is also different from previous studies on exchanging quantized gradients [13] since we are only using the quantized relative state information. Therefore, most algorithms in the literature, particularly the ones in the references cited above, cannot handle the case of one-bit information. It is worth noting that another advantage of our algorithm, in addition to using only one bit of relative information, is that it does not require the interaction matrix of the agents to be doubly-stochastic. A doubly-stochastic adjacency matrix is a common assumption in many existing algorithms [14, 16, 20], but it is restrictive in the distributed setting. For example, the Metropolis method [20] to construct a doubly-stochastic matrix requires each node to know its neighbors' degrees, which may be impractical in applications, especially when the graph is time-varying.

Designing an algorithm using one bit of information often involves nonlinear systems analysis, which is substantially different from the commonly applied graph Laplacian theory in the aforementioned works. There are, however, some exceptions [5, 9, 12]. In [5], the authors designed a consensus algorithm using only sign information of the relative state. A similar algorithm was also proposed in [9] to distributedly compute a sample median. The algorithm in [12] is the most relevant to the one in this chapter except that it is a continuous-time algorithm, which adopts a completely different analysis tool than ours. We will return to this point, and discuss more extensively later.

In fact, all the aforementioned works that use one bit of information focused on continuous-time algorithms. However, a discrete-time algorithm is worth studying because many distributed optimization applications involve communication between agents and control of agents, which are typically discrete in nature. Besides, a discrete-time algorithm is easier to implement. What is more, a continuous-time algorithm cannot be extended to the discrete-time case that easily, since the methods used to analyze continuous-time algorithms in the above works are often based on Lyapunov theory. We know that some general stepsize rules (e.g., constant, diminishing) in discrete-time gradient-based algorithms cannot guarantee the nonincreasingness of a latent Lyapunov function, and some special stepsize rules (e.g., line minimization rule) often fail to meet the requirement of distributed computation, which renders the Lyapunov analysis difficult to extend to the discrete-time case. Therefore, an alternative method is urgently needed, which is what this chapter does.

More precisely, we propose in this chapter a distributed optimization algorithm using only one bit of information in the discrete-time case. Different from most of the previous works, our analysis is based on optimization theory rather than algebraic graph theory or Lyapunov theory. There are two underlying advantages of this. First, compared to many existing approaches which first propose an algorithm, and then find a Lyapunov function to prove its convergence, the intuition behind our algorithm appears to be more natural and reasonable, as it aims to minimize a well-designed objective function. Second, a wealth of research in convex optimization theory ensures our algorithm more easily extensible to more general cases. For example, our algorithm over time-varying graphs is a direct extension of that over static graphs. Specifically, we extend our algorithm to both *deterministically* time-varying graphs and *randomly* time-varying graphs. The former can model the time-varying topology of agents in applications [17, 22], while the latter can be used to describe the gossip networks [10], random package losses in communication networks, etc. Based on optimization theory, our methods to analyze the cases of deterministically time-varying graphs and randomly time-varying graphs take advantage of incremental gradient methods and stochastic gradient descent methods, respectively.

For a connected static graph, each node of the distributed optimization algorithm is shown to converge asymptotically to the same optimal point of the optimization without any reduction in the convergence rate. For deterministically time-varying graphs, the convergence of the distributed optimization algorithm is established if the graphs are uniformly jointly connected. For randomly time-varying graphs, we show the convergence of the distributed optimization algorithm in the almost sure sense under the so-called randomly activated connected graph assumption.

The rest of the chapter is organized as follows. Section 2 provides some preliminaries and introduces the distributed optimization problem. In Sect. 3, we present our discrete-time distributed optimization algorithm using one bit of information. Section 4 includes our main results on convergence and convergence rate of the algorithm over static graphs. Section 5 provides the convergence results over uniformly jointly connected graphs and randomly activated graphs. Finally, we perform simulations to validate the theoretical results in Sect. 6, and draw some concluding remarks in Sect. 7.

Notation: We use a, $\mathbf{a}$, A, and $\mathscr{A}$ to denote a scalar, vector, matrix, and set, respectively. $\mathbf{a}^\mathsf{T}$ and A^T denote the transposes of $\mathbf{a}$ and A, respectively. $[A]_{ij}$ denotes the element in row i and column j of A. $\mathbb{R}$ denotes the set of real numbers and $\mathbb{R}^n$ denotes the set of all n-dimensional real vectors. $\mathbb{1}$ denotes the vector with all ones, the dimension of which depends on the context. We let $\|\cdot\|_1$, $\|\cdot\|$ and $\|\cdot\|_\infty$ denote the l_1-norm, l_2-norm and l_∞-norm of a vector or matrix, respectively. We define

$$\operatorname{sgn}(x) = \begin{cases} 1, & \text{if } x \geq 0, \\ -1, & \text{otherwise.} \end{cases}$$

With a slight abuse of notation, $\nabla f(x)$ denotes *any* subgradient of $f(x)$ at x, i.e., $\nabla f(x)$ satisfies

$$f(y) \geq f(x) + (y - x)^{\mathsf{T}} \nabla f(x), \ \forall y \in \mathbb{R}. \tag{1}$$

The subdifferential $\partial f(x)$ is the set of all subgradients of $f(x)$ at x. If $f(x)$ is differentiable at x, then $\partial f(x)$ includes only the gradient of $f(x)$ at x.

We call $\inf_{\mathbf{x} \in \mathbb{R}^n} f(\mathbf{x})$ the optimal value of $f(\mathbf{x})$. Any element from the set $\arg\inf_{\mathbf{x} \in \mathbb{R}^n} f(\mathbf{x})$ is called an optimal solution or optimal point of $f(\mathbf{x})$.

Superscripts are usually used to represent sequence indices, i.e., x^k represents the value of the sequence $\{x^k\}$ at time k.

2 Problem Formulation

This section introduces some basics of graph theory, and presents the distributed optimization problem in multi-agent networks.

2.1 Basics of Graph Theory

A graph (network) is represented as $\mathscr{G} = (\mathscr{V}, \mathscr{E})$, where $\mathscr{V} = \{1, ..., n\}$ is the set of nodes and $\mathscr{E} \subseteq \mathscr{V} \times \mathscr{V}$ is the set of edges. Let $\mathscr{N}_i = \{j \in \mathscr{V} | (i, j) \in \mathscr{E}\}$ be the set of neighbors of node i, and $A = [a_{ij}]$ be the weighted adjacency matrix of $\mathscr{G}$, where $a_{ij} > 0$ if and only if there exists an edge connecting nodes i and j, and otherwise, $a_{ij} = 0$. If $A = A^{\mathsf{T}}$, the associated graph is undirected. This chapter focuses only on undirected graphs.

In the case of time-varying graphs, we use $\mathscr{G}^k = (\mathscr{V}, \mathscr{E}^k, A^k)$ to represent the graph at time k. Let $\mathscr{G}^{k_1} \cup \mathscr{G}^{k_2}$ denote the graph $(\mathscr{V}, \mathscr{E}^{k_1} \cup \mathscr{E}^{k_2}, A^{k_1} + A^{k_2})$. Let $\mathscr{N}_i^k = \{j \in \mathscr{V} | (i, j) \in \mathscr{E}^k\}$ denote the set of neighbors of node i at time k. The incidence matrix $B \in \mathbb{R}^{n \times m}$ of $\mathscr{G}$ is defined by

$$B_{ie} = \begin{cases} 1, \text{ if node } i \text{ is the source node of edge } e, \\ -1, \text{ if node } i \text{ is the sink node of edge } e, \\ 0, \text{ otherwise.} \end{cases}$$

For any $\mathbf{x} = [x_1, ..., x_n]^{\mathsf{T}}$, we have that

$$\mathbf{b}_e^{\mathsf{T}} \mathbf{x} = x_i - x_j$$

where $\mathbf{b}_e, e \in \mathscr{E}$ is the e-th column of B, and i and j are the source and the sink nodes of edge e, respectively.

A path is a sequence of consecutive edges that connect a set of different nodes. We say a graph is *connected* if there exists a path between any pair of two nodes. To evaluate the intensity of the graph's connectivity, we introduce an important concept called l-connected graph below.

Definition 1 (l-*connected graph*) A connected graph is l-connected ($l \geq 1$) if it remains connected whenever fewer than l edges are removed.

Clearly, a connected graph is at least 1-connected and each node of an l-connected graph has at least l neighbors.

2.2 Distributed Optimization Problem

With only the sign of relative state, our objective is to distributedly solve the multi-agent optimization problem

$$\underset{x\in\mathbb{R}}{\text{minimize}} \;\; f(x) := \sum_{i=1}^{n} f_i(x) \tag{2}$$

where for each $i \in \mathscr{V}$, the local objective function $f_i(x)$ is continuously convex but not necessarily differentiable, and is only known by node i. The number of nodes is set to be $n > 1$. We first make a mild assumption.

Assumption 1 (Nonempty optimal set and bounded subgradients)

(a) The set $\mathscr{X}^\star$ of optimal solutions of problem (2) is nonempty, i.e., for any $x^\star \in \mathscr{X}^\star$, it holds that $f^\star := f(x^\star) = \inf_{x\in\mathbb{R}} f(x)$.
(b) There exists a constant $c > 0$ such that

$$|\nabla f_i(x)| \leq c,\;\; \forall i \in \mathscr{V}, x \in \mathbb{R}. \tag{3}$$

Assumption 1 is common in the literature, see e.g., [16, 25]. In particular, the second part is often made to guarantee the convergence of a subgradient method [16], and obviously holds if the decision variable x is restricted to a compact set.

3 The Distributed Optimization Algorithm Over Static Graphs

In this section, we provide the discrete-time distributed optimization algorithm that uses only the sign information of the relative state of the neighboring nodes (hence one-bit information), and then interpret it via the penalty method in optimization theory.

This section only focuses on static graphs, which are important to the analysis of time-varying cases in following sections.

3.1 The Distributed Optimization Algorithm

The discrete-time distributed algorithm to solve (2) over a static network $\mathscr{G}$ is given in Algorithm 1.

Algorithm 1: Distributed Algorithm Using the Sign of Relative State

```
**Initialization**: Every node $i$ sets $x_i^0 = 0$ for all $i \in \mathscr{V}$.
**Repeat**
**Information collection**: Each node $i$ collects the sign of the relative state to its neighbor
    $j \in \mathscr{N}_i$ and obtain $r_i^k$, which is given below
        $r_i^k = \sum_{j \in \mathscr{N}_i} a_{ij} \operatorname{sgn}(x_j^k - x_i^k)$.
**Local update**: The decision variable in each node is locally updated as
        $x_i^{k+1} = x_i^k + \rho^k \left(\lambda \cdot r_i^k - \nabla f_i(x_i^k)\right)$,
    where $\lambda$ and $\rho^k$ are to be given, and $\nabla f_i(x_i^k)$ is any subgradient of $f_i(x)$ at $x_i^k$.
**Set** $k = k + 1$.
**Until** a predefined stopping rule (e.g., a maximum iteration number) is satisfied.
```

The continuous-time version of Algorithm 1 is also given in (4) of [12] and is proved to be convergent by using the non-smooth analysis tool [6]. To ensure a valid algorithm, it is important to choose both λ and ρ_k, which, for the discrete-time case, requires a completely different approach from that of [12], as it will be evident in Sect. 3.2.

Compared with the celebrated distributed gradient descent (DGD) algorithm, see e.g.,[16],

$$x_i^{k+1} = x_i^k + \sum_{j \in \mathscr{N}_i} \tilde{a}_{ij}(x_j^k - x_i^k) - \rho^k \nabla f_i(x_i^k). \tag{4}$$

Algorithm 1 has at least two advantages. First, each node i in Algorithm 1 only uses the binary information of the relative state $(x_j^k - x_i^k)$, instead of the exact relative state from each of its neighbors j, which is essential in some cases where $\operatorname{sgn}(x_j^k - x_i^k)$ is the only available information. Second, Algorithm 1 does not require the adjacency matrix A^k to be doubly-stochastic, while associated adjacency matrix $\tilde{A}^k$ must be doubly-stochastic in DGD [16], where $[\tilde{A}^k]_{ij} := \tilde{a}^k_{ij}$. This is very restrictive in the distributed setting.

Remark 1 Algorithm 1 also works if x is a vector by applying $\text{sgn}(\cdot)$ to each element of the relative state vector. All the results on the scalar case continue to hold with such an adjustment.

3.2 Penalty Method Interpretation of Algorithm 1

In this subsection, we interpret Algorithm 1 via the penalty method and show that it is the subgradient iteration of a penalized optimization problem.

Notice that problem (2) can be essentially reformulated as follows:

$$\begin{aligned} &\underset{\mathbf{x}\in\mathbb{R}^n}{\text{minimize}} && g(\mathbf{x}) := \sum_{i=1}^{n} f_i(x_i) \\ &\text{subject to} && x_i = x_j,\ \forall i, j \in \{1, ..., n\} \end{aligned} \tag{5}$$

where $\mathbf{x} = [x_1, ..., x_n]^\mathsf{T}$. It is easy to see that the optimal value of problem (5) is also $f^\star$, and the set of optimal solutions is $\{x^\star \mathbb{1} | x^\star \in \mathscr{X}^\star\}$.

Define a penalty function by

$$h(\mathbf{x}) = \frac{1}{2}\sum_{i=1}^{n}\sum_{j\in\mathscr{N}_i} a_{ij}|x_i - x_j|. \tag{6}$$

If the associated network $\mathscr{G}$ is connected, then $h(\mathbf{x}) = 0$ is equivalent to that $x_i = x_j,\ \forall i, j \in \{1, ..., n\}$. Thus, a penalized optimization problem of (5) can be given as

$$\underset{\mathbf{x}\in\mathbb{R}^n}{\text{minimize}}\ \widetilde{f}_\lambda(\mathbf{x}) := g(\mathbf{x}) + \lambda h(\mathbf{x}) \tag{7}$$

where $\lambda > 0$ is the penalty factor.

We show below that Algorithm 1 is just the subgradient iteration of the penalized problem (7) with stepsizes ρ^k. Recall that $\text{sgn}(x)$ is a subgradient of $|x|$ for any $x \in \mathbb{R}$. It follows from (6) that a subgradient $\nabla h(\mathbf{x}) = [\nabla h(\mathbf{x})_1, ..., \nabla h(\mathbf{x})_n]^\mathsf{T}$ of $h(\mathbf{x})$ is given element-wise by

$$\nabla h(\mathbf{x})_i = \sum_{j\in\mathscr{N}_i} a_{ij}\text{sgn}(x_i - x_j),\ i \in \mathscr{V}.$$

Similarly, a subgradient $\nabla g(\mathbf{x}) = [\nabla g(\mathbf{x})_1, ..., \nabla g(\mathbf{x})_n]^\mathsf{T}$ of $g(\mathbf{x})$ is given element-wise by $\nabla g(\mathbf{x})_i = \nabla f_i(x_i)$. Then, the i-th element of a subgradient of $\widetilde{f}_\lambda(\mathbf{x})$ is given as

$$\nabla \widetilde{f}_\lambda(\mathbf{x})_i = \lambda \sum_{j \in \mathscr{N}_i} a_{ij}\mathrm{sgn}(x_i - x_j) + \nabla f_i(x_i), i \in \mathscr{V}.$$

Finally, the subgradient method for solving (7) is given as

$$\mathbf{x}^{k+1} = \mathbf{x}^k - \rho^k \nabla \widetilde{f}_\lambda(\mathbf{x}^k),$$

which is exactly the vector form of the local update in Algorithm 1. By [2], it follows that the subgradient method converges to an optimal solution of problem (7) if ρ^k is appropriately chosen.

For a finite $\lambda > 0$, the optimization problems (5) and (7) are generally not equivalent. Under mild conditions, our main result shows that they actually become equivalent if the penalty factor λ is strictly greater than an explicit lower bound. To this end, we define

$$\begin{aligned}\bar{x} &= \frac{1}{n}\mathbb{1}^\mathsf{T}\mathbf{x}, \\ v(\mathbf{x}) &= \max_i(x_i) - \min_i(x_i),\end{aligned} \tag{8}$$

and let $a_{\min}^{(l)}$ be the sum of the l smallest edges' weights, i.e.,

$$a_{\min}^{(l)} = \sum_{e=1}^{l} a_{(e)} \tag{9}$$

where $a_{(1)}, a_{(2)}, \ldots$ are given as an ascending order of the positive weights a_{ij} for any edge $(i, j) \in \mathscr{E}$.

Theorem 1 (Lower bound for the penalty factor, [28]) *Suppose that Assumption 1 holds, and that the multi-agent network is l-connected. If the penalty factor is selected as*

$$\lambda > \underline{\lambda} := \frac{nc}{2a_{min}^{(l)}}, \tag{10}$$

where c and $a_{min}^{(l)}$ are defined in (3) and (9), then

(a) The optimization problems (2) and (7) are equivalent in the sense that the set of optimal solutions and optimal value of (7) are given by $\widetilde{\mathscr{X}}^\star = \{x^\star\mathbb{1} | x^\star \in \mathscr{X}^\star\}$ and $f^\star$, respectively.

(b) For any $\mathbf{x} \notin \{\alpha\mathbb{1} | \alpha \in \mathbb{R}\}$, it holds that

$$\|\nabla \widetilde{f}_\lambda(\mathbf{x})\|_\infty \geq \frac{2\lambda a_{min}^{(l)}}{n} - c.$$

Proof (of part (a)) Consider the inequalities below

$$\begin{aligned}\widetilde{f}_\lambda(\mathbf{x}) &= \lambda h(\mathbf{x}) + g(\mathbf{x} - \bar{x}\mathbb{1} + \bar{x}\mathbb{1}) \\ &\geq \lambda h(\mathbf{x}) + g(\bar{x}\mathbb{1}) + (\mathbf{x} - \bar{x}\mathbb{1})^\mathsf{T}\nabla g(\bar{x}\mathbb{1}) \\ &\geq \lambda h(\mathbf{x}) + f(\bar{x}) - \|\mathbf{x} - \bar{x}\mathbb{1}\|\|\nabla g(\bar{x}\mathbb{1})\|\end{aligned} \tag{11}$$

where the equality follows from the definition of $\widetilde{f}_\lambda(\mathbf{x})$, the first inequality is from (1), and the second inequality results from the Cauchy–Schwarz inequality [2] as well as the fact that $g(a\mathbb{1}) = f(a)$.

Then, we can show that

$$h(\mathbf{x}) \geq a_{\min}^{(l)} v(\mathbf{x}). \tag{12}$$

Since the multi-agent network is l-connected, it follows from Menger's theorem [8] that there exist at least l disjoint paths (two paths are disjoint if they have no common edge) between any two nodes of the graph. Therefore, letting $x_{\max}$ and $x_{\min}$ be two nodes associated with the maximum element and the minimum element of $\mathbf{x}$, respectively, we can find l disjoint paths from $x_{\max}$ to $x_{\min}$.

Let $x_{(p,1)}, ..., x_{(p,n_p)}$ denote the nodes of path p in order, where n_p is the number of nodes in path p, and $x_{(p,1)} = x_{\max}, x_{(p,n_p)} = x_{\min}$ for all $p \in \{1, ..., l\}$. Since these l paths are disjoint, it follows that

$$\begin{aligned}h(\mathbf{x}) &\geq \sum_{p=1}^{l}\sum_{i=1}^{n_p-1} a_{(p,i,i+1)}|x_{(p,i)} - x_{(p,i+1)}| \\ &\geq \sum_{p=1}^{l}\sum_{i=1}^{n_p-1} \min_i a_{(p,i,i+1)}|x_{(p,i)} - x_{(p,i+1)}| \\ &\geq \sum_{p=1}^{l} \min_i a_{(p,i,i+1)} \sum_{i=1}^{n_p-1}(x_{(p,i)} - x_{(p,i+1)}) \\ &\geq \sum_{p=1}^{l} \min_i a_{(p,i,i+1)}(x_{\max} - x_{\min}) \geq a_{\min}^{(l)} v(\mathbf{x})\end{aligned} \tag{13}$$

where $a_{(p,i,i+1)}$ is the weight of the edge connecting nodes $x_{(p,i)}$ and $x_{(p,i+1)}$.

Letting $\tilde{x} = \frac{1}{2}(\max_i(x_i) + \min_i(x_i))$, we have

$$\begin{aligned}\|\mathbf{x} - \bar{x}\mathbb{1}\|\|\nabla g(\bar{x}\mathbb{1})\| &\leq \|\mathbf{x} - \tilde{x}\mathbb{1}\|\|\nabla g(\bar{x}\mathbb{1})\| \\ &\leq \sqrt{n}\|\mathbf{x} - \tilde{x}\mathbb{1}\|_\infty \cdot \sqrt{n}\|\nabla g(\bar{x}\mathbb{1})\|_\infty \\ &\leq \frac{nc}{2} v(\mathbf{x}).\end{aligned} \tag{14}$$

where the first inequality follows from the fact that $\bar{x}$ minimizes $\|\mathbf{x} - \alpha\mathbb{1}\|$ with respect to (w.r.t.) α for all $\mathbf{x}$. Equations (11), (12) and (14) jointly imply the following inequality

$$\widetilde{f}_\lambda(\mathbf{x}) - f^\star \geq f(\bar{x}) - f^\star + (\lambda a_{\min}^{(l)} - \frac{cn}{2})v(\mathbf{x}). \tag{15}$$

Since $\lambda > nc/(2a_{\min}^{(l)}), v(\mathbf{x}) \geq 0, \ \forall\mathbf{x} \in \mathbb{R}^n$ and $f(\bar{x}) \geq f^\star, \forall\bar{x} \in \mathbb{R}$, then the right-hand side of (15) is nonnegative. That is, $\widetilde{f}_\lambda(\mathbf{x}) \geq f^\star$ for all $\mathbf{x} \in \mathbb{R}^n$.

Moreover, it follows from (7) that $\widetilde{f}_\lambda(x^\star\mathbb{1}) = f^\star$ for any $x^\star \in \mathscr{X}^\star$, i.e., $\widetilde{f}_\lambda(\mathbf{x}) = f^\star$ for any $\mathbf{x} \in \widetilde{\mathscr{X}}^\star$. What remains to be shown is that $\widetilde{f}_\lambda(\mathbf{x}) > f^\star$ for all $\mathbf{x} \notin \widetilde{\mathscr{X}}^\star$, which includes

Case (a): $\mathbf{x} \neq \alpha\mathbb{1}$ for any $\alpha \in \mathbb{R}$,
Case (b): $\mathbf{x} = \alpha\mathbb{1}$ for some $\alpha \notin \mathscr{X}^\star$.

For Case (a), $v(\mathbf{x})$ is strictly positive, and hence we know that $\widetilde{f}_\lambda(\mathbf{x}) > f^\star$ from (15). For Case (b), we have $v(\mathbf{x}) = 0$. By (15) we have that $\widetilde{f}_\lambda(\mathbf{x}) \geq f(\bar{x}) = f(\alpha) > f^\star$. Thus, $\widetilde{f}_\lambda(\mathbf{x}) > f^\star$ for all $\mathbf{x} \notin \widetilde{\mathscr{X}}^\star$, which completes the proof of part (a).

The proof of part (b) is very involved and the interested readers are referred to [28] for details. □

Algorithm 1(b) can also be modified to deal with objective functions with unbounded subgradients, e.g., quadratic functions, see [28] for details. Theorem 1 provides a sufficient condition for the equivalence between problems (5) and (7), and allows us to focus only on problem (7). Notice that this result is nontrivial even though the penalty method has been widely studied in optimization theory [2]. For example, a well-known result is that the gap between the optimal values of the penalized problem (7) and the problem (5) gets smaller as λ becomes larger, which however cannot always guarantee the existence of a finite penalty factor λ to eliminate the gap. A large λ may have negative effects on the transient performance of Algorithm 1.

Remark 2 It is worth mentioning that (10) in Theorem 1 also holds for the multidimensional case if Assumption 1(b) is replaced with $\|\nabla f_i(\mathbf{x})\| \leq c$ for all i and $\mathbf{x}$.

In view of the duality theory [2], a potential lower bound for λ could be the absolute value of the associated Lagrange multiplier. However, a Lagrange multiplier usually cannot be obtained before solving its dual problem. Theorem 1 gives an explicit lower bound for λ in terms of the network size and its connectivity, and is tighter than the bounds in [9] and [12].

In fact, the lower bound can be tight in some cases as shown in the following example. Note that [9] does not consider a generic optimization problem.

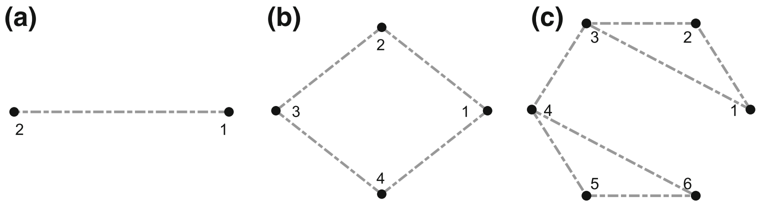


Fig. 1 Some graphs

Example 1 ([28]) Consider the graph in Fig. 1b with unit edge weights, i.e., $a_{ij} = 1$ for all $(i, j) \in \mathscr{V}$. Let $f_1(x) = |x|$, $f_2(x) = |x - 2|$, $f_3(x) = |x - 4|$, $f_4(x) = |x - 6|$ and $f(x) = \sum_{i=1}^{4} f_i(x)$. It is not difficult to compute that the optimal value of $f(x)$ is 8 and the set of optimal solutions is a closed interval $[2, 4]$. By (7), the corresponding penalized problem is given as

$$\begin{aligned}\widetilde{f}_\lambda(\mathbf{x}) = |x_1| + |x_2 - 2| + |x_3 - 4| + |x_4 - 6| + \\ \lambda(|x_1 - x_2| + |x_2 - x_3| + |x_3 - x_4| + |x_4 - x_1|).\end{aligned}$$

Theorem 1 implies that $\widetilde{f}_\lambda(\mathbf{x})$ has the same optimal value as $f(x)$ and the set of optimal solutions is $\widetilde{\mathscr{X}}^\star = \{x^\star \mathbb{1} | x^\star \in [2, 4]\}$, provided that $\lambda > 4 \cdot 1/(2 \cdot 2) = 1$.

Given any $\lambda \leq 1$, consider $\mathbf{x} = [2, 2, 4, 4]^\mathsf{T} \notin \widetilde{\mathscr{X}}^\star$. Clearly,

$$\widetilde{f}_\lambda(\mathbf{x}) = 4 + 4\lambda \leq f^\star = 8,$$

which implies that the set of optimal solutions of the penalized problem is not $\widetilde{\mathscr{X}}^\star$. Thus for any $\lambda \leq 1$, the original problem $f(x)$ cannot be solved via the penalized problem $\widetilde{f}_\lambda(\mathbf{x})$, and the lower bound in (10) is tight in this example. □

The lower bound in (10) is in a simple form and $a_{\min}^{(l)}$ cannot be easily replaced. One may consider to use the minimum degree of the network, i.e., $d_m = \min_{i \in \mathscr{V}} \sum_{j=1}^{n} a_{ij}$. This is impossible in some cases. Consider the 1-connected graph in Fig. 1c with unit edge weights. Then, $a_{\min}^{(1)} = 1$ and $d_m = 2$. Let $[s_1, ..., s_6] = [1, 2, 3, 4, 5, 6]$ and $f_i(x) = |x - s_i|$, $\forall i \in \{1, ..., 6\}$. Set

$$\mathbf{x} = [x_1, ..., x_6]^\mathsf{T} = [3, 3, 3, 4, 4, 4]^\mathsf{T}.$$

Then, using similar arguments as in Example 1, one can infer that the lower bound $\underline{\lambda}$ in (10) cannot be reduced to $nc/(2d_m) = 3/2$.

A similar penalty method interpretation of (4) with constant ρ^k is provided in [14], where the penalty function is chosen as

$$\mathbf{x}^\mathsf{T} L \mathbf{x} = \frac{1}{2} \sum_{i,j} a_{ij}(x_i - x_j)^2$$

and L is the graph Laplacian matrix. However, such a quadratic penalty function cannot always guarantee the existence of a finite λ for the equivalence of the two problems. We provide a concrete example to illustrate this.

Example 2 Consider the graph in Fig. 1a with unit edge weights. Let $f_1(x) = (x-1)^2$ and $f_2(x) = (x-3)^2$. Clearly, the optimal solution of $f(x) = f_1(x) + f_2(x)$ is $x^\star = 2$. Then a corresponding penalized problem using $x^\mathsf{T} Lx$ is

$$\underset{x_1, x_2 \in \mathbb{R}}{\text{minimize}} \;\; f_1(x_1) + f_2(x_2) + \lambda(x_1 - x_2)^2. \tag{16}$$

The optimal solution of (16) is $x_1^\star = (1+4\lambda)/(1+2\lambda)$ and $x_2^\star = (3+4\lambda)/(1+2\lambda)$, and there does not exist a finite value of λ which makes both of them equal to $x^\star = 2$. □

By [2], $\mathbf{x}^\star$ is an optimal solution of (7) if and only if $0 \in \partial \widetilde{f}_\lambda(\mathbf{x}^\star)$. Part (b) of Theorem 1 shows that for any $\mathbf{x} \notin \{\alpha \mathbb{1} | \alpha \in \mathbb{R}\}$, the norm of the corresponding subgradient is uniformly greater than a positive lower bound, which clearly shows non-optimality of $\mathbf{x}$.

4 Convergence Analysis of Algorithm 1 Over Static Graphs

In this section, we examine the convergence behavior of Algorithm 1 over static graphs. If ρ^k is diminishing, all agents converge to the same optimal solution of problem (2) under Algorithm 1. With a constant stepsize, all agents eventually converge to a neighborhood of an optimal solution, where the error size is proportional to the stepsize. For both cases, we perform the non-asymptotic analysis to quantify their convergence rates.

Before providing the convergence results of $\{\mathbf{x}^k\}$, we recall from Proposition A.4.6 in [2] a well-known result on the convergence of a sequence of vectors.

Lemma 1 ([2]) *Let $\mathscr{X}^\star$ be a nonempty subset of $\mathbb{R}^n$, and let $\{\mathbf{x}^k\} \in \mathbb{R}^n$ be a sequence satisfying for some $p > 0$ and for all k,*

$$\|x^{k+1} - x^\star\|^p \le \|x^k - x^\star\|^p - \gamma^k \phi(x^k) + \delta^k, \;\; \forall \mathbf{x}^\star \in \mathscr{X}^\star,$$

where $\{\gamma^k\}$ and $\{\delta^k\}$ are nonnegative sequences satisfying

$$\sum_{k=0}^{\infty} \gamma^k = \infty, \;\; \sum_{k=0}^{\infty} \delta^k < \infty.$$

Suppose that $\phi(\cdot)$ is continuous, nonnegative, and satisfies $\phi(x) = 0$ if and only if $x \in \mathscr{X}^\star$. Then $\{\mathbf{x}^k\}$ converges to an optimal point in $\mathscr{X}^\star$.

The first result in this section is on the convergence of Algorithm 1 under the assumption of diminishing stepsize, which is given as follow:

Assumption 2 The sequence $\{\rho^k\}$ satisfies

$$\sum_{k=0}^{\infty} \rho^k = \infty, \text{ and } \sum_{k=0}^{\infty} (\rho^k)^2 < \infty.$$

Proof of the convergence of Algorithm 1 under Assumption 2 is now given below.

Theorem 2 (Convergence, [28]) *Suppose that the conditions in Theorem 1 and Assumption 2 hold. Let* $\{\mathbf{x}^k\}$ *be generated by Algorithm 1. Then, there exists some optimal point* $x^\star \in \mathscr{X}^\star$ *such that* $\lim_{k\to\infty} \mathbf{x}^k = x^\star \mathbb{1}$.

Proof Under Assumption 1, we have that

$$\|\nabla \widetilde{f}_\lambda(\mathbf{x})\| \leq c_a, \forall \mathbf{x} \in \mathbb{R}^n \tag{17}$$

where $c_a = \sqrt{n}(c + \lambda \|A\|_\infty)$. Since Algorithm 1 is the exact iteration of the subgradient method of problem (7), this implies that

$$\begin{aligned}
&\|\mathbf{x}^{k+1} - x^\star \mathbb{1}\|^2 \\
&= \|\mathbf{x}^k - x^\star \mathbb{1}\|^2 - 2\rho^k (\mathbf{x}^k - x^\star \mathbb{1})^\mathsf{T} \nabla \widetilde{f}_\lambda(\mathbf{x}^k) + (\rho^k)^2 \|\nabla \widetilde{f}_\lambda(\mathbf{x}^k)\|^2 \\
&\leq \|\mathbf{x}^k - x^\star \mathbb{1}\|^2 - 2\rho^k (\widetilde{f}_\lambda(\mathbf{x}^k) - \widetilde{f}_\lambda(x^\star \mathbb{1})) + (\rho^k)^2 c_a^2 \\
&\leq \|\mathbf{x}^k - x^\star \mathbb{1}\|^2 - 2\rho^k (\widetilde{f}_\lambda(\mathbf{x}^k) - f^\star) + (\rho^k)^2 c_a^2, \ \forall x^\star \in \mathscr{X}^\star
\end{aligned} \tag{18}$$

where the first inequality follows from (1) and (17), and the second inequality is from Theorem 1.

By virtue of Lemma 1 and Theorem 1, the result follows immediately. □

Our next result provides a non-asymptotic result to evaluate the convergence rate for $\rho^k = k^{-\alpha}, \alpha \in (0.5, 1]$. To this end, we first define

$$d(\mathbf{x}) = \min_{x^\star \in \mathscr{X}^\star} \|\mathbf{x} - x^\star \mathbb{1}\|. \tag{19}$$

Then, it follows from (8) that

$$\begin{aligned}
v(\mathbf{x}^k) &= \max_i (x_i^k) - \min_i (x_i^k) \\
\bar{x}^k &= \frac{1}{n} \mathbb{1}^\mathsf{T} \mathbf{x}^k.
\end{aligned}$$

Clearly, $d(\mathbf{x})$ is the distance between $\mathbf{x}$ and the set of optimal solutions, v^k is the maximum divergence between agents' states at time k, and $\bar{x}^k$ is the mean of all agents' states at time k. Intuitively, we can use the rates that $f(\bar{x}^k)$ approaches $f^\star$ and v^k reduces to 0 to evaluate the convergence rate of Algorithm 1.

Theorem 3 *Suppose that the conditions in Theorem 1 hold, and let* $\{\mathbf{x}^k\}$ *be generated by Algorithm 1. If* $\rho^k = k^{-\alpha}, \alpha \in (0.5, 1]$, *then*

$$\min_{1<k\leq\bar{k}} f(\bar{x}^k) - f^\star \leq \frac{(2\alpha-1)d(\mathbf{x}^0)^2 + 2\alpha c_a^2}{2(2\alpha-1)s(\bar{k})}, \tag{20}$$

$$\min_{1<k\leq\bar{k}} v(\mathbf{x}^k) \leq \frac{(2\alpha-1)d(\mathbf{x}^0)^2 + 2\alpha c_a^2}{(2\lambda a_{min}^{(l)} - cn)(2\alpha-1)s(\bar{k})},$$

where $\mathbf{x}^0$ *is the initial point,* $\bar{x}^k$ *and* $v(\mathbf{x}^k)$ *are defined in* (8)*, and*

$$s(\bar{k}) = \begin{cases} \frac{1}{1-\alpha}(\bar{k}^{1-\alpha} - 1), & if\, \alpha \in (0.5, 1), \\ ln(\bar{k}), & if\, \alpha = 1. \end{cases}$$

Proof By Theorem 2, $\{\mathbf{x}^k\}$ is a convergent sequence. For any $x^\star \in \mathscr{X}^\star$, it follows from (18) that

$$2\rho^k(\widetilde{f}_\lambda(\mathbf{x}^k) - f^\star) \leq \|\mathbf{x}^k - x^\star\mathbb{1}\|^2 - \|\mathbf{x}^{k+1} - x^\star\mathbb{1}\|^2 + (\rho^k)^2 c_a^2.$$

Summing the above relation over $k \in \{1, ..., \bar{k}\}$ yields

$$\begin{aligned} 2\sum_{k=1}^{\bar{k}} \rho^k(\widetilde{f}_\lambda(\mathbf{x}^k) - f^\star) &\leq \|\mathbf{x}^0 - x^\star\mathbb{1}\|^2 - \|\mathbf{x}^{\bar{k}+1} - x^\star\mathbb{1}\|^2 + \sum_{k=1}^{\bar{k}}(\rho^k)^2 c_a^2 \\ &\leq d(\mathbf{x}^0)^2 + \sum_{k=1}^{\bar{k}}(\rho^k)^2 c_a^2 \end{aligned}$$

where the last inequality holds by choosing $x^\star = \text{argmin}_{x\in\mathscr{X}^\star}\|\mathbf{x}^0 - x\mathbb{1}\|$. Then, we arrive at

$$\min_{0\leq k\leq\bar{k}} \widetilde{f}_\lambda(\mathbf{x}^k) - f^\star \leq \frac{d(\mathbf{x}^0)^2 + \sum_{k=1}^{\bar{k}}(\rho^k)^2 c_a^2}{2\sum_{k=1}^{\bar{k}} \rho^k}. \tag{21}$$

Since $\int_1^{\bar{k}} x^{-\alpha}dx < \sum_{k=1}^{\bar{k}} k^{-\alpha} < \int_1^{\bar{k}} x^{-\alpha}dx + 1$, we have that

$$\sum_{k=1}^{\bar{k}}(\rho^k)^2 < \int_1^{\bar{k}} x^{-2\alpha}dx + 1 = \frac{1 - \bar{k}^{1-2\alpha}}{2\alpha - 1} + 1 < \frac{2\alpha}{2\alpha - 1},$$

and $\sum_{k=1}^{\bar{k}} \rho^k > \int_1^{\bar{k}} x^{-\alpha}dx = s(\bar{k})$. Using the above and (21) leads to

$$\min_{0\le k\le \bar{k}} \widetilde{f}_{\lambda}(\mathbf{x}^k) - f^\star \le \frac{(2\alpha-1)d(\mathbf{x}^0)^2 + 2\alpha c_a^2}{2(2\alpha-1)s(\bar{k})}. \tag{22}$$

Since $f(\bar{x}^k) - f^\star > 0$ and $\lambda a_{\min}^{(l)} - \frac{1}{2}cn > 0$, it follows from (15) and (22) that (20) holds. □

The first inequality in (20) quantifies the decreasing rate of the gap between $f(\bar{x}^k)$ and the optimal value $f^\star$, while the second one shows that the largest difference between agents' states is reduced at a comparable rate. Thus, Theorem 3 reveals that the convergence rate lies between $O(1/\ln(k))$ and $O(\ln(k)/\sqrt{k})$, depending on the choice of ρ^k.

We also provide an alternative evaluation of the convergence rate, which uses a robust form and is presented in the following Corollary 1.

Corollary 1 (Non-asymptotic convergence, [28]) *Suppose that the conditions in Theorem 3 hold. Then*

$$\min_{1<k\le\bar{k}} \max_{i\in\mathscr{V}} f(x_i^k) - f^\star \le \frac{(2\alpha-1)d(\mathbf{x}^0)^2 + 2\alpha c_a^2}{2(2\alpha-1)s(\bar{k})}$$

where all notations are the same as those in Theorem 3.

Proof For all k and any $x_m \in [\min_{i\in\mathscr{V}} x_i^k, \max_{i\in\mathscr{V}} x_i^k]$, it follows from (11) that

$$f(x_m) \le \widetilde{f}_{\lambda}(\mathbf{x}^k) - \lambda h(\mathbf{x}^k) + \|\mathbf{x}^k - x_m\mathbb{1}\|\|\nabla g(x_m\mathbb{1})\|$$

which together with

$$\|\mathbf{x}^k - x_m\mathbb{1}\|\|\nabla g(x_m\mathbb{1})\| \le \sqrt{n}\|\mathbf{x}^k - x_m\mathbb{1}\|_\infty \cdot \sqrt{n}\|\nabla g(x_m\mathbb{1})\|_\infty \le ncv(\mathbf{x}^k)$$

and (13) yields that

$$\begin{aligned} f(x_m) &\le \widetilde{f}_{\lambda}(\mathbf{x}^k) - \lambda h(\mathbf{x}^k) + \frac{nc}{a_{\min}^{(l)}} h(\mathbf{x}^k) \\ &= g(\mathbf{x}^k) + \frac{nc}{a_{\min}^{(l)}} h(\mathbf{x}^k) \\ &\le \widetilde{f}_{2\lambda}(\mathbf{x}^k) \end{aligned}$$

where the last inequality follows from $\lambda > nc/(2a_{\min}^{(l)})$.

Noting that (22) implies

$$\min_{0\le k\le \bar{k}} \widetilde{f}_{2\lambda}(\mathbf{x}^k) - f^\star \le \frac{(2\alpha-1)d(\mathbf{x}^0)^2 + 2\alpha c_a^2}{2(2\alpha-1)s(\bar{k})}$$

the result follows immediately. □

If $f(x)$ is non-differentiable, the objective function of the classical distributed algorithm (4) converges at a rate of $O(\ln(k)/\sqrt{k})$ when $\rho^k = 1/\sqrt{k}$ [20], which is comparable to Algorithm 1 when α approaches 0.5. Thus using only the sign of relative state essentially does not lead to any reduction in the convergence rate. However, if $f(x)$ is more smooth, e.g., differentiable or strongly convex, Algorithm 1 may converge at a rate slower than that of (4).

For a constant stepsize, Algorithm 1 approaches a neighborhood of an optimal solution as fast as $O(1/k)$ and the error size is proportional to the stepsize. These are formally stated in Theorems 4 and 5.

Theorem 4 (Constant Stepsize, [28]) *Suppose that the conditions in Theorem 1 hold, and let* $\{\mathbf{x}^k\}$ *be generated by Algorithm 1. If* $\rho^k = \rho$*, then*

$$\limsup_{k\to\infty} d(\mathbf{x}^k) \le 2\sqrt{n} \max\left\{\widetilde{d}(\rho), \frac{\rho c_a^2}{2\lambda a_{min}^{(l)} - cn}\right\} + \rho c_a$$

where $\widetilde{\mathscr{X}}(\rho) = \{x | f(x) \le f^\star + \rho c_a^2/2\}$ *and* $\widetilde{d}(\rho) = \max_{x\in\widetilde{\mathscr{X}}(\rho)} d(x) < \infty$.

Proof See the Appendix. □

In Theorem 4, $\widetilde{d}(0) = 0$ and $\widetilde{d}(\rho)$ is increasing in ρ. Thus, Algorithm 1 under a constant stepsize finally approaches a neighborhood of $x^\star \mathbb{1}$ for some $x^\star \in \mathscr{X}^\star$, the size of which decreases to zero as ρ tends to zero. If the order of growth of f near the set of optimal solutions is available, then $\widetilde{d}(\rho)$ can even be determined explicitly, which is illustrated in Corollary 2.

Corollary 2 ([28]) *Suppose that the conditions in Theorem 4 hold, and that* $f(x)$ *satisfies*

$$f(x) - f^\star \ge \gamma (d(x))^\alpha$$

where $\gamma > 0$ *and* $\alpha \ge 1$*. Then, it holds that*

$$\limsup_{k\to\infty} d(\mathbf{x}^k) \le 2\sqrt{n} \max\left\{\left(\frac{\rho c_a^2}{2\gamma}\right)^{\frac{1}{\alpha}}, \frac{\rho c_a^2}{2\lambda a_{min}^{(l)} - cn}\right\} + \rho c_a$$

Proof Noting that $\widetilde{d}(\rho) \le (\rho c_a^2/2\gamma)^{\frac{1}{\alpha}}$, the result follows directly from Theorem 4. □

The following theorem evaluates the convergence rate when the stepsize is set to be constant.

Theorem 5 ([28]) *Suppose that the conditions in Theorem 4 hold. Then*

$$\min_{0\le k\le \bar{k}} f(\bar{x}^k) - f^\star \le \frac{\rho c_a^2}{2} + \frac{d(\mathbf{x}^0)^2}{2\rho\bar{k}}, \tag{23}$$

$$\min_{0\le k\le \bar{k}} v(\mathbf{x}^k) \le \frac{\rho c_a^2}{2\lambda a_{min}^{(l)} - cn} + \frac{d(\mathbf{x}^0)^2}{\rho\bar{k}(2\lambda a_{min}^{(l)} - cn)}.$$

Proof From (21) we know that

$$\min_{0\le k\le \bar{k}} \widetilde{f}_\lambda(\mathbf{x}^k) - f^\star \le \frac{d(\mathbf{x}^0)^2 + \bar{k}\rho^2 c_a^2}{2\rho\bar{k}},$$

which together with (15) implies the result. □

Remark 3 The following conclusions can be easily drawn from Theorem 5.

(a) $\min_{0\le k\le \bar{k}} f(\bar{x}^k)$ approaches the interval $[f^\star, f^\star + \frac{\rho c_a^2}{2}]$ at a rate of $O(1/\bar{k})$.

(b) Given $\bar{k}$ iterations, let $\rho = \frac{1}{c_a}\frac{d(\mathbf{x}^0)}{\sqrt{\bar{k}}}$, which minimizes the right-hand side of (23). Then

$$\min_{0\le k\le \bar{k}} f(\bar{x}^k) - f^\star \le c_a \frac{d(\mathbf{x}^0)}{\sqrt{\bar{k}}},$$

$$\min_{0\le k\le \bar{k}} v(\mathbf{x}^k) \le \frac{c_a}{2\lambda a_{\min}^{(l)} - cn}\frac{d(\mathbf{x}^0)}{\sqrt{\bar{k}}}.$$

The multi-agent network converges only to a point that is close to an optimal solution with an error size $O(\bar{k}^{-1/2})$.

Algorithm 2: Distributed Algorithm Using the Sign of Relative State

```
1. Initialization: Every node $i$ sets $x_i^0 = 0$ for all $i \in \mathscr{V}$.
2. Repeat
3.     Information collection: Each node $i$ collects the sign of the relative state to its neighbors
       at time $k$, e.g., node $j \in \mathscr{N}_i^k$ and obtain $r_i^k$, which is given below
           $r_i^k = \sum_{j\in\mathscr{N}_i^k} a_{ij}^k \mathrm{sgn}(x_j^k - x_i^k).$
4.     Local update: The decision variable in each node is locally updated as
           $x_i^{k+1} = x_i^k + \rho^k \left(\lambda \cdot r_i^k - \nabla f_i(x_i^k)\right),$
       where $\lambda$ and $\rho^k$ are to be given, and $\nabla f_i(x_i^k)$ is any subgradient of $f_i(x)$ at $x_i^k$.
5.     Set $k = k + 1$.
6. Until a predefined stopping rule (e.g., a maximum iteration number) is satisfied.
```

5 The Distributed Optimization Algorithm over Time-varying Graphs

When the graphs are time-varying, Algorithm 1 is revised and we provide the details in Algorithm 2. In this section, we study the convergence of Algorithm 2 over two types of time-varying graphs: uniformly jointly connected time-varying graphs and *randomly* activated graphs.

5.1 Uniformly Jointly Connected Time-varying Graphs

Now we introduce the concept of uniformly jointly connected time-varying graphs. First we define the union of the graphs $\mathcal{G}^{(k,b)}$ for integers $k \geq 0$ and $b > 0$ below

$$\mathcal{G}^{(k,b)} = (\mathcal{V}, \mathcal{E}^{(k,b)}, A^{(k,b)}) := \mathcal{G}^{k} \cup \mathcal{G}^{k+1} \cup \cdots \cup \mathcal{G}^{k+b-1}$$

and $A^{(k,b)}$ is the associated adjacency matrix of $\mathcal{G}^{(k,b)}$. We make the following assumption.

Assumption 3 Assume that

(a) For some $\eta > 0$, it holds that

$$\begin{cases} a_{ij}^{k} \geq \eta, \text{ if } (i, j) \in \mathcal{E}^{k}, \\ a_{ij}^{k} = 0, \text{ otherwise.} \end{cases} \tag{24}$$

(b) There exists an integer $b \geq 1$ such that $A^{(tb,b)}$ is l-connected for each $t = 0, 1, 2, \ldots$

Assumption 3 is commonly made in dealing with deterministically time-varying graphs. The first part requires that either an edge is not connected at some time, or the edge is connected with a weight larger than some fixed value. The second part assumes the joint graph in time intervals with length b to be connected. We call time-varying graphs satisfying Assumption 3 *uniformly jointly connected graphs*, which are also sometimes referred to as b-connected graphs [15, 17].

We are now ready to present the convergence result of Algorithm 2 over uniformly jointly connected graphs.

Theorem 6 (Convergence, [26]) *Suppose that Assumptions 1-3 hold, and that there exists a constant $c_\rho > 0$ such that for all $k > 0$,*

$$\max_{t\in[k,k+b)} \rho^{t} \leq c_\rho \min_{t\in[k,k+b)} \rho^{t}. \tag{25}$$

Select

$$\lambda > \frac{nbcc_\rho}{2l\eta},$$

where n is the number of agents, c is given in Assumption 1, c_ρ is given in Assumption 2, and b, l, η are given in Assumption 3. Let $\{\mathbf{x}^k\}$ be generated by Algorithm 2. Then, $\lim_{k\to\infty}\mathbf{x}^k = x^\star\mathbb{1}$ for some $x^\star \in \mathscr{X}^\star$.

Proof We first consider the subsequence $\{\mathbf{x}^{tb}, t = 0, 1, 2, ...\}$, i.e., we let $k = tb$, $t \in \{0, 1, 2, ...\}$. Define

$$\widetilde{f}_\lambda^k(\mathbf{x}) := \frac{\lambda}{2}\sum_{i,j\in\mathscr{V}} a_{ij}^k|x_i - x_j| + \sum_{i=1}^n f_i(x_i)$$

and

$$\begin{aligned}\widetilde{f}_\lambda^{(k,b)}(\mathbf{x}) &:= \frac{1}{\rho^k}\sum_{t=k}^{b+k-1}\rho^t\widetilde{f}_\lambda^t(\mathbf{x})\\ &= \frac{\lambda}{2\rho^k}\sum_{i,j\in\mathscr{V}}\sum_{t=k}^{b+k-1}\rho^t a_{ij}^t|x_i - x_j| + \frac{1}{\rho^k}\sum_{t=k}^{b+k-1}\rho^t\sum_{i=1}^n f_i(x_i)\\ &= \bar{\rho}^k\left[\frac{\lambda}{2}\sum_{i,j\in\mathscr{V}}\bar{a}_{ij}^k|x_i - x_j| + \sum_{i=1}^n f_i(x_i)\right]\end{aligned}$$

where

$$\bar{\rho}^k = \sum_{t=k}^{b+k-1}\frac{\rho^t}{\rho^k}, \text{ and } \bar{a}_{ij}^k = \frac{\sum_{t=k}^{b+k-1}\rho^t a_{ij}^t}{\sum_{t=k}^{b+k-1}\rho^t}.$$

Let $[\bar{A}^k]_{ij} := \bar{a}_{ij}^k$ and $\bar{a}_{\min}^{k,(l)}$ be the sum of the l smallest nonzero elements of $\bar{A}^k$. Note that $\bar{a}_{\min}^{k,(l)}$ is well defined because for any (i, j), if $[A^{(k,b)}]_{ij}$ is nonzero, then $[\bar{A}^k]_{ij}$ is also nonzero, and $A^{(k,b)}$ has at least l nonzero elements by Assumption 3.

Then, we obtain from (25) that

$$\bar{a}_{ij}^k \geq \frac{\min_{t\in[k,k+b)}\rho^t\sum_{t=k}^{b+k-1}a_{ij}^t}{b\max_{t\in[k,k+b)}\rho^t} \geq \frac{\sum_{t=k}^{b+k-1}a_{ij}^t}{bc_\rho}.$$

Thus, if $\bar{a}_{ij}^k \neq 0$, then it follows from (24) that $\bar{a}_{ij}^k$ must be larger than η/bc_ρ, which means that any nonzero element of $\bar{A}^k$ is larger than η/bc_ρ, and hence $\bar{a}_{\min}^{k,(l)} \geq l\eta/bc_\rho$.

By virtue of that $\lambda > nbcc_\rho/(2l\eta)$ and Theorem 1, we know that the problem

$$\underset{\mathbf{x}\in\mathbb{R}^n}{\text{minimize}}\ \frac{1}{\bar{\rho}^k}\widetilde{f}_\lambda^{(k,b)}(\mathbf{x})$$

is equivalent to the original problem for all $k = tb$, $t \in \{0, 1, 2, ...\}$. That is, we have $\widetilde{f}_\lambda^{(k,b)}(\mathbf{x}) \geq \bar{\rho}^k f^\star$ for all $\mathbf{x} \in \mathbb{R}^n$, and $\widetilde{f}_\lambda^{(k,b)}(\mathbf{x}) = \bar{\rho}^k f^\star$ if and only if $\mathbf{x} \in \{a\mathbb{1}|a \in \mathscr{X}^\star\}$.

Let $\mathbf{d}^k = [d_1^k, ..., d_n^k]^\mathsf{T}$, where

$$d_i^k = -\lambda \sum_{j\in\mathscr{N}_i^k} a_{ij}^k \operatorname{sgn}(x_j^k - x_i^k) + \nabla f_i(x_i^k).$$

Then, Algorithm 2 can be written in a compact form as

$$\mathbf{x}^{k+1} = \mathbf{x}^k - \rho^k \mathbf{d}^k.$$

Note that $\mathbf{d}^k$ is a subgradient of $\widetilde{f}_\lambda^k(\mathbf{x})$ at $\mathbf{x}^k$, and $\|\nabla \widetilde{f}_\lambda^k(\mathbf{x})\| \leq c_a$ for any $\mathbf{x} \in \mathbb{R}^n$ by (17). Hence $\|\mathbf{d}^k\| \leq c_a$ for any k. Let $x^\star$ be an arbitrary element of $\mathscr{X}^\star$. We have the following relation

$$\begin{aligned}\|\mathbf{x}^{k+b} - x^\star\mathbb{1}\|^2 &= \left\|\mathbf{x}^k - \sum_{t=k}^{b+k-1} \rho^t \mathbf{d}^t - x^\star\mathbb{1}\right\|^2 \\ &= \|\mathbf{x}^k - x^\star\mathbb{1}\|^2 + 2(x^\star\mathbb{1} - \mathbf{x}^k)^\mathsf{T} \sum_{t=k}^{b+k-1} \rho^t \mathbf{d}^t + \left\|\sum_{t=k}^{b+k-1} \rho^t \mathbf{d}^t\right\|^2 \\ &\leq \|\mathbf{x}^k - x^\star\mathbb{1}\|^2 + 2(x^\star\mathbb{1} - \mathbf{x}^k)^\mathsf{T} \sum_{t=k}^{b+k-1} \rho^t \mathbf{d}^t + bc_a^2 \sum_{t=k}^{b+k-1} (\rho^t)^2.\end{aligned} \tag{26}$$

Consider the second term of the right-hand-side of (26); then

$$\begin{aligned}&(x^\star\mathbb{1} - \mathbf{x}^k)^\mathsf{T} \sum_{t=k}^{b+k-1} \rho^t \mathbf{d}^t = \sum_{t=k}^{b+k-1} \rho^t (x^\star\mathbb{1} - \mathbf{x}^k)^\mathsf{T} \mathbf{d}^t \\ &= \sum_{t=k}^{b+k-1} \rho^t (x^\star\mathbb{1} - \mathbf{x}^t)^\mathsf{T} \mathbf{d}^t + \sum_{t=k}^{b+k-1} \rho^t (\mathbf{x}^t - \mathbf{x}^k)^\mathsf{T} \mathbf{d}^t \\ &\leq \sum_{t=k}^{b+k-1} \rho^t (\widetilde{f}_\lambda^t(x^\star\mathbb{1}) - \widetilde{f}_\lambda^t(\mathbf{x}^t)) + \sum_{t=k}^{b+k-1} \rho^t (\mathbf{x}^t - \mathbf{x}^k)^\mathsf{T} \mathbf{d}^t \\ &= \sum_{t=k}^{b+k-1} \rho^t (f^\star - \widetilde{f}_\lambda^t(\mathbf{x}^k)) + \sum_{t=k}^{b+k-1} \rho^t (\widetilde{f}_\lambda^t(\mathbf{x}^k) - \widetilde{f}_\lambda^t(\mathbf{x}^t)) + \sum_{t=k}^{b+k-1} \rho^t (\mathbf{x}^t - \mathbf{x}^k)^\mathsf{T} \mathbf{d}^t\end{aligned} \tag{27}$$

$$
\begin{aligned}
&= \rho^k(f^\star - \widetilde{f}_\lambda^{(k,b)}(\mathbf{x}^k)) + \sum_{t=k}^{b+k-1} \rho^t(\widetilde{f}_\lambda^t(\mathbf{x}^k) - \widetilde{f}_\lambda^t(\mathbf{x}^t)) + \sum_{t=k}^{b+k-1} \rho^t(\mathbf{x}^t - \mathbf{x}^k)^\mathsf{T}\mathbf{d}^t \\
&\le \rho^k(f^\star - \widetilde{f}_\lambda^{(k,b)}(\mathbf{x}^k)) + \sum_{t=k}^{b+k-1} \rho^t \left(\|\mathbf{x}^t - \mathbf{x}^k\|\|\mathbf{d}^t\| + \|\mathbf{x}^k - \mathbf{x}^t\|\|\nabla \widetilde{f}_\lambda^t(\mathbf{x}^k)\|\right) \\
&= \rho^k(f^\star - \widetilde{f}_\lambda^{(k,b)}(\mathbf{x}^k)) + \sum_{t=k}^{b+k-1} \rho^t \|\mathbf{x}^k - \sum_{u=k}^{t-1} \rho^u \mathbf{d}^u - \mathbf{x}^k\|(\|\mathbf{d}^t\| + \|\nabla \widetilde{f}_\lambda^t(\mathbf{x}^k)\|) \\
&\le \rho^k(f^\star - \widetilde{f}_\lambda^{(k,b)}(\mathbf{x}^k)) + (\sum_{t=k}^{b+k-1} \rho^t)^2 \\
&\le \rho^k(f^\star - \widetilde{f}_\lambda^{(k,b)}(\mathbf{x}^k)) + 2bc_a^2 \sum_{t=k}^{b+k-1} (\rho^t)^2.
\end{aligned}
$$

Combining (27) with (26) yields that

$$
\|\mathbf{x}^{k+b} - x^\star \mathbb{1}\|^2 \le \|\mathbf{x}^k - x^\star \mathbb{1}\|^2 + 2\rho^k(f^\star - \widetilde{f}_\lambda^{(k,b)}(\mathbf{x}^k)) + 5bc_a^2 \sum_{t=k}^{b+k-1} (\rho^t)^2. \quad (28)
$$

Noting that $k = tb$, $t \in \{0, 1, ...\}$, the above relation becomes

$$
\begin{aligned}
&\|\mathbf{x}^{(t+1)b} - x^\star \mathbb{1}\|^2 \\
&\quad \le \|\mathbf{x}^{tb} - x^\star \mathbb{1}\|^2 + 2\rho^{tb}(f^\star - \widetilde{f}_\lambda^{(tb,b)}(\mathbf{x}^k)) + 5bc_a^2 \sum_{u=tb}^{(t+1)b-1} (\rho^u)^2.
\end{aligned}
$$

Note that $\widetilde{f}_\lambda^{(tb,b)}(\mathbf{x}^k)$ is nonnegative and $\widetilde{f}_\lambda^{(tb,b)}(\mathbf{x}) = 0$ if and only if $\mathbf{x} \in \{a\mathbb{1} | a \in \mathscr{X}^\star\}$, and that $\sum_{t=1}^\infty \rho^{tb} = \infty$, $\sum_{t=1}^\infty (\rho^{tb})^2 < \infty$. It follows from Lemma 1 that there exists $\bar{x} \in \mathscr{X}^\star$ such that the subsequence $\{\mathbf{x}^{tb}\}$, $t \in \{0, 1, 2, ...\}$ must converge to $\bar{x}\mathbb{1}$. This, combined with $\lim_{k\to\infty} \rho^k = 0$, implies that $\{\mathbf{x}^k\}$ converges to $\bar{x}\mathbb{1}$. □

Compared with the convergence result on static graphs (Theorem 2), the major difference on uniformly jointly connected graphs is that λ should be bc_ρ times larger than that in the case of static graphs.

Next, we evaluate the convergence rate of Algorithm 2 over uniformly jointly connected graphs when $\rho^k = k^{-\alpha}$, $\alpha \in (0.5, 1]$. As in Theorem 3, we evaluate the rates that $f(\bar{x}^k)$ approaches $f^\star$ and $v(\mathbf{x}^k)$ tends to 0 to quantify the convergence rate.

Theorem 7 (Non-asymptotic result, [26]) *Let the assumptions in Theorem 6 hold, and further assume that* $\lambda > nbc/l\eta$. *Let* $\{\mathbf{x}^k\}$ *be generated by Algorithm 2. If* $\rho^k = k^{-\alpha}$ *with some* $\alpha \in (0.5, 1]$, *then for any* $k_0 > 2b$,

$$\min_{1<k\leq k_0} f(\bar{x}^k) - f^\star \leq \frac{(2\alpha-1)(d(\mathbf{x}^0))^2 + 10\alpha bc_a^2}{b(2\alpha-1)s(k_0)} \tag{29}$$
$$\min_{1<k\leq k_0} v(\mathbf{x}^k) \leq \frac{2(2\alpha-1)(d(\mathbf{x}^0))^2 + 12\alpha bc_a^2}{(\lambda l\eta - nbc)(2\alpha-1)s(k_0)}$$

where $\mathbf{x}^0$ *is the initial point, and*

$$s(k_0) = \begin{cases} \dfrac{(k_0-b)^{1-\alpha} - b^{1-\alpha}}{b(1-\alpha)}, & \alpha \in (0.5, 1), \\ \dfrac{1}{b}[ln(k_0-b) - ln(b)], & \alpha = 1. \end{cases}$$

Proof Note that λ and $\{\rho^k\}$ satisfy the conditions in Theorem 6 with $c_\rho = 2$, and $\|\nabla \widetilde{f}_\lambda^k(\mathbf{x})\| \leq c_a$ for any $\mathbf{x}$ and k. Let $x^\star$ be an arbitrary optimal solution of problem (2) and $t_0 = \lfloor k_0/b \rfloor$, where $\lfloor x \rfloor$ denotes the nearest integer to $(\cdot)$ that is smaller than $(\cdot)$. It then follows from (28) that

$$2\rho^{tb}(\widetilde{f}_\lambda^{(tb,b)}(\mathbf{x}^k) - f^\star) \leq \|\mathbf{x}^{tb} - x^\star\mathbb{1}\|^2 - \|\mathbf{x}^{(t+1)b} - x^\star\mathbb{1}\|^2 + 5bc_a^2 \sum_{u=tb}^{tb+b-1} (\rho^u)^2.$$

Summing the above relation over $t = 0, 1, ..., t_0$ yields

$$\begin{aligned} &2\sum_{t=0}^{t_0} \rho^{tb}(\widetilde{f}_\lambda^{(tb,b)}(\mathbf{x}^k) - f^\star) \\ &\leq \|\mathbf{x}^0 - x^\star\mathbb{1}\|^2 - \|\mathbf{x}^{t_0 b+1} - x^\star\mathbb{1}\|^2 + 5bc_a^2 \sum_{t=0}^{t_0} \sum_{u=tb}^{(t+1)b-1} (\rho^u)^2 \\ &\leq d(\mathbf{x}^0) + 5bc_a^2 \sum_{k=1}^{k_0} (\rho^k)^2. \end{aligned}$$

Therefore, we have

$$\min_{0\leq k\leq k_0} \widetilde{f}_\lambda^{(k,b)}(\mathbf{x}^k) - f^\star \leq \frac{d(\mathbf{x}^0) + 5bc_a^2 \sum_{k=1}^{k_0}(\rho^k)^2}{2\sum_{t=0}^{t_0} \rho^{tb}}. \tag{30}$$

Since

$$\int_1^{k_0} \frac{1}{x^\alpha} dx < \sum_{k=1}^{k_0} \frac{1}{k^\alpha} < \int_1^{k_0} \frac{1}{x^\alpha} dx + 1,$$

we obtain that

$$\sum_{k=1}^{k_0}(\rho^k)^2 < \int_1^{k_0}\frac{1}{x^{2\alpha}}dx + 1 = \frac{1-k_0^{1-2\alpha}}{2\alpha-1} + 1 < \frac{2\alpha}{2\alpha-1}$$

and for $\alpha \in (0.5, 1)$,

$$\begin{aligned}\sum_{t=0}^{t_0}\rho^{tb} &> b^{-a}\sum_{t=0}^{t_0}\rho^{t} > b^{-a}\int_1^{t_0}\frac{1}{x^{\alpha}}dx = \frac{t_0^{1-\alpha}-1}{b^a(1-\alpha)} \\ &> \frac{(k_0/b-1)^{1-\alpha}-1}{b^a(1-\alpha)} = s(k_0).\end{aligned}$$

We also obtain $\sum_{t=0}^{t_0}\rho^{tb} = s(k_0)$ using similar arguments. Substituting these two inequalities into (30) yields

$$\min_{0\le k\le k_0}\widetilde{f}_\lambda^{(k,b)}(\mathbf{x}^k) - f^\star \le \frac{(2\alpha-1)d(\mathbf{x}^0)+10bc_a\alpha}{2(2\alpha-1)s(k_0)}. \tag{31}$$

Noticing that $\bar{\rho}^k \ge c_\rho/2 \ge b/2$ for all k, we have

$$\begin{aligned}\widetilde{f}_\lambda^{(k,b)}(\mathbf{x}^k) &= \lambda\bar{\rho}^k h(\mathbf{x}^k) + \bar{\rho}^k g(\mathbf{x}^k - \bar{x}^k\mathbb{1} + \bar{x}^k\mathbb{1}) \\ &\ge \frac{b}{2}\left[\lambda h(\mathbf{x}^k) + g(\bar{x}^k\mathbb{1}) + (\mathbf{x}^k - \bar{x}^k\mathbb{1})^{\mathsf{T}}\nabla g(\bar{x}^k\mathbb{1})\right] \\ &\ge \frac{b}{2}\left[\lambda h(\mathbf{x}^k) + f(\bar{x}^k) - \|\mathbf{x}^k - \bar{x}^k\mathbb{1}\|\|\nabla g(\bar{x}^k\mathbb{1})\|\right]\end{aligned} \tag{32}$$

where the first equality follows from the definition of $\widetilde{f}_\lambda^{(k,b)}(\mathbf{x})$, the second inequality is from the definition of a subgradient, and the last inequality is the result of the Cauchy–Schwarz inequality as well as the fact that $g(a\mathbb{1}) = f(a)$.

Recall from (13) and (14) that

$$h(\mathbf{x}^k) \ge \frac{l\eta}{2b}v^k, \text{ and } \|\mathbf{x}^k - \bar{x}^k\mathbb{1}\|\|\nabla g(\bar{x}^k\mathbb{1})\| \le \frac{nc}{2}v(\mathbf{x}^k).$$

These two relations together with (32) yield

$$\widetilde{f}_\lambda^{(k,b)}(\mathbf{x}^k) - f^\star \ge \frac{b}{2}\left[f(\bar{x}^k) - f^\star + (\frac{\lambda l\eta}{2b} - \frac{nc}{2})v(\mathbf{x}^k)\right].$$

Since $f(\bar{x}^k) - f^\star > 0$ and $\lambda l\eta - bcn > 0$, the above inequality combined with (31) implies (29) immediately. □

Theorem 2 reveals that from the worst-case point of view, the convergence rate over uniformly jointly connected time-varying graphs is about b times slower than that of a static graph (Theorem 3), which is reasonable.

5.2 Randomly Activated Graphs

This subsection studies the convergence of Algorithm 2 over randomly activated graphs, which can model many networks such as gossip social networks and random measurement losses in networks. The definition is given as follows.

Definition 2 (*Randomly Activated Graphs*) The sequence of graphs $\{\mathscr{G}^k\}$ are randomly activated if for all $i, j \in \mathscr{V}, i \neq j$, $\{a_{ij}^k\}$ is an i.i.d. Bernoulli process with $\mathbb{P}\{a_{ij}^k = 1\} = p_{ij}$, where $\mathbb{P}(\mathscr{X})$ denotes the probability of an event $\mathscr{X}$ and $0 \leq p_{ij} \leq 1$, $\forall i, j \in \mathscr{V}$.

Remark 4 For brevity, we assume here that the weight of each edge a_{ij}^k is taken to be either zero or one at each time k in randomly activated graphs.

We call $P = [p_{ij}]$ the activation matrix of $\mathscr{G}^k$, and the graph associated with P is denoted as $\mathscr{G}_P$, which is also the mean graph of $\mathscr{G}^k$, i.e.,

$$\mathscr{G}_P := \mathbb{E}(\mathscr{G}^k). \tag{34}$$

Recall that Algorithm 1 is the iteration of subgradient methods of (7). Similarly, Algorithm 2 is just the iteration of the *stochastic* subgradient method of the following optimization problem

$$\underset{\mathbf{x}\in\mathbb{R}^n}{\text{minimize}} \ \widehat{f}_\lambda(\mathbf{x}) := g(\mathbf{x}) + \lambda \widehat{h}(\mathbf{x}) \tag{35}$$

where $g(x)$ is given in (5) and

$$\widehat{h}(\mathbf{x}) = \frac{1}{2} \sum_{i,j=1}^{n} p_{ij} |x_i - x_j|.$$

To exposit it, notice that $\mathbb{E}(a_{ij}^k) = p_{ij}$, and thus a stochastic subgradient $\nabla_s \widehat{h}(\mathbf{x}) = [\nabla_s \widehat{h}(\mathbf{x})_1, ..., \nabla_s \widehat{h}(\mathbf{x})_n]^\mathsf{T}$ of $\widehat{h}(\mathbf{x})$ is given element-wise by

$$\nabla_s \widehat{h}(\mathbf{x})_i = \sum_{j=1}^{n} a_{ij}^k \text{sgn}(x_i - x_j) = \sum_{j \in \mathscr{N}_i^k} \text{sgn}(x_i - x_j).$$

Since $\mathbb{E}\{\nabla_s \widehat{h}(\mathbf{x})_i\} = \sum_j p_{ij} \text{sgn}(x_i - x_j)$, $\mathbb{E}\{\nabla_s \widehat{h}(\mathbf{x})\}$ is a subgradient of $\widehat{h}(\mathbf{x})$. Hence, the almost sure convergence of Algorithm 2 follows from the following lemma.

Lemma 2 (Convergence of Stochastic Subgradient Method, [3]) *Consider the optimization problem*

$$\underset{\mathbf{x}\in\mathbb{R}^n}{\textit{minimize}} \ \mathbb{E}\{F(\mathbf{x}, w)\} \tag{36}$$

where w is a random variable and $F(\mathbf{x}, w) : \mathbb{R}^n \times \mathbb{R} \to \mathbb{R}$ is continuous and convex w.r.t. $\mathbf{x}$ for any given w. Let $\mathscr{X}^\star$ be the set of optimal solutions and assume that $\mathscr{X}^\star$ is not empty.

The stochastic subgradient method for (36) is given by

$$\mathbf{x}^{k+1} = \mathbf{x}^k - \rho^k r(\mathbf{x}^k, w^k)$$

where $r(\mathbf{x}, w^k)$ is bounded and $\mathbb{E}(r(\mathbf{x}, w^k))$ is a subgradient of $\mathbb{E}\{F(\mathbf{x}, w^k)\}$ for all $\mathbf{x} \in \mathbb{R}^n$. If $\{\rho^k\}$ is chosen such that

$$\sum_{k=0}^{\infty} \rho^k = \infty, \quad \sum_{k=0}^{\infty} (\rho^k)^2 < \infty,$$

then it holds almost surely that $\lim_{k\to\infty} \mathbf{x}^k = \mathbf{x}^\star$ for some $\mathbf{x}^\star \in \mathscr{X}^\star$.

The following theorem summarizes the above analysis, and is the main result of this subsection.

Theorem 8 ([28]) *Suppose that Assumptions 1 and 2 hold, and that the multi-agent network $\mathscr{G}_P$ is l-connected. Select*

$$\lambda > \frac{nc}{2p_{\min}^{(l)}},$$

where $\mathscr{G}_P$ is given in (34), $p_{\min}^{(l)}$ denotes the sum of the l smallest nonzero elements of P. Let $\{\mathbf{x}^k\}$ be generated by Algorithm 2. Then, it holds almost surely that $\lim_{k\to\infty} \mathbf{x}^k = x^\star \mathbb{1}$ for some $x^\star \in \mathscr{X}^\star$.

Proof By Theorem 1, it follows that problem (35) has the same set of optimal solutions and optimal value as problem (2). Combined with Lemma 2, the proof follows. □

6 Numerical Examples

In this section, we apply our algorithms to distributedly find the geometric median of a couple of points in a two-dimensional plane. The geometric median of n points is defined as the point which minimizes the sum of Euclidean distances to these points [7]. In other words, it is the optimal solution of the following convex optimization problem:

$$\underset{\mathbf{x}\in\mathbb{R}^2}{\text{minimize}} \;\; f(\mathbf{x}) := \sum_{i=1}^{n} f_i(\mathbf{x}) = \sum_{i=1}^{n} \|\mathbf{x} - \mathbf{x}_i\|. \tag{37}$$

The local function $f_i(\mathbf{x}) := \|\mathbf{x} - \mathbf{x}_i\|$ is convex but non-differentiable, the subdifferential of which is given as

$$\partial f_i(\mathbf{x}) = \begin{cases} \dfrac{\mathbf{x} - \mathbf{x}_i}{\|\mathbf{x} - \mathbf{x}_i\|}, & \text{if } \mathbf{x} \neq \mathbf{x}_i \\ \{g \mid \|g\| \leq 1\}, & \text{otherwise.} \end{cases}$$

Apparently, problem (37) satisfies Assumption 1, and hence it can be solved by Algorithms 1 and 2. Notice that $\mathbf{x}$ in (37) is 2-dimensional and Algorithms 1 and 2 should be modified accordingly as stated in Remark 1.

The geometric median problem is a special case of least square problems in statistics and Weber problems in location theory. Here we provide a possible application in distributed settings. Consider n base stations under the sea, and we want to find a place to build a communication center, which should have the minimum distances to these stations to save the costs of cables. Since global positioning is very difficult under seas, a feasible distributed approach to find the desired place is for each station to send an agent, which however can only measure the distance to the station and know rough relative positions to its neighbor agents. Clearly, we can use the proposed algorithms to achieve this goal.

In this example, we consider five stations (hence five agents), the positions of which are randomly generated on a rectangular area with size 100 × 100. We run three simulations over a static graph, uniformly jointly connected graphs, and randomly activated graphs, respectively. We choose the stepsize $\rho^k = 5/(k + 10)$ in all simulations. The topology of the five agents is a ring graph as shown in Fig. 2a. The λ in Algorithm 1 used in the static graph's case is chosen to be 2, which satisfies the condition in Remark 2. For the uniformly jointly connected graphs' case, we let only one edge in the graph of Fig. 2a connect at each time, and each edge connects once and only once in each cycle, the order of which is determined by a random permutation of {1, ..., 5} at the beginning of each cycle. The λ in Algorithm 2 used in the case of uniformly jointly connected graphs is chosen to be 6. We generate randomly activated graphs by letting each edge in the graph of Fig. 2a connect with probability 0.5 at each time, and we choose λ to be 4.

Fig. 2b, c, d depict respectively the trajectories of the 5 agents from $k = 1$ to 1500 over the static graph, uniformly jointly connected graphs and randomly activate graphs, where the filled circles are the initial positions of the agents and the black triangle is the geometric median of these circles computed by Weiszfeld's method [1]. As shown in the figures, agents in all cases converge to the geometric median with however slightly different transient performances.

If λ is smaller than the lower bound provided in Theorem 1, consensus may not be achieved among agents. Figure 3 shows the trajectories of agents with $\lambda = 0.8, 2, 1.5$ over a static graph, uniformly jointly connected graphs, randomly activated graphs, respectively. Other settings remain the same except that we increase the times of iterations to 5000. Clearly, agents fail to converge to the geometric median.

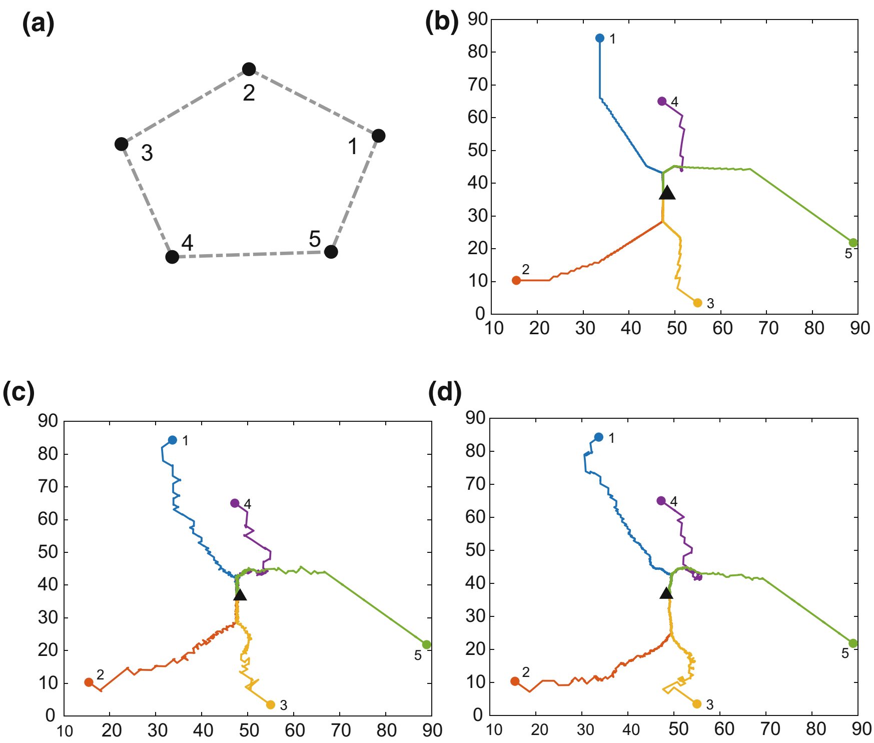


Fig. 2 **a** The topology of the agents. **b** The trajectories of the agents in a static graph, where the filled circles are the initial positions of the agents and the black triangle is the geometric median of these circles. **c** The trajectories of the agents in uniformly jointly connected graphs. **d** The trajectories of the agents in randomly activated graphs

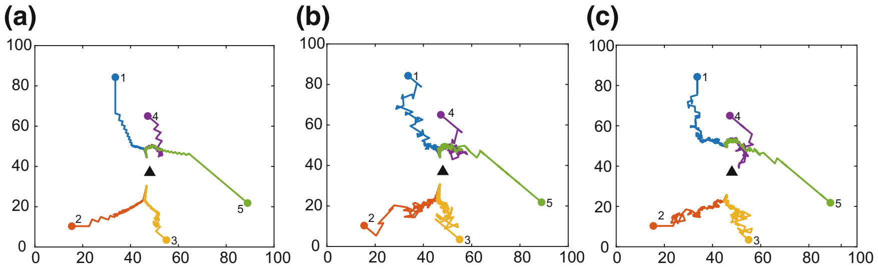


Fig. 3 The trajectories of agents with smaller λ over a static graph, uniformly jointly connected graphs, randomly activated graphs, respectively

7 Conclusions

In this chapter, we have proposed a distributed optimization algorithm to solve the additive cost optimization problem in multi agent networks. Each agent in the algorithm uses only the sign of relative state to each of its neighbor agents. The network was allowed to be static or time-varying. For the former case, we have first provided a penalty method interpretation of our algorithm, and then studied its convergence under diminishing stepsizes as well as a constant stepsize. We have shown that the convergence rate varies from $O(1/\ln(k))$ to $O(1/\sqrt{k})$, depending on the stepsize. For the latter case, we studied the performance of our algorithm over the so-called uniformly jointly connected graphs and randomly activated graphs, the convergence of which is also guaranteed. Finally, we have applied our algorithm to solve a geometric median problem. All the theoretical results have been corroborated via simulations.

Acknowledgements The authors would very much like to thank Professor Tamer Başar for the stimulating discussions on this topic. This work was supported by the National Natural Science Foundation of China under Grant No. 61722308, and National Key Research and Development Program of China under Grant No. 2017YFC0805310.

Appendix: Proof of Theorem 4

We first show that $\widetilde{d}(\rho) < \infty$. Since $\widetilde{f}_\lambda(x)$ is convex, $\widetilde{\mathscr{X}}(\rho)$ is convex and $\mathscr{X}^\star \subseteq \widetilde{\mathscr{X}}(\rho)$ for any $\rho > 0$. One can verify that $\widetilde{\mathscr{X}}(\rho) - \mathscr{X}^\star$ is bounded. If $\widetilde{\mathscr{X}}(\rho) - \mathscr{X}^\star$ is empty, then $\widetilde{d}(\rho) = 0$, otherwise $0 \leq \widetilde{d}(\rho) = \max_{x \in \widetilde{\mathscr{X}}(\rho)} \min_{x^\star \in \mathscr{X}^\star} |x - x^\star| = \max_{x \in \widetilde{\mathscr{X}}(\rho) - \mathscr{X}^\star} \min_{x^\star \in \mathscr{X}^\star} |x - x^\star| < \infty$.

Then, we claim the following.

Claim 1: If $\|\mathbf{x}^k - x^\star \mathbb{1}\| > c_\rho$ for all $x^\star \in \mathscr{X}^\star$, then $\widetilde{f}_\lambda(\mathbf{x}^k) - f^\star > \rho c_a^2/2$.

Recall from (15) that

$$\widetilde{f}_\lambda(\mathbf{x}^k) - f^\star \geq f(\bar{x}^k) - f^\star + (\lambda a_{\min}^{(l)} - \frac{1}{2}cn)v(\mathbf{x}^k), \forall k.$$

This implies that if either $f(\bar{x}^k) - f^\star > \rho c_a^2/2$ or $v(\mathbf{x}^k) > \frac{\rho c_a^2}{2\lambda a_{\min}^{(l)} - cn}$, then $\widetilde{f}_\lambda(\mathbf{x}^k) - f^\star > \rho c_a^2/2$. Let

$$c_\rho := 2\sqrt{n} \max\{\widetilde{d}(\rho), \frac{\rho c_a^2}{2\lambda a_{\min}^{(l)} - cn}\}.$$

Since

$$\begin{aligned} c_\rho &< \|\mathbf{x}^k - x^\star \mathbb{1}\| \leq \|\mathbf{x}^k - \bar{x}^k \mathbb{1}\| + \|\bar{x}^k \mathbb{1} - x^\star \mathbb{1}\| \\ &\leq \sqrt{n} v(\mathbf{x}^k) + \sqrt{n} |\bar{x}^k - x^\star| \end{aligned}$$

we obtain that $v(\mathbf{x}^k) > c_\rho/(2\sqrt{n}) \geq \frac{\rho c_a^2}{2\lambda a_{\min}^{(l)} - cn}$ or $|\bar{x}^k - x^\star| > c_\rho/(2\sqrt{n}) \geq \widetilde{d}(\rho)$. For the former case we have $\widetilde{f}_\lambda(\mathbf{x}^k) - f^\star > \rho c_a^2/2$. For the latter case, $\bar{x}^k \notin \widetilde{\mathscr{X}}(\rho)$, which by the definition of $\widetilde{\mathscr{X}}(\rho)$ implies $\widetilde{f}_\lambda(\mathbf{x}^k) - f^\star > \rho c_a^2/2$.

Claim 2: There is $x_0^\star \in \mathscr{X}^\star$ such that $\liminf_{k\to\infty} \|\mathbf{x}^k - x_0^\star \mathbb{1}\| \leq c_\rho$.

Otherwise, there exists $k_0 > 0$ such that

$$\|\mathbf{x}^k - x^\star \mathbb{1}\| > c_\rho, \forall x^\star \in \mathscr{X}^\star, \forall k > k_0.$$

By Claim 1, there exists some $\varepsilon > 0$ such that $\widetilde{f}_\lambda(\mathbf{x}^k) - f^\star > \rho c_a^2/2 + \varepsilon$ for all $k > k_0$. Together with (18), it yields that

$$\begin{aligned} \|\mathbf{x}^{k+1} - x^\star \mathbb{1}\|^2 &\leq \|\mathbf{x}^k - x^\star \mathbb{1}\|^2 - 2\rho(\widetilde{f}_\lambda(\mathbf{x}^k) - f^\star) + \rho^2 c_a^2 \\ &\leq \|\mathbf{x}^k - x^\star \mathbb{1}\|^2 - 2\rho(\frac{\rho c_a^2}{2} + \varepsilon) + \rho^2 c_a^2 \\ &= \|\mathbf{x}^k - x^\star \mathbb{1}\|^2 - 2\rho\varepsilon. \end{aligned} \tag{38}$$

Summing this relation implies that for all $k > k_0$,

$$\|\mathbf{x}^{k+1} - x^\star \mathbb{1}\|^2 \leq \|\mathbf{x}^{k_0} - x^\star \mathbb{1}\|^2 - 2(k + 1 - k_0)\rho\varepsilon,$$

which clearly cannot hold for a sufficiently large k. Thus, we have verified Claim 2.

Claim 3: There is $x^\star \in \mathscr{X}^\star$ such that $\limsup_{k\to\infty} \|\mathbf{x}^k - x^\star \mathbb{1}\| \leq c_\rho + \rho c_a$.

Otherwise, for *any* $x^\star \in \mathscr{X}^\star$, there must exist a subsequence $\{\mathbf{x}^k\}_{k\in\mathscr{K}}$ (which depends on $x^\star$) such that for all $k \in \mathscr{K}$,

$$\|\mathbf{x}^k - x^\star \mathbb{1}\| > c_\rho + \rho c_a. \tag{39}$$

Notice that the penalty function $h(\mathbf{x})$ can be represented as

$$h(\mathbf{x}) = \sum_{e=1}^{m} a_e |\mathbf{b}_e^\mathsf{T} \mathbf{x}|.$$

where a_e is the weight of edge e. The subdifferential of $h(\mathbf{x})$ is then given by

$$\partial h(\mathbf{x}) = \sum_{e=1}^{m} a_e \text{SGN}(\mathbf{b}_e^\mathsf{T} \mathbf{x}) \mathbf{b}_e = B A_e \text{SGN}(B^\mathsf{T} \mathbf{x}) \tag{40}$$

where $A_e = \text{diag}\{a_1, ..., a_m\}$. Then, it follows from (40) that

$$\begin{aligned} \|\mathbf{x}^{k+1} - x^\star \mathbb{1}\| &= \|\mathbf{x}^k - x^\star \mathbb{1} - \rho\lambda B A_e \text{sgn}(B^\mathsf{T} \mathbf{x}^k) - \rho \nabla g(\mathbf{x}^k)\| \\ &\leq \|\mathbf{x}^k - x^\star \mathbb{1}\| + \lambda\rho \|B A_e \text{sgn}(B^\mathsf{T} \mathbf{x}^k)\| + \rho \|\nabla g(\mathbf{x}^k)\| \\ &\leq \|\mathbf{x}^k - x^\star \mathbb{1}\| + \rho\sqrt{n}(\lambda \|A\|_\infty + c) \end{aligned}$$

$$= \|\mathbf{x}^k - x^\star \mathbb{1}\| + \rho c_a, \forall k$$

where the second inequality follows from

$$\begin{aligned}\|BA_e \text{sgn}(B^\mathsf{T}\mathbf{x}^k)\| &\leq \sqrt{n}\|BA_e \text{sgn}(B^\mathsf{T}\mathbf{x}^k)\|_\infty \\ &\leq \sqrt{n}\|BA_e\|_\infty \|\text{sgn}(B^\mathsf{T}\mathbf{x}^k)\|_\infty \\ &\leq \sqrt{n} \max_i \sum_{j=1}^n a_{ij} = \sqrt{n}\|A\|_\infty.\end{aligned}$$

Thus, we obtain that for all $k \in \mathscr{K}$,

$$\|\mathbf{x}^{k-1} - x^\star \mathbb{1}\| \geq \|\mathbf{x}^k - x^\star \mathbb{1}\| - \rho c_a > c_\rho. \tag{41}$$

By Claim 2, there must exist some $k_1 \in \mathscr{K}$ and $k_1 > k_0$ such that

$$\|\mathbf{x}^{k_1-1} - x_0^\star \mathbb{1}\| \leq c_\rho + \rho c_a.$$

Together with (41), it implies that

$$c_\rho < \|\mathbf{x}^{k_1-1} - x_0^\star \mathbb{1}\| \leq c_\rho + \rho c_a. \tag{42}$$

Hence, it follows from Claim 1 that $\widetilde{f}_\lambda(\mathbf{x}^{k_1-1}) - f^\star > \rho c_a^2/2$, which together with (38) and (42) yields that

$$\|\mathbf{x}^{k_1} - x_0^\star \mathbb{1}\| \leq \|\mathbf{x}^{k_1-1} - x_0^\star \mathbb{1}\| \leq c_\rho + \rho c_a. \tag{43}$$

Setting $x^\star = x_0^\star$ in (39), we have $\|\mathbf{x}^{k_1} - x_0^\star \mathbb{1}\| > c_\rho + \rho c_a$. This contradicts (43), and hence verifies Claim 3.

In view of (19), the proof is completed. □

References

1. Beck A, Sabach S (2015) Weiszfelds method: Old and new results. Journal of Optimization Theory and Applications 164(1):1–40
2. Bertsekas DP (2015) Convex Optimization Algorithms. Athena Scientific Belmont
3. Borkar VS (2008) Stochastic approximation: a dynamical systems viewpoint. Baptism's 91 Witnesses
4. Cevher V, Becker S, Schmidt M (2014) Convex optimization for big data: Scalable, randomized, and parallel algorithms for big data analytics. IEEE Signal Processing Magazine 31(5):32–43
5. Chen G, Lewis FL, Xie L (2011) Finite-time distributed consensus via binary control protocols. Automatica 47(9):1962–1968
6. Clarke FH, Ledyaev YS, Stern RJ, Wolenski PR (2008) Nonsmooth Analysis and Control Theory, vol 178. Springer Science & Business Media

7. Cohen MB, Lee YT, Miller G, Pachocki J, Sidford A (2016) Geometric median in nearly linear time. In: Proceedings of the forty-eighth annual ACM symposium on Theory of Computing, ACM, pp 9–21
8. Deo N (1974) Graph Theory with Applications to Engineering and Computer Science. Courier Dover Publications
9. Franceschelli M, Giua A, Pisano A (2017) Finite-time consensus on the median value with robustness properties. IEEE Transactions on Automatic Control 62(4):1652–1667
10. Kan Z, Shea JM, Dixon WE (2016) Leader–follower containment control over directed random graphs. Automatica 66:56–62
11. Li T, Fu M, Xie L, Zhang J (2011) Distributed consensus with limited communication data rate. IEEE Transactions on Automatic Control 56(2):279–292
12. Lin P, Ren W, Farrell JA (2017) Distributed continuous-time optimization: nonuniform gradient gains, finite-time convergence, and convex constraint set. IEEE Transactions on Automatic Control 62(5):2239–2253
13. Magnússon S, Enyioha C, Li N, Fischione C, Tarokh V (2017) Convergence of limited communications gradient methods. IEEE Transactions on Automatic Control 63(5):1356–1371
14. Mokhtari A, Ling Q, Ribeiro A (2017) Network Newton distributed optimization methods. IEEE Transactions on Signal Processing 65(1):146–161
15. Nedić A, Olshevsky A (2015) Distributed optimization over time-varying directed graphs. IEEE Transactions on Automatic Control 60(3):601–615
16. Nedic A, Ozdaglar A (2009) Distributed subgradient methods for multi-agent optimization. IEEE Transactions on Automatic Control 54(1):48–61
17. Nedić A, Olshevsky A, Rabbat MG (2017) Network topology and communication-computation tradeoffs in decentralized optimization. In: Proceedings of the IEEE, vol 106, no 5, pp 953–976, May 2018
18. Olfati-Saber R, Murray RM (2004) Consensus problems in networks of agents with switching topology and time-delays. IEEE Transactions on Automatic Control 49(9):1520–1533
19. Pu Y, Zeilinger MN, Jones CN (2017) Quantization design for distributed optimization. IEEE Transactions on Automatic Control 62(5):2107–2120
20. Shi W, Ling Q, Wu G, Yin W (2015) Extra: An exact first-order algorithm for decentralized consensus optimization. SIAM Journal on Optimization 25(2):944–966
21. Wang H, Li C (2017) Distributed quantile regression over sensor networks. IEEE Transactions on Signal and Information Processing over Networks pp 1–1, 10.1109/TSIPN.2017.2699923
22. Xie P, You K, Tempo R, Song S, Wu C (2018) Distributed convex optimization with inequality constraints over time-varying unbalanced digraphs. IEEE Transactions on Automatic Control PP(99):1–1, 10.1109/TAC.2018.2816104
23. Yi P, Hong Y (2014) Quantized subgradient algorithm and data-rate analysis for distributed optimization. IEEE Transactions on Control of Network Systems 1(4):380–392
24. You K, Xie L (2011) Network topology and communication data rate for consensusability of discrete-time multi-agent systems. IEEE Transactions on Automatic Control 56(10):2262–2275
25. You K, Tempo R, Xie P (2018) Distributed algorithms for robust convex optimization via the scenario approach. IEEE Transactions on Automatic Control
26. Zhang J, You K (2018) Distributed optimization with binary relative information over deterministically time-varying graphs. To appear in the 57th IEEE Conference on Decision and Control, Miami Beach, FL, USA
27. Zhang J, You K, Başar T (2017) Distributed discrete-time optimization by exchanging one bit of information. In: 2018 annual American Control Conference (ACC), IEEE, pp 2065–2070
28. Zhang J, You K, Başar T (2018) Distributed discrete-time optimization in multi-agent networks using only sign of relative state. Accepted by IEEE Transactions on Automatic Control

Analysis of a Distributed Consensus Based Economic Dispatch Algorithm

Raghuraman Mudumbai, Soura Dasgupta and M. Muhammad Mahboob Ur Rahman

Abstract We present a consensus-based approach to the optimal economic dispatch of power generators in a smart microgrid. Under the proposed approach, generators independently make adjustments to their power frequency primary controller set points using three pieces of information: (a) their own marginal cost of generation, (b) the measured frequency deviation, and (c) marginal generation cost of a subset of other generators obtained using local message exchanges. We show that in the absence of power losses, these independent adjustments can be designed in such a way that frequency deviations are reduced to zero, and additionally, the overall cost of generation is minimized; that a slight modification to enforce harp power constraints on power generation achieves the same as long as the global optimum is within those bounds. When power losses are taken into account, our algorithm still reduces the frequency deviations to zero; however, in this case, the total cost of generation is only approximately minimized, though the resulting suboptimality can be shown to be negligible for typical levels of power losses. The proposed approach can be thought of as a gradient search of a carefully chosen objective function, whose minimization only requires the frequency deviation and message exchanges between neighboring generators. We prove that the algorithm uniformly converges to the global optimum and illustrate its performance using numerical simulations.

Rahman is with This work was partly supported by US NSF grants CAREER award ECCS-1150801, and CCF-1302456.

R. Mudumbai · S. Dasgupta (✉)
Department of Electrical and Computer Engineering, University of Iowa, Iowa City, IA 52242, USA
e-mail: dasgupta@engineering.uiowa.edu

R. Mudumbai
e-mail: rmudumbai@engineering.uiowa.edu

S. Dasgupta
Shandong Computer Science Center, Shandong Provincial Key Laboratory of Computer Networks, Jinan, China

M. Muhammad Mahboob Ur Rahman
Department of Electrical Engineering, Information Technology University, Lahore, Pakistan
e-mail: mahboob.rahman@itu.edu.pk

© Springer Nature Switzerland AG 2018

T. Başar (ed.), *Uncertainty in Complex Networked Systems*,
Systems & Control: Foundations & Applications,
https://doi.org/10.1007/978-3-030-04630-9_14

Keywords Consensus · Smart grid · Microgrids · Optimal dispatch · Constrains

1 Introduction

We present a simple, distributed *consensus based* algorithm for optimal economic dispatch [1], of power generators in a smart electric grid. To place the work in the context, we refer to our earlier papers [2, 3], which describe a distributed scheme where each generator adjusts its power frequency primary controller set points using only the measured frequency deviation, which eventually drives the set points to the optimal minimum cost allocations without any explicit communication between the generators. However, since the only information used by the generators is the frequency deviation, the resulting algorithm requires persistent nonzero frequency deviations in order to progress towards optimality. In practice, if initialized far from optimal allocations, the convergence of the algorithm in [2, 3] can be slow. This chapter is based on the idea that additional information exchange between the generators can significantly speedup the progress to optimality.

Accordingly, in this chapter, we consider a grid where the generators are connected by a communication network that forms a connected, potentially undirected, graph and are able to exchange information over this network regarding their marginal cost of generation with their neighbors. The generators independently make adjustments to their power frequency primary controller set points using three pieces of information: (a) their own marginal cost of generation, (b) the measured frequency deviation, and (c) marginal generation cost of a subset of other generators obtained using local message exchanges. The algorithm we propose globally uniformly asymptotically erases the frequency deviation, while achieving near-optimal allocations, under mild assumptions. More precisely, in the absence of power losses, the proposed algorithm achieves the exact minimum cost if the max and min limits on the generators are inactive, and achieves a cost arbitrarily close to the minimum when max and min limit constraints are active. With typical levels of power losses, the algorithm achieves near-optimal allocations. These properties are retained by a simple modification when there are hard bounds on the power outputs of each generator, as long as the optimum is within these bounds.

1.1 Motivation

Just like our previous work [3], this work is motivated by the anticipated needs of the next-generation electric grid which is expected to have smart consumer end nodes [4] and alternative energy generators. Thus while in the traditional electric grid, only a small number of large generation units are dispatchable, the future smart grid potentially involves a large number of small distributed generation (DG) [6], storage and demand response units that can all contribute to dispatch; integrating these units is expected to be crucial to accommodate a high penetration of intermittent [5] alternative energy generators.

Under this vision of the next-generation grid, a centralized control approach does not scale: the communication and computational overheads associated with learning and controlling the real-time state of a large number of dispatch units in a highly intermittent setting would be prohibitive. Thus a decentralized approach with a limited use of communication infrastructure and using local message exchanges is very attractive, and there is now a growing body of research into such decentralized methods (see e.g., [13–15]). In this chapter, we describe a novel distributed solution to the dispatch problem that leverages techniques from the distributed consensus literature. While we consider the dispatch problem in this chapter, we believe the distributed consensus approach provides a powerful set of *tools for other control loops in the electric grid such as reactive power control, voltage regulation, and so on.*

For convenience, we adopt the terminology of the traditional economic dispatch problem in this chapter, thus we use the term "generators" to represent all dispatchable units. We do assume that all dispatchable units have primary controllers that follow a power frequency droop characteristic with negative slope, just like traditional generators. While the droop curve was originally conceived to ensure stable interconnection of synchronous generators [33], and modern generators with power electronic converters do not necessarily require such a relationship in their primary controllers, recent studies [34] have shown that the droop curve is useful and effective and it is advantageous to retain this mechanism even for modern microgrids.

1.2 Background

In a traditional electric grid, control of generators is accomplished on multiple time scales using multiple different mechanisms [7]. Primary control is implemented in a distributed fashion at the generators, but secondary and tertiary control (corresponding to load frequency control (LFC) and economic dispatch (ED), respectively) are implemented from a centralized control station at the transmission system operator (TSO) and load serving entity (LSE) [8]. While the primary control operates over short time scales of up to 30 s [32], secondary and tertiary control operate over longer time scales on the order of minutes. Traditionally, an *ad hoc* allocation is used by the secondary controller to return ACE to zero without consideration of cost minimization; the latter function is the responsibility of the tertiary control process or economic dispatch (ED). The economic dispatch process periodically re-allocates the total generation power among generators to minimize total cost; the power allocations once set by the dispatch algorithm may over time deviate from their optimal values because of cumulative load fluctuations and the actions of the secondary controller until it is again optimally re-allocated by the dispatch process.

Recent studies [9] have shown that Centralized Security Constrained Economic Dispatch (SCED) methods [10] are effective for dispatch in a large-scale grid while taking into account congestion management, electricity market operations, transmission line constraints and so on. We would like to emphasize that *the distributed approach presented in this chapter is not intended as an alternative to these methods*. Rather, we envision a process by which small autonomous "microgrids" with smart grid capabilities are gradually integrated into the larger electric grid; our work is targeted towards the needs of such microgrids, specifically, ones with distributed generation, storage and/or demand response needs that are unsuitable for centralized management. The control systems of microgrids are carefully decoupled from, and operate alongside, the corresponding control systems in the larger grid. The concept of a microgrid is described in detail in [11].

It is important to emphasize that our proposed technique of distributed dispatch combines the functions of load frequency control and economic dispatch; however, it retains the same relationship between the primary and secondary control as in the traditional grid. Specifically, the dispatch algorithm operates at a slower time scale (min) as compared to the primary control which typically operates over time scales of up to 30 s or so. The dispatch algorithm does not directly change the power generated by each unit—doing so might cause instability in the form of frequency hunting, see e.g., [32]; rather our dispatch algorithm changes the *power frequency set point* on the droop curve of each unit, which indirectly changes the power generated by each unit through the operation of the primary controller.

Thus, for example, suppose that there is a small increase in load in a system that is currently at its optimal operating point (minimum cost allocations across its generators and zero frequency deviation). The immediate effect of the additional load would be to activate the AGC loops in the generators which would together increase their generations to match the extra load; this process typically occurs over a period of several seconds, and at the end of this process, after the transients have died down and steady-state is achieved, there is a small negative frequency deviation on the grid proportional to the extra load power. Note that with the increased load, the original allocations among the generators is no longer optimal. Our algorithm is designed to work with the frequency deviation to restore optimality. In this example, on the next iteration of the dispatch algorithm, each of the generators will make a small adjustment to their *power frequency set point* that will have the effect of reducing the steady-state frequency deviation. The cumulative effect of several iterations will be to drive the frequency deviation to zero while reallocating the power among the generators to achieve minimum cost.

1.3 Relation to Previous Work

Economic dispatch is a classical and very well-studied problem in the literature on power systems. Traditionally economic dispatch has been formulated and implemented as a multivariable constrained optimization problem [1]. A centralized network control center with detailed knowledge of cost curves of each generator along with models of line losses and constraints for the whole grid periodically calculates the optimal allocation of power across all the generating units, and then disseminates these allocations to the generators [9, 10]. Earlier academic work on this problem frequently use complex numerical optimization methods such as neural networks, particle swarm optimization or Monte Carlo methods [31, 38] to find the optimal solution.

Recently, the growing interest in "smart grid" technologies [12] has driven a surge of interest in distributed approaches to various optimization and control functions in the electric grid including secondary frequency and voltage control [16], adaptive scheduling [17], optimal power flow [18] and other related topics [19].

There is also previous work on distributed control methods for dispatch [13–15]; however, this chapter contains several novel features compared to this previous work as we now detail. In [13] the nodes exchange and update their estimates of the optimal marginal costs; however this procedure needs a globally consistent initial estimate of the generator allocations that add up to a sum that equals the total load as does the approach proposed in [15]. The authors in [13, 15] do not specify how this information on the load power is obtained by the entities participating in the dispatch process. By contrast, each node in our algorithm only needs to locally measure the frequency deviation from which the power imbalance is inferred using the power frequency droop curve.

The major difference between the work reported here and each of [13–15], *however is that they require the generator cost functions to be convex and quadratic.* These algorithms simply cannot be implemented without this property, as they critically require the closed-form relationship between the marginal cost and the power value, that is available for quadratic functions, but not necessarily for more general functions. *By contrast our algorithm permits arbitrary convex cost functions.* Unlike the general optimization framework in [19], which requires local optimization at each node augmented by message passing, we focus on the specific problem of dispatch and are able to take advantage of the structure of the problem to obtain a much simpler algorithm.

The work presented here also represents a significant advance over our own previous work of [2, 3]. This previous work is based on the following simple idea. When there is a positive power imbalance (i.e., instantaneous load exceeds rated generation), it is intuitively reasonable for a generator with a lower marginal cost to increase its generation by a larger amount than one with a higher marginal cost. The key limitation of this approach is that since the generators move slowly and incrementally towards the optimal allocations in response to small frequency deviations, it can take a long time to achieve optimality. This new work improves on [2, 3] by ensuring that the generators keep making adjustments to their generation even when there are no power imbalances on the grid. They do so by taking advantage of a separate communication infrastructure. In other words, in addition to the implicit signal available from the frequency deviation, we now assume that there is a communication network that allows each generator to directly query its neighbors regarding their instantaneous marginal costs and make adjustments accordingly.

Note, however, our assumptions on the communication network are minimal: We do not require that each generator talk to all other generators or that communication be bidirectional. Nor do we assume that there is any central authority to coordinate all the generators. We only require that the communication links between the generators form a connected graph. While this work has clear parallels to the rich literature on consensus and multi-agent theory, [21–27], there are notable distinctive features. In particular, should we apply traditional consensus algorithms the result would be the equalization of the marginal costs, without necessarily erasing the load imbalance. Our approach on the other hand can be viewed as a constrained consensus technique where consensus of marginal costs is achieved subject to zero load imbalance. A preliminary version of this work was presented in [30]. A related argument was studied in [29], but the proof given assumed a continuous time update. The discrete time analysis here is significantly harder. Neither [30] nor [29] tackle constraints.

The rest of this chapter is organized as follows. We first introduce the dispatch problem considered in this chapter in Sect. 2 and our proposed distributed algorithm is presented in Sect. 3. Section 4 has the stability analysis. Section 5 addresses issues of hard bounds and power losses. We illustrate the performance of the algorithm using simulations in Sect. 6 and conclude in Sect. 7.

2 The Dispatch Problem

We model the economic dispatch problem as follows. We assume that there are N generators supplying power to the network. We denote the total power consumed by P_L and the active power set point for generator i at the rated system frequency by $P_i(k),\ i \in 1 \ldots N$. As a result, the power imbalance in the system is given by

$$\varDelta(k) = P_L - \sum_{i=1}^{N} P_i(k) \tag{1}$$

The $P_i(k)$ represents the active power *set point*; the actual active power produced by each generator is determined by its primary controller which uses $P_i(k)$ as a reference. More precisely, the primary controller on each generator responds to a power deficit (or surplus) $\varDelta(k)$ by increasing (or decreasing) its generated power above (or below) its generation set point $P_i(k)$ until the total generated power matches the total load. This action by the controller has the side effect of introducing a small frequency deviation that is proportional to the original imbalance $\varDelta(k)$.

In other words, the total imbalance between the rated generation power and the load, after the controllers have reached steady-state, results in a proportional frequency deviation $\varDelta f(k) = \beta \varDelta(k)$ on the grid. This frequency deviation can be monitored continuously by each generator which thus can directly monitor the power imbalance $\varDelta(k)$. This is analogous to the area control error (ACE) signal observed by the secondary controller in a traditional load frequency control (LFC) implementation [28]. We assume that β remains constant for all values of $P_i(k)$ and $\varDelta(k)$. This is a reasonable assumption for small frequency deviations.

Our analysis assumes that the load P_L is constant and $J_i(P_i)$ the cost function for generator i, is strictly convex, an assumption that is standard in the power systems literature. With $P = [P_1, \dots, P_N]^\top$, define P_i^* to be power allocations that minimize the total cost:

$$\begin{aligned} &\textstyle\sum_{i\in V} J_i(P_i) \\ \text{subject to } &\textstyle\sum_{i\in V} P_i = P_L,\ 0 < p_i^- \le P_i \le p_i^+. \end{aligned} \tag{2}$$

Observe that as is often the case in practice the formulation also restricts the power generation of each generator. In the sequel, we will call the problem *without generation limits (WGL)* for all $i \in V$, the constraint

$$p_i^- \le P_i \le p_i^+ \tag{3}$$

is removed.

Assumption 2.1 The optimum solutions of the WGL problem and the constrained problem (2) are identical.

In Sect. 3 we provide a distributed algorithm that achieves

$$\lim_{k\to\infty} P_i(k) = P_i^*, \tag{4}$$

without respecting (3). Subsequently we will provide a simple modification that achieves (4) while respecting (3), as long as Assumption 2.1 holds.

3 The Algorithm for WGL

We assume that the network of generators form an underlying, *possibly directed graph* $G = (V, E)$, where $V = \{1, \dots, N\}$ is the vertex set comprising the generators. The directed edge $\{i, j\} \in E$ if generator i has access to generator j's marginal cost $J_j'(P_j)$, where P_j is the power generated by agent j. In the sequel $\mathscr{N}(i)$ represents the set of neighbors of i. In particular:

$$\mathscr{N}(i) = \{j \,|\{i, j\} \in E\}. \tag{5}$$

In the sequel we assume a constant load P_L, and a load deficit:

$$\varDelta = P_L - \sum_{i=1}^{N} P_i. \tag{6}$$

The following constitutes a standing assumption of this chapter.

Assumption 3.1 The load P_L is constant. For all $i \in V$, the cost $J_i(\cdot) : \mathbb{R} \to \mathbb{R}^+$ is analytic everywhere and strictly convex and the marginal cost $J_i'(\cdot)$ positive. In particular for all $x \in \mathbb{R}$, and $i \in V$, there holds,

$$J_i''(x) > 0. \tag{7}$$

Further, on every compact subset $S \subset \mathbb{R}^N$, and all i, J_i and all its derivatives are uniformly bounded and there exists $\omega(S) > 0$ such that

$$J_i''(x) > \omega(S),\ \forall x \in S. \tag{8}$$

Finally, for every $i \in V$

$$\lim_{P_i \to \infty} J_i'(P_i) = \infty. \tag{9}$$

We now comment on these assumptions. Observe, (8) is a convexity assumption on the generator costs. This is standard for most cost functions used in the power systems literature. Sometimes, these can be manufactured by interpolating tabulated data. Such data are themselves of a form that permits a convex interpolant, [20]. The convexity condition of course implicitly reflects the appealing reality that the marginal cost increases with production. Likewise, (9) is in accord with intuition. The marginal cost is unbounded and positive. Observe, our conditions require the marginal costs to be always positive, which is again a reality. For technical reasons, we have not restricted the P_i to be nonnegative, though, in reality they would be. We comment more on this fact after presenting the technical results in the next section.

The next Lemma shows that the solution to (2) is unique.

Theorem 1 *Under Assumptions 2.1 and 3.1, the solution to (2) is unique and obeys*

$$J_i'(P_i^*) = J_j'(P_j^*),\ \forall \{i, j\} \subset V. \tag{10}$$

Proof As under Assumption 2.1 the solution to (2) is identical to the the WGL problem we focus on the latter where (3) does not hold. In this case (10) under

$$P_L = \sum_{i \in V} P_i^*. \tag{11}$$

is a necessary condition for optimality. We now assert that P_i^* that satisfy (10) and (11) are unique. Indeed if a second set of P_i obey

$$P_L = \sum_{i \in V} P_i$$

then for some i, j, $P_i > P_i^*$, $P_j < P_j^*$. Then from Assumption 3.1,

$$J'(P_i) > J'(P_i^*) = J'(P_j^*) > J'(P_j^*).$$

Thus this set of P_i cannot be optimizing.

Thus, effectively we must find P_i that equalize the marginals subject to (2). Equalization of the marginals through their local exchange has similarities to the goals of consensus algorithms, [22]. A defining departure is in the additional requirement of (2). To achieve this modified objective, we resort to the gradient descent optimization of an alternative cost function.

Specifically, with $P(k) = [P_1(k), \ldots, P_N(k)]^\top : \mathbb{Z} \rightarrow \mathbb{R}^N$, define the auxilliary cost, for some $\alpha > 0$,

$$J(P) = \alpha \Delta^2 + \sum_{\{i,j\}\in E} \left(J_i'(P_i) - J_j'(P_j)\right)^2 . \tag{12}$$

Minimization of $J(P)$ achieves (2) and the equalization of at least the marginal costs of neighbors. Connectedness ensures the equalization of all marginals.

Then our algorithm is the following gradient descent law: for $\mu > 0$,

$$P_i(k+1) = P_i(k) - \mu \left.\frac{\partial J(P)}{\partial P_i}\right|_{P=P(k)} . \tag{13}$$

Observe,

$$\frac{\partial J(P)}{\partial P_i} = -2\alpha\Delta + 2J_i''(P_i) \sum_{j\in\mathcal{N}(i)} \left(J_i'(P_i) - J_j'(P_j)\right) . \tag{14}$$

Thus, to implement its update law the ith generator needs to know its neighbors' marginal costs, and a quantity proportional to the load deficit that as noted earlier is supplied by local measurement of the frequency deviation. This law does not account for the constraint (3). In Sect. 5, we address modifications that address (3).

4 Stability

We begin with a fairly general result on gradient descent the proof of which requires two additional results. The first is the multivariable Taylor's theorem, [35]. A clean proof of this result can be found in [36].

Theorem 2 *Consider an analytic function $f : \mathbb{R}^N \rightarrow \mathbb{R}$. For $x, h \in \mathbb{R}^N$. Define the Hessian, $H_f : \mathbb{R}^N \rightarrow \mathbb{R}^{N\times N}$, whose ijth element obeys*

$$\left[H_f(x)\right]_{ij} = \frac{\partial^2 f(x)}{\partial x_i \partial x_j}$$

where x_i is the ith element of x. Then there exists $R_2(x, h)$ with the property that

$$\lim_{h\to 0} \frac{R_2(x,h)}{\|h\|^2} = 0, \tag{15}$$

for which

$$f(x+h) = f(x) + h^\top \frac{\partial f(x)}{\partial x} + h^\top H(x)h + R_2(x,h).$$

The second result is a consequence of Lasalle's invariance principle, a recent proof of which is in [37].

Theorem 3 *Consider the state equation*

$$\rho[k+1] = g(\rho[k]), \quad \forall k \geq k_0 \tag{16}$$

where k and k_0 are integers, and $g(\rho[k])$ has no explicit dependence on k. Suppose the following conditions hold:

(a) $\rho[k]$ is uniformly bounded for every finite $\rho[k_0]$.
(b) There exists a nonnegative function $f(\rho[k])$ such that the following holds for all $k \geq k_0$ along the trajectories of (16):

$$f(\rho[k+1]) \leq f(\rho[k]) \tag{17}$$

(c) For all finite $\rho[k_0]$, $f(\rho[k])$ is uniformly bounded.

Then $\rho[k]$ uniformly converges to a trajectory of (16) on which $f(\rho[k])$ is a constant.

Now the promised result on gradient descent.

Lemma 1 *Consider an analytic, nonnegative function $f(\cdot) : \mathbb{R}^N \to \mathbb{R}_+$ and the algorithm for $k \geq k_0$:*

$$x(k+1) = x(k) - \mu \left.\frac{\partial f(x)}{\partial x}\right|_{x=x(k)}, \quad x(k_0) = x_0. \tag{18}$$

Suppose for every $f_0 > 0$, the set $S(f_0) = \left\{x \in \mathbb{R}^N \mid f(x) \leq f_0\right\}$ is compact and each derivative of f in this compact set is uniformly bounded. Then there exists a μ^* such that for every $0 < \mu < \mu^*$, there exists $a(\mu) > 0$ such that for all $k > k_0$, there holds:

$$f(x(k+1)) \leq f(x(k)) - a(\mu) \left\| \left.\frac{\partial f(x)}{\partial x}\right|_{x=x(k)} \right\|^2. \tag{19}$$

Further, for every x_0, there exists a μ^*, such that for all $0 < \mu < \mu^*$, the following holds, uniformly in k_0,

$$\lim_{k\to\infty} \left.\frac{\partial f(x)}{\partial x}\right|_{x=x(k)} = 0.$$

Proof From Theorem 2, in the notation of Theorem 2, there holds for suitable

$$C\left(x, \mu\, \frac{\partial f(x)}{\partial x}\bigg|_{x=x(k)}\right)$$

$$\begin{aligned} f(x(k+1)) &= f\left(x(k) - \mu\, \frac{\partial f(x)}{\partial x}\bigg|_{x=x(k)}\right) \\ &= f(x[k]) - \mu \left\| \frac{\partial f(x)}{\partial x}\bigg|_{x=x(k)} \right\|^2 \\ &+ \mu^2\, \frac{\partial f(x)}{\partial x}\bigg|_{x=x(k)}^{\top} H_f(x(k))\, \frac{\partial f(x)}{\partial x}\bigg|_{x=x(k)} \\ &+ C\left(x, \mu\, \frac{\partial f(x)}{\partial x}\bigg|_{x=x(k)}\right) \end{aligned}$$

Because of Theorem 2 and the assumptions embedded in the hypothesis of this lemma, there is an M, such that

$$\begin{aligned} &\frac{\partial f(x)}{\partial x}\bigg|_{x=x(k)}^{\top} H_f(x(k)) \quad \frac{\partial f(x)}{\partial x}\bigg|_{x=x(k)} \\ &\qquad\qquad \le M \left\| \frac{\partial f(x)}{\partial x}\bigg|_{x=x(k)} \right\|^2 \end{aligned}$$

and

$$\lim_{\mu \to 0} \frac{C\left(x, \mu\, \frac{\partial f(x)}{\partial x}\Big|_{x=x(k)}\right)}{\mu^2 \left\| \frac{\partial f(x)}{\partial x}\Big|_{x=x(k)} \right\|^2} = 0. \tag{20}$$

Consequently, there holds:

$$\begin{aligned} f(x[k+1]) &\le f(x[k]) - \mu \left\| \frac{\partial f(x)}{\partial x}\bigg|_{x=x(k)} \right\|^2 + \mu^2 M \left\| \frac{\partial f(x)}{\partial x}\bigg|_{x=x(k)} \right\|^2 + C\left(x, \mu\, \frac{\partial f(x)}{\partial x}\bigg|_{x=x(k)}\right) \\ &= f(x[k]) - \mu \left\| \frac{\partial f(x)}{\partial x}\bigg|_{x=x(k)} \right\|^2 \mu^2 M \left\| \frac{\partial f(x)}{\partial x}\bigg|_{x=x(k)} \right\|^2 \left(1 + \frac{C\left(x, \mu\, \frac{\partial f(x)}{\partial x}\Big|_{x=x(k)}\right)}{\mu^2 \left\| \frac{\partial f(x)}{\partial x}\Big|_{x=x(k)} \right\|^2} \right) \end{aligned}$$

Thus, because of (20) there exists a μ^* such that for every $0 < \mu < \mu^*$, there exists $a(\mu) > 0$ such that for all $k > k_0$, (19) holds. As $f(.)$ is nonnegative, $f(x(k))$ is bounded. Thus, as $S(f_0)$ is compact for all finite f_0, for every x_0, $x(k)$ is uniformly bounded. The result follows from Theorem 3.

Thus, under the right conditions, gradient descent forces the gradient to converge to zero. The next Lemma shows that our auxiliary cost satisfies the conditions imposed on $f(\cdot)$ in Lemma 1.

Lemma 2 *With the various quantities as defined above suppose Assumption 3.1 holds and the graph G is connected. Under (12) consider for $B > 0$, the set:*

$$\Omega_B = \left\{P \in \mathbb{R}^N \,|J(P) \leq B\right\}. \tag{21}$$

Then Ω_B is compact.

Proof Clearly Ω_B is closed. It remains to show that it is bounded. To establish a contradiction, suppose it is unbounded. Then for every M, there exists an i and $P_i \in \Omega_B$, such that $|P_i| > M$. Consider such an i and M. Observe

$$|\Delta| \leq \frac{B}{\sqrt{\alpha}}.$$

From (6), there exists $j \in V$ such that:

$$|P_j| \geq \frac{M}{N-1} - \frac{B}{\sqrt{\alpha}(N-1)} \tag{22}$$

and P_j and P_i have opposite signs. Thus, from (9) and (8), by choosing M arbitrarily large one can make $|J_i'(P_i) - J_j'(P_j)|$ arbitrarily large. Now, because of (21) and (12) for all $\{k, l\} \in E$

$$|J_k'(P_k) - J_l'(P_l)| \leq B$$

As G is connected, for all $i, j \in V$, and there are N nodes in the network, there are at most $N-1$ hops in a path from any node i to any node j. Thus for aribitrary i, j:

$$|J_j'(P_j) - J_i'(P_i)| \leq (N-1)B$$

establishing a contradiction.

Taken together, Lemmas 1 and 2 show that (13) forces:

$$\lim_{k\to\infty} \left.\frac{\partial J(P)}{\partial P}\right|_{P=P(k)} = 0.$$

To show that this ensures that $J(P)$ is minimized, we next expose a property of the gradient of $J(\cdot)$.

Lemma 3 *With the various quantitities defined as above, consider (14). Then*

$$\frac{\partial J(P)}{\partial P_i} = 0$$

iff

$$\varDelta = 0$$

and for all $i \in V$

$$\sum_{\{i,j\}\in E} \left(J_i'(P_i) - J_j'(P_j)\right) = 0.$$

Proof Observe that (8) and (14) ensure that the result clearly holds when $\varDelta = 0$. To establish a contradiction suppose first that $\varDelta < 0$. Choose any $i^* \in V$ such that:

$$i^* = \arg\min_{i\in V} \left\{J_i'(P_i)\right\}.$$

Note i^* need not be unique. Consider (14) with $i = i^*$. Then the convexity part of Assumption 3.1 ensures that:

$$J_{i^*}''(P_{i^*}) \sum_{j\in\mathscr{N}(i^*)} \left(J_{i^*}'(P_{i^*}) - J_j'(P_j)\right) \le 0.$$

Consequently, $\varDelta = 0$. Likewise if $\varDelta > 0$. Choose any $i^* \in V$ such that:

$$i^* = \arg\max_{i\in V} \left\{J_i'(P_i)\right\}.$$

Then again $\varDelta = 0$ as

$$\sum_{j\in\mathscr{N}(i^*)} \left(J_{i^*}'(P_{i^*}) - J_j'(P_j)\right) \ge 0,$$

completing the proof.

This brings us to the main result of this section.

Theorem 4 *With the various quantitities defined above, consider (12) under Assumption 3.1 and with G connected. Define $P^* = [P_1^*, P_2^*, \ldots, P_N^*]^\top$ whose elements satisfy Theorem 1. Then for every $P(0)$ there exists a μ^*, such that for all $0 < \mu \le \mu^*$, under (13) there holds uniformly:*

$$\lim_{k\to\infty} P(k) = P^*.$$

Proof From Lemma 2 and Assumption 3.1 that $J(\cdot)$ satisfies the conditions imposed on $f(\cdot)$ in Lemma 1. Then from Lemma 1

$$\lim_{k\to\infty} \left.\frac{\partial J(P)}{\partial P_i}\right|_{P=P(k)} = 0.$$

Because of Lemma 3 this means that

$$\lim_{k\to\infty} \Delta(k) = 0,$$

and for all $i \in V$ and $j \in \mathscr{N}(i)$

$$\lim_{k\to\infty} \left(J_j'(P_j(k)) - J_i'(P_i(k))\right) = 0. \tag{23}$$

As G is connected this means (23) holds *for all* $\{i, j\} \subset V$. Then the result follows from arguments used in the proof Theorem 1. Uniformity is a consequence of the autonomous nature of (13).

5 Incorporation of (3) and Consequences of Power Loss

Observe as presented, the $P_i(k)$ generated by (13) need not respect (3). We now present an additional reasonable assumption and argue that subject to its satisfaction a simple fix suffices to ensure the satisfaction of (3).

Assumption 5.1 In (3), there exists $\delta > 0$ such that the following holds for all $\{i, j\} \subset V$,

$$J_i'(p_i^-) + \delta < J_j'(p_j^+), \tag{24}$$

and

$$p_i^- < p_j^+. \tag{25}$$

Thus the marginal costs at the smallest allowable power are uniformly over bounded by those at the largest. Then we argue that the following algorithm converges to the optimum.

$$P_i(k+1) = \begin{cases} P_i(k) - \mu \left.\frac{\partial J(P)}{\partial P_i}\right|_{P=P(k)} & p_i^- \le P_i(k+1) \le p_i^+ \\ P_i(k) & \text{else} \end{cases} \tag{26}$$

Thus this algorithm employs the simplest possible projection: Do not move if you violate constrains. Then we have the following theorem.

Theorem 5 *Under the conditions of Theorem 4, suppose Assumptions 2.1, 5.1 and*

$$p_i^- \le P_i(0) \le p_i^+$$

hold. Then there exists $\mu^* > 0$ such that for all $0 < \mu \le \mu^*$ so does (4).

Proof We first prove that there exists no k such that

$$\psi_i(k) = P_i(k) - \mu \left. \frac{\partial J(P)}{\partial P_i} \right|_{P=P(k)} \notin [p_i^-, p_i^+], \ \forall i \in V \tag{27}$$

holds. To establish a contradiction suppose at some k (27) holds. Suppose first $\Delta(k) \ge 0$. Then because of Assumption 5.1 there must be an $i \in V$ such that $\psi_i(k) < p_i^-$, and because of Assumption 5.1, and sufficiently small $\mu > 0$, $J_i'(P_i(k)) < J_j'(P_j(k))$, for all $j \in \mathscr{N}(i)$. Then from (14), $\psi_i(k) > P_i(k) \ge p_i^-$. This establishes a contradiction. The $\Delta(k) < 0$ case is similarly handled. Thus (27) cannot hold. Now suppose $\bar{P}(k)$ comprise all $P_i(k)$ that violate (27). Then

$$\bar{P}_i(k+1) = \bar{P}_i(k) - \mu \left. \frac{\partial J(P)}{\partial \bar{P}_i} \right|_{P=\bar{P}(k)}.$$

Then (19) holds for sufficiently small μ with $f(\cdot) = J(\cdot)$. Then a slight variation of the arguments leading to the proof of Theorem 4 yields the result.

We now address the issue of power losses. Assume the total loss amounts to $P_{\text{loss}}(k)$. Then as shown in [3]

$$\Delta(k) = P_L - \sum_{i \in V} P_i(k) + P_{\text{loss}}(k).$$

In this case convergence still occurs to a point where

$$\Delta = 0 \text{ and } J_i'(P_i) = J_j'(P_j), \ \forall \{i, j\} \subset V. \tag{28}$$

This is however, no longer the optimum. However, as in [3] assume

$$L(P) = P_{\text{loss}}. \tag{29}$$

Note this permits a coupled dependence of the net loss on the P_i. As in [3], assume that

$$\gamma_i(P) = \frac{\partial L(P)}{\partial P_i} \le \gamma_0 < 1.$$

Observe γ_0 is the extra unit of power from generator i lost in the grid. Then as shown in [3], that the suboptimality induced by (28) is proportional to γ_0^2 and for small γ_0 is small.

6 Simulations

Simulations are conducted with 100 generating units to examine the scalability of this approach to larger grids, and to study the effect of power losses. To demonstrate the efficacy of the algorithm for more general non-quadratic cost functions, cubic cost functions are used.

A randomly generated adjacency graph models the communication links in the smart grid. This graph is generated by placing all the generators units uniformly and randomly in a square grid and connecting any two nodes within a threshold distance. This model is suitable, for instance, to model the communication links when the network comprises isotropic wireless links. Figure 1 shows the connectivity graph. In this graph, each node has on average 6 neighbors. The cubic cost functions have the form $J_i(P_i) = d_i P_i^3 + c_i P_i^2 + b_i P_i + a_i$. The parameters $d_i,\ c_i,\ b_i,\ a_i$ are chosen randomly (independent and identically distributed across the generators).

The first simulation of the 100 unit system is without any power losses. The parameter values for this simulation are $\mu = 10$ and $\alpha = 0.0001$. The results are shown in Fig. 2. Near-optimal cost is reached and power imbalance is removed within less than 50 iterations underscoring the scalability of the algorithm.

The next simulation studies the effect of power losses on this 100 unit system. The power losses in the grid is modeled using the popular model in the literature on dispatch (see e.g., [31]) as $P_{loss} = \sum_{i,j=1}^{N} BB(i,j)P_i P_j + \sum_{i=1}^{N} B(i)P_i$, where $BB(i,j)$ is a positive-semidefinite matrix and B a nonnegative vector of quadratic and linear loss terms respectively. The loss parameters $BB,\ B$ were chosen randomly to give power losses of approximately 1% in this simulation.

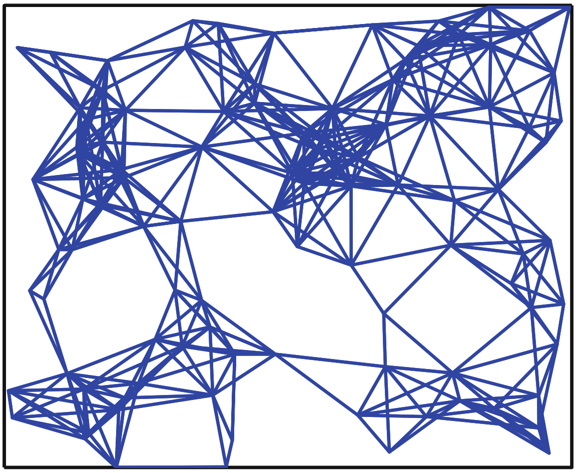


Fig. 1 Connectivity graph of a 100 node network

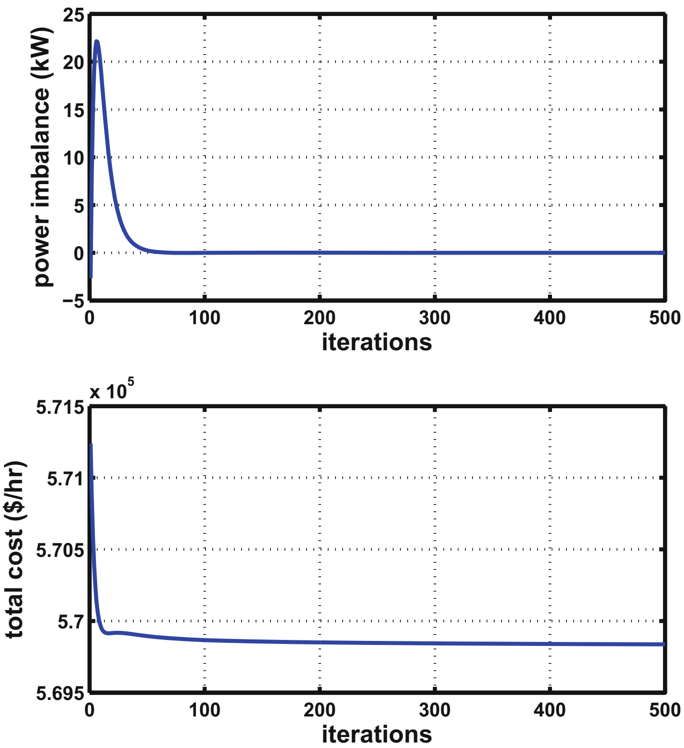


Fig. 2 Convergence of dispatch algorithm for a 100 node network

The simulation results in Fig. 3 show the convergence behavior of the algorithm for the 100 node system with losses. All parameters for the results in Fig. 3 are the same as for Fig. 2 except for the power losses. It can be seen that the total generation cost decreases to a steady-state value within 100 iterations and the algorithm is able to restore power balance even in the presence of power losses. Our analysis in [3] suggests, however, that the equilibrium point of this algorithm is no longer optimal in the presence of power losses, though the sub-optimality can be shown to be small for typical levels of power losses. We can see by comparing Figs. 2 and 3 that the steady-state total generation cost is slightly higher with losses that reflects the cost of generating additional power to compensate for the losses. Since the (equal marginal cost) solution to which our algorithm converges in the presence of losses is the same solution considered in [3], the analysis of losses presented in [3] extends also to the algorithm presented in this chapter.

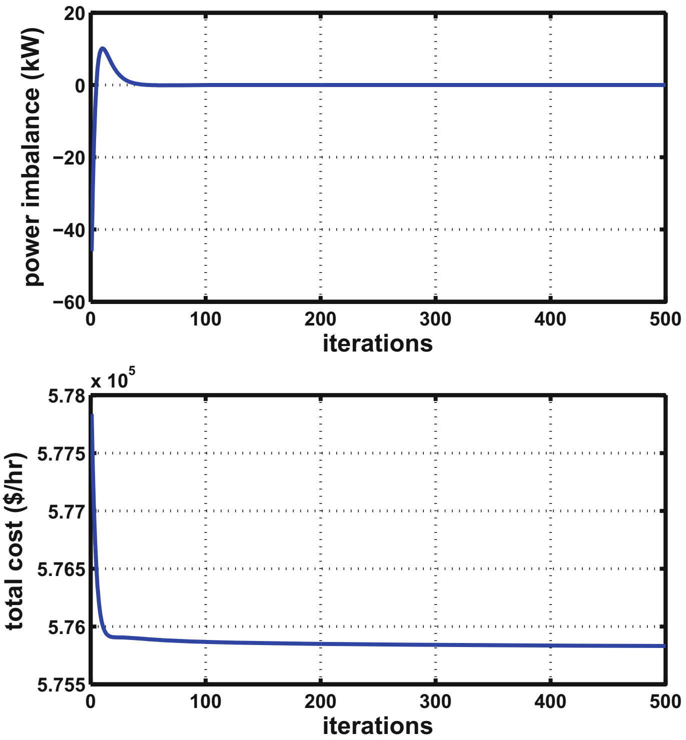


Fig. 3 Simulation of distributed dispatch algorithm with non-quadratic cost functions with power losses

7 Conclusion

We have presented a decentralized consensus-based algorithm for achieving optimal dispatch, that uniformly achieves optimal power allocation while meeting load requirements. Our algorithm relies on local frequency deviation measurements and on a connected network over which neighboring generators exchange their marginal costs. The underlying network graph is permitted to be directed.

Several areas of future research can be pursued. The algorithm as presented involves synchronous communication. Asynchronous communication, in the tradition of, [21] is worthy of exploration. Other kinds of information exchange that can further enhance the convergence rate of the algorithm is also an interesting topic for further study.

References

1. B. Chowdhury and S. Rahman, “A review of recent advances in economic dispatch,” *Power Systems, IEEE Transactions on*, pp. 1248–1259, 1990.
2. R. Mudumbai, S. Dasgupta and B. Cho, “Distributed control for optimal economic dispatch of power generators: the heterogenous case,” in *Proc. of the IEEE CDC*, 2011.
3. R. Mudumbai, S. Dasgupta and B. Cho, “Distributed control for optimal economic dispatch of a network of heterogeneous power generators,” *IEEE Trans. on Power Systems* pp 1750–1760, 2012.
4. S. Marvin, H. Chappells, and S. Guy, “Pathways of smart metering development: shaping environmental innovation,” *Computers, Environment and Urban Systems*, pp. 109–126, 1999.
5. D. Milborrow, “Penalties for intermittent sources of energy,” *Cabinet Office, London*, p. 17, 2001.
6. R. Dugan and T. McDermott, “Distributed generation,” *IEEE Industry Applications Magazine*, vol. 8, pp. 19–25, Mar/Apr 2002.
7. Y. G. Rebours, D. S. Kirschen, M. Trotignon, and S. Rossignol, “A survey of frequency and voltage control ancillary services part i: Technical features,” *IEEE Transactions on Power Systems*, vol. 22, pp. 350–357, Feb. 2007.
8. F. Wu, K. Moslehi, and A. Bose, “Power system control centers: past, present, and future,” *Proceedings of the IEEE*, vol. 93, no. 11, pp. 1890–1908, 2005.
9. U. S. D. of Energy, “Economic dispatch of electric generation capacity,” *A REPORT TO CONGRESS AND THE STATES*, 2007.
10. L. Vargas, V. Quintana, and A. Vannelli, “A tutorial description of an interior point method and its applications to security-constrained economic dispatch,” *Power Systems, IEEE Transactions on*, vol. 8, no. 3, pp. 1315–1324, 2002.
11. J. Lopes, C. Moreira, A. Madureira, F. Resende, X. Wu, N. Jayawarna, Y. Zhang, N. Jenkins, F. Kanellos, and N. Hatziargyriou, “Control strategies for microgrids emergency operation,” in *International Conference on Future Power Systems, Amsterdam, Netherlands*, 2005.
12. M. Amin and B. Wollenberg, “Toward a smart grid: power delivery for the 21st century,” *IEEE Power and Energy Magazine*, vol. 3, no. 5, pp. 34–41, 2005.
13. S. Kar and G. Hug, “Distributed robust economic dispatch in power systems: A consensus and innovations approach,” in *Power and Energy Society General Meeting, 2012 IEEE*, 2012.
14. Z. Zhang and M.-Y. Chow, “Convergence analysis of the incremental cost consensus algorithm under different communication network topologies in a smart grid,” *Power Systems, IEEE Transactions on*, vol. 27, no. 4, pp. 1761–1768, 2012.
15. S. Yang, S. Tan, and J.-X. Xu, “Consensus based approach for economic dispatch problem in a smart grid,” *Power Systems, IEEE Transactions on*, 2013.
16. A. Bidram, A. Davoudi, F. Lewis, and Z. Qu, “Secondary control of microgrids based on distributed cooperative control of multi-agent systems,” *Generation, Transmission Distribution, IET*, vol. 7, no. 8, pp. –, 2013.
17. M. Fathi and H. Bevrani, “Adaptive energy consumption scheduling for connected microgrids under demand uncertainty,” *Power Delivery, IEEE Transactions on*, pp. 1576–1583, 2013.
18. E. Dall’Anese, H. Zhu, and G. Giannakis, “Distributed optimal power flow for smart microgrids,” *Smart Grid, IEEE Transactions on*, vol. 4, no. 3, pp. 1464–1475, 2013.
19. M. Kraning, E. Chu, J. Lavaei, and S. Boyd, “Dynamic network energy management via proximal message passing,” *Optimization*, vol. 1, no. 2, pp. 1–54, 2013.
20. M. Aganagic and S. Mokhtari, “Security constrained economic dispatch using nonlinear dantzig-wolfe decomposition,” *Power Systems, IEEE Transactions on*, pp. 105–112, 1997.
21. L. Moreau, “Stability of multi-agent systems with time-dependent communication links”, *IEEE Trans. Autom. Control,* pp. 169–182, 2005.
22. R. Olfati-Saber and R. M. Murray, “Consensus problems in networks of agents with switching topology and time-delays”, *IEEE Trans. Autom. Control*, pp. 1520–1533, 2004.

23. S. Guler, B. Fidan, S. Dasgupta, B. D. O. Anderson and I. Shames "Adaptive Source Localization Based Station Keeping of Autonomous Vehicles", *IEEE Transactions on Automatic Control,* pp. 3122–3135, 2017.
24. B. Fidan, S. Dasgupta and B. D. O. Anderson, "Adaptive rangemeasurementbased target pursuit", *International Journal of Adaptive Control and Signal Processing,* pp. 66–81, 2013.
25. M. Cao, C. Yu, A. S. Morse, B. D. O. Anderson and S. Dasgupta, "Generalized controller for directed triangle formations", in *Proceedings of the IFAC World Congress*, vol. 41, pp. 6590–6595, 2008.
26. M. Fu, S. Dasgupta and Y. C. Soh, "Integral quadratic constraint approach vs. multiplier approach", *Automatica*, pp. 281–287, 2005.
27. M. Fu and S. Dasgupta, "Parametric Lyapunov functions for uncertain systems: The multiplier approach", *Advances in linear matrix inequality methods in control,* pp. 95–108, 2000.
28. Christie, R.D., and Bose, A., "Load frequency control issues in power system operations after deregulation", *IEEE Transactions on Power Systems,* pp. 1191–1200, 1996.
29. M. Basu, R. Mudumbai and S. Dasgupta, "Intelligent Distributed Economic Dispatch in Smart Grids", *Intelligent Systems Technologies and Applications*, pp. 285–295, S. Berretti, S. M. Thampi and S. Dasgupta Eds, Springer 2015.
30. R. Mudumbai, S. Dasgupta, and R. Mahboob, "A distributed consensus based algorithm for optimal dispatch in smart power grids," Proceedings of the 32nd IASTED International Conference on Modeling, Identification and Control (MIC), Feb 2013.
31. C. Chen, "Economic dispatch using simplified personal best oriented particle swarm optimizer," in *Electric Utility Deregulation and Restructuring and Power Technologies, 2008. DRPT 2008. Third International Conference on*. IEEE, 2008, pp. 572–576.
32. N. Jaleeli, L. VanSlyck, D. Ewart, L. Fink, and A. Hoffmann, "Understanding automatic generation control," *Power Systems, IEEE Transactions on*, vol. 7, no. 3, pp. 1106–1122, Aug 1992.
33. F. Clough, "Stability of large power systems," *Journal of the Institution of Electrical Engineers,* vol. 65, no. 367, pp. 653–659, 1927.
34. A. Dobakhshari, S. Azizi, and A. Ranjbar, "Control of microgrids: Aspects and prospects," in IEEE International Conference on Networking, Sensing and Control (ICNSC), april 2011, pp. 38–43.
35. R. C. Buck, *Advanced Calculus*, 3rd Ed, McGraw Hill, 1978.
36. http://people.brandeis.edu/~igusa/Math22bS12/Notes22b_d32
37. Sundarapandian, V., "An invariance principle of discrete-time nonlinear systems", *Applied Mathematics Letters*, pp. 85–91, 2003.
38. A. Mohammadi, M. Varahram, and I. Kheirizad, "Online solving of economic dispatch problem using neural network approach and comparing it with classical method," in *Emerging Technologies, 2006. ICET'06. International Conference on*. IEEE, 2007, pp. 581–586.

Impact of Quantized Inter-agent Communications on Game-Theoretic and Distributed Optimization Algorithms

Ehsan Nekouei, Tansu Alpcan and Robin J. Evans

Abstract Quantized inter-agent communications in game-theoretic and distributed optimization algorithms generate uncertainty that affects the asymptotic and transient behavior of such algorithms. This chapter uses the information-theoretic notion of *differential entropy power* to establish universal bounds on the maximum exponential convergence rates of primal-dual and gradient-based Nash seeking algorithms under quantized communications. These bounds depend on the inter-agent data rate and the local behavior of the agents' objective functions, and are independent of the quantizer structure. The presented results provide trade-offs between the speed of exponential convergence, the agents' objective functions, the communication bit rates, and the number of agents and constraints. For the proposed Nash seeking algorithm, the transient performance is studied and an upper bound on the average time required to settle inside a specified ball around the Nash equilibrium is derived under uniform quantization. Furthermore, an upper bound on the probability that the agents' actions lie outside this ball is established. This bound decays double exponentially with time.

1 Introduction

Modern societies are heavily dependent on networking technologies for almost every type of activity. The Internet, smart phones, and cloud computing could not exist without networking. In all networked systems, a limited number of resources, e.g., bandwidth and computing power, are shared among the interconnected devices, hereafter called the *agents*. The performance of the networked system is highly depen-

E. Nekouei
KTH Royal Institute of Technology, Osquldas väg 10, KTH Main Campus, stockholm, Sweden
e-mail: nekouei@kth.se

T. Alpcan (✉) · R. J. Evans
The University of Melbourne, Parkville, Melbourne, Australia
e-mail: tansu.alpcan@unimelb.edu.au

R. J. Evans
e-mail: robinje@unimelb.edu.au

© Springer Nature Switzerland AG 2018

T. Başar (ed.), *Uncertainty in Complex Networked Systems*,
Systems & Control: Foundations & Applications,
https://doi.org/10.1007/978-3-030-04630-9_15

dent on how these resources are shared among the agents. Hence, resource allocation algorithms play an important role in networking technologies. The network resource allocation problem between the agents can be formulated as a global optimization problem with a team-optimal solution, or modeled as a non-cooperative game. The solution in the former case is team-optimal, whereas the resources are shared according to the equilibrium of the game among the selfish agents in the latter.

This chapter studies two distinct scenarios for the network resource sharing problem related to these cases. In the first scenario, the resource allocation problem is posed as a network utility maximization (NUM) problem, and the agents deploy a distributed, iterative primal-dual optimization algorithm to solve the NUM problem. In the second scenario, the interaction between the agents is modeled as a non-cooperative game and the agents compute the Nash equilibrium (NE) solution of the game using a gradient-based Nash seeking algorithm. The communication channels between the agents are modeled as digital ones since the agents may be far away from each other. The actions of the agents are hence quantized into discrete-valued symbols that may are represented as bits. The finite capacity of practical communication leads to an upper bound on the average number of bits transmitted per unit time. Consequently, the agents' local variables can only be transmitted in a quantized form using a finite number of bits per time interval. It is known in the literature that such data rate limitations can have detrimental impacts on the performance of control and optimization algorithms. For example, a communication channel deployed in a feedback control system can destabilize the system if its data rate is too low, e.g., see [1, 2]. Moreover, in distributed optimization as well as Nash equilibrium seeking algorithms, the quantized communications results in information ambiguity since each agent receives only quantized information from the other agents (which is typically different from unquantized information).

The aim of this chapter is to quantify the impact of quantized communications in NUM problems and non-cooperative games by making use of information-theoretic ideas. The results presented integrate and summarize those in [3, 4].

The remainder of the chapter is organized as follows. The next section introduces the quantized primal-dual algorithm for NUMs and studies its asymptotic performance under quantized inter-agent communications. In Sect. 3, a quantized gradient-based Nash seeking algorithm is proposed and its asymptotic and non-asymptotic behaviors are analyzed under quantized communications. Section 3.3 presents the numerical results, which is followed by the concluding remarks of Sect. 4.

2 Primal-Dual Algorithm Under Quantized Communications

In the seminal work [5], Kelly et al. introduced the network utility maximization (NUM) approach, which provides decentralized frameworks in the form of primal, dual, and primal-dual (PD) decomposition methods, for solving network resource allocation problems. Each decomposition method distributes the computational burden of solving the resource allocation problem among the agents, while the task of information transfer between the agents is handled by the underlying communication network. The problem of devising efficient decomposition methods for NUM problems has been extensively studied in the literature, e.g., see [6] and the references therein. While the performance of distributed optimization algorithms, in particular of NUM algorithms, is well understood under perfect communication channels, investigation of the impact of imperfect communications on these optimization algorithms is relatively a new research area that has attracted much interests in the recent years, e.g., see [7, 8].

This section focuses on a NUM problem in which a group of agents maximize the sum of their local concave objective functions subject to a set of linear constraints using a quantized PD algorithm with a random initial condition. Following the conventions of the NUM literature, e.g., see [5, 6] and the references therein, it is assumed that the primal variables are updated by the agents, whereas each dual variable is updated by a network node (NN) that has access to the knowledge of the constraint associated with that specific dual variable. Thus, the agents and NNs need to exchange the quantized values of the primal and dual variables to execute the PD algorithm. The impact of quantized communications between the agents and NNs on the convergence rate of the PD algorithm under quantization is investigated in this setup.

The rest of this section first studies the system model, and then describes the communication graph, the structure of quantizers, as well as the underlying standing assumptions. Asymptotic and non-asymptotic results on the convergence of PD algorithm under quantization are presented.

2.1 NUM Model

A specific formulation of the NUM problem as one of convex optimization involves M agents, who maximize the sum of their individual objective functions subject to a set of linear equality constraints. Let x^i and $U_i\left(x^i\right)$ represent the decision variable of the agent i and its objective function, respectively. Assume that the objective function of each agent is concave in its decision variable. The agents then collectively solve the following NUM problem:

$$\begin{aligned} &\underset{\boldsymbol{x}}{\text{maximize}} \sum_{i}^{M} U_i\left(x^i\right), \\ &\text{Subject to} \;\; A\boldsymbol{x} = \boldsymbol{b}, \end{aligned} \tag{1}$$

where, M is the number of agents, $\boldsymbol{b} \in \mathbb{R}^N$, $A \in \mathbb{R}^{N\times M}$, N is the number of constraints, and $\boldsymbol{x} = \left[x^1, \cdots, x^M\right]^\top$. The condition $N < M$ is imposed to ensure that the feasible set of the optimization problem (1) is nonempty. The matrix A is assumed to be full rank, i.e., rank $(A) = N$, to ensure the uniqueness of the dual optimal solution. The objective function in (1) is concave and the constraints are linear. Thus, the centralized optimization problem (1) can be solved using the standard convex optimization techniques.

When solving the problem in a distributed manner using the PD algorithm [6], the primal and dual variables are updated according to

$$\begin{aligned} x_k^i &= x_{k-1}^i + \mu_{k-1}\left(\frac{d}{dx^i}U_i\left(x_{k-1}^i\right) - A_i^\top \boldsymbol{\lambda}_{k-1}\right), 1 \le i \le M \\ \lambda_k^j &= \lambda_{k-1}^j + \mu_{k-1}\left(\bar{A}_j \boldsymbol{x}_{k-1} - b_j\right) \quad 1 \le j \le N, \end{aligned} \tag{2}$$

where μ_{k-1} is the step size of the algorithm at iteration $k-1$, x_k^i and λ_k^j denote the values of ith primal variable and jth dual variable at iteration k, respectively. In terms of notation, $\boldsymbol{\lambda}_{k-1} = \left[\lambda_{k-1}^1, \ldots, \lambda_{k-1}^N\right]^\top$, A_i denotes the ith column of the matrix A, and $\bar{A}_j$ denotes the jth row of matrix A. The solution of the optimization problem (1) is obtained following a primal-dual (PD) decomposition approach in which the primal variables, i.e., agents' decision variables, are updated by the agents at each iteration. In addition, at each iteration of the PD algorithm, the jth dual variable, i.e., λ^j, is updated by the specific jth network node (NN) with knowledge of the parameters characterizing the constraint associated with λ^j, i.e., A_j and b_j. The vector of PD variables at iteration k, i.e., $\boldsymbol{y}_k$, is defined as the vector concatenation of $\boldsymbol{x}_k$ and $\boldsymbol{\lambda}_k$, i.e.,

$$\boldsymbol{y}_k = \left[\, \boldsymbol{x}_k,\, \boldsymbol{\lambda}_k \,\right].$$

It is assumed that the initial primal and dual variables, i.e., $\boldsymbol{x}_0$ and $\boldsymbol{\lambda}_0$, are chosen randomly according to the probability density functions $p_{\boldsymbol{x}_0}(\boldsymbol{x})$ and $p_{\boldsymbol{\lambda}_0}(\boldsymbol{\lambda})$, respectively. By allowing the initial condition to be random, the primal and dual variables become random variables. This facilitates the use information-theoretic tools for studying the speed of exponential convergence of the primal-dual algorithm under quantized communications. Furthermore, the following **assumptions** are imposed on the objective functions of the agents, the step size μ_k, $p_{\boldsymbol{x}_0}(\boldsymbol{x})$, and $p_{\boldsymbol{\lambda}_0}(\boldsymbol{\lambda})$.

1. The agents' objective functions are concave and twice continuously differentiable.
2. $U_i^{\min} \le \frac{d^2}{dx^{i^2}}U_i\left(x^i\right) \le U_i^{\max} < 0$ for $x_i \in \mathbb{R}$ and all i.
3. $\mu_k \le \min_i \frac{1}{\left|U_i^{\min}\right|}$ for all k.
4. The sequence $\{\mu_k\}_k$ converges to $\mu^\star > 0$.
5. The random vectors $\boldsymbol{x}_0$ and $\boldsymbol{\lambda}_0$ are mutually independent and the distributions of $\boldsymbol{x}_0$ and $\boldsymbol{\lambda}_0$ have finite differential entropies, i.e.,

$$\left|-\int p_{\boldsymbol{x}_0}(\boldsymbol{x}) \log\left(p_{\boldsymbol{x}_0}(\boldsymbol{x})\right) d\boldsymbol{x}\right| < \infty$$

$$\left|-\int p_{\boldsymbol{\lambda}_0}(\boldsymbol{\lambda}) \log\left(p_{\boldsymbol{\lambda}_0}(\boldsymbol{\lambda})\right) d\boldsymbol{\lambda}\right| < \infty$$

Assumptions 1 and 2 above are standard in the optimization literature. Assumption 2 implies that the objective functions of agents are strongly concave and the first derivative of each objective function is Lipschitz continuous. Assumption 4 implies that the unquantized update rule does not employ a diminishing step size rule as the PD update may not converge exponentially with such a step size rule. Assumptions 3 and 4, which are not commonly used in the literature, allow usage of the entropy power method from information theory. Assumption 5 implies that the initial condition injects a minimum amount of uncertainty to the PD algorithm, and the amount of uncertainty due to the initial condition is bounded. Variants of Assumption 5 are used in the quantized feedback control literature [1].

2.1.1 Communication Topology and Cost

The inter-agent communication topology is represented as a bipartite graph induced by the $N \times M$ constraint matrix A. In this graph, the edges exist only between the agents and the network nodes (NNs), which form two disjoint sets of vertices. There exists an edge between agent i and NN j in the communication graph if and only if $A_{ji} \neq 0$.

The communication mechanism is broadcast in nature, with each vertex ‘listening’ and broadcasting only to those other vertices with which it shares an edge. This is implemented by uniquely assigning every vertex in the graph one of $N + M$ disjoint transmission radio-frequency bands (*frequency division multiplexing*) or one of $N + M$ disjoint time slots per cycle (*time division multiplexing*), before the system is deployed. Any other vertex that needs to listen to a transmission just tunes in to the appropriate frequency band or time slot dedicated to the corresponding transmitter. Note that the edges do not represent individual one-to-one channels, but indicate the broadcast transmitter-receiver structure of the system.

Under typical digital modulation formats, the width of the frequency band/time slot allocated to agent i and/or the average transmission power it consumes to broadcast its encoded symbols to all NNs j with $A_{ji} \neq 0$ will be proportional to its average data rate $R_{\boldsymbol{x}}^{i} := \lim_{k\to\infty} \frac{1}{k} \sum_{t=0}^{k-1} \log \left|\mathscr{A}_{i,t}^{\boldsymbol{x}}\right|$. Similarly, the band/slot-width and/or transmission power used by NN j to broadcast its encoded dual symbols to all agents i with $A_{ji} \neq 0$ is typically proportional to $R_{\boldsymbol{\lambda}}^{j} := \lim_{k\to\infty} \frac{1}{k} \sum_{t=0}^{k-1} \log \left|\mathscr{A}_{j,t}^{\boldsymbol{\lambda}}\right|$. Equation (5) in the upcoming section, which can be intuitively interpreted as $\sum_{i=1}^{M} R_{\boldsymbol{x}}^{i} + \sum_{j=1}^{N} R_{\boldsymbol{\lambda}}^{j}$, then captures the total amount of physical resources, i.e., time, bandwidth, or transmission power, required for the system to communicate. It can be seen that this communication cost scales with $O(N + M)$ as the network grows in size. Note that due to the broadcast nature of the system, every transmission can be heard by multiple receivers, without the transmitter having to use up extra resources.

2.1.2 A Quantizer Structure for NUM

To execute the PD update rule (2), the agents and NNs require the knowledge of dual and primal variables, respectively. Since the agents and NNs are not necessarily co-located, the information exchange between NNs and agents is performed via broadcast communication channels, as described in the next subsection. Due to the capacity limitations of these channels, only quantized versions of the primal and dual variables can be exchanged between NNs and agents.

At iteration k, agent i encodes x_k^i to $\hat{Q}_{i,k}^{\boldsymbol{x}}$ using an adaptive encoder mapping of the form

$$\hat{Q}_{i,k}^{\boldsymbol{x}} = E_{i,k}^{\boldsymbol{x}} \left(\left\{ x_n^i \right\}_{n=0}^{k}, \left\{ \hat{Q}_{i,n}^{\boldsymbol{x}} \right\}_{n=0}^{k-1} \right).$$

It then broadcasts $\hat{Q}_{i,k}^{\boldsymbol{x}}$ to all NNs j with $A_{ji} \neq 0$. The output of the encoder of agent i at iteration k, i.e., $\hat{Q}_{i,k}^{\boldsymbol{x}}$, belongs to the finite alphabet set $\mathscr{A}_{i,k}^{\boldsymbol{x}}$. Thus, agent i requires $\log_2 \left| \mathscr{A}_{i,k}^{\boldsymbol{x}} \right|$ bits to transmit its encoded symbol to NNs. A large value of $\left| \mathscr{A}_{i,k}^{\boldsymbol{x}} \right|$ indicates that agent i transmits its decision variable with high precision to NNs whereas a low $\left| \mathscr{A}_{i,t}^{\boldsymbol{x}} \right|$ indicates low quality communication between agent i and NNs. Upon receiving $\hat{Q}_{i,k}^{\boldsymbol{x}}$, all NNs j with $A_{ji} \neq 0$ reconstruct the quantized estimate of x_k^i, i.e., $Q_{i,k}^{\boldsymbol{x}}$, using the decoder mapping $Q_{i,k}^{\boldsymbol{x}} = D_{i,k}^{\boldsymbol{x}} \left(\left\{ \hat{Q}_{i,n}^{\boldsymbol{x}} \right\}_{n=0}^{k} \right)$.

Similarly, at iteration k, NN j chooses symbol $\hat{Q}_{j,k}^{\boldsymbol{\lambda}}$ from the finite alphabet set $\mathscr{A}_{j,k}^{\boldsymbol{\lambda}}$ according to the adaptive encoding map

$$\hat{Q}_{j,k}^{\boldsymbol{\lambda}} = E_k^{\lambda} \left(\left\{ \lambda_n^j \right\}_{n=0}^{k}, \left\{ \hat{Q}_{j,n}^{\boldsymbol{\lambda}} \right\}_{n=0}^{k-1} \right),$$

and broadcasts $\hat{Q}_{j,k}^{\boldsymbol{\lambda}}$ to all the agents with index i, where $A_{ji} \neq 0$. Next, all agents i with $A_{ji} \neq 0$ construct the quantized version of λ_k^j, i.e., $Q_{j,k}^{\boldsymbol{\lambda}}$, using the decoding map $Q_{j,k}^{\boldsymbol{\lambda}} = D_{j,k}^{\boldsymbol{\lambda}} \left(\left\{ \hat{Q}_{j,n}^{\boldsymbol{\lambda}} \right\}_{n=0}^{k} \right)$. Note that this formulation allows the encoded symbol at iteration k to depend on the current and past values of the primal/dual variables as well as the past outputs of the encoder.

Let $\mathscr{Q} = \left\{ \left\{ E_{i,k}^{\boldsymbol{x}}(\cdot), D_{i,k}^{\boldsymbol{x}}(\cdot) \right\}_i, \left\{ E_{j,k}^{\boldsymbol{\lambda}}(\cdot), D_{j,k}^{\boldsymbol{\lambda}}(\cdot) \right\}_j \right\}_{k=0}^{\infty}$ be a quantization scheme. Then, the quantized versions of the PD variables at iteration k under the quantization scheme $\mathscr{Q}$ are denoted by Q_k, i.e.,

$$Q_k = \left[\, Q_k^{\boldsymbol{x}}, \;\; Q_k^{\boldsymbol{\lambda}} \, \right],$$

where $Q_k^{\boldsymbol{x}} = \left[Q_{1,k}^{\boldsymbol{x}}, \ldots, Q_{M,k}^{\boldsymbol{x}} \right]^{\top}$ and $Q_k^{\boldsymbol{\lambda}} = \left[Q_{1,k}^{\boldsymbol{\lambda}}, \ldots, Q_{N,k}^{\boldsymbol{\lambda}} \right]^{\top}$.

Next, three notions of data rate for a given quantization scheme $\mathscr{Q}$ are defined and later used to study the convergence behavior of primal, dual, and PD variables. The average aggregate data rate per unit time for transmitting the primal variables to NNs under the quantization scheme $\mathscr{Q}$, $R_{\boldsymbol{x}}$, is defined as

$$R_{\boldsymbol{x}} = \lim_{k\to\infty} \frac{1}{k} \sum_{t=0}^{k-1} \left(\sum_{i=1}^{M} \log \left| \mathscr{A}_{i,t}^{\boldsymbol{x}} \right| \right) \tag{3}$$

Similarly, define the average aggregate data rate per unit time for broadcasting the dual variables to agents under the quantization scheme $\mathscr{Q}$, $R_{\boldsymbol{\lambda}}$, as

$$R_{\boldsymbol{\lambda}} = \lim_{k\to\infty} \frac{1}{k} \sum_{t=0}^{k-1} \left(\sum_{j=1}^{N} \log \left| \mathscr{A}_{j,t}^{\boldsymbol{\lambda}} \right| \right) \tag{4}$$

Finally, the average total data rate per unit time under the quantization scheme $\mathscr{Q}$, i.e., $R_{\mathscr{Q}}$, is defined as

$$R_{\mathscr{Q}} = \lim_{k\to\infty} \frac{1}{k} \sum_{t=0}^{k-1} \left(\left(\sum_{i=1}^{M} \log \left| \mathscr{A}_{i,t}^{\boldsymbol{x}} \right| \right) + \sum_{j=1}^{N} \log \left| \mathscr{A}_{j,t}^{\boldsymbol{\lambda}} \right| \right) \tag{5}$$

The quantized PD update rule under the quantization scheme $\mathscr{Q}$ is formulated as

$$\begin{aligned} x_k^i &= x_{k-1}^i + \mu_{k-1} \left(\frac{d}{dx^i} U_i \left(x_{k-1}^i \right) - A_i^\top Q_{k-1}^{\boldsymbol{\lambda}} \right), \\ \lambda_k^j &= \lambda_{k-1}^j + \mu_{k-1} \left(\bar{A}_j Q_{k-1}^{\boldsymbol{x}} - b_j \right) \end{aligned} \tag{6}$$

Let $\boldsymbol{x}^\star$, $\boldsymbol{\lambda}^\star$ be the primal optimal and dual optimal solutions, respectively. Further, let $\boldsymbol{y}^\star$ be the vector concatenation of $\boldsymbol{x}^\star$, $\boldsymbol{\lambda}^\star$. Define $\varepsilon_k = \boldsymbol{y}_k - \boldsymbol{y}^\star$ as the difference between the PD variables at iteration k and the optimal solution. Let $\|\boldsymbol{\varepsilon}_k\|_2$ denote the distance of the PD variables at iteration k from the optimal solution, i.e.,

$$\|\boldsymbol{\varepsilon}_k\|_2 = \sqrt{\sum_{i=1}^{M} \left(x_k^i - x^{i\star} \right)^2 + \sum_{j=1}^{N} \left(\lambda_k^j - \lambda^{j\star} \right)^2}, \tag{7}$$

where $x^{i\star}$ and $\lambda^{j\star}$ are the optimal values of the primal variable x^i and the dual variable λ^j, respectively. Then, the mean square distance (MSD) of the PD variables from the optimal solution at iteration k under the quantization scheme $\mathscr{Q}$ is defined as $\mathsf{E}\left[\|\boldsymbol{\varepsilon_k}\|_2^2\right]$. Define the MSD of the primal variables from the optimal primal solution at iteration k as $\mathsf{E}\left[\left\|\boldsymbol{\varepsilon}_k^{\boldsymbol{x}}\right\|_2^2\right]$ where $\boldsymbol{\varepsilon}_k^{\boldsymbol{x}} = \boldsymbol{x}_k - \boldsymbol{x}^\star$. Similarly, the MSD of the dual variables at iteration k from the optimal dual solution is defined as $\mathsf{E}\left[\left\|\boldsymbol{\varepsilon}_k^{\boldsymbol{\lambda}}\right\|_2^2\right]$ where

$\boldsymbol{\varepsilon}_k^\lambda = \boldsymbol{\lambda}_k - \boldsymbol{\lambda}^\star$. Next, the class of optimum achieving (OA) quantization schemes are defined.

Definition 1 The quantization scheme $\mathscr{Q}$ is called an OA quantization scheme if, under $\mathscr{Q}$, the primal and dual variables converge to their optimal values $\boldsymbol{x}^\star$ and $\boldsymbol{\lambda}^\star$. That is:

$$\lim_{k\to\infty} \boldsymbol{x}_k = \boldsymbol{x}^\star$$

$$\lim_{k\to\infty} \boldsymbol{\lambda}_k = \boldsymbol{\lambda}^\star$$

Definition 1 implies that, under an OA quantization scheme, the quantization error does not impede the convergence of the PD algorithm to the optimal solution. Thus, under an OA quantization scheme, the PD algorithm converges to the optimal solution of the optimization problem regardless of the quantized communication between agents and NNs.

2.2 *Distributed Optimization Results and Discussion*

This section analyzes the impact of quantized communications on the mean square distance (MSD) from the optimal solution of the primal and dual variables generated by the primal-dual algorithm (PD) for two different regimes: (i) Asymptotic regime, (ii) Non-asymptotic regime. In the asymptotic regime, the behavior of the MSD under OA quantization schemes is studied as the number of iterations k increases to infinity. To this end, the notion of distance decay exponent (DDE) is introduced, which captures the rate of exponential convergence of the MSD to zero. Universal lower bounds on the DDE of PD variables, namely the primal variables and dual variables, are established in Theorems 1, 2, 3, and 4. In the non-asymptotic regime, the behavior of the MSD is investigated for any finite k. Here, the results provide universal lower bounds on the MSD for any finite k (see Corollaries 1 and 2 for more details).

2.2.1 Asymptotic Behavior of PD Algorithm Under Quantization

This subsection first introduces the notion of the distance decay exponent (DDE) for the primal and dual variables in PD. Subsequently, universal lower bounds on the DDE of PD, primal, and dual variables are derived.

Definition 2 Let $\mathscr{Q}$ be an OA quantization scheme. Then, the DDE of the PD, primal, and dual variables under $\mathscr{Q}$ are defined as

$$\liminf_{k\to\infty} \frac{1}{k} \log \mathsf{E}\left[\|\boldsymbol{\varepsilon}_k\|_2^2\right],$$
$$\liminf_{k\to\infty} \frac{1}{k} \log \mathsf{E}\left[\left\|\boldsymbol{\varepsilon}_k^x\right\|_2^2\right],$$
$$\liminf_{k\to\infty} \frac{1}{k} \log \mathsf{E}\left[\left\|\boldsymbol{\varepsilon}_k^\lambda\right\|_2^2\right],$$

respectively.

The DDEs capture the speed of exponential mean square convergence of the PD primal and dual variables to their corresponding optimal solutions. These are non-positive quantities, where a more negative DDE indicates faster convergence to the optimal solution. Moreover, a zero DDE implies slower-than-exponential convergence. In this subsection, the information-theoretic notion of entropy power is used to establish universal lower bounds on the DDE of the primal and dual variables.

The next theorem provides a universal lower bound on the DDE of the PD variables under OA quantization schemes. The proof uses the information-theoretic notion of *differential entropy power*, which has been previously applied to study control with communication constraints; see, e.g., [9].

Theorem 1 *Let $\mathscr{Q}$ be an OA quantization scheme. Then, the DDE of the PD variables under $\mathscr{Q}$ can be bounded from below*

$$\liminf_{k\to\infty} \frac{1}{k} \log \mathsf{E}\left[\|\boldsymbol{\varepsilon}_k\|_2^2\right] \geq \frac{2}{N+M}\left(\sum_{i=1}^{M} \log\left(1+\mu^\star \frac{d^2}{d{x^i}^2} U_i\left({x^i}^\star\right)\right) - R_{\mathscr{Q}}\right), \tag{8}$$

where ${x^i}^\star$ is the optimal value of the primal variable x^i.

Proof. See the Appendix.

Theorem 1 establishes an explicit universal lower bound on the DDE of PD variables under OA quantization schemes. This bound is universal in the sense that it is independent of the structure of the quantizer, and is thus applicable to all quantization schemes which are OA.

According to Theorem 1, for a given average total data rate $R_{\mathscr{Q}}$, *the PD variables converge to the optimal solution at most exponentially fast*. The speed of this exponential convergence is bounded by the average total data rate under the quantization scheme, i.e., $R_{\mathscr{Q}}$, and also by the behavior of the objective functions of agents around the optimal solution. As stated in Theorem 1, the lower bound on the DDE for PD variables decreases linearly with $R_{\mathscr{Q}}$. Note that as $R_{\mathscr{Q}}$ becomes large, the NNs and agents have more precise information about the primal and dual variables. The lower bound on the DDE also increases with the second derivatives of the agents' objective functions at the optimal solution. As these second derivatives become less negative, the objective function becomes flatter near the optimal solution and the quantized PD algorithm can be expected to converge more slowly. Theorem 1 is in concordance with this intuition.

The next theorem establishes a universal lower bound on the DDE of primal variables in the quantized PD update rule under an OA quantization scheme.

Theorem 2 ([10]) *Under an OA quantization scheme $\mathcal{Q}$, the DDE of the primal variables is lower bounded by*

$$\liminf_{k\to\infty} \frac{1}{k} \log \mathsf{E}\left[\left\|\boldsymbol{\varepsilon}_k^{\boldsymbol{x}}\right\|_2^2\right] \geq \frac{2}{M}\left(\sum_{i=1}^{M} \log\left(1+\mu^\star \frac{d^2}{d{x^i}^2} U_i\left(x^{i\star}\right)\right) - R_{\boldsymbol{\lambda}}\right). \tag{9}$$

According to Theorem 2, the exponential convergence speed of the primal variables is limited by (i) the behavior of objective functions of the agents around the optimal solution, (ii) the average aggregate data rate for transmission of dual variables, and (iii) the number of agents. Different from the PD bound in Theorem 1, this lower bound on the DDE of the primal variables depends only on the average aggregate data rate for transmission of dual variables, i.e., $R_{\boldsymbol{\lambda}}$, rather than on the average total data rate under the quantization scheme $\mathcal{Q}$. This observation signifies the role of the quantized dual variables on the convergence of the primal variables.

The next theorem presents a result on DDE for dual variables.

Theorem 3 ([10]) *The DDE of dual variables under an OA quantization scheme $\mathcal{Q}$ satisfies*

$$\liminf_{k\to\infty} \frac{1}{k} \log \mathsf{E}\left[\left\|\boldsymbol{\varepsilon}_k^{\boldsymbol{\lambda}}\right\|_2^2\right] \geq -\frac{2}{N} R_{\boldsymbol{x}}. \tag{10}$$

Theorem 3 establishes *a universal bound on the fastest possible exponential convergence rate of the dual variables under any OA quantization scheme $\mathcal{Q}$*. The lower bound in Theorem 3 is controlled by the number of constraints and the average aggregate data rate for transmission of primal variables to NNs. Compared to the PD lower bound, it does not depend on the behavior of the objective functions of agents and is only limited by the average aggregate data rate for transmission of the primal variables, i.e., $R_{\boldsymbol{x}}$, rather than the average total data rate $R_{\mathcal{Q}}$.

Next, a lower bound on the DDE of the PD algorithm is derived for quadratic NUM problems under zoom-in quantization schemes (see Definition 3). This bound is tighter than the lower bound in Theorem 1 for the high data rate regime. In a quadratic NUM problem, the objective function of agent i is given by $U_i\left(x^i\right) = -\frac{a_i}{2}\left(x^i\right)^2 + c_i x^i + f_i$ where a_i is a positive constant. The unquantized PD algorithm for quadratic NUM problems can be written as

$$\begin{aligned} x_k^i &= (1-\mu a_i)\, x_{k-1}^i + \mu\left(c_i - A_i^\top \boldsymbol{\lambda}_{k-1}\right), 1 \leq i \leq M \\ \lambda_k^j &= \lambda_{k-1}^j + \mu\left(\bar{A}_j \boldsymbol{x}_{k-1} - b_j\right) \quad 1 \leq j \leq N \end{aligned} \tag{11}$$

Let $\boldsymbol{y}_k$ be the vector concatenation of $\boldsymbol{x}_k$ and $\boldsymbol{\lambda}_k$. Then, (11) can be written as

$$\boldsymbol{y}_k = T\boldsymbol{y}_{k-1} + \mu \begin{bmatrix} \boldsymbol{c} \\ -\boldsymbol{b} \end{bmatrix}$$

where $\boldsymbol{c} = [c_1 \ldots, c_M]^\top$ and the matrix T is defined as

$$T = \begin{bmatrix} \text{Diag}\,(1-\mu a_1, \ldots, 1-\mu a_M) & -\mu A^\top \\ \mu A & I_N \end{bmatrix} \tag{12}$$

in which I_N denotes an N-by-N identity matrix and $\text{Diag}\,(1-\mu a_1, \ldots, 1-\mu a_M)$ is a diagonal matrix with the ith diagonal element equal to $1-\mu a_i$.

Let $\tilde{Q}_k = \left\{\hat{Q}^{\boldsymbol{x}}_{1,n}, \ldots, \hat{Q}^{\boldsymbol{x}}_{M,n}, \hat{Q}^{\boldsymbol{\lambda}}_{1,n}, \ldots, \hat{Q}^{\boldsymbol{\lambda}}_{N,n}\right\}_{n=0}^{k}$ be the collection of encoders' outputs up to iteration k, respectively. The quantized PD update rule is

$$x^i_k = T_{ii}x^i_{k-1} + \sum_{j=M+1}^{M+N} T_{ij}Q^{\lambda^{(j-M)}}_k + \mu c_i$$

$$\lambda^j_k = \lambda^j_{k-1} + \sum_{i=1}^{M} T_{ji}Q^{x^i}_k - \mu b_j$$

The quantized update rule is denoted by $\boldsymbol{y}_{k+1} = \hat{T}\left(\boldsymbol{y}_k, \tilde{q}_k\right)$ where $\tilde{q}_k$ is a realization of $\tilde{Q}_k$. We use $C_k\left(\tilde{q}_k\right)$ to represent the quantization cell corresponding to $\tilde{q}_k$, i.e., the set of points in $\mathbb{R}^{N+M}$ which are mapped to the same output by the encoder when $\tilde{Q}_k = \tilde{q}_k$. Next, a zoom-in quantization scheme is defined.

Definition 3 Consider the quantization scheme $\mathscr{Q}$, and let $C_k\left(\tilde{q}_k\right)$ be the quantization cell at iteration k which contains $\boldsymbol{y}_k$. Then, $\mathscr{Q}$ is a zoom-in quantization scheme if at time $k+1$ the image of $C_k\left(\tilde{q}_k\right)$ under $\hat{T}\left(\cdot, \tilde{q}_k\right)$ is quantized for all $k \in \mathbb{N}_0 = \{0, 1, 2, \ldots\}$.

In addition to the assumptions stated in Sect. 2.1 we also require

1. The matrix T is invertible and all its eigenvalues are inside the unit circle in the complex plane.
2. A zoom-in quantization scheme is employed and each primal/dual variable is independently quantized.
3. The distributions of initial primal and dual variables, i.e., $p_{\boldsymbol{x}_0}\left(\boldsymbol{x}\right)$ and $p_{\boldsymbol{\lambda}_0}\left(\boldsymbol{\lambda}\right)$, are bounded and have finite support sets.

Theorem 4 ([10]) *Consider any zoom-in quantization scheme $\mathscr{Q}$ with $\rho = \frac{\delta^{\max}_k}{\delta^{\min}_k}$ (for all k) where $\delta^{\max}_k$ and $\delta^{\min}_k$ are the maximum and minimum quantization steps under $\mathscr{Q}$ at iteration k, respectively. Let* B *be the hypercube centered at the origin with the ith side length equal to $4\rho\left|T_{ii}\right| + 2\left\|T\right\|_\infty$ where $\left\|\cdot\right\|_\infty$ denotes the norm infinity*

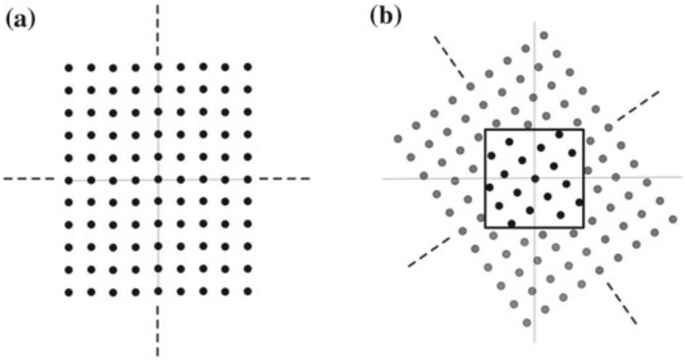


Fig. 1 Two-dimensional lattice of integers $\mathbb{Z}^2$ (**a**) and the lattice $T\mathbb{Z}^2$ (**b**)

and T_{ii} is the i diagonal entry of matrix T. Let β_T be the number elements in the set $\mathrm{B} \cap T\mathbb{Z}^{N+M}$ where the lattice $T\mathbb{Z}^{N+M}$ is defined as $T\mathbb{Z}^{N+M} = \left\{ T\boldsymbol{I}, \quad \boldsymbol{I} \in \mathbb{Z}^{N+M} \right\}$ and $\mathbb{Z}^{N+M}$ is the lattice of integers in $\mathbb{R}^{N+M}$. Then, the DDE of the PD variables under $\mathscr{Q}$ for quadratic NUM problems is lower bounded as

$$\liminf_{k\to\infty} \frac{1}{k+1} \log \mathsf{E}\left[\|\boldsymbol{\varepsilon}_{k+1}\|_2^2\right] \geq -\frac{2}{M+N} \log \left(\frac{\beta_T}{\left(\prod_{i=1}^{M+N} |T_{ii}|\right)} \right). \tag{13}$$

Theorem 4 establishes a bound on the fastest possible exponential convergence speed of quantized PD algorithms in quadratic NUM problems, under any zoom-in quantization scheme which is OA. The lower bound in Theorem 4 depends on the number of agents, number of constraints, and β_T. The constant β_T depends on the dynamics of the unquantized PD algorithm, i.e., matrix T, and can be interpreted as the number of lattice points in $\mathbb{Z}^{N+M}$ which lie in B after applying the linear transformation T to $\mathbb{Z}^{N+M}$. Figure 1 shows the two dimensional lattice of integers $\mathbb{Z}^2$ and its image after applying a linear transformation. In Fig. 1b, the number of lattice points in the square is equal to β_T. Since the transformation T is linear, $\mathbf{0}$ always lies in B which implies $\beta_T \geq 1$.

Consider the PD algorithm in a quadratic NUM problem under the zoom-in quantization scheme $\mathscr{Q}$ with $\rho = \frac{\delta_k^{\max}}{\delta_k^{\min}}$. For the quadratic PD algorithms, Theorems 1 and 4 can be combined into

$$\liminf_{k\to\infty} \frac{1}{k+1} \log \mathsf{E}\left[\left\|\boldsymbol{\varepsilon}_{k+1}\right\|_2^2\right] \geq \frac{2}{M+N} \left(\left(\sum_{i=1}^{M} \log\left(1-\mu a_i\right) \right) - \min\left(\log\left(\beta_T\right), R_{\mathscr{Q}} \right) \right) \tag{14}$$

If the quantization intervals for each primal/dual variable are divided into $K \geq 2$ equal length intervals, the data rate under quantization scheme $\mathscr{Q}$ i.e., $R_{\mathscr{Q}}$, will increase by $(N+M)\log(K)$ bits and ρ does not change. Hence, according to (14), the lower bound in Theorem 4 becomes tighter when compared to that in Theorem 1 as $R_{\mathscr{Q}}$ (or K) becomes large. This observation shows that the exponential convergence speed of the quantized PD algorithm in quadratic NUM problems cannot be made arbitrarily fast by increasing $R_{\mathscr{Q}}$.

An upper bound on β_T can be obtained by finding the number of lattice points of $\mathbb{Z}^{N+M}$ which lie in the smallest hypercube containing the image of B under T^{-1}. Let T^{-1} (B) be the image of the hypercube B under linear transformation T^{-1}. Let $\mathrm{B}^{\star}_{T^{-1}}$ be the smallest hypercube containing T^{-1} (B). Then, β_T is upper bounded by $\prod_i \left(\lfloor l_i^{\star} \rfloor + 1\right)$ where $l_i^{\star}$ is the ith side length of $\mathrm{B}^{\star}_{T^{-1}}$. In the numerical analysis, this upper bound on β_T is used to compute the lower bound in Theorem 4.

2.2.2 PD Algorithm in the Non-asymptotic Regime

This subsection establishes universal lower bounds on the mean square distance (MSD) of primal-dual (PD), primal, and dual variables from their corresponding optimal solutions at any finite time instance k. Unlike Theorems 1, 2 and 3, the following results are not limited to optimum achieving (OA) quantization schemes. Thus, they give rise to universal lower bounds on the MSD of PD, primal and dual variables from their corresponding optimal solutions, under arbitrary quantization schemes. The results in this subsection indicate that the distance between the optimization variables and the optimal solution cannot be made arbitrarily close to zero at a given time instance k. The following corollary presents a non-asymptotic lower bound on the MSD of the PD variables.

Corollary 1 ([10]) *Consider the PD algorithm under the quantization scheme $\mathscr{Q}$. Then, the MSD of the PD variables from the optimal solution at iteration k can be lower bounded as*

$$\log \mathsf{E}\left[\|\boldsymbol{\varepsilon}_k\|_2^2\right] \geq \log\left(\frac{\mathrm{e}^{1-\frac{1}{M+N}}}{2\pi\mathrm{e}}\right) + \frac{2}{N+M}\left(\sum_{i=1}^{M}\sum_{n=0}^{k-1}\log\left(1+\mu_n U_i^{\min}\right)\right.$$
$$\left. + \mathsf{h}\left[\boldsymbol{y}_0\right] - \sum_{t=0}^{k-1}\left(\left(\sum_{i=1}^{M}\log\left|\mathscr{A}_{i,t}^{\boldsymbol{x}}\right|\right) + \sum_{j=1}^{N}\log\left|\mathscr{A}_{j,t}^{\boldsymbol{\lambda}}\right|\right)\right), \tag{15}$$

Corollary 1 provides a universal lower bound on the MSD of PD variables under quantized communications between agents and NNs. This result indicates that at a given time the PD variables cannot be arbitrarily close to the optimal solution (in the mean square sense), and imposes a lower bound on the MSD of PD variables from the optimal solution at a given time. According to Corollary 1, the MSD of PD variables from the optimal solution at iteration k is bounded from below by the behavior of the second derivative of the objective functions of agents along the trajectories of primal variables up to time $k-1$, the total number of bits exchanged between agents and NNs up to time $k-1$, the differential entropy of distribution of initial PD variables, i.e., $\mathsf{h}\left[\boldsymbol{y}_0\right]$, and the number of constraints and agents. The impact of objective functions of agents and the data rate between agents and NNs on the lower bound in (15) are similar to those in Theorem 1.

Note that the entropy power of $\boldsymbol{y}_0$, i.e., $\frac{1}{2\pi \mathrm{e}} \mathrm{e}^{\frac{2}{N+M}\mathsf{h}[\boldsymbol{y}_0]}$ is a measure of effective support volume of the random vector $\boldsymbol{y}_0$. Thus, as $\mathsf{h}\left[\boldsymbol{y}_0\right]$ becomes large, the size of the effective support set of $\boldsymbol{y}_0$ increases, i.e., $\boldsymbol{y}_0$ will be distributed on a larger region of $\mathbb{R}^{N+M}$. As a result, the MSD of the PD variables from the optimal solution increases since $\boldsymbol{y}_0$ effectively takes value from a larger set, a behavior predicted by Corollary 1.

The next corollary establishes a lower bound on the MSD of primal and dual variables:

Corollary 2 ([10]) *Let* $\mathsf{E}\left[\left\|\boldsymbol{\varepsilon}^{\boldsymbol{x}}_k\right\|_2^2\right]$ *and* $\mathsf{E}\left[\left\|\boldsymbol{\varepsilon}^{\boldsymbol{\lambda}}_k\right\|_2^2\right]$ *be the MSD of the primal variables and dual variables, respectively, at iteration k from the optimal solution. Then,*

$$\log \mathsf{E}\left[\left\|\boldsymbol{\varepsilon}^{\boldsymbol{x}}_k\right\|_2^2\right] \geq \log\left(\frac{\mathrm{e}^{1-\frac{1}{M}}}{2\pi \mathrm{e}}\right) + \frac{2}{M}\left(\sum_{i=1}^{M}\sum_{n=0}^{k-1}\log\left(1+\mu_n U_i^{\min}\right) + \mathsf{h}\left[\boldsymbol{x}_0\right] - \sum_{t=0}^{k-1}\sum_{j=1}^{N}\log\left|\mathscr{A}^{\boldsymbol{\lambda}}_{j,t}\right|\right),$$

$$\log \mathsf{E}\left[\left\|\boldsymbol{\varepsilon}^{\boldsymbol{\lambda}}_k\right\|_2^2\right] \geq \log\left(\frac{\mathrm{e}^{1-\frac{1}{N}}}{2\pi \mathrm{e}}\right) + \frac{2}{N}\left(\mathsf{h}\left[\boldsymbol{\lambda}_0\right] - \sum_{t=0}^{k-1}\sum_{i=1}^{M}\log\left|\mathscr{A}^{\boldsymbol{x}}_{i,t}\right|\right),$$

2.3 An Optimum Achieving Quantization Scheme for NUM

This section presents a zoom-in uniform optimum achieving (OA) quantization scheme for the PD algorithm, denoted as $\mathscr{Q}_{\mathrm{a}}$. It is also proven that the PD algorithm under the quantization scheme $\mathscr{Q}_{\mathrm{a}}$ converges to the optimal solution of the optimization problem (1). To this end, assume that the unquantized PD algorithm forms a contraction map with contraction constant $\alpha \in [0,\ 1)$. Further assume that α is known by all agents and NNs. Under the quantization scheme $\mathscr{Q}_{\mathrm{a}}$, the quantization step at iteration k, i.e., δ_k, is set to $\delta_k = \alpha^{k+1}$.

At time $k = 0$, the agent i generates x^i_0 according to a uniform distribution on the interval $(-L\alpha, L\alpha)$ where L is a positive integer. Similarly, NN j generates λ^j_0 using a uniform distribution on $(-L\alpha, L\alpha)$. Next, agents and NNs quantize the initial primal and dual variables, respectively, using a midpoint uniform quantizer on $(-L\alpha, L\alpha)$ with quantization step $\delta_0 = \alpha$. Thus, the quantizer employed by agents and NNs at time $k = 0$, is given by $Q_{\mathrm{a},0}(z) = \left\lfloor \frac{z}{\alpha} \right\rfloor \alpha + \frac{\alpha}{2}$ for $z \in (-L\alpha, L\alpha)$ where $\lfloor\cdot\rfloor$ is the floor function. Each agent (NN) only needs $\left\lceil \log_2(2L)\right\rceil$ bits to communicate its initial primal (dual) variable where $\lceil\cdot\rceil$ is the ceiling function.

At time $k+1$, agent i first encodes x^i_{k+1} using the encoder $\hat{\mathrm{Q}}\left(\frac{x^i_{k+1} - C^{x^i}_{k+1}}{\delta_{k+1}}\right)$ where $C^{x^i}_{k+1} = Q^{\boldsymbol{x}}_{i,k} + \left\lfloor \frac{x^i_{k+1} - x^i_k}{\delta_k} \right\rfloor \delta_k$, $Q^{\boldsymbol{x}}_{i,k}$ is the quantized version of x^i_k, and $\hat{\mathrm{Q}}(\cdot)$ is given by

$$\hat{\mathrm{Q}}\left(z\right)=\begin{cases}\left\lceil\frac{2}{\alpha}\right\rceil-1 & \left(\left\lceil\frac{2}{\alpha}\right\rceil-1\right)\leq z\leq\left\lceil\frac{2}{\alpha}\right\rceil\\ \lfloor z\rfloor & -\left\lceil\frac{2}{\alpha}\right\rceil\leq z\leq\left(\left\lceil\frac{2}{\alpha}\right\rceil-1\right)\end{cases}\tag{16}$$

Let $I_{k+1}^{x^i}$ be the interval centered at $C_{k+1}^{x^i}$ with length $2\left\lceil\frac{2}{\alpha}\right\rceil\delta_{k+1}$. It can be shown that x_{k+1}^i belongs to this interval, which implies that the encoder mapping is always well defined (see the proof of Theorem 5 for more details).

Next, agent i transmits $\hat{\mathrm{Q}}\left(\frac{x_{k+1}^i-C_{k+1}^{x^i}}{\delta_{k+1}}\right)$ to its neighboring NNs in the communication graph using $\left\lceil\log_2\left(2\left\lceil\frac{2}{\alpha}\right\rceil\right)\right\rceil$ bits. Agent i also transmits $\left\lfloor\frac{x_{k+1}^i-x_k^i}{\delta_k}\right\rfloor$ to its neighboring NNs. This will allow the neighboring NNs of agent i to compute $C_{k+1}^{x^i}$, and update their decoders at time $k+1$. Note that $\left\lfloor\frac{x_{k+1}^i-x_k^i}{\delta_k}\right\rfloor$ is an integer which can be transmitted using finite number of bits. Finally, the neighboring NNs of agent i construct the quantized version of x_{k+1}^i using the decoder mapping $Q_{i,k+1}^{\boldsymbol{x}}=C_{k+1}^{x^i}+\hat{\mathrm{Q}}\left(\frac{x_{k+1}^i-C_{k+1}^{x^i}}{\delta_{k+1}}\right)\delta_{k+1}+\frac{\delta_{k+1}}{2}$.

At time $k+1$, NN j first encodes λ_{k+1}^j using the encoder mapping $\hat{\mathrm{Q}}\left(\frac{\lambda_{k+1}^j-C_{k+1}^{\lambda^j}}{\delta_{k+1}}\right)$ where $C_{k+1}^{\lambda^j}=Q_{j,k}^{\boldsymbol{\lambda}}+\left\lfloor\frac{\lambda_{k+1}^j-\lambda_k^j}{\delta_k}\right\rfloor\delta_k$, $Q_{j,k}^{\boldsymbol{\lambda}}$ is the quantized version of λ_k^j and $\hat{\mathrm{Q}}\left(\cdot\right)$ is given by (16). Let $I_{k+1}^{\lambda^j}$ be the interval centered at $C_{k+1}^{\lambda^j}$ with the length $2\left\lceil\frac{2}{\alpha}\right\rceil\delta_{k+1}$. It can be shown that λ_{k+1}^j belongs to $I_{k+1}^{\lambda^j}$ which indicates that the encoder mapping is always well defined.

Next, NN j transmits $\hat{\mathrm{Q}}\left(\frac{\lambda_{k+1}^j-C_{k+1}^{\lambda^j}}{\delta_{k+1}}\right)$ and $\left\lfloor\frac{\lambda_{k+1}^j-\lambda_k^j}{\delta_k}\right\rfloor$ to its neighboring agents in the communication graph. Finally, the neighboring agents of the NN j construct the quantized version of λ_{k+1}^j using the decoder mapping $Q_{j,k+1}^{\boldsymbol{\lambda}}=C_{k+1}^{\lambda^j}+\hat{\mathrm{Q}}\left(\frac{\lambda_{k+1}^j-C_{k+1}^{\lambda^j}}{\delta_{k+1}}\right)\delta_{k+1}+\frac{\delta_{k+1}}{2}$.

The next theorem shows that the quantized PD algorithm under $\mathscr{Q}_{\mathrm{a}}$ converges to the optimal solution.

Theorem 5 ([10]) *The PD algorithm under the quantization scheme $\mathscr{Q}_{\mathrm{a}}$ converges exponentially to the optimal solution of the optimization problem* (1).

3 Gradient-Based Nash Seeking Algorithms Under Quantized Communications

Game theory has been established to be of ubiquitous importance in engineering and used to analyze numerous problems, e.g., power control in wireless networks [11], wind energy harvesting, or sensor coverage [12]. In *non-cooperative games*, multiple agents aim to maximize individual utility functions by taking actions that are not necessarily coordinated with one another. The *Nash equilibrium (NE)* is one of the most important solution concepts in such games. It is a point in the action space at which no agent can increase its own utility by unilaterally changing its action.

The problem of finding Nash equilibria is an active research area that has attracted much attention, e.g., see [13, 14] and references therein. Gradient-based equilibrium-seeking (ES) algorithms are popular techniques for finding the NE of games with continuous action spaces and differentiable utility functions. In such algorithms, each agent modifies its current action according to the partial derivative of its utility function with respect to its action. The computation of this derivative implicitly requires communication between agents, since it typically depends on the actions of other agents.

This section investigates the effect of quantized communication on gradient-based, Nash seeking algorithms. More specifically, the following questions are addressed: (i) How does the communication data rate generally affect the convergence speeds achievable by ES algorithms? (ii) Given a uniform quantization scheme, on average how many time-steps are required for the ES algorithm to settle inside a ball around the NE? (iii) Given a uniform quantization scheme, what is the probability that agents' actions lie outside this ball at a given time?.

Section 3.1 introduces the non-cooperative game among agents and the distributed Nash seeking algorithm under quantized communications. The main results on the asymptotic and non-asymptotic behaviors of the Nash seeking algorithm are discussed in Sect. 3.2.

3.1 Game Model

Consider a game with M agents, indexed by $i \in \mathscr{M} := \{1, \dots, M\}$. Let $x^i \in \mathbb{R}$ be the action of the ith agent, $\boldsymbol{x}^{-i} := \left[x^1, \dots, x^{i-1}, x^{i+1}, x^M\right]^\top \in \mathbb{R}^{M-1}$, the vector of all agents' actions except the ith, and $U_i\left(x^i, \boldsymbol{x}^{-i}\right) \in \mathbb{R}$ the utility of the ith agent. Refer to this game as $\mathscr{G} = \langle \mathscr{M}, \left\{x^i\right\}_i, \{U_i(\cdot)\}_i\rangle$. Assume that each utility function $U_i\left(x^i, \boldsymbol{x}^{-i}\right)$ is twice continuously differentiable and concave with respect to x^i.

Ideally, each agent in the game would like to make its own utility as large as possible. However, since the global maximizers of the utility functions will not generally coincide, a compromise is needed. This is provided by the *Nash equilibrium (NE)*, already. If all agents play their NE strategies, denoted by x^i_{NE}, $i \in \mathscr{M}$, then no agent can increase its individual utility by unilaterally changing its action, i.e.,

$$x^i_{\mathrm{NE}} = \arg\max_{x^i} U_i\left(x^i, \boldsymbol{x}^{-i}_{\mathrm{NE}}\right), \forall i \in \mathscr{M}.$$

Throughout this section, it is assumed that the game admits a unique NE. This can be easily satisfied by imposing some additional mild conditions on the utility functions of agents, e.g., see [15]. Games arising in many engineering applications often admit a unique NE, which is associated with the desired operating point.

3.1.1 Gradient-Based Equilibrium Seeking with Quantized Communication

Gradient-based, equilibrium-seeking (ES) algorithms are among the most popular iterative techniques for finding the NE of a game with continuous action spaces and differentiable utility functions. In the absence of quantization, such algorithms take the general form

$$x_{k+1}^i = x_k^i + \mu_k \frac{\partial}{\partial x^i} U_i\left(x_k^i, \boldsymbol{x}_k^{-i}\right), k \in \mathbb{N}_0 := \{0, 1, 2 \ldots\} \tag{17}$$

where x_k^i is the action of the ith agent at iteration k, $\boldsymbol{x}_k^{-i}$ is the vector of all agent actions at iteration k except the ith, and $\mu_k > 0$ is a time-varying *step size*.

In order to implement this update rule, each agent does not need to know other agents' utility functions, which may be kept private, but only their latest actions. However, agents in a distributed game are often located far from each other, e.g., power plants competing in a wholesale electricity market for maximizing their individual profits. The long distance between agents, combined with finite transmission power and bandwidth, limit the communication capacity between agents. Consequently, the agents in distributed games cannot transmit their actions with infinite resolution, but instead exchange quantized versions that are representable with finite numbers of bits.

Assuming that each ith agent knows its own action x_k^i perfectly, let $D_{i,k}\left(x_k^i\right) \in \mathscr{A}_{i,k}$ represent the quantized action broadcast by it to all other agents at iteration k. Here $\mathscr{A}_{i,k} \subset \mathbb{R}$ is a finite set and $\left|\mathscr{A}_{i,k}\right|$ is the number of quantization levels used by the ith agent at iteration k. A large value of $\left|\mathscr{A}_{i,k}\right|$ implies that the ith agent transmits its action with high precision, whereas a low value reflects poor communication capacity and low precision.

Let $D_k\left(\boldsymbol{x}_k\right) \in \mathscr{A}_k$ represent the component-wise quantized version of the vector $\boldsymbol{x}_k$, that is $D_k\left(\boldsymbol{x}_k\right) = \left[D_{i,k}\left(x_k^i\right)\right]_i^\top$. Note that $\log|\mathscr{A}_k|$ denotes the aggregate number of bits used by agents to represent their actions in the kth iteration, where $|\mathscr{A}_k| = \prod_{i=1}^M \left|\mathscr{A}_{i,k}\right|$. With a slight abuse of notation, let Q_k denote the quantized version of the vector $\boldsymbol{x}_k^{-i}$. The ES algorithm with quantization then takes the form

$$x_{k+1}^i = x_k^i + \mu_k \frac{\partial}{\partial x^i} U_i\left(x_k^i, Q_k\right), \quad \forall i \in \mathscr{M}. \tag{18}$$

Let $\mathscr{D} = \{D_k\}_{k=0}^\infty$ be a quantization scheme. Given $\mathscr{D}$, the average *aggregate data rate* per unit time is defined as

$$R_{\mathscr{D}} := \limsup_{k \longrightarrow \infty} \frac{1}{k} \sum_{j=0}^{k-1} \log\left|\mathscr{A}_j\right|. \tag{19}$$

The next subsection studies the asymptotic and transient performance of this algorithm.

3.2 Equilibrium-Seeking Algorithm Results and Discussion

This section presents three performance measures for gradient-based equilibrium-seeking (ES) algorithms under quantization (18): (i) the asymptotic rate of exponential mean square convergence to the Nash equilibrium (NE), (ii) the expected time for agent actions to settle inside a specified neighborhood of the NE, under uniform quantization, and (iii) the probability that agent actions at a given iteration k lie outside this neighborhood, also under uniform quantization. The first criterion measures the long-term performance of the system, whereas the other two criteria characterize its transient performance. It is assumed that $\boldsymbol{x}_0$, the vector of initial agent actions, is drawn randomly according to a probability distribution on $\mathbb{R}^M$. This assumption allows application of stochastic methods to analyze performance, under mild assumptions on the initial distribution.

3.2.1 Lower Bound on the Asymptotic Mean Square Convergence Rate

This subsection presents a universal lower bound on the asymptotic convergence rate of any quantized ES scheme of the form (18). In this asymptotic analysis, it is assumed that

- the joint probability density function (pdf) $p_{\boldsymbol{x}_0}$ of initial actions has finite *differential entropy*, i.e., $\left|-\int p_{\boldsymbol{x}_0}(\boldsymbol{x}_0)\log\left(p_{\boldsymbol{x}_0}(\boldsymbol{x}_0)\right)d\boldsymbol{x}_0\right| < \infty$.
- the second partial derivatives of the utility functions are uniformly bounded above and below as

$$c_i \le \frac{\partial^2}{\partial {x^i}^2} U_i\left(x^i, \boldsymbol{x}^{-i}\right) \le b_i < 0, \quad \forall \boldsymbol{x} \in \mathbb{R}^M \tag{20}$$

- the step sizes $\mu_k > 0$ converge to $\mu^\star > 0$ as $k \to \infty$ and also satisfy $\sup_{k\in\mathbb{N}_0} \mu_k < \frac{1}{\max_i |c_i|}$, where c_i is the lower bound (20) on the second derivative of the ith agent's utility function with respect to its action.

Next, *equilibrium-achieving (EA)* quantization schemes are defined.

Definition 4 A quantization scheme $\mathscr{D}$ is *equilibrium-achieving* if all quantized and unquantized actions converge to the NE with time for any initial condition in the support of $p_{\boldsymbol{x}_0}$, i.e.

$$\begin{aligned} \lim_{k\longrightarrow\infty} \boldsymbol{x}_k &= \boldsymbol{x}_{\mathrm{NE}}, \\ \lim_{k\longrightarrow\infty} Q_k &= \boldsymbol{x}_{\mathrm{NE}}. \end{aligned} \tag{21}$$

The notion of distance decay exponent (DDE) for the ES algorithm under an EA quantization scheme is defined next.

Definition 5 For a given equilibrium-achieving quantization scheme $\mathscr{D}$, let $\boldsymbol{\varepsilon}_k$ be the difference between agent actions and the NE at iteration k, i.e., $\boldsymbol{\varepsilon}_k = \boldsymbol{x}_k - \boldsymbol{x}_{\mathrm{NE}}$. Then the *distance decay exponent (DDE)* is defined as

$$\liminf_{k \longrightarrow \infty} \frac{1}{k} \log \mathsf{E}\left[\|\boldsymbol{\varepsilon}_k\|_2^2\right].$$

The DDE gives the speed of exponential mean square convergence of the agents' actions to NE under $\mathscr{D}$, where the expectation is taken with respect to the initial distribution of actions. A more negative exponent indicates faster convergence. The first main result of this section is stated now

Theorem 6 ([4]) *Let $\mathscr{D}$ be any equilibrium-achieving quantization scheme with average aggregate data rate $R_{\mathscr{D}}$* (19)*. Then, the error decay exponent is lower bounded as*

$$\liminf_{k \longrightarrow \infty} \frac{1}{k} \log \mathsf{E}\left[\|\boldsymbol{\varepsilon}_k\|_2^2\right] \geq \frac{2}{M}\left(\sum_{i=1}^{M} \log\left(1 + \mu^{\star} \left.\frac{\partial^2 U_i}{\partial {x^i}^2}\right|_{\boldsymbol{x}_{\mathrm{NE}}}\right) - R_{\mathscr{D}}\right). \tag{22}$$

Theorem 6 establishes a universal lower bound on the rate of exponential mean square convergence that holds for any EA quantization scheme. This lower bound depends on the average aggregate date-rate $R_{\mathscr{D}}$, the second derivatives of the utility functions at the NE, and the number of agents. Recall that a more negative DDE corresponds to faster convergence.

Based on (22), the lower bound decreases (linearly) as $R_{\mathscr{D}}$ increases. This reflects the fact that each agent has more accurate information about the actions of the others and hence can make better decisions. Furthermore, the bound increases with the second derivatives of the utility functions. This is because a less negative second derivative indicates a flatter utility function, hence slower convergence to the NE.

Though the bound above may be conservative, unlike previous work it does not impose any particular structure on the quantization scheme, and delineates a universal trade-off between convergence rate, utility functions, data rate, and the number of agents.

3.2.2 Transient Performance

This subsection investigates the transient behavior of the equilibrium-seeking (ES) algorithm (18) under a uniform, time-invariant quantization scheme. The following assumptions are made on the game $\mathscr{G}$ and the ES algorithm (17):

- The NE of the game $\mathscr{G}$ belongs to the open, bounded and connected set $\mathscr{R} \subset \mathbb{R}^M$ which has non-zero Lebesgue measure.
- $\boldsymbol{x}_0$, i.e., the initial action of agents, is randomly drawn from $\mathscr{R}$.
- $\frac{\partial}{\partial x^i} U_i\left(x^i, \boldsymbol{x}^{-i}\right)$ is twice continuously differentiable for all i.
- The ES algorithm under perfect communication, i.e., the update rule (17), is a pseudo-contraction mapping. That is

$$\left\| \boldsymbol{x}_k - \boldsymbol{x}_{\mathrm{NE}} \right\|_2 \leq \alpha \left\| \boldsymbol{x}_{k-1} - \boldsymbol{x}_{\mathrm{NE}} \right\|_2,$$

where $\alpha \in [0, 1)$ and

$$x_k^i = x_{k-1}^i + \mu_{k-1} \frac{\partial}{\partial x^i} U_i \left(x_{k-1}^i, \boldsymbol{x}_{k-1}^{-i} \right).$$

- The sequence $\{\mu_k\}_{k=0}^{\infty}$ is assumed to be bounded.

Let d be the diameter of $\mathscr{R}$, that is, $d = \sup \left\{ \| \boldsymbol{x} - \boldsymbol{y} \|_2 : \boldsymbol{x}, \boldsymbol{y} \in \mathscr{R} \right\}$. Let $\mathrm{B}(\boldsymbol{x}_c, d/2)$ be the smallest ball containing $\mathscr{R}$ where $\mathrm{B}(\boldsymbol{x}_c, d/2)$ represents a closed ball in Euclidean norm centered at $\boldsymbol{x}_c$ with radius $d/2$. Let $\mathrm{Q}(\boldsymbol{x}_c, 3d/2)$ be the cube centered at $\boldsymbol{x}_c$ with side length $3d$. In this subsection, assume that the agents employ a uniform, time-invariant quantization scheme denoted by $\mathscr{D}_{\mathrm{u}}$. Under $\mathscr{D}_{\mathrm{u}}$, the intersection of $\mathrm{Q}(\boldsymbol{x}_c, 3d/2)$ and action space of each agent is uniformly quantized with the quantization step δ. The ES update rule under the uniform quantization scheme $\mathscr{D}_{\mathrm{u}}$ is given by

$$x_{k+1}^i = x_k^i + \mu_k \frac{\partial}{\partial x^i} U_i \left(x_k^i, \mathscr{D}_{\mathrm{u}} \left(\boldsymbol{x}_k^{-i} \right) \right), \forall i. \tag{23}$$

Since the quantization scheme $\mathscr{D}_{\mathrm{u}}$ is only defined on $\mathrm{Q}(\boldsymbol{x}_c, 3d/2)$, the actions of agents have to stay in $\mathrm{Q}(\boldsymbol{x}_c, 3d/2)$. Note that if $\boldsymbol{x}_0$, the initial action of agents, belongs to $\mathscr{R}$ and the quantization step δ is sufficiently small, then, the actions of agents will always stay in $\mathrm{Q}(\boldsymbol{x}_c, 3d/2)$. A sufficient condition for δ is given by

$$\begin{aligned} \sup_k \mu_k \left(\delta \sqrt{\sum_i \Phi_i^2} + \frac{1}{2} \delta^2 \left(M \sqrt{\sum_i \Psi_i^2} + \sqrt{\sum_i \eta_i^2} \right) \right) & \\ \leq (1 - \alpha) d, & \end{aligned} \tag{24}$$

where Φ_i, Ψ_i and η_i are given by

$$\begin{aligned} \Phi_i &= \sup_{\boldsymbol{x} \in \mathrm{Q}(\boldsymbol{x}_c, 3d/2)} \sum_{j \neq i} \left| \frac{\partial^2}{\partial^2 x^j x^i} U_i \left(x^i, \boldsymbol{x}^{-i} \right) \right|, \\ \Psi_i &= \sup_{\boldsymbol{x} \in \mathrm{Q}(\boldsymbol{x}_c, 3d/2)} \left\| \nabla^2 \frac{\partial}{\partial x^i} U_i \left(x^i, \boldsymbol{x}^{-i} \right) \right\|_2, \\ \eta_i &= \sup_{\boldsymbol{x} \in \mathrm{Q}(\boldsymbol{x}_c, 3d/2)} \left| \frac{\partial^3}{\partial {x^i}^3} U_i \left(x^i, \boldsymbol{x}^{-i} \right) \right|. \end{aligned}$$

respectively, where $\nabla^2 (\cdot)$ is the Hessian operator. The left hand side of (24) is an upper bound on the distortion induced by the quantization scheme $\mathscr{D}_{\mathrm{u}}$. Thus, (24) essentially implies that the quantization scheme $\mathscr{D}_{\mathrm{u}}$ is well defined if the distortion caused by the quantization scheme at each time-step is small enough.

The term Φ_i represents the sensitivity of update rule of the ith agent to the actions of other agents. When agents are less sensitive to each other's actions, according to (24), a relatively large quantization step δ can be chosen without introducing a large amount of distortion in the evolution of the ES algorithm. However, when agents are highly sensitive to each other's actions, a high-resolution quantization scheme should be employed to avoid a large amount of distortion. Moreover, according to (24), small values of the step size result in small distortion values. Since each agent modifies its action by adding the term $\mu_k \frac{\partial}{\partial x^i} U_i \left(x_k^i, \mathscr{D}_{\mathrm{u}}\left(\boldsymbol{x}_k^{-i}\right)\right)$ to its previous action, a small value of step size results in a small value of distortion at the cost of a slow convergence speed.

Let $\mathsf{E}\left[\mathscr{N}\right]$ denote the expected time required for $\boldsymbol{x}_k$ to settle inside $\mathrm{B}\left(\boldsymbol{x}_{\mathrm{NE}}, r\right)$. A small value of $\mathsf{E}\left[\mathscr{N}\right]$ indicates that the ES algorithm, on average, quickly approaches the NE whereas a large value of the $\mathsf{E}\left[\mathscr{N}\right]$ indicates a relatively slow convergence. Due to the quantization distortion, the radius of $\mathrm{B}\left(\boldsymbol{x}_{\mathrm{NE}}, r\right)$ cannot be arbitrarily small. If r is less than the total quantization distortion, one cannot guarantee that agents' actions will eventually settle inside $\mathrm{B}\left(\boldsymbol{x}_{\mathrm{NE}}, r\right)$ as k becomes large. Here, it is assumed that $r > \theta$ where θ is given by

$$\theta = \frac{\sup_k \mu_k}{1-\alpha}\left(\delta\sqrt{\sum_i \Phi_i^2} + \frac{1}{2}\delta^2\left(M\sqrt{\sum_i \Psi_i^2} + \sqrt{\sum_i \eta_i^2}\right)\right).$$

Note that, θ represents an upper bound on the aggregate distortion caused by the quantization scheme $\mathscr{D}_{\mathrm{u}}$ over time (see [4] for more details). The next theorem provides an upper bound on $\mathsf{E}\left[\mathscr{N}\right]$.

Theorem 7 ([4]) *Consider the uniform quantization scheme $\mathscr{D}_{\mathrm{u}}$ with the quantization step δ satisfying* (24)*. Let $\mathsf{E}\left[\mathscr{N}\right]$ denote the expected time required for the ES algorithm under $\mathscr{D}_{\mathrm{u}}$ to settle in $\mathrm{B}\left(\boldsymbol{x}_{\mathrm{NE}}, r\right)$ with $r > \theta$. Then, $\mathsf{E}\left[\mathscr{N}\right]$ is upper bounded as*

$$\mathsf{E}\left[\mathscr{N}\right] \leq \frac{1}{\log\left(\frac{1}{\alpha}\right)}\left(\mathsf{E}\left[\log\left(\frac{\left\|\boldsymbol{x}_0 - \boldsymbol{x}_{\mathrm{NE}}\right\|_2}{r-\theta}\right)\mathsf{I}_{\left\{\frac{\left\|\boldsymbol{x}_0 - \boldsymbol{x}_{\mathrm{NE}}\right\|_2}{r-\theta} > 1\right\}}\right]\right.$$

Theorem 7 provides an upper bound on the expected time required for the actions of agents to settle inside a ball of radius r centered at the NE. This upper bound is controlled by α, θ, r and the distribution of the initial actions of agents. According to this theorem, the effect of α on the expected time is manifested through the multiplicative factor $\frac{1}{\log\left(\frac{1}{\alpha}\right)}$ with $\alpha \in [0, 1)$. As α becomes closer to zero, the distance between the actions of agents and the NE decays faster due to the pseudo-contraction property of the non-quantized update rule. Thus, the average time required to settle inside $\mathrm{B}\left(\boldsymbol{x}_{\mathrm{NE}}, r\right)$ becomes smaller as α decreases.

The function $\mathsf{E}[\mathscr{N}]$ is non-increasing in r. That is, as r becomes small, it takes more time for the ES algorithm (23) to settle in $\mathrm{B}\left(\boldsymbol{x}_{\mathrm{NE}}, r\right)$. This observation is also consistent with our result in Theorem 7, i.e., the upper bound on the $\mathsf{E}[\mathscr{N}]$ increases as r becomes small. Finally, Theorem 7 suggests that the expected time required to settle inside $\mathrm{B}\left(\boldsymbol{x}_{\mathrm{NE}}, r\right)$ is influenced by the distribution of initial actions of agents, $p_{\boldsymbol{x}_0}(\boldsymbol{x})$. Observe that, when $p_{\boldsymbol{x}_0}(\boldsymbol{x})$ is highly concentrated around the Nash equilibrium, the ES algorithm (23) requires less time to settle inside $\mathrm{B}\left(\boldsymbol{x}_{\mathrm{NE}}, r\right)$ compared to the case that $p_{\boldsymbol{x}_0}(\boldsymbol{x})$ has a low degree of concentration around the NE.

The upper bound in Theorem 7 depends on the distance between the initial action of agents and the NE, i.e., $\left\|\boldsymbol{x}_0 - \boldsymbol{x}_{\mathrm{NE}}\right\|_2$. Since both $\boldsymbol{x}_0$ and $\boldsymbol{x}_{\mathrm{NE}}$ belong to $\mathscr{R}$, one can use the fact that $\left\|\boldsymbol{x}_0 - \boldsymbol{x}_{\mathrm{NE}}\right\|_2 \leq d$ to obtain an upper bound on the $\mathsf{E}[\mathscr{N}]$ which is independent of $\boldsymbol{x}_0$ and $\boldsymbol{x}_{\mathrm{NE}}$. This result is stated in the next corollary.

Corollary 3 ([4]) *The expected time required for the ES algorithm under $\mathscr{D}_{\mathrm{u}}$ to settle in $\mathrm{B}\left(\boldsymbol{x}_{\mathrm{NE}}, r\right)$ can be upper bounded as*

$$\mathsf{E}[\mathscr{N}] \leq \frac{1}{\log\left(\frac{1}{\alpha}\right)} \log\left(\frac{d}{r-\theta}\right).$$

Another performance measure for the transient behavior of the ES algorithm (23) under the uniform quantization scheme $\mathscr{D}_{\mathrm{u}}$ is investigated next. In this case, the probability that $\boldsymbol{x}_k$ lies outside a ball of radius $r > \theta$ around the NE, i.e.,

$$\mathsf{Pr}\left\{\left\|\boldsymbol{x}_k - \boldsymbol{x}_{\mathrm{NE}}\right\|_2 > r\right\}.$$

Note that, $\mathsf{Pr}\left\{\left\|\boldsymbol{x}_k - \boldsymbol{x}_{\mathrm{NE}}\right\|_2 > r\right\}$ is a function of k, and decays to zero as k tends to infinity. For a given k, a small value of $\mathsf{Pr}\left\{\left\|\boldsymbol{x}_k - \boldsymbol{x}_{\mathrm{NE}}\right\|_2 > r\right\}$ indicates that $\boldsymbol{x}_k$ approaches the NE at a higher speed compared to a large value of $\mathsf{Pr}\left\{\left\|\boldsymbol{x}_k - \boldsymbol{x}_{\mathrm{NE}}\right\|_2 > r\right\}$. The next theorem provides an upper bound on the probability that $\boldsymbol{x}_k$ lies outside $\mathrm{B}\left(\boldsymbol{x}_{\mathrm{NE}}, r\right)$.

Theorem 8 ([4]) *Consider the uniform quantization scheme $\mathscr{D}_{\mathrm{u}}$ with the quantization step δ satisfying (24). Then, the probability that $\boldsymbol{x}_k$ lies outside $\mathrm{B}\left(\boldsymbol{x}_{\mathrm{NE}}, r\right)$ with $r > \theta$ is upper bounded as*

$$\mathsf{Pr}\left\{\left\|\boldsymbol{x}_k - \boldsymbol{x}_{\mathrm{NE}}\right\|_2 > r\right\} \leq \min\left(1, \mathrm{e}^{-\frac{r-\theta}{\alpha^k}} \mathsf{E}\left[\mathrm{e}^{\left\|\boldsymbol{x}_0 - \boldsymbol{x}_{\mathrm{NE}}\right\|_2}\right]\right). \tag{25}$$

Theorem 8 provides an upper bound on the probability that $\boldsymbol{x}_k$ lies outside of the ball radius r around the NE at a given time. According to Theorem 8, this probability decays to zero at least double exponentially with k. Also, the decay rate of this probability depends on the contraction constant α. As α becomes small, the distance between agents' actions and the NE decays faster. Hence, the probability that $\boldsymbol{x}_k$ lies outside $\mathrm{B}\left(\boldsymbol{x}_{\mathrm{NE}}, r\right)$ decays faster to zero, and $\boldsymbol{x}_k$ with high probability lies inside $\mathrm{B}\left(\boldsymbol{x}_{\mathrm{NE}}, r\right)$. The term $\mathsf{E}\left[\mathrm{e}^{\left\|\boldsymbol{x}_0 - \boldsymbol{x}_{\mathrm{NE}}\right\|_2}\right]$ in (25) indicates the effect of the distribution of $\boldsymbol{x}_0$ on $\Pr\left\{\left\|\boldsymbol{x}_k - \boldsymbol{x}_{\mathrm{NE}}\right\|_2 > r\right\}$. That is, when the distribution of $\boldsymbol{x}_0$ is more concentrated around the NE, we expect $\boldsymbol{x}_k$ to approach the NE at a faster speed. Note that for a given k, $\Pr\left\{\left\|\boldsymbol{x}_k - \boldsymbol{x}_{\mathrm{NE}}\right\|_2 > r\right\}$ is a non-increasing function of r. That is, $\boldsymbol{x}_k$ lies outside $\mathrm{B}\left(\boldsymbol{x}_{\mathrm{NE}}, r\right)$ with high probability as r becomes small. This behavior is consistent with the upper bound in Theorem 8.

One can use the fact that $\left\|\boldsymbol{x}_0 - \boldsymbol{x}_{\mathrm{NE}}\right\|_2 \leq d$ to obtain an upper bound on $\Pr\left\{\left\|\boldsymbol{x}_k - \boldsymbol{x}_{\mathrm{NE}}\right\|_2 > r\right\}$ which is independent of $\boldsymbol{x}_0$ and $\boldsymbol{x}_{\mathrm{NE}}$. This result is stated in the next corollary.

Corollary 4 *The probability that* $\boldsymbol{x}_k$ *lies outside* $\mathrm{B}\left(\boldsymbol{x}_{\mathrm{NE}}, r\right)$ *can be upper bounded as*

$$\Pr\left\{\left\|\boldsymbol{x}_k - \boldsymbol{x}_{\mathrm{NE}}\right\|_2 > r\right\} \leq \min\left(1, \mathrm{e}^{d - \frac{r-\theta}{\alpha^k}}\right). \tag{26}$$

Finally, the upper bound in Corollary 4 can be used to obtain an upper bound on the number of time-steps required for $\boldsymbol{x}_k$ to lie in $\mathrm{B}\left(\boldsymbol{x}_{\mathrm{NE}}, r\right)$ with a given certainty level. Let N_p be the number of time-steps required for $\boldsymbol{x}_k$ to lie in $\mathrm{B}\left(\boldsymbol{x}_{\mathrm{NE}}, r\right)$ with the probability at least equal to p. Then, using (26), N_p can be upper bounded by the smallest positive integer satisfying

$$d - (r - \theta) \leq \alpha^k \log(1 - p).$$

Theorem 8 and Corollary 4 can be used to obtain bounds on the required number of quantization levels to guarantee that the agents' actions at iteration k lie inside a ball of radius r around the NE with a given probability. In addition, Theorem 7 and Corollary 3 give rise to bounds on the required number of quantization levels to guarantee an average settling time. Finally, Theorem 6 provides guidelines on the required average aggregate data rate for achieving a desired speed of exponential convergence to the NE.

3.2.3 An Adaptive EA Quantization Scheme

Based on the previous analysis of the uniform quantization scheme $\mathscr{D}_{\mathrm{u}}$, this subsection presents an adaptive EA quantization scheme under which the ES algorithm converges to the NE. This quantization scheme is denoted as $\mathscr{D}_{\mathrm{a}}$. Later, in the numerical result section, the error decay exponent of $\mathscr{D}_{\mathrm{a}}$ is studied. Recall that, the NE belongs to the region $\mathscr{R}$ with the diameter d. The basic idea behind the adaptive quantization scheme $\mathscr{D}_{\mathrm{a}}$ is to reduce the size of the known region around the NE in each time-step.

Let $\mathscr{R}_k$ denote the region which the NE belongs to at iteration k under the quantization scheme $\mathscr{D}_{\mathrm{a}}$. The quantization scheme $\mathscr{D}_{\mathrm{a}}$ is designed such that the diameter of $\mathscr{R}_k$ converges to zero as k tends to infinity. Under the quantization scheme $\mathscr{D}_{\mathrm{a}}$, initially, the intersection of action space of each agent with $\mathrm{Q}\left(\boldsymbol{x}_c, 3d/2\right)$ is quantized with the quantization step δ_0 which satisfies the following inequality

$$\sup_k \mu_k \left(\delta_0 \sqrt{\sum_i \Phi_i^2} + \frac{1}{2}\delta_0^2 \left(M \sqrt{\sum_i \Psi_i^2} + \sqrt{\sum_i \eta_i^2} \right) \right) \leq \hat{\alpha} d, \tag{27}$$

where $\hat{\alpha}$ is a constant arbitrarily selected from the interval $(0, 1-\alpha)$. It is straightforward to show that the distance between $\boldsymbol{x}_1$ and the NE under $\mathscr{D}_\mathrm{a}$ can be upper bounded as

$$\begin{aligned} \left\|\boldsymbol{x}_1 - \boldsymbol{x}_\mathrm{NE}\right\|_2 &\leq \alpha \left\|\boldsymbol{x}_0 - \boldsymbol{x}_\mathrm{NE}\right\|_2 + \sup_k \mu_k \left(\delta_0 \sqrt{\sum_i \Phi_i^2} + \frac{1}{2}\delta_0^2 \left(M \sqrt{\sum_i \Psi_i^2} + \sqrt{\sum_i \eta_i^2} \right) \right) \\ &\leq \left(\alpha + \hat{\alpha}\right) d, \end{aligned}$$

which implies that the NE belongs to the ball of radius $\left(\alpha + \hat{\alpha}\right) d$ around $\boldsymbol{x}_1$. In the second time-step, $\mathrm{Q}\left(\boldsymbol{x}_1, \left(\alpha + \hat{\alpha}\right) d\right) \cap \mathrm{Q}\left(\boldsymbol{x}_c, 3d/2\right)$ is considered as $\mathscr{R}_1$ and the intersection of each agent's action space with the $\mathscr{R}_1$ is quantized. Similarly, at iteration k, $k \geq 1$, we have $\mathscr{R}_k = \mathrm{Q}\left(\boldsymbol{x}_k, d_k\right) \cap \mathrm{Q}\left(\boldsymbol{x}_c, 3d/2\right)$ where $d_i = \left(\alpha + \hat{\alpha}\right) d_{i-1}$ with $d_0 = d$. Then, the intersection of action space of each agent with $\mathscr{R}_k$ is quantized. Also, the quantization step at iteration k, δ_k, is chosen such that the following inequality is satisfied:

$$\sup_k \mu_k \left(\delta_k \sqrt{\sum_i \Phi_i^2} + \frac{1}{2}\delta_k^2 \left(M \sqrt{\sum_i \Psi_i^2} + \sqrt{\sum_i \eta_i^2} \right) \right) \leq \hat{\alpha} d_k, \tag{28}$$

Since d_k converges to zero as k tends to infinity, the actions of agents and their quantized versions, under the quantization scheme $\mathscr{D}_\mathrm{a}$, converge to the NE as k tends to infinity which implies that $\mathscr{D}_\mathrm{a}$ is an EA quantization scheme. Algorithm 1 shows the different steps of the adaptive quantization scheme $\mathscr{D}_\mathrm{a}$.

Algorithm 1 The adaptive quantization scheme $\mathscr{D}_\mathrm{a}$

```
$k \leftarrow 0$. ($k$ is the time index.)
$d_k \leftarrow d$. ($d$ and $d_k$ are the radii of $\mathscr{R}$ and $\mathscr{R}_k$, respectively.)
Set $\delta_k$ as the solution of (27). ($\delta_k$ is the quantization step at iteration $k$.)
Quantize the intersection of action space of each agent with the $\mathrm{Q}(\boldsymbol{x}_c, 3d/2)$.
repeat
    Update the actions of agents.
    $k \leftarrow k + 1$.
    $d_k \leftarrow \left(\alpha + \hat{\alpha}\right) d_{k-1}$.
    Choose $\delta_k$ such that (28) is satisfied.
    Quantize the intersection of action space of each agent with $\mathrm{Q}\left(\boldsymbol{x}_k, d_k\right) \cap \mathrm{Q}(\boldsymbol{x}_c, 3d/2)$.
until The ES algorithm converges to the NE.
```

3.3 Numerical Results

This section presents a set of numerical results for a non-cooperative game with five agents seeking to maximize their utility functions. The utility function of ith agent is given by

$$U_i\left(x^i, \boldsymbol{x}^{-i}\right) = \frac{t_{ii}}{2}\left(x^i\right)^2 + x^i\left(\sum_{j\neq i} t_{ij}x^j - l_i\right), \tag{29}$$

where $t_{ii} < 0$ for all i and $t_{ij}, l_i \in \mathbb{R}$. Utility functions of the form (29) arise in many engineering applications such as analyzing the bidding behavior of a group of generators competing for maximizing their profits in an electricity market, e.g., see [16]. Let T be an M-by-M matrix with the (i, j)th entry equal to t_{ij}. Assume that T is negative definite. Since T is invertible, it can be easily verified, using the Karush–Kuhn–Tucker conditions, that the quadratic game with the utility functions (29) admits a unique Nash equilibrium. For this quadratic game, the ES algorithm under perfect communication condition can be written as

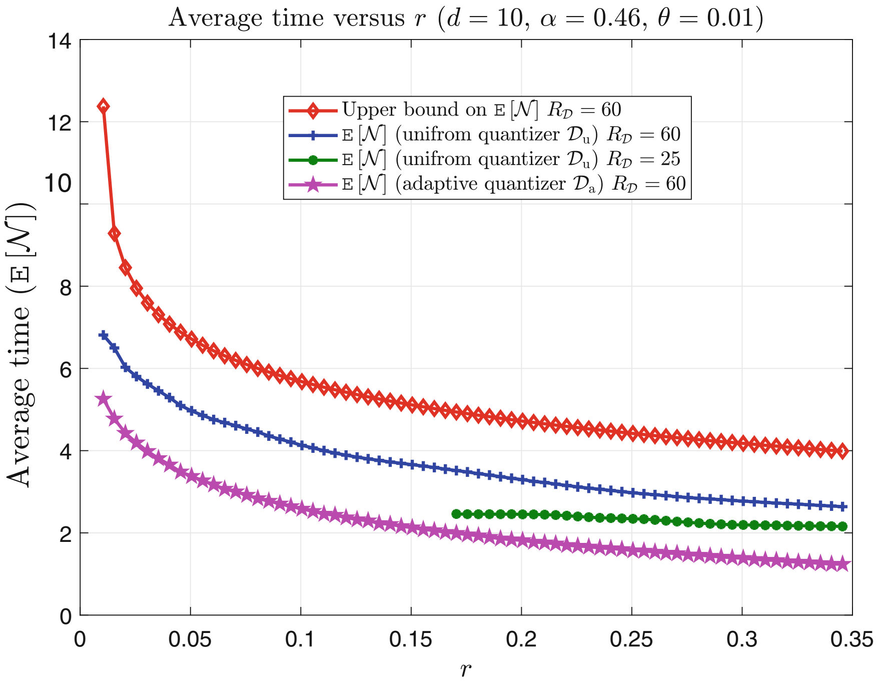


Fig. 2 The average time required for the ES algorithm to settle in a ball of radius r around the NE for the fixed and adaptive quantizers as a function of r

$$\boldsymbol{x}_{k+1} = (I + \mu T)\,\boldsymbol{x}_k - \mu \boldsymbol{l},$$

where $\boldsymbol{l} = [l_1, \ldots, l_M]^\top$. The step size μ is selected such that the spectral radius of $I + \mu T$ is strictly less than one. In the numerical results, it is assumed that the NE belongs to a hypercube $\mathscr{R}$, whose side length is equal to $\frac{10}{\sqrt{5}}$. Additionally, the vector of initial actions of agents, $\boldsymbol{x}_0$, is assumed to be uniformly distributed on $\mathscr{R}$.

Figure 2 illustrates the expected time required for $\boldsymbol{x}_k$ to settle inside $\mathrm{B}\left(\boldsymbol{x}_{\mathrm{NE}}, r\right)$ as a function of r for different quantization schemes and different values of average aggregate data rates $R_{\mathscr{D}}$. In this figure, θ, d and α are set to 10^{-2}, 10 and 0.46, respectively. As r becomes large, the ES algorithm under both $\mathscr{D}_{\mathrm{u}}$ and $\mathscr{D}_{\mathrm{a}}$ requires less time to settle inside $\mathrm{B}\left(\boldsymbol{x}_{\mathrm{NE}}, r\right)$, and as a result, the expected time, under both $\mathscr{D}_{\mathrm{u}}$ and $\mathscr{D}_{\mathrm{a}}$, decreases as r becomes large. As shown in Fig. 2, the expected time under the fixed quantization scheme $\mathscr{D}_{\mathrm{u}}$ is limited by the upper bound provided by Theorem 7. According Fig. 2, the ES algorithm under the EA quantization scheme $\mathscr{D}_{\mathrm{a}}$, on average, requires less time to settle inside $\mathrm{B}\left(\boldsymbol{x}_{\mathrm{NE}}, r\right)$ compared to the fixed quantization scheme $\mathscr{D}_{\mathrm{u}}$. The fast convergence of the ES algorithm under $\mathscr{D}_{\mathrm{a}}$ is due to the flexible structure of the EA quantization scheme $\mathscr{D}_{\mathrm{a}}$.

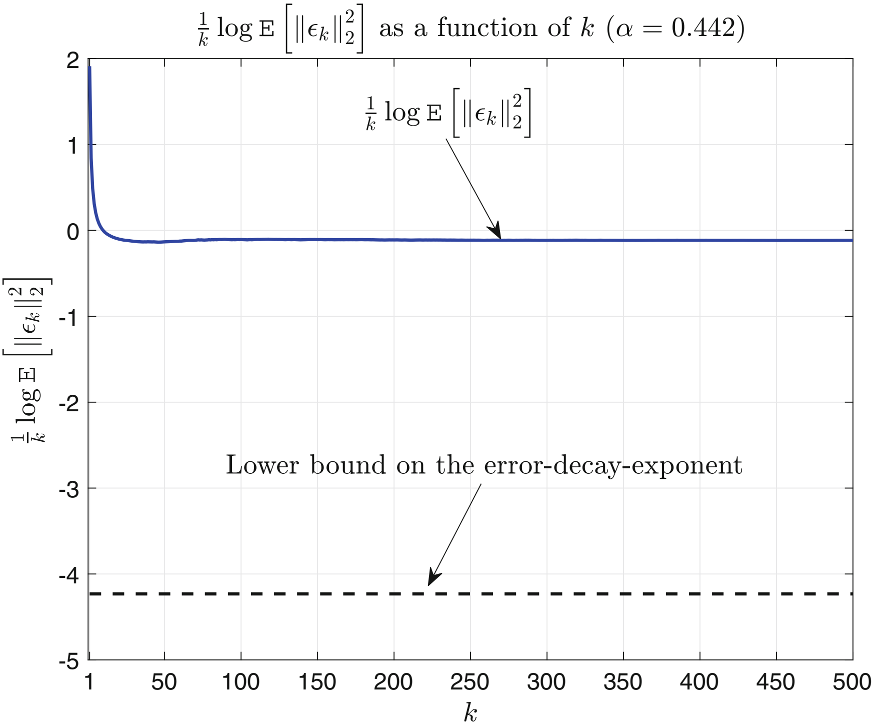


Fig. 3 log-mean-square-error-norm divided by k under the adaptive quantization scheme $\mathscr{D}_{\mathrm{a}}$ as a function of k

In Fig. 2, the ES algorithm under $\mathscr{D}_{\mathrm{u}}$ with coarse quantization scheme $R_{\mathscr{D}} = 25$ cannot settle inside the ball $\mathrm{B}\left(\boldsymbol{x}_{\mathrm{NE}}, r\right)$ when r is small which is due to the large amount of distortion caused by the coarse quantization scheme. However, as r becomes large, the ES algorithm under the coarse quantization scheme settles inside the ball $\mathrm{B}\left(\boldsymbol{x}_{\mathrm{NE}}, r\right)$ faster than the fine quantization scheme $R_{\mathscr{D}} = 60$. Note that any quantization scheme introduces an extra displacement to the agents' action at each time-step. A coarse quantization scheme causes a bigger displacement, compared to a fine quantization scheme, which results in a lower expected time to settle inside $\mathrm{B}\left(\boldsymbol{x}_{\mathrm{NE}}, r\right)$ (when r is large enough).

Figure 3 demonstrates the behavior of log-mean-square-error-norm divided by k, i.e., $\frac{1}{k} \log \mathsf{E}\left[\|\boldsymbol{\varepsilon}_k\|_2^2\right]$, under the adaptive quantizer $\mathscr{D}_{\mathrm{a}}$ as a function of the number of time-steps. As Fig. 3 shows $\frac{1}{k} \log \mathsf{E}\left[\|\varepsilon_k\|_2^2\right]$ stays above -4.2318, the predicted lower bound by Theorem 6, as k becomes large.

4 Conclusion

This chapter presented a set of results on the convergence behavior of a quantized primal-dual (PD) algorithm as well as a gradient-based Nash seeking algorithm under quantized inter-agent communications. First, using the information-theoretic notion of entropy power, universal bounds were derived on the fastest speed of exponential mean square convergence of PD, primal, and dual variables to the optimal solution under optimum achieving quantization schemes. These results highlight the universal trade-offs between the speed of convergence of the quantized PD algorithm, data rate under the quantization, objective functions of agents, the number of agents, and the number of constraints. Next, universal lower bounds were established on the mean square distance of PD, primal, and dual variables from the optimal solution of the NUM problem for any finite time index.

Subsequently, the impact of quantized inter-agent communications on the convergence behavior of the gradient-based Nash equilibrium seeking (ES) algorithm in non-cooperative games was studied. The information-theoretic notion of entropy power helped establishing a universal lower bound on the rate of exponential mean square convergence of such algorithms, assuming equilibrium-achieving quantizers. This lower bound signifies the impact of inter-agent communication data rates on the convergence speed of the ES algorithm to the Nash equilibrium (NE). Next, the transient behavior of the ES algorithm under quantized message passing among agents was examined. To this end, an upper bound was derived on the expected time required for the ES algorithm to settle inside a ball centered at the NE under a uniform quantization scheme. Moreover, an upper bound was obtained on the probability that agents' actions at a given time lie outside a ball around the Nash equilibrium. It is worth noting that these last two results only concern the behavior of the ES algorithm until it reaches a neighborhood of the Nash equilibrium, and do not make any assumption on the convergence of the ES algorithm.

Acknowledgements The authors would like to thank Prof. Girish Nair from The University of Melbourne for his contributions and fruitful discussions. This work was supported by the Australian Research Councils Discovery Projects funding scheme (DP140100819).

Appendix

Proof of Theorem 1

This appendix presents the main steps of the proof of Theorem 1. To this end, first, the notion of conditional differential entropy power of a random vector is defined. Then, the notion of entropy power facilitates establishing a universal lower bound on the DDE of the PD variables. The differential entropy power of the random vector $z \in \mathbb{R}^{N+M}$ conditioned on the event $A = a$, denoted by $\mathsf{N}\left[z\right| A = a]$, is defined as

$$\mathsf{N}\left[z\right| A = a] = \frac{1}{2\pi \mathrm{e}} \mathrm{e}^{\frac{2}{M+N} \mathsf{h}[z|A=a]},$$

where $\mathsf{h}\left[z\right| A = a]$ is the conditional differential entropy of z given $A = a$ defined as

$$\mathsf{h}\left[z\right| A = a] = -\int \log\left(p\left(z\right| A = a)\right) p\left(z\right| A = a)\, dz,$$

where $p\left(z\right| A = a)$ is the conditional distribution of z given $A = a$. Using the entropy maximizing property of Gaussian distributions, the conditional entropy power of z given $A = a$ can be upper bounded [1] as

$$\mathsf{N}\left[z\right| A = a] \leq \mathrm{e}^{1/(M+N)-1} \mathsf{E}\left[\left\|z\right\|_2^2 \right| A = a\Big], \tag{30}$$

where $\mathsf{E}\left[z\right| A = a]$ is conditional expectation of z given $A = a$. Let $\mathsf{E}_A\left[\mathsf{N}\left[z\right| A = a]\right]$ denote the average conditional entropy power of z given $A = a$. Using (30), $\mathsf{E}_A\left[\mathsf{N}\left[z\right| A = a]\right]$ can be upper bounded as

$$\mathsf{E}_A\left[\mathsf{N}\left[z\right| A]\right] \leq \mathrm{e}^{1/(M+N)-1} \mathsf{E}\left[\left\|z\right\|_2^2 \right]. \tag{31}$$

Next, the inequality (31) is used to establish the universal lower bound on the DDE of the PD variables under OA quantization schemes. To this end, let $\mathscr{D}_{k-1} = \left\{ \hat{Q}_n = \hat{\boldsymbol{q}}_n \right\}_{n=0}^{k-1}$ where $\hat{Q}_n = \left[\hat{Q}^{\boldsymbol{x}}_{1,n}, \ldots, \hat{Q}^{\boldsymbol{x}}_{M,n}, \hat{Q}^{\boldsymbol{\lambda}}_{1,n}, \ldots, \hat{Q}^{\boldsymbol{\lambda}}_{N,n} \right]$ and $\hat{\boldsymbol{q}}_n$ is a possible realization of $\hat{Q}_n$. Using (31), $\mathsf{E}\left[\left\|\boldsymbol{\varepsilon}_k\right\|_2^2 \right]$ can be lower bounded as $\frac{\mathrm{e}^{1-\frac{1}{M+N}}}{2\pi \mathrm{e}} \mathrm{e}^{\frac{2}{M+N} \mathsf{E}\left[\mathsf{h}\left[\boldsymbol{\varepsilon}_k | \mathscr{D}_{k-1}\right]\right]}$

$$\begin{aligned}\mathsf{E}\left[\|\boldsymbol{\varepsilon}_k\|_2^2\right] &\geq \mathrm{e}^{1-\frac{1}{M+N}}\,\mathsf{E}\left[\mathsf{N}\left[\boldsymbol{\varepsilon}_k \,|\, \mathscr{D}_{k-1}\right]\right]\\ &\overset{(*)}{\geq} \frac{\mathrm{e}^{1-\frac{1}{M+N}}}{2\pi\,\mathrm{e}}\,\mathrm{e}^{\frac{2}{M+N}\mathsf{E}\left[\mathsf{h}\left[\,\boldsymbol{\varepsilon}_k | \mathscr{D}_{k-1}\right]\right]},\end{aligned} \tag{32}$$

where $(*)$ is obtained using the Jensen inequality. The term $\mathsf{h}\left[\boldsymbol{\varepsilon}_k \,|\, \mathscr{D}_{k-1}\right]$ on the right hand side of (32) can be expanded as

$$\begin{aligned}\mathsf{h}\left[\boldsymbol{\varepsilon}_k \,|\, \mathscr{D}_{k-1}\right] &= \mathsf{h}\left[\boldsymbol{y}_k - \boldsymbol{y}^{\star} \,\middle|\, \mathscr{D}_{k-1}\right]\\ &\overset{(*)}{=} \mathsf{h}\left[\boldsymbol{y}_k \,\middle|\, \mathscr{D}_{k-1}\right],\end{aligned} \tag{33}$$

where $(*)$ follows from the translation invariance property of differential entropy as $\boldsymbol{y}^{\star}$ is a constant vector (see [17] Theorem 8.6.3 page 253).

The next lemma establishes a useful expression between $\mathsf{h}\left[\boldsymbol{y}_n \,\middle|\, \mathscr{D}_{k-1}\right]$ and $\mathsf{h}\left[\boldsymbol{y}_{n-1} \,\middle|\, \mathscr{D}_{k-1}\right]$ for $n \leq k$, which is used to further expand $\mathsf{h}\left[\boldsymbol{y}_k \,\middle|\, \mathscr{D}_{k-1}\right]$.

Lemma 1 *For $n \leq k$, $\mathsf{h}\left[\boldsymbol{y}_n \,\middle|\, \mathscr{D}_{k-1}\right]$ can be expanded as*

$$\mathsf{h}\left[\boldsymbol{y}_n \,\middle|\, \mathscr{D}_{k-1}\right] = \mathsf{h}\left[\boldsymbol{y}_{n-1} \,\middle|\, \mathscr{D}_{k-1}\right] + \mathsf{E}\left[\sum_{j=1}^{M} \log\left(1 + \mu_{n-1}\,\frac{d^2}{d{x^j}^2} U_j\left(\boldsymbol{x}^j_{n-1}\right)\right)\middle|\, \mathscr{D}_{k-1}\right] \tag{34}$$

Proof. Let $\tilde{x}^i_n = x^i_n + \mu_n\left(\frac{d}{dx^i} U_i\left(x^i_n\right)\right)$ and $\tilde{\boldsymbol{x}}_n = \left[\tilde{x}^i_1, \ldots, \tilde{x}^i_M\right]^{\top}$. Let $\tilde{\boldsymbol{y}}_n$ be the vector concatenation of $\tilde{\boldsymbol{x}}_n$ and $\boldsymbol{\lambda}_n$. This lemma is proved in two steps. First, it is shown that the conditional differential entropy of $\boldsymbol{y}_n$ given $\mathscr{D}_k$ is equal to that of $\tilde{\boldsymbol{y}}_{n-1}$ given $\mathscr{D}_k$ (see (35)). Next, a relation between the conditional differential entropy of $\tilde{\boldsymbol{y}}_{n-1}$ given $\mathscr{D}_k$ and that of $\boldsymbol{y}_{n-1}$ given $\mathscr{D}_k$ is established. Note that, $\mathsf{h}\left[\boldsymbol{y}_n \,\middle|\, \mathscr{D}_{k-1}\right]$ can be written as

$$\begin{aligned}\mathsf{h}\left[\boldsymbol{y}_n \,\middle|\, \mathscr{D}_{k-1}\right] &= \mathsf{h}\left[\boldsymbol{x}_n, \boldsymbol{\lambda}_n \,|\, \mathscr{D}_{k-1}\right]\\ &\overset{*}{=} \mathsf{h}\left[\tilde{\boldsymbol{x}}_{n-1}, \boldsymbol{\lambda}_{n-1} \,\middle|\, \mathscr{D}_{k-1}\right]\\ &= \mathsf{h}\left[\tilde{\boldsymbol{y}}_{n-1} \,\middle|\, \mathscr{D}_{k-1}\right]\end{aligned} \tag{35}$$

where $(*)$ follows from the translation invariance property of the differential entropy and the fact that Q_{k-1} is fixed given $\mathscr{D}_{k-1} = \left\{\hat{Q}_n = \hat{\boldsymbol{q}}_n\right\}_{n=0}^{k-1}$. Next, we derive an expression for the probability density function (PDF) of $\tilde{\boldsymbol{y}}_n$ in terms of the PDF of $\boldsymbol{y}_n$. Let $p_{\tilde{\boldsymbol{y}}_n}\left(\boldsymbol{y} \,|\mathscr{D}_{k-1}\right)$ and $p_{\boldsymbol{y}_n}\left(\boldsymbol{y} \,|\mathscr{D}_{k-1}\right)$ to denote the PDFs of $\tilde{\boldsymbol{y}}_n$ and $\boldsymbol{y}_n$, respectively, conditioned on $\mathscr{D}_{k-1}$. Let $\boldsymbol{F}\left(\cdot\right)$ represent the mapping between $\tilde{\boldsymbol{y}}_n$ and $\boldsymbol{y}_n$, i.e., $\tilde{\boldsymbol{y}}_n = \boldsymbol{F}\left(\boldsymbol{y}_n\right)$. Note that $0 < 1 + \mu_n \frac{d^2}{d{x^i}^2} U_i\left(x^i\right) < 1$ since $0 < \mu_n < \min_i \frac{1}{\left|U_i^{\min}\right|}$ which implies that the mapping $\boldsymbol{F}\left(\cdot\right)$ is invertible. Thus, the change-of-variables formula for invertible diffeomorphisms of random vectors (see e.g., (4.63) in [18]) can be applied to write

$$p_{\tilde{\boldsymbol{y}}_{n-1}}(\boldsymbol{y}\,|\mathscr{D}_{k-1}) = \frac{1}{\det J_{\boldsymbol{F}}\big[\boldsymbol{F}^{-1}(\boldsymbol{y})\big]}\, p_{\boldsymbol{y}_{n-1}}\big(\boldsymbol{F}^{-1}(\boldsymbol{y})\,|\mathscr{D}_{k-1}\big), \tag{36}$$

where $J_{\boldsymbol{F}}[\boldsymbol{x}]$ is Jacobian of $\boldsymbol{F}(\boldsymbol{x})$ evaluated at $\boldsymbol{x}$. Using (36), the conditional entropy of $\tilde{\boldsymbol{y}}_{n-1}$ given $\mathscr{D}_{k-1}$ can be written as

$$\begin{aligned}
\mathsf{h}\left[\tilde{\boldsymbol{y}}_{n-1}\middle|\,\mathscr{D}_{k-1}\right] &= \int \log\left(\det J_{\boldsymbol{F}}\left[\boldsymbol{F}^{-1}(\boldsymbol{y})\right]\right) \frac{1}{\det J_{\boldsymbol{F}}\left[\boldsymbol{F}^{-1}(\boldsymbol{y})\right]}\, p_{\boldsymbol{y}_{n-1}}\left(F^{-1}(\boldsymbol{y})\,|\mathscr{D}_{k-1}\right) d\boldsymbol{y} \\
&\quad - \int \log\left(p_{\boldsymbol{y}_{n-1}}\left(F^{-1}(\boldsymbol{y})\,|\mathscr{D}_{k-1}\right)\right) \frac{1}{\det J_{\boldsymbol{F}}\left[\boldsymbol{F}^{-1}(\boldsymbol{y})\right]}\, p_{\boldsymbol{y}_{n-1}}\left(\boldsymbol{F}^{-1}(\boldsymbol{y})\,|\mathscr{D}_{k-1}\right) d\boldsymbol{y}, \\
&\overset{(*)}{=} \int \log\left(\det J_{\boldsymbol{F}}[\boldsymbol{z}]\right) p_{\boldsymbol{y}_{n-1}}(\boldsymbol{z}\,|\mathscr{D}_{k-1})\, d\boldsymbol{z} - \int \log\left(p_{\boldsymbol{y}_{n-1}}(\boldsymbol{z}\,|\mathscr{D}_{k-1})\right) p_{\boldsymbol{y}_{n-1}}(\boldsymbol{z}\,|\mathscr{D}_{k-1})\, d\boldsymbol{z}, \\
&= \sum_{j=1}^{M} \mathsf{E}\left[\log\left(1+\mu_{n-1}\frac{d^2}{d{x^j}^2}U_j\left(x_{n-1}^{j}\right)\right)\middle|\,\mathscr{D}_{k-1}\right] + \mathsf{h}\left[\boldsymbol{y}_{n-1}\middle|\,\mathscr{D}_{k-1}\right],
\end{aligned} \tag{37}$$

where $(*)$ follows from the change of variable $\boldsymbol{z} = F^{-1}(\boldsymbol{x})$.

Using Lemma 1, $\mathsf{h}\left[\boldsymbol{y}_k\middle|\,\mathscr{D}_{k-1}\right]$ can be further expanded as

$$\mathsf{h}\left[\boldsymbol{y}_k\middle|\,\mathscr{D}_{k-1}\right] = \mathsf{h}\left[\boldsymbol{y}_0\middle|\,\mathscr{D}_{k-1}\right] + \sum_{j=1}^{M}\sum_{n=0}^{k-1} \mathsf{E}\left[\log\left(1+\mu_n\frac{d^2}{d{x^j}^2}U_j\left(x_n^j\right)\right)\middle|\,\mathscr{D}_{k-1}\right] \tag{38}$$

Using (38), $\mathsf{E}\left[\mathsf{h}\left[\boldsymbol{y}_k\middle|\,\mathscr{D}_{k-1}\right]\right]$ can be written as

$$\mathsf{E}\left[\mathsf{h}\left[\boldsymbol{y}_k\middle|\,\mathscr{D}_{k-1}\right]\right] = \sum_{j=1}^{M}\sum_{n=0}^{k-1} \mathsf{E}\left[\log\left(1+\mu_n\frac{d^2}{d{x^j}^2}U_j\left(x_n^j\right)\right)\right] + \mathsf{E}\left[\mathsf{h}\left[\boldsymbol{y}_0\middle|\,\mathscr{D}_{k-1}\right]\right], \tag{39}$$

The following lemma, adapted from [1], establishes a lower bound on $\mathsf{E}\left[\mathsf{h}\left[\boldsymbol{y}_k\middle|\,\mathscr{D}_{k-1}\right]\right]$:

Lemma 2 *The average conditional entropy of $\boldsymbol{y}_0$ given $\mathscr{D}_{k-1}$, i.e., $\mathsf{E}\left[\mathsf{h}\left[\boldsymbol{y}_0\middle|\,\mathscr{D}_{k-1}\right]\right]$, can be lower bounded as*

$$\mathsf{E}\left[\mathsf{h}\left[\boldsymbol{y}_0\middle|\,\mathscr{D}_{k-1}\right]\right] \geq \mathsf{h}\left[\boldsymbol{y}_0\right] - \sum_{t=0}^{k-1}\left(\left(\sum_{i=1}^{M}\log\left|\mathscr{A}_{i,t}^{\boldsymbol{x}}\right|\right) + \sum_{j=1}^{N}\log\left|\mathscr{A}_{j,t}^{\boldsymbol{\lambda}}\right|\right).$$

Proof. Follows directly from the first inequality in appendix C in [1]; alternatively, it can be derived from (8.48) and (8.89) in [17].

Applying Lemma 2 to (39) yields

$$\mathsf{E}\left[\mathsf{h}\left[\,\boldsymbol{y}_k \,\middle|\, \mathscr{D}_{k-1}\right]\right] \geq \sum_{j=1}^{M}\sum_{n=0}^{k-1}\mathsf{E}\left[\log\left(1+\mu_n\frac{d^2}{dx^{j^2}}U_j\left(x_n^j\right)\right)\right] + \mathsf{h}\left[\boldsymbol{y}_0\right] - \sum_{t=0}^{k-1}\left(\left(\sum_{i=1}^{M}\log\left|\mathscr{A}_{i,t}^{\boldsymbol{x}}\right|\right) + \sum_{j=1}^{N}\log\left|\mathscr{A}_{j,t}^{\boldsymbol{\lambda}}\right|\right), \tag{40}$$

Since $\boldsymbol{x}_0$ and $\boldsymbol{\lambda}_0$ are independent, the differential entropy of $\boldsymbol{y}_0$ can be written as $\mathsf{h}\left[\boldsymbol{y}_0\right] = \mathsf{h}\left[\boldsymbol{x}_0\right] + \mathsf{h}\left[\boldsymbol{\lambda}_0\right]$ which implies that $\boldsymbol{y}_0$ has finite differential entropy. Using (32), (33), (40) and the fact that $\boldsymbol{y}_0$ has a finite entropy, the DDE can be lower bounded as

$$\liminf_{k\longrightarrow\infty}\frac{1}{k}\log\mathsf{E}\left[\|\boldsymbol{\varepsilon}_k\|_2^2\right] \geq \frac{2}{M+N}\left(\liminf_{k\longrightarrow\infty}\sum_{j=1}^{M}\frac{1}{k}\sum_{n=0}^{k-1}\mathsf{E}\left[\log\left(1+\mu_n\frac{d^2}{dx^{j^2}}U_j\left(x_n^j\right)\right)\right] - R_{\mathscr{Q}}\right). \tag{41}$$

The next lemma presents the asymptotic behavior of the first term in the right hand side of equation (41).

Lemma 3 ([10]) *Consider the primal-dual update rule* (6) *under an OA quantization scheme. Then,*

$$\lim_{k\longrightarrow\infty}\sum_{j=1}^{M}\frac{1}{k}\sum_{n=0}^{k-1}\mathsf{E}\left[\log\left(1+\mu_n\frac{d^2}{dx^{j^2}}U_j\left(x_n^j\right)\right)\right] = \sum_{j=1}^{M}\log\left(1+\mu^\star\frac{d^2}{dx^{j^2}}U_j\left(x^{j^\star}\right)\right).$$

Applying Lemma 3 to (41) yields

$$\liminf_{k\to\infty}\frac{1}{k}\log\mathsf{E}\left[\|\boldsymbol{\varepsilon}_k\|_2^2\right] \geq \frac{2}{N+M}\left(\sum_{i=1}^{m}\log\left(1+\mu^\star\frac{d^2}{dx^{i^2}}U_i\left(x^{i^\star}\right)\right) - R_{\mathscr{Q}}\right). \tag{42}$$

which completes the proof.

References

1. G. N. Nair and R. J. Evans, "Stabilizability of Stochastic Linear Systems with Finite Feedback Data Rates," *SIAM Journal on Control and Optimization*, vol. 43, no. 2, pp. 413–436, 2004.
2. G. N. Nair, F. Fagnani, S. Zampieri, and R. J. Evans, "Feedback Control Under Data Rate Constraints: An Overview," *Proceedings of the IEEE*, vol. 95, no. 1, pp. 108–137, 2007.
3. E. Nekouei, T. Alpcan, G. Nair, and R. J. Evans, "Convergence Analysis of Quantized Primal-dual Algorithms in Network Utility Maximization Problems," *IEEE Transactions on Control of Network Systems*, vol. PP, no. 99, pp. 1–1, 2016.
4. E. Nekouei, G. N. Nair, and T. Alpcan, "Performance Analysis of Gradient-Based Nash Seeking Algorithms Under Quantization," *IEEE Transactions on Automatic Control*, vol. 61, no. 12, pp. 3771–3783, 2016.
5. F. Kelly, A. Maulloo, and D. Tan, "Rate control in communication networks: shadow prices, proportional fairness and stability," in *Journal of the Operational Research Society*, vol. 49, 1998.

6. S. Shakkottai and R. Srikant, “Network Optimization and Control,” *Found. Trends Netw.*, vol. 2, no. 3, pp. 271–379, 2007.
7. A. Nedić, A. Olshevsky, A. Ozdaglar, and J. N. Tsitsiklis, “Distributed subgradient methods and quantization effects,” in *47th IEEE Conference on Decision and Control (CDC)*, Dec 2008, pp. 4177–4184.
8. P. Yi and Y. Hong, “Quantized Subgradient Algorithm and Data-Rate Analysis for Distributed Optimization,” *IEEE Transactions on Control of Network Systems*, vol. 1, no. 4, pp. 380–392, 2014.
9. J. S. Freudenberg, R. H. Middleton, and V. Solo, “Stabilization and Disturbance Attenuation Over a Gaussian Communication Channel,” *IEEE Transactions on Automatic Control*, vol. 55, no. 3, pp. 795–799, 2010.
10. E. Nekouei, T. Alpcan, G. Nair, and R. J. Evans, “Convergence Analysis of Quantized Primal-dual Algorithm in Network Utility Maximization Problems,” arXiv:1604.00723, Tech. Rep., Apr 2016.
11. C. U. Saraydar, N. B. Mandayam, and D. J. Goodman, “Efficient power control via pricing in wireless data networks,” *IEEE Transactions on Communications*, vol. 50, no. 2, pp. 291–303, 2002.
12. J. R. Marden and J. S. Shamma, “Chapter 16 - Game Theory and Distributed Control,” ser. Handbook of Game Theory with Economic Applications, H. P. Young and S. Zamir, Eds. Elsevier, 2015, vol. 4, pp. 861–899.
13. N. D. Stein, “Characterization and Computation of Equilibria in Infinite Games,” Master’s thesis, M.I.T., June 2007.
14. S. Li and T. Basar, “Distributed algorithms for the computation of noncooperative equilibria,” *Automatica*, vol. 23, no. 4, pp. 523–533, 1987.
15. J. B. Rosen, “Existence and Uniqueness of Equilibrium Points for Concave N-Person Games,” *Econometrica*, vol. 33, no. 3, pp. 520–534, 1965.
16. E. Nekouei, T. Alpcan, and D. Chattopadhyay, “Game-Theoretic Frameworks for Demand Response in Electricity Markets,” *IEEE Transactions on Smart Grid*, vol. 6, no. 2, pp. 748–758, 2015.
17. T. M. Cover and J. A. Thomas, *Elements of Information Theory*. Wiley-Interscience, 2006.
18. A. Leon-Garcia, *Probability, Statistics, and Random Processes for Electrical Engineering*, 2nd ed. Massachusetts: Addison-Wesley, 1994.

Fault Diagnosis for Uncertain Networked Systems

Francesca Boem, Christodoulos Keliris, Thomas Parisini and Marios M. Polycarpou

Abstract Fault diagnosis has been at the forefront of technological developments for several decades. Recent advances in many engineering fields have led to the networked interconnection of various systems. The increased complexity of modern systems leads to a larger number of sources of uncertainty which must be taken into consideration and addressed properly in the design of monitoring and fault diagnosis architectures. This chapter reviews a model-based distributed fault diagnosis approach for uncertain nonlinear large-scale networked systems to specifically address: (a) the presence of measurement noise by devising a filtering scheme for dampening the effect of noise; (b) the modeling of uncertainty by developing an adaptive learning scheme; (c) the uncertainty issues emerging when considering networked systems such as the presence of delays and packet dropouts in the communication networks. The proposed architecture considers in an integrated way the various components of complex distributed systems such as the physical environment, the sensor level, the fault diagnosers, and the communication networks. Finally, some actions taken after the detection of a fault, such as the identification of the fault location and its magnitude or the learning of the fault function, are illustrated.

This work has been supported by the European Union's Horizon 2020 research and innovation programme under grant agreement No 739551 (KIOS CoE).

F. Boem
University College London, Torrington Place, London WC1E 7JE, UK
e-mail: f.boem@ucl.ac.uk

C. Keliris · T. Parisini · M. M. Polycarpou
KIOS Research and Innovation Center of Excellence, University of Cyprus, 1 Panepistimiou Avenue, 2109 Aglantzia, Nicosia, Cyprus
e-mail: keliris.chris@gmail.com

M. M. Polycarpou
e-mail: mpolycar@ucy.ac.cy

T. Parisini (✉)
Imperial College London, London, UK
e-mail: t.parisini@gmail.com

T. Parisini
University of Trieste, Trieste, Italy

© Springer Nature Switzerland AG 2018

T. Başar (ed.), *Uncertainty in Complex Networked Systems*,
Systems & Control: Foundations & Applications,
https://doi.org/10.1007/978-3-030-04630-9_16

1 Introduction: From Centralized to Distributed Fault Diagnosis

In systems and control engineering, the adoption of models describing the behavior of systems is ubiquitous and of fundamental importance. However, such models are usually affected by some uncertainty and the sources of uncertainty may vary quite a lot. For instance, the derivation of an accurate mathematical model may be very difficult to obtain or even entail increased financial costs and so, less accurate models are used. Other sources of uncertainty include the measurement noise, the system disturbances, and the changing system parameters due to the components degradation over time. The presence of uncertainty is especially important when considering complex large-scale systems, such as Systems of Systems (SoS) [79] or Cyber-Physical Systems (CPS) [4], where it is difficult to understand and model the relationships that exist among the (possibly large) number of interconnected subsystems. Therefore, uncertainty represents an important challenge for many control applications, thus motivating the research and the development of robust methods able to manage its presence and effect on the control task performance [25, 67, 97, 109]. In some situations, the mismatch between the considered model and the actual system behavior becomes major, due to the presence of undesired or unexpected behaviors, possibly leading to negative consequences such as instabilities, failures in the system, or deterioration of performance. Therefore, it is important to take into consideration modeling uncertainty at the design stage, so that if any unexpected behavior is observed during the system operation, it will be feasible to identify the presence of a fault, avoiding, at the same time, the occurrence of false alarms.

Reliability is a key requirement for modern systems. It can be defined as the ability of a system to perform its intended function over a given period of time [7]. The inability to perform the intended function is called a failure, and it can be due to the effects of a fault. A fault is a change in the behavior of a system, or part of it, from the behavior that was set at design time.

As practical systems become more complex and more interconnected, the need for enhanced robustness, fault tolerance and sustainability becomes of essential importance. Potential faults could lead to major catastrophes and consequently could trigger a chain of failing dependent systems, such as electric power systems, communication and water networks, along with production plants, causing tremendous economic and social damage. Therefore, safe and reliable operation of such systems through the early detection of any "small" fault before they become serious failures is a crucial component of the overall system performance and sustainability.

For these reasons, fault diagnosis is a research field that has been in the forefront of the technological evolution for a few decades and has attracted the attention from the research and industrial communities, as testified by many important survey papers [33, 37, 43, 99–101] and books [9, 18, 44, 65].

Generally, fault diagnosis is comprised of several steps: *detection* of a fault, *isolation* and *identification* of the fault and fault *accommodation*, or reconfiguration of the system.

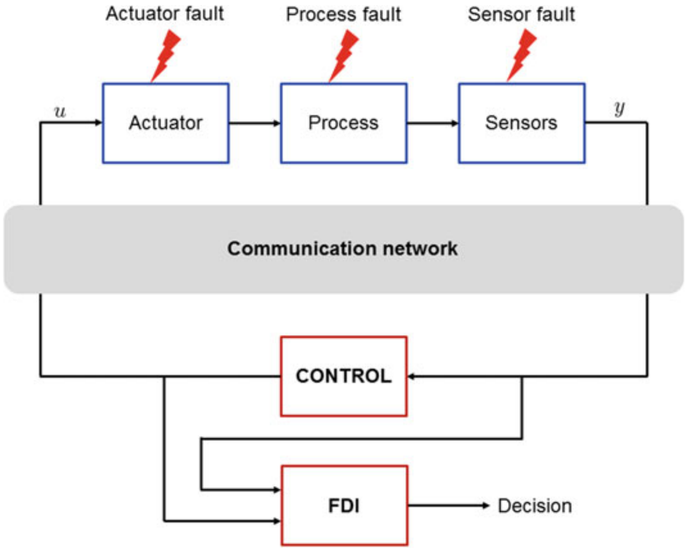


Fig. 1 Fault types and FDI

Fault detection consists of understanding whether a fault has occurred or not, while the isolation task refers to pinpointing the type of fault and its location. Fault identification is an extra step that is carried on after isolation in order to quantify the extent to which a fault is present. Fault accommodation addresses the problem of how the system actively responds to the fault: for example, after a successful fault diagnosis, the controller parameters may be adjusted to accommodate changed plant dynamics in order to prevent failure at the system level.

A control system is comprised of mainly three parts: the actuators, the plant components, and the sensors, therefore a fault may appear in any of these (see Fig. 1). Specifically, process faults (on the plant components) alter the dynamics of the system, sensor faults alter the measurement readings and actuator faults modify the controllers' influence on the system.

Apart from the fault source, we can further distinguish between abrupt or incipient faults. *Abrupt faults* are sudden, step-like changes that appear almost instantaneously and can lead to immediate component or even general system failure. On the other hand, *incipient faults* are slowly developing faults that occur due to parameter changes of the components because of their continuous operation and diminishing lifetime. These changes develop slowly and are initially small, thus harder to detect and may be better prevented through system maintenance.

There are mainly two methods to address the possible presence of a fault. The first one is *physical redundancy* (or hardware redundancy), that is the fact that critical components of the system are replicated in a greater number than what is strictly necessary. This is effective but implies a highly expensive solution and can be justified only for critical, potentially life-threatening systems (i.e., aviation applications).

The second method is the *analytical redundancy* approach which is based on a mathematical model of the system under healthy system behavior. In this approach, the actual physical signals that are measured, are compared to the corresponding signals given by the mathematical model of the process under healthy state; their difference constitutes the *residuals* (*residual generation stage*). Under the ideal conditions of no faults, no modeling uncertainties and no measurement noise nor disturbances, the residuals are zero. In real applications, after the residual generation stage, the information given by residuals is processed to take a decision regarding the health status of the system and determine the potential occurrence of faults (*decision making stage*). If the fault decision is positive, then further analysis is conducted to identify the fault's type and location, and possibly its size. Although this approach is more affordable, it is computationally intensive and may be sensitive to false alarms due to inaccuracies in the mathematical modeling of the system which may be mistakenly passed as faults. This model-based approach was born during the 1970s thanks to the seminal works of Beard, Jones and Clark [5, 22, 47] among others (see the survey papers [33, 37, 45, 100]).

An alternative approach to model-based methods is represented by the signal-based techniques, in which known features of signals, such as spectral components or statistical features, are compared to nominal ones [37, 44]. These methods though, require some knowledge of previous behavior of the system during healthy operation and that is the reason why they are classified into the wider class of process history fault diagnosis approaches (i.e., see [99] and the references therein).

Under the analytical redundancy framework, there are various methods to generate the residual vector, which can be divided into two main approaches: the state estimation techniques (such as parity space approach, observer schemes, and detection filters) and the parameter identification techniques. Moreover, in order to ease the fault isolation task, residuals can be designed so as to contain specific isolation properties. The main residual enhancement techniques are represented by structured and directional residuals [38, 100]. In the *structured residuals* scheme, each fault affects a specific subset of the residuals and any residual responds only to a specific subset of faults. Therefore, due to the dependence of the residuals on the faults, certain patterns appear on the residual vector that can be used for fault isolation. In the *directional residuals* scheme, each fault amounts to a specific direction in the residual space, and thus fault isolation is concluded by selecting the direction that the generated residual vector lies closest to. More information regarding these techniques can be found in the books by Gertler [39] and Isermann [44]. In the literature, many methods have been proposed for the generation of residuals, which can mainly be classified according the following approaches:

- *Parity space approach*. This method consists of checking the consistency of the mathematical equations by using the actual measurements: a fault is declared whenever predetermined error thresholds are exceeded. Further information can be found in [38] and the references therein.
- *Observer schemes*. In this category lie many approaches, starting from the *Fault detection filter* (FDF), first proposed by Beard and Jones in the early 70s, to the

Diagnostic Observer approach, which has been widely adopted in the literature. According to this approach, observers are used to reconstruct the output $\hat{y}$ of the system from measurements y and the residual is represented by the output estimation error $e = y - \hat{y}$. In the case of stochastic systems, the observers may be substituted by Kalman filters and the residual is the innovation which under the fault-free case should be white noise with zero mean and known covariance. The isolation of faults can be enhanced with the use of a bank of residual generators under the *Dedicated Observer Scheme (DOS)* proposed by Clark [22] or the *Generalized Observer Scheme (GOS)* [33, 34]. In both schemes as many residuals as the number of possible faults are generated. The difference is that in the DOS scheme, each residual is sensitive to only a single fault, while in the GOS, each residual is sensitive to every but one fault. The DOS scheme is appealing as it can also isolate concurrent faults, but it cannot always be designed. Instead, the GOS can be always applied, but can only isolate non-concurrent faults. It is important to note that, as pointed out in [34], the observers used in fault diagnosis are primarily output observers which simply reconstruct the measurable part of the state variables, rather than state observers which are required for control purposes. The use of state observers for nonlinear systems has not been used extensively for the FDI problem, even though analytical results regarding the stability of the nonlinear observers and design procedures have been established. The main issue with the observer approach is that the design of observers for nonlinear systems with asymptotically stable error dynamics is not an easy task even when the nonlinearities are fully known. As a result, the research in fault diagnosis for nonlinear systems utilizing state observers is more limited [1, 36, 41, 51].

- *Parameter estimation*. This method is particularly suited to the detection of incipient faults and it is extensively studied in the survey papers by Isermann [45] and Frank [33] and the books by Patton et al. [65] and Isermann [44]. Using system identification methods (utilizing the input and output signals), the parameters of a mathematical model of the system can be obtained (recursively and online) across different time intervals and compared to their respective values based on a nominal model. Any significant difference could indicate the occurrence of a fault and a relation between parameter changes and faults can be formed with the use of pattern recognition methods.

An important aspect to be considered when monitoring controlled systems relates to the possibly *conflicting* dynamic behaviors of the FDI scheme and the reconfigurable controller, namely the feedback controller may hide the presence of faults by compensating their effects (see as example the simulation analysis in [78]) thus making the FDI task much more difficult or even impossible [3, 21, 35, 100]. This is particularly eminent in passive FDI methods, in which the status of health of the system is analyzed by comparing input–output data for the closed-loop system with a process model or historical data. A possible solution has been proposed for this problem when considering application use-cases allowing to affect the closed-loop dynamics by acting at run time on the control inputs. This paves the way to the so-called *active FDI methodologies*. Active FDI approaches consist of suitably modifying the control input to improve fault detectability and isolability capabilities [2, 6, 20, 42, 71, 73, 82, 87, 92]. The typical main limitation of active FDI techniques concerns high computational cost and complexity. This drawback restricts quite a bit the applicability of this approach to low-dimensional systems [30, 73, 85, 86, 104, 105], even though some approaches have been suggested in the literature to alleviate the computational complexity (see as examples [6, 62]).

An obvious problem in the practical implementation of model-based FDI schemes consists of deriving accurate mathematical models of engineering systems. This is a challenging task and thus, due to the presence of uncertainties and modeling errors, the resulting residual vectors are never identically zero. In addition, generally in the literature, the presence of measurement noise and modeling uncertainty is often overseen. In most real-world applications, such uncertainties may influence significantly the performance of fault detection schemes by causing, for example, false alarms. Therefore, bounds on the residuals must be defined, but still the proper choice remains a major problem. If bounds are chosen too narrow, this may lead to false alarms, whilst if they are chosen too wide faults may pass undetected. Therefore, dealing with the uncertainty in Fault Detection and Isolation architectures is of fundamental importance. As a result, there is a growing demand for robust residual generation to reduce the sensitivity of the residual against the effect of modeling errors, noise and disturbances. This issue can be tackled either by the use of enhanced techniques for robust residual generation or by choosing appropriately the level of the error threshold which can also change adaptively as discussed in the book by Patton et al. [65]. A line of research tried to overcome the problem of accurate mathematical modeling by using qualitative models, where only qualitative information, such as sign or trend of measured variables, are used [101] as well as classification techniques and inference methods. A more successful approach, anyway, is based on the use of adaptive online approximators, such as neural networks as example, to learn online the unknown or uncertain parts of the system dynamical model or the fault model [15, 16, 28, 31, 50, 53, 69, 98, 107].

1.1 Distributed and Networked Large-Scale Systems

In the literature, FDI methods have been historically designed for *centralized* frameworks, where information about the state of the system is gathered and processed centrally. From a practical perspective, gathering the distributed information into a central processing unit to implement a centralized approach for the fault diagnosis task is counterproductive due to communication overload and the requirement for higher computational power. Moreover, the processing of the information at a centralized station imposes several risks since the station constitutes a *single point of failure*, thus making the architecture possibly fragile to faults. Recent advances in communications and distributed sensing have allowed the transition from centralized fault diagnosis approaches [9, 18, 33, 65, 100] toward the development of hierarchical, decentralized and distributed schemes [8, 13–15, 23, 29, 31, 40, 48, 49, 52, 54–56, 66, 75, 76, 78, 84, 89, 90, 96, 102, 108].

In many cases, a distributed FDI framework is not an option but a necessity, since many factors contribute to this formulation such as the large-scale nature of the system to be monitored, its spatial distribution, the inability to access certain parts of the system from a remote monitoring component. Specifically, recent research efforts are focused on decentralized, distributed, networked systems, Cyber-Physical Systems (CPS) [4] and Systems of Systems (SoS) [80]. Examples of these systems include power networks, water distribution networks, transportation systems, smart buildings and complex industrial plants. The term CPS refers to systems with integrated computational and physical capabilities that can interact with humans through many new modalities [4], expanding the capabilities of the physical world through computation, communication, and control. On the other hand, a SoS can be considered as a composition, made of components that are themselves systems, which is characterized by two properties that the whole must possess for it [61]: operational and managerial independence of components. This means that the component systems fulfill their own purposes and continue to operate to fulfill those purposes even if disassembled from the overall system; besides, the component systems are managed (at least in part) for their own purposes rather than the purposes of the whole.

In this chapter, we will use the term *networked* with two meanings: the considered system can be represented as a network of physically interconnected subsystems, and the monitoring agents operate and collaborate using input–output information obtained through a communication network.

When monitoring this kind of systems, distributed or decentralized algorithms are usually necessary due to computational, communication, scalability and reliability limits. The main benefits of using a distributed fault diagnosis scheme can be summarized as follows: (a) enhanced robustness of the monitoring architecture, since centralized approaches are subject to single point of failure, (b) reduced computation costs, and (c) scalability benefits; the distributed scheme allows for more flexibility in adding subsystems with respective fault detection modules requiring fewer and possibly local modifications in the already existing architecture. Moreover, an emerging requirement is the design of monitoring architectures that are robust to changes that may occur in the dynamic topology of the large-scale systems, allowing the addition/disconnection of subsystem to/from the network of interconnected subsystems only requiring local operations (see for example [11, 13, 78]).

Concerning Cyber-Physical Systems, in the literature, many contributions deal with the description of the technical challenges and design and modeling issues that need to be addressed in order to interface with these modern systems, the technological impact deriving by CPS and the requirements emerging by them [4, 46, 57–59, 74, 83, 93, 103, 106]. With regards to reliability, safety and security of CPS, some methods have been proposed ([77], including some recent works dealing with the topic of the detection of cyber-physical attacks and attacks against process control systems [17, 19, 26, 63, 64, 81, 84, 88, 95]. An interesting approach for distributed fault diagnosis is based on exploiting sensor networks [32, 110].

Another important direction of research related to the control and monitoring of large-scale distributed networked systems is the design of distributed Fault-Tolerant Control (FTC) architectures based on passive [8, 10, 78, 91] or active FDI methods [72].

1.2 Outline of the Chapter

Motivated by the issues raised above, in this chapter, we present a distributed FDI architecture specifically designed for uncertain networked nonlinear large-scale systems. We will consider different sources of uncertainty, namely modeling uncertainty, measurement noise, and network-related uncertainties, such as communication delays, packet losses, and asynchronous measurements, and the presence of possibly unknown anomalies. In Sect. 2 the problem formulation is given and the objectives and contributions of this chapter are explained in detail. In Sect. 3, the development of a fault detection scheme is presented in a continuous-time framework based on [48], where a filtering technique, which is embedded in the design of the residual and threshold signals, is used to attenuate the measurement noise. This allows for the design of tight thresholds, and thus enhances fault detectability whilst guaranteeing the absence of false alarms. This filtering approach for fault detection is rigorously investigated, providing results regarding the class of detectable faults, the magnitude of detectable faults and the filtering impact (according to the poles' location and filters' order) on the detection time.

Section 4 addresses the need for integration between the different levels composing CPS systems, which are deeply correlated in modern systems, by presenting a comprehensive architecture, based on [14], where all the parts of complex distributed systems are considered: the physical environment, the sensor level, the diagnosers layer, and the communication networks. Based on the problem formulation given in Sect. 2 and on the filtering approach explained in Sect. 3, a distributed fault diagnosis approach is designed for distributed uncertain nonlinear large-scale systems to specifically address the issues emerging when considering networked diagnosis systems, such as the presence of delays and packet dropouts in the communication networks that degrade performance and could be a source of instability, misdetection, and false alarms.

Section 5 discusses some issues regarding fault diagnosis, that is the actions taken after the detection of a fault, for identifying its location and its magnitude or even learning the fault function so that it can be used for fault accommodation schemes. Finally, in Sect. 6, some concluding remarks are given.

2 Problem Formulation

Consider a large-scale distributed nonlinear dynamic system composed of N subsystems Σ_I, $I \in \{1, ..., N\}$, each of which is described by the differential equation:

$$\Sigma_I : \begin{cases} \dot{x}_I(t) = f_I(x_I(t), u_I(t)) + g_I(x_I(t), z_I(t), u_I(t)) + \eta_I(x_I(t), z_I(t), u_I(t)) \\ \qquad +\beta_I(t - T_0)\phi_I(x_I(t), z_I(t), u_I(t)) & (1) \\ m_I(t) = x_I(t) + w_I(t), & (2) \end{cases}$$

where $x_I \in \mathbb{R}^{n_I}$, $u_I \in \mathbb{R}^{l_I}$ and $m_I \in \mathbb{R}^{n_I}$ are the state, input and measured output vectors of the I-th subsystem respectively, $z_I \in \mathbb{R}^{\bar{n}_I}$ is the vector of interconnection variables which are the state variables of the other subsystems $J \in \{1, \ldots, N\} \setminus \{I\}$ that affect the I-th subsystem, $f_I : \mathbb{R}^{n_I} \times \mathbb{R}^{l_I} \mapsto \mathbb{R}^{n_I}$ is the known local function dynamics of the I-th subsystem and $g_I : \mathbb{R}^{n_I} \times \mathbb{R}^{\bar{n}_I} \times \mathbb{R}^{l_I} \mapsto \mathbb{R}^{n_I}$ is the known part of the interconnection function between the I-th and the other subsystems. The vector function $\eta_I : \mathbb{R}^{n_I} \times \mathbb{R}^{\bar{n}_I} \times \mathbb{R}^{l_I} \mapsto \mathbb{R}^{n_I}$ is the overall modeling uncertainty associated with the known local and interconnection function dynamics and $w_I \in \mathscr{D}_{w_I} \subset \mathbb{R}^{n_I}$ ($\mathscr{D}_{w_I}$ is a compact set) represents the measurement noise. The state vectors x_I, $I \in \{1, ..., N\}$ are considered unknown whereas their noisy counterparts m_I are known. Analogously, in the case of the interconnection variable z_I, only its noisy counterpart $m_{zI}(t) = z_I(t) + \varsigma_I(t)$ is available, where $\varsigma_I(t)$ is composed by the components of w_J affecting the relevant components of m_J (as before J refers to a neighboring subsystem). The term $\beta_I(t - T_0)\phi_I(x_I, z_I, u_I)$ characterizes the fault function dynamics affecting the I-th subsystem including its time evolution. More specifically, the term $\phi_I : \mathbb{R}^{n_I} \times \mathbb{R}^{\bar{n}_I} \times \mathbb{R}^{l_I} \mapsto \mathbb{R}^{n_I}$ is the unknown fault function and the term $\beta_I(t - T_0) : \mathbb{R} \mapsto \mathbb{R}^+$ denotes the time evolution of the fault, where T_0 is the unknown time of the fault occurrence [70]. Note that the fault function ϕ_I may depend on the interconnection state variable vector z_I and not only on the local state vector x_I. In this work, we consider the case of a single fault that occurs in a subsystem (hence there is only one function $\phi_I(\cdot)$) and not the case of a distributed fault that spans across several subsystems. Of course, the fault that occurs in a subsystem Σ_I can affect neighboring subsystems Σ_J through the interconnection terms z_J. The fault time profile $\beta_I(t - T_0)$ can be used to model abrupt faults or incipient faults using a decaying exponential type function:

$$\beta_I(t - T_0) \triangleq \begin{cases} 0 & \text{if } t < T_0 \\ 1 - e^{-b_I(t-T_0)} & \text{if } t \geq T_0 \end{cases} \qquad (3)$$

where $b_I > 0$ is typically an unknown parameter which denotes the fault evolution rate. Abrupt faults correspond to the limit $b_I \to \infty$, in this case, the time profile $\beta_I(t - T_0)$ becomes a step function. In general, small values of b_I indicate slowly developing faults (incipient faults), whereas large values of b_I make the time profile $\beta_I(t - T_0)$ approach a step function (abrupt faults).

In this work, subsystem Σ_J is said to affect subsystem Σ_I (or in other words Σ_J is a "neighbor" of Σ_I), if the interconnection variables of Σ_I, i.e., $z_I(t)$, contains at least one of the state variables of Σ_J, i.e., $x_J(t)$.

The notation $|\cdot|$ used in this chapter indicates the absolute value of a scalar function or the 2-norm in case of a vector. In addition, the notation $y(t) = H(s)\big[x(t)\big]$ (which is used extensively in the adaptive control literature) denotes the output $y(t)$ of a linear system represented by the transfer function $H(s)$ with $x(t)$ as input. In terms of more rigorous notation, let $h(t)$ be the impulse response associated with $H(s)$; i.e., $h(t) \triangleq \mathscr{L}^{-1}\,[H(s)]$, where $\mathscr{L}^{-1}$ is the inverse Laplace transform operator. Then $y(t) = H(s)\big[x(t)\big] = \int_0^t h(\tau)x(t-\tau)\,\mathrm{d}\tau$.

The following assumptions are used throughout the chapter:

Assumption 1 For each subsystem Σ_I, $I \in \{1, ..., N\}$, the local state variables $x_I(t)$ and the local inputs $u_I(t)$ belong to a known compact region $\mathscr{D}_{x_I}$ and $\mathscr{D}_{u_I}$, respectively, before and after the occurrence of a fault, i.e., $x_I(t) \in \mathscr{D}_{x_I}$, $u_I(t) \in \mathscr{D}_{u_I}$ for all $t \geq 0$. □

Assumption 2 The modeling uncertainty $\eta_I^{(i)}$ (i denotes the i-th component of η_I) in each subsystem is an unstructured and possibly unknown nonlinear function of x_I, z_I, and u_I but uniformly bounded by a known positive function $\bar{\eta}_I^{(i)}$, i.e.,

$$|\eta_I^{(i)}(x_I, z_I, u_I)| \leq \bar{\eta}_I^{(i)}(m_I, m_{zI}, u_I), \quad i = 1, 2, \ldots, n_I \tag{4}$$

for all $t \geq 0$ and for all $(x_I, z_I, u_I) \in \mathscr{D}_I$, where $m_{zI} = z_I + \varsigma_I$ is the measurable noisy counterpart of z_I, $\varsigma_I \in \mathscr{D}_{\varsigma_I} \subset \mathbb{R}^{\bar{n}_I}$ and $\bar{\eta}_I^{(i)}(m_I, m_{zI}, u_I) \geq 0$ is a known bounding function in some region of interest $\mathscr{D}_I = \mathscr{D}_{x_I} \times \mathscr{D}_{z_I} \times \mathscr{D}_{u_I} \subset \mathbb{R}^{n_I} \times \mathbb{R}^{\bar{n}_I} \times \mathbb{R}^{l_I}$. The regions $\mathscr{D}_{\varsigma_I}$ and $\mathscr{D}_I$ are compact sets. □

Assumption 1 is required for well posedness since here we do not address the control design and fault accommodation. Assumption 2 characterizes the class of modeling uncertainties being considered. In practice, the system can be modeled more accurately in certain regions of the state space. Therefore, the fact that the bound $\bar{\eta}_I$ is a function of m_I, m_{zI} and u_I provides more flexibility by allowing the designer to take into consideration any prior knowledge of the system. Moreover, the bound $\bar{\eta}_I$ is required in order to distinguish the effects between modeling uncertainty and faults. For example if the bound $\bar{\eta}_I$ is not set properly and it is too low so that (4) does not hold, then false alarms may occur. On the other hand, if the bound $\bar{\eta}_I$ is set too high, so that (4) holds, then this might lead to conservative detection thresholds which may never be crossed, leading to undetected faults. Therefore, the handling of the modeling uncertainty is a key design issue in fault diagnosis architectures, which creates a trade-off between false alarms and conservative fault detection. In Sect. 4.4, adaptive approximation methods will be used to learn the modeling uncertainty η_I and we will use the learned function in order to obtain even tighter detection thresholds and enhance fault detectability.

Each sensor is associated with exactly one subsystem (see Fig. 2). The *local sensor* $S_I^{(i)}$ associated with the I-th subsystem provides a measurement $m_I^{(i)}$ of the i-th

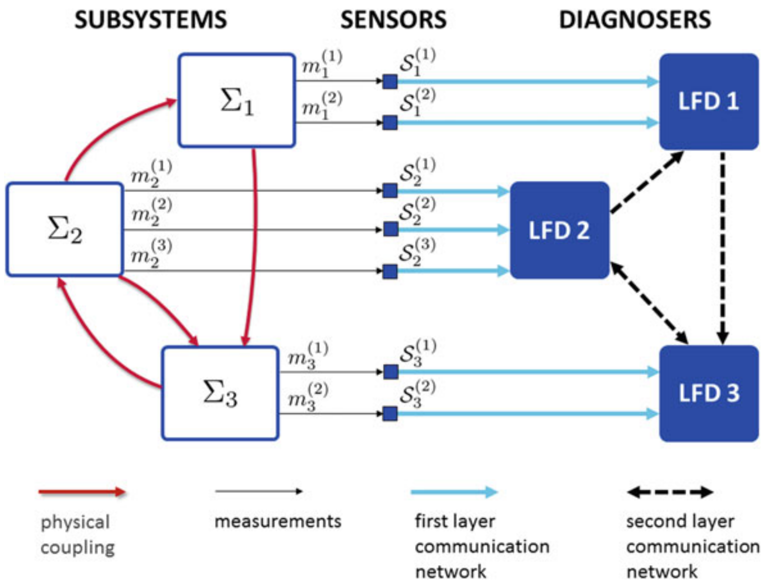


Fig. 2 An example of the proposed multi-layer fault detection architecture. The local state variables for each subsystem (physical layer, left) are measured by the sensor layer (center). The sensors communicate their measurements to the LFDs by means of the first level communication network. The second level communication network (right) allows the diagnosers to communicate with each other exchanging information

component of the local state vector x_I according to the output equation

$$S_I^{(i)} : m_I^{(i)}(t) = x_I^{(i)}(t) + w_I^{(i)}(t) , \quad i = 1, \ldots, n_I , \tag{5}$$

where $w_I^{(i)}$ denotes the noise affecting the i-th sensor of the I-th subsystem.

Assumption 3 For each i-th measurement $m_I^{(i)}$, with $i = 1, \ldots, n_I$, being the vector component index, the measurement uncertainty term $w_I^{(i)}$ is an unstructured and unknown function of time, but it is bounded by a known positive time function $\bar{w}_I^{(i)}(t)$ such that $\left|w_I^{(i)}(t)\right| \leq \bar{w}_I^{(i)}(t)$, $i = 1, \ldots, n_I$, $I = 1, \ldots, N$, $t \geq 0$. □

We assume that the control input is available without any error or delay (it is assumed that there exist feedback controllers yielding a local control action u_I such that some desired control objectives are achieved). Each subsystem is monitored by its respective Local Fault Diagnoser (LFD). The objective is to design and analyze a distributed fault detection scheme, with each subsystem Σ_I being monitored by a LFD that receives local measurements through the first communication network (see Fig. 2) and partial information (i.e., the measurements m_{zI} of the interconnection variables) from neighboring LFDs through the second communication network. In general, the distributed fault detection scheme is composed of N LFDs $\mathscr{S}_I$, one for each subsystem $\varSigma_I$. Each LFD $\mathscr{S}_I$ requires the input and output measurements of the subsystem $\varSigma_I$ that it is monitoring and also the measurements of all interconnecting subsystems $\varSigma_J$ that affect $\varSigma_I$. Note that these last measurements are communicated by neighboring LFDs $\mathscr{S}_J$, and not by the subsystems $\varSigma_J$. Therefore, there is the need of communication between the LFDs depending on their interconnections. It is important to note that, the second layer communication network mirrors the physical coupling morphology. Note that, the information exchanged among the subsystems is readily available since it is constituted by quantities z_I that are measurable with some uncertainty as $m_{zI}(t) = z_I(t) + \varsigma_I(t)$ (the noisy counterpart of z_I). Therefore, the distributed nature of the scheme stems from the fact that there is communication between the LFDs depending on their interconnections. More specifically, each LFD receives from its local sensors the noisy state measurements forming the vector $m_I = \mathrm{col}(m_I^{(i)}, i = 1, \ldots, n_I)$ (see (5)) and from the J-th neighboring LFD the noisy measurements $m_{zI}^{(i)}$, $i = 1, \ldots, \bar{n}_I$ of the local state variables components $x_J^{(i)}$ that influence the I-th subsystem (i.e., the variables $x_J^{(i)}$ belonging to the interconnection vector z_I). Each LFD computes a local state estimate $\hat{x}_I(t)$ based on the local I-th model, by communicating the interconnection variables (and possibly other information) to neighboring LFDs. The LFD implements a model-based fault detection method: the local residual error vector $r_I(t)$ is compared, component by component, to a time-varying detection threshold vector $\bar{r}_I(t)$, well-suited guarantee the absence of false alarms.

2.1 Objectives and Contributions

In this chapter, a distributed fault diagnosis methodology is presented to address the sources of uncertainty mentioned in the introduction. More specifically:

(a) a filtering-based design is embedded in a distributed fault diagnosis methodology to dampen the effect of the measurement noise and enhance fault detection robustness by facilitating less conservative conditions for fault detectability;
(b) an adaptive learning approach is adopted to reduce the modeling uncertainty and thus, further enhance fault detectability;
(c) a delay compensation strategy is devised to address delays and packet losses in the communication network between the LFDs using Time stamps and a buffer, called *diagnosis buffer* (see Fig. 4);
(d) a model-based re-synchronization algorithm is embedded in the diagnosis procedure to manage asynchronous measurements. This algorithm is based on *virtual sensors* implemented in the LFDs and on the use of a *measurements buffer* (see Fig. 4);

In the following, we will first present in Sect. 3 the distributed filtering approach in a continuous-time framework under the assumptions of (i) global synchronization, i.e., subsystems, sensors, and LFDs are assumed to share the same clock and sampling frequency and (ii) perfect information exchange, i.e., it is assumed that information exchanged between LFDs and communicated from the system to the LFDs is without any error nor delay and it is immediately available at any point of the diagnosis system. The effect of the filtering on the detectability performance is rigorously analyzed. After that, in Sect. 4, the filtering design is adapted in a discrete-time formulation to allow to analyze the more realistic networked scenarios, where different strategies for managing modeling uncertainty and network-related issues will be integrated in a comprehensive framework.

3 Filtering-Based Distributed Fault Detection

In this section, we present a filtering framework for the detection of faults in a class of interconnected, nonlinear, continuous-time systems with modeling uncertainty and measurement noise (see [48] for more details). In order to address the measurement noise issue which can lead to conservative detection thresholds or even false alarms if not dealt with properly, filtering is used by embedding the filters into the design in a way that takes advantage of the filtering noise suppression properties. Essentially, filtering dampens the effect of measurement noise in a certain frequency range allowing to set less conservative adaptive fault detection thresholds and thus enhancing fault detectability. As a result, a robust fault detection scheme is designed which guarantees no false alarms. The distributed fault detection scheme is comprised of a set of interacting LFDs, in which each subsystem is monitored by its respective detection agent.

To dampen the effect of measurement uncertainty $w_I(t)$, each measured variable $m_I^{(i)}$ is filtered by $H(s)$, where $H(s)$ is a p-th order filter with strictly proper transfer function

$$H(s) = sH_p(s), \tag{6}$$

$$H_p(s) = \frac{d_{p-2}s^{p-2} + d_{p-3}s^{p-3} + \ldots + d_0}{s^p + c_{p-1}s^{p-1} + \ldots + c_1 s + c_0}. \tag{7}$$

Note that the strictly proper requirement is important. If the transfer function $H(s)$ is proper, then the noise would appear in the filter output and the noise dampening would not be effective.

The choice of a particular type of filter to be used is application dependent, and it is made according to the available a priori knowledge on the noise properties. Usually, measurement noise is constituted by high frequency components and therefore the use of low-pass filter for dampening noise is well justified. On other occasions, one may want to focus the fault detectability on a prescribed frequency band of the measurement signals and hence choose the filter accordingly.

Generally, each measured variable $m_I^{(i)}(t)$ can be filtered by a different filter. In this chapter, without loss of generality, we consider $H(s)$ to be the same for all the output variables in order to simplify the notation and presentation.

The filters $H(s)$ and $H_p(s)$ are asymptotically stable and hence BIBO stable. Therefore, for bounded measurement noise $w_I(t)$ (see Assumption 3), the filtered measurement noise $\varepsilon_{w_I}(t) \triangleq H(s)\,[w_I(t)]$ is uniformly bounded as follows:

$$|\varepsilon_{w_I}^{(i)}(t)| \leq \bar{\varepsilon}_{w_I}^{(i)} \quad i = 1, 2, \ldots, n_I, \tag{8}$$

where $\bar{\varepsilon}_{w_I}^{(i)}$ are known bounding constants. Depending on the noise characteristics, $H(s)$ can be selected to reduce the bound $\bar{\varepsilon}_{w_I}^{(i)}$.

3.1 Distributed Fault Detection

In this section, we explain in detail the fault filtering framework in order to obtain the residual signals $r_I(t)$ to be used for fault detection and the corresponding detection thresholds $\bar{r}_I(t)$. The fault detection logic is based on deriving suitable detection thresholds so that in the absence of a fault the residual signals are bounded by their corresponding detection threshold signals, guaranteeing no false alarms. To state this formally: in the absence of a fault (i.e., for $t \in [0, T_0)$), it is guaranteed that $|r_I^{(i)}(t)| \leq \bar{r}_I^{(i)}(t)$, $\forall i = 1, \ldots, n_I$ and $\forall I = 1, \ldots, N$. The detection decision of a fault in the overall system is made when $|r_I^{(i)}(t)| > \bar{r}_I^{(i)}(t)$ at some time t for at least one component i in any subsystem Σ_I. Note that, in this chapter, only a single fault ϕ_I is considered to occur in the large-scale distributed system.

By locally filtering the output signal $m_I(t)$, we obtain the filtered output $y_{I,f}(t)$:

$$\begin{aligned} y_{I,f}(t) &= H(s)\,[m_I(t)] \\ &= H(s)\,[x_I(t) + w_I(t)]\,. \end{aligned} \tag{9}$$

By using $\varepsilon_{w_I}(t) = H(s)\,[w_I(t)]$ and the fact that $s[x_I(t)] = \dot{x}_I(t) + x_I(0)\delta(t)$ (where $\delta(t)$ is the delta function), we obtain

$$\begin{aligned} y_{I,f}(t) &= H(s)\,[x_I(t)] + \varepsilon_{w_I}(t) \\ &= H_p(s)\,[\dot{x}_I(t)] + H_p(s)\,[x_I(0)\delta(t)] + \varepsilon_{w_I}(t) \\ &= H_p(s)\big[f_I\big(x_I(t), u_I(t)) + g_I(x_I(t), z_I(t), u_I(t)\big) \\ &\qquad + \eta_I\big(x_I(t), z_I(t), u_I(t)\big) + \beta_I(t - T_0)\phi_I\big(x_I(t), z_I(t), u_I(t)\big)\big] \\ &\qquad + \varepsilon_{w_I}(t) + h_p(t)x_I(0), \end{aligned} \tag{10}$$

where $h_p(t)$ is the impulse response of the filter $H_p(s)$, i.e., $h_p(t) \triangleq \mathscr{L}^{-1}\left[H_p(s)\right]$. The estimation model $\hat{x}_I(t)$ for $x_I(t)$ under fault-free operation is generated based on (1) by considering only the known components and by using the measurements

m_I and m_{zI} as follows:

$$\dot{\hat{x}}_I = f_I(m_I(t), u_I(t)) + g_I(m_I(t), m_{zI}(t), u_I(t)), \tag{11}$$

with the initial condition $\hat{x}_I(0) = m_I(0)$.
The corresponding estimation model for $y_{I,f}(t)$, denoted by $\hat{y}_{I,f}(t)$, is given by

$$\hat{y}_{I,f}(t) = H(s)\big[\hat{x}_I(t)\big], \tag{12}$$

and by using (11) and following a similar procedure as in the derivation of (10), $\hat{y}_{I,f}(t)$ becomes

$$\hat{y}_{I,f}(t) = H_p(s)\big[f_I\big(m_I(t), u_I(t)\big) + g_I\big(m_I(t), m_{zI}(t), u_I(t)\big)\big] + h_p(t)m_I(0). \tag{13}$$

The local residual error $r_I(t)$ to be used for fault detection is defined as

$$r_I(t) \triangleq y_{I,f}(t) - \hat{y}_{I,f}(t), \tag{14}$$

and it is readily computable from Eqs. (9), (11) and (12).

Prior to the fault ($t < T_0$), the local residual error can be written using Eqs. (10), (13) and (14) as

$$r_I(t) = H_p(s)\,[\chi_I(t)] + \varepsilon_{w_I}(t) \tag{15}$$

where the total uncertainty term $\chi_I(t)$ is defined as

$$\chi_I(t) \triangleq \varDelta f_I(t) + \varDelta g_I(t) + \eta_I\big(x_I(t), z_I(t), u_I(t)\big), \tag{16}$$

$$\varDelta f_I(t) \triangleq f_I\big(x_I(t), u_I(t)\big) - f_I\big(x_I(t) + w_I(t), u_I(t)\big), \tag{17}$$

$$\varDelta g_I(t) \triangleq g_I\big(x_I(t), z_I(t), u_I(t)\big) - g_I\big(x_I(t) + w_I(t), z_I(t) + \varsigma_I(t), u_I(t)\big). \tag{18}$$

For simplicity, in the derivation of (15), the initial conditions $x_I(0) = m_I(0)$ are assumed to be known. If there is uncertainty in the initial conditions (i.e., $x_I(0) \neq m_I(0)$) then that introduces the extra term $h_p(t)(x_I(0) - m_I(0))$ in (15) which however converges to zero exponentially (since $h_p(t)$ is exponentially decaying [24]) and thus does not affect significantly the subsequent analysis.

By taking bounds on (15) and by using the triangle inequality for each component i of the residual, we obtain

$$\begin{aligned}|r_I^{(i)}(t)| &\leq |H_p(s)\left[\chi_I^{(i)}(t)\right]| + |\varepsilon_{w_I}^{(i)}(t)| = |\int_0^t h_p(t-\tau)\chi_I^{(i)}(\tau)\,\mathrm{d}\tau| + |\varepsilon_{w_I}^{(i)}(t)| \\ &\leq \int_0^t |h_p(t-\tau)||\chi_I^{(i)}(\tau)|\,\mathrm{d}\tau + |\varepsilon_{w_I}^{(i)}(t)|\end{aligned}$$

$$\leq \int_0^t \bar{h}_p(t-\tau)\bar{\chi}_I^{(i)}(\tau)\,\mathrm{d}\tau + \bar{\varepsilon}_{w_I}^{(i)} \tag{19}$$

where $\bar{h}_p(t)$ is the impulse response (of the filter $\bar{H}_p(s)$) that satisfies $|h_p(t)| \leq \bar{h}_p(t)$ for all $t > 0$ (details for selecting $\bar{H}_p(s)$ will be given in Sect. 3.2) and $\bar{\chi}_I^{(i)}(t)$ is the bound on the total uncertainty term $\chi_I^{(i)}(t)$, i.e., $|\chi_I^{(i)}(t)| \leq \bar{\chi}_I^{(i)}(t)$.
Using Assumption 2, the bound $\bar{\chi}_I^{(i)}(t),\ \ i = 1, 2, \ldots, n_I$ is defined as

$$\bar{\chi}_I^{(i)}(t) \triangleq \overline{\Delta f}_I^{(i)} + \overline{\Delta g}_I^{(i)} + \bar{\eta}_I^{(i)}\big(m_I(t), m_{zI}(t), u_I(t)\big), \tag{20}$$

where

$$\overline{\Delta f}_I^{(i)} \triangleq \sup_{\substack{(x_I,u_I)\in\mathscr{D}_{x_I}\times\mathscr{D}_{u_I}\\ w_I\in\mathscr{D}_{w_I}}} |f_I^{(i)}\big(x_I, u_I\big) - f_I^{(i)}\big(x_I + w_I, u_I\big)| \tag{21}$$

$$\overline{\Delta g}_I^{(i)} \triangleq \sup_{\substack{(x_I,z_I,u_I)\in\mathscr{D}_I\\ (w_I,\varsigma_I)\in\mathscr{D}_{w_I}\times\mathscr{D}_{\varsigma_I}}} |g_I^{(i)}\big(x_I, z_I, u_I\big) - g_I^{(i)}\big(x_I + w_I, z_I + \varsigma_I, u_I\big)|. \tag{22}$$

Since the regions $\mathscr{D}_I$, $\mathscr{D}_{w_I}$ and $\mathscr{D}_{\varsigma_I}$ are compact sets, the suprema in (21) and (22) are finite. In addition, note that the bound $\bar{\chi}_I^{(i)}(t)$ in (20) depends on t because of the bounding function $\bar{\eta}_I^{(i)}$.

Finally, a suitable detection threshold $\bar{r}_I^{(i)}(t)$ can be selected as the right-hand side of (19) which can be rewritten as

$$\bar{r}_I^{(i)}(t) = \bar{H}_p(s)\left[\bar{\chi}_I^{(i)}(t)\right] + \bar{\varepsilon}_{w_I}^{(i)}. \tag{23}$$

A practical issue that requires consideration is the derivation of the bound $\bar{\chi}_I^{(i)}(t)$ given in (20). Specifically, the derivation of $\bar{\chi}_I^{(i)}(t)$ requires the bounds $\overline{\Delta f}_I^{(i)}$ and $\overline{\Delta g}_I^{(i)}$ on $\Delta f_I^{(i)}(t)$ and $\Delta g_I^{(i)}(t)$, respectively. One approach for deriving the bound $\overline{\Delta f}_I^{(i)}$ in (21) is to consider a local Lipschitz assumption:

$$|f_I^{(i)}(x_I, u_I) - f_I^{(i)}(x_I + w_I, u_I)| \leq L_{f_I^{(i)}}|w_I| \tag{24}$$

where $L_{f_I^{(i)}}$ is the Lipschitz constant for the function $f_I^{(i)}(x_I, u_I)$ with respect to x_I in the region $\mathscr{D}_{x_I}$. Therefore, if we have a bound w_I^M on the measurement noise, i.e., $|w_I(t)| \leq w_I^M \quad \forall t > 0$, then we can obtain a bound on $\Delta f_I^{(i)}(t)$. A similar approach can be followed for $\Delta g_I^{(i)}(t)$.

Another way of obtaining a less conservative bound than $\bar{\chi}_I^{(i)}$ and therefore further enhance fault detectability, is by exploiting the use of filtering which can be proved beneficial for dampening the mismatch function $\Delta f_I(t) + \Delta g_I(t)$ which results due to the measurement noise. Among the various filters one can select, some may lead to less conservative detection thresholds. Therefore, a significantly less conservative detection threshold without the need for the Lipschitz constants can be obtained by observing that the residual (15) can be written as

$$r_I(t) = H_p(s)\left[\eta_I\big(x_I(t), z_I(t), u_I(t)\big)\right] + H_p(s)\left[\Delta f_I(t) + \Delta g_I(t)\right] + \varepsilon_{w_I}(t) \quad (25)$$

and by making the following assumptions:

Assumption 4 The filtered function mismatch term $\varepsilon_{\Delta_I}(t) \triangleq H_p(s)\left[\Delta f_I(t) + \Delta g_I(t)\right]$ is uniformly bounded as follows:

$$|\varepsilon_{\Delta_I}^{(i)}(t)| \leq \bar{\varepsilon}_{\Delta_I}^{(i)} \quad i = 1, 2, \ldots, n_I, \quad (26)$$

where $\bar{\varepsilon}_{\Delta_I}^{(i)}$ is a known bounding constant. □

Assumption 4 is based on the fact that filtering dampens the effect of measurement noise present in the function mismatch term $\Delta f_I(t) + \Delta g_I(t)$. A suitable selection of $\bar{\varepsilon}_{\Delta_I}^{(i)}$ can be made through the use of simulations (i.e., Monte Carlo methods) by filtering the function mismatch term $\Delta f_I(t) + \Delta g_I(t)$ using the known function dynamics and the available noise characteristics (recall that the measurement noise is assumed to take values in a compact set).

Therefore, the detection threshold becomes

$$\bar{r}_I^{(i)}(t) = \bar{H}_p(s)\left[\bar{\eta}_I^{(i)}\big(m_I(t), m_{zI}(t), u_I(t)\big)\right] + \bar{\varepsilon}_{\Delta_I}^{(i)} + \bar{\varepsilon}_{w_I}^{(i)}. \quad (27)$$

Figure 3 illustrates the I-th LFD which includes the implementation of the local filtered fault detection scheme for the I-th subsystem resulting from Eqs. (9), (11), (12), (14) and (23).

3.2 Selection of Filter $\bar{H}_p(s)$

Two methods for selecting a suitable transfer function $\bar{H}_p(s)$ with impulse response $\bar{h}_p(t)$ such that $|h_p(t)| \leq \bar{h}_p(t)$ for all $t \geq 0$ are illustrated.

In general though, note that if the impulse response $h_p(t)$ is nonnegative, i.e., $h_p(t) \geq 0$, for all $t \geq 0$, then the calculation of $\bar{H}_p(s)$ can be omitted. In this case $H_p(s)$ can be used instead of $\bar{H}_p(s)$ in (23), as it can easily be seen from (19) since $|h_p(t-\tau)| = h_p(t-\tau)$. Necessary and sufficient conditions for nonnegative impulse response for a specific class of filters are given in [60].

- *First method.*

The first method relies on the following Lemma, which describes a methodology for finding $\bar{H}_p(s)$. For notational convenience, for any $m \times n$ matrix A we define

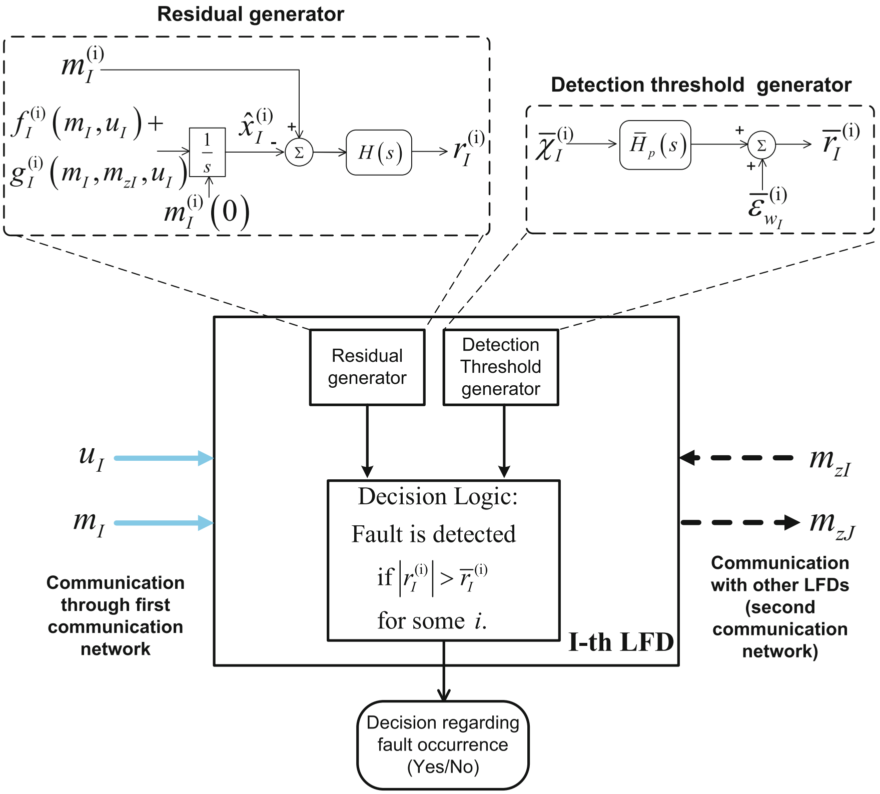

Residual generator
$m_I^{(i)}$
$f_I^{(i)}(m_I, u_I)+$
$g_I^{(i)}(m_I, m_{zI}, u_I)$
$\frac{1}{s}$
$m_I^{(i)}(0)$
$\hat{x}_I^{(i)}$
+
-
Σ
$H(s)$
$r_I^{(i)}$
Detection threshold generator
$\overline{\chi}_I^{(i)}$
$\bar{H}_p(s)$
+
+
Σ
$\overline{r}_I^{(i)}$
$\overline{\varepsilon}_{w_I}^{(i)}$
Residual generator
Detection Threshold generator
u_I
m_I
m_{zI}
m_{zJ}
Decision Logic: Fault is detected if $|r_I^{(i)}| > \overline{r}_I^{(i)}$ for some i.
I-th LFD
Communication through first communication network
Communication with other LFDs (second communication network)
Decision regarding fault occurrence (Yes/No)

Fig. 3 Local filtered fault detection scheme

$|A|_{\mathscr{E}}$ as the matrix whose elements correspond to the modulus of the element $a_{i,j}$, $i = 1, \ldots, m$ and $j = 1, \ldots, n$ of the matrix A.

Lemma 1 ([48]). *Let $w(t) = Ce^{At}B$ be the impulse response of a strictly proper SISO transfer function $W(s)$ with state space representation (A, B, C). Then, for any signal $v(t) \geq 0$, the following inequality holds for all $t \geq 0$:*

$$\int_0^t |w(t-\tau)| v(\tau)\,\mathrm{d}\tau \leq \overline{W}(s)\,[v(t)]\,,$$

where $\overline{W}(s)$ is given by

$$\overline{W}(s) \triangleq |CT|_{\mathscr{E}}\,(sI - Re[J])^{-1}\,\left|T^{-1}B\right|_{\mathscr{E}} \tag{28}$$

and $J = T^{-1}AT$ is the Jordan form of the matrix A.

Therefore, by using Lemma 1 with $w(t) = h_p(t)$, the transfer function $\bar{H}_p(s)$ such that its impulse response satisfies $|h_p(t)| \leq \bar{h}_p(t)$ can be obtained from (28).

- *Second method.*

The second method is by using the following well-known result (see, for instance [24]).

Lemma 2 *The impulse response $h_p(t)$ of a strictly proper and asymptotically stable transfer function $H_p(s)$ decays exponentially; i.e., $|h_p(t)| \leq \kappa e^{-\upsilon t}$ for some $\kappa > 0$, $\upsilon > 0$, for all $t \geq 0$.*

By using Lemma 2, a suitable impulse response $\bar{h}_p(t)$ such that $|h_p(t)| \leq \bar{h}_p(t)$ for all $t \geq 0$ is given by $\bar{h}_p(t) = \kappa e^{-\upsilon t}$ and can be implemented using linear filtering techniques as $\bar{H}_p(s) = \frac{\kappa}{s+\upsilon}$.

3.3 Fault Detectability and Detection Time Analysis

3.3.1 Fault Detectability Analysis

The design and analysis of the fault detection scheme in the previous sections were based on the derivation of suitable thresholds $\bar{r}_I^{(i)}(t)$ such that in the absence of any fault, the residual signals $r_I^{(i)}(t)$ are bounded by $\bar{r}_I^{(i)}(t)$. An important related question is what class of faults can be detected. This is referred to as *fault detectability analysis*. In this section, fault detectability conditions for the aforementioned fault detection scheme are derived. The fault detectability analysis constitutes a theoretical result that characterizes quantitatively the class of faults detectable by the proposed scheme.

Theorem 1 *Consider the nonlinear system (1), (2) with the distributed fault detection scheme described in (9), (11), (12), (14) and (23) in the general case of $H(s)$ given by (6). A sufficient condition for a fault $\phi_I^{(i)}(x_I, z_I, u_I)$ in the I-th subsystem initiated at T_0 to be detectable at time $T_d > T_0$ is that for some $i = 1, 2, \ldots, n_I$:*

$$|H_p(s)\big[\beta_I(T_d - T_0)\phi_I^{(i)}\big(x_I(T_d), z_I(T_d), u_I(T_d)\big)\big]| > 2\bar{r}_I^{(i)}(T_d). \tag{29}$$

Proof In the presence of a fault that occurs at T_0, Eq. (15) becomes

$$r_I^{(i)}(t) = H_p(s)\big[\chi_I^{(i)}(t) + \beta_I(t - T_0)\phi_I^{(i)}\big(x_I(t), z_I(t), u_I(t)\big)\big] + \varepsilon_{w_I}^{(i)}(t).$$

By using the triangle inequality, for $t > T_0$, the residual $r_I^{(i)}(t)$ satisfies

$$\begin{aligned} |r_I^{(i)}(t)| \geq & - |H_p(s)\big[\chi_I^{(i)}(t)\big]| - |\varepsilon_{w_I}^{(i)}(t)| \\ & + |H_p(s)\big[\beta_I(t - T_0)\phi_I^{(i)}\big(x_I(t), z_I(t), u_I(t)\big)\big]| \end{aligned}$$

$$
\begin{aligned}
&\geq - \int_0^t |h_p(t-\tau)||\chi_I^{(i)}(\tau)|\,\mathrm{d}\tau - |\varepsilon_{w_I}^{(i)}(t)| \\
&\quad + \big|H_p(s)\big[\beta_I(t-T_0)\phi_I^{(i)}\big(x_I(t), z_I(t), u_I(t)\big)\big]\big| \\
&\geq - \int_0^t \bar{h}_p(t-\tau)\bar{\chi}_I^{(i)}(\tau)\,\mathrm{d}\tau - \bar{\varepsilon}_{w_I}^{(i)} \\
&\quad + \big|H_p(s)\big[\beta_I(t-T_0)\phi_I^{(i)}\big(x_I(t), z_I(t), u_I(t)\big)\big]\big| \\
&\geq - \bar{r}_I^{(i)}(t) + \big|H_p(s)\big[\beta_I(t-T_0)\phi_I^{(i)}(x_I(t), z_I(t), u_I(t))\big]\big|.
\end{aligned}
$$

For fault detection, the inequality $|r_I^{(i)}(t)| > \bar{r}_I^{(i)}(t)$ must hold at some time $t = T_d$ for some $i = 1, \ldots, n_I$, so the final fault detectability condition given by (29) is obtained. □

Although Theorem 1 is based on threshold (23), it can be readily shown that the same result holds in the case where threshold (27) is used. Clearly, the fault functions $\phi_I(x_I, z_I, u_I)$ are typically unknown and therefore this condition cannot be checked a priori. However, it provides useful intuition about the type of faults that are detectable. The detectability condition given in Theorem 1 is a sufficient condition, but not a necessary one and hence, the class of detectable faults can be significantly larger. The use of filtering is of crucial importance in order to derive tighter detection thresholds that guarantee no false alarms. As it can be seen in the detectability condition given by (29), the detection of the fault depends on the filtered fault function ϕ_I and as a result, the selection of the filter is very important. Since the fault function is usually comprised of lower frequency components, it is not affected that much by low-pass filtering in comparison to the measurement noise which is usually of higher frequency. In addition, filtering allows the derivation of tighter detection thresholds and as a result, the fault detectability condition can be met more easily. Obviously, some filter selections may lead to less conservative thresholds than others.

The detectability properties of the proposed filtering approach are further investigated by considering a specific case for the filter $H_p(s)$

$$H_p(s) = \frac{\alpha^p}{(s+\alpha)^p}. \tag{30}$$

This type of filter is well suited for gaining further intuition since it contains two parameters p and α that denote the order of the filter and the pole location, respectively. More specifically, the order p of the filter regulates the damping effect of the high frequency noise, whereas the value α of the filter determines the cutoff frequency at which the damping begins. In general, more selective filter implementations can be made (i.e., Butterworth filters) which may have some implications in the filters required for the detection threshold implementation (due to the fact that the impulse response may not be always positive). But, the particular filter $H_p(s)$ given by (30) is perfectly suited for the investigation of the analytical properties of the filtering scheme. Note also that $H_p(s)$ has a nonnegative impulse response $h_p(t)$ and therefore $\bar{H}_p(s)$ can be selected simply as $H_p(s)$.

In order to conduct this fault detectability analysis, we simplify Assumption 2 by considering a constant bounding condition. It is important to note that the constant bounding of the uncertainty may introduce additional conservativeness, thus reducing the advantage given by the tighter conditions obtained through the filtering.

Assumption 5 The modeling uncertainty $\eta_I^{(i)}$ in each subsystem is an unstructured and possibly unknown nonlinear function of x_I, z_I and u_I but uniformly bounded by a known positive scalar $\bar{\eta}_I^{(i)}$, i.e.,

$$|\eta_I^{(i)}(x_I, z_I, u_I)| \leq \bar{\eta}_I^{(i)}, \quad i = 1, 2, \dots, n_I \tag{31}$$

for all $t \geq 0$ and for all $(x_I, z_I, u_I) \in \mathscr{D}_I$, where $\bar{\eta}_I^{(i)} \geq 0$ is a known bounding scalar in some region of interest $\mathscr{D}_I = \mathscr{D}_{x_I} \times \mathscr{D}_{z_I} \times \mathscr{D}_{u_I} \subset \mathbb{R}^{n_I} \times \mathbb{R}^{\bar{n}_I} \times \mathbb{R}^{l_I}$. □

By using the Lipschitz assumption stated in (24), along with the known constant bound w_I^M of the measurement uncertainty $|w_I|$ and the constant bound on the modeling uncertainty $\bar{\eta}_I^{(i)}$, as stated in Assumption 5, the bound of the total uncertainty term $\bar{\chi}_I^{(i)}(t)$ takes a constant value $\bar{\chi}_I^{(i)}$. Then, Theorem 2, which follows, can be obtained (its proof can be found in [48]).

It must be pointed out that, although we use (23) for the detection threshold, the adaptation of the subsequent results in the case where the threshold is given by (27) is straightforward by simply replacing $\bar{\chi}_I^{(i)}$ with $\bar{\eta}_I^{(i)}$ and adding the term $\bar{\varepsilon}_{\Delta_I}^{(i)}$ along the term $\bar{\varepsilon}_{w_I}^{(i)}$ in what follows.

Theorem 2 *Consider the nonlinear system (1), (2) with the distributed fault detection scheme described in (9), (11), (12), (14) and (23) in the special case of $H_p(s)$ given by (30) and with $\bar{H}_p(s) = H_p(s)$. Suppose at least one component $\phi_I^{(i)}(x_I, z_I, u_I)$ of the fault vector $\phi_I(x_I, z_I, u_I)$ satisfies the condition*

$$|\phi_I^{(i)}(x_I(t'), z_I(t'), u_I(t'))| \geq M, \quad \forall\, t' \in [T_0, t]\,, \tag{32}$$

for sufficiently large $t > T_0$ and is continuous in the time interval $t' \in [T_0, t]$. If $M > 2(\bar{\chi}_I^{(i)} + \bar{\varepsilon}_{w_I}^{(i)})$, then the fault will be detected, that is $|r_I^{(i)}(t)| > \bar{r}_I^{(i)}(t)$.

The aforementioned theorem is conceptually different from Theorem 1. More specifically, the detectability condition (29) of Theorem 1 allows the fault function $\phi_I^{(i)}$ to change sign. On the other hand, Theorem 2 states that if the fault function $\phi_I^{(i)}$ maintains the same sign over time and its magnitude is larger than $2(\bar{\chi}_I^{(i)} + \bar{\varepsilon}_{w_I}^{(i)})$ for sufficiently long, then the fault is guaranteed to be detected.

3.3.2 Detection Time Analysis

The detection time of a fault, that is, the time interval between the fault occurrence and its detection, plays a crucial role in fault diagnosis and it constitutes a form of performance criterion. When a fault is detected faster, then timely actions can be undertaken to avoid more serious or even disastrous consequences. It is worth noting that incipient faults are more difficult to detect, especially during their early stages, and as a result the detection time of an incipient fault is generally larger than that of an abrupt fault. In this section, an upper bound of the detection time is obtained in the case where a fault is detected according to Theorem 2. Moreover, we investigate the influence of the filter's order p and the pole location α on the upper bound of the detection time in order to derive some insight regarding the selection of p and α. The results are obtained for the general case of an incipient fault; concerning the dependence of the detection time on the filter's order p, only the abrupt fault case is addressed for the sake of simplicity.

Theorem 3 *Consider the nonlinear system (1), (2) with the distributed fault detection scheme described in (9), (11), (12), (14) and (23) in the special case of $H_p(s)$ given by (30) and with $\bar{H}_p(s) = H_p(s)$. If at least one component $\phi_I^{(i)}(x_I, z_I, u_I)$ of the fault vector $\phi_I(x_I, z_I, u_I)$ satisfies the condition*

$$\left|\phi_I^{(i)}\big(x_I(t'), z_I(t'), u_I(t')\big)\right| \geq M, \quad \forall\, t' \in [T_0, t] \tag{33}$$

where $M > 2(\bar{\chi}_I^{(i)} + \bar{\varepsilon}_{w_I}^{(i)})$ for sufficiently large $t > T_0$ and is continuous in the time interval $t' \in [T_0, t]$ such that the fault can be detected according to Theorem 2, then:

(a) A sufficient condition for fault detectability is given by

$$q(t, T_0, \alpha) > \frac{2(p-1)!}{M}(\bar{\chi}_I^{(i)} + \bar{\varepsilon}_{w_I}^{(i)}). \tag{34}$$

where

$$q(t, T_0, \alpha) \triangleq q_1(t, T_0, \alpha) - q_2(t, T_0, \alpha) \tag{35}$$

$$q_1(t, T_0, \alpha) = \gamma\big(p, \alpha(t - T_0)\big), \tag{36}$$

$$q_2(t, T_0, \alpha) = \begin{cases} \frac{\alpha^p}{p}(t - T_0)^p e^{-\alpha(t-T_0)} & \textit{if } \alpha = b_I \\ \frac{\alpha^p e^{-b_I(t-T_0)}}{(a-b_I)^p}\gamma\big(p, (\alpha - b_I)(t - T_0)\big) & \textit{else,} \end{cases} \tag{37}$$

and $\gamma(\cdot)$ indicates the lower incomplete Gamma function, defined as $\gamma\big(p, z\big) \triangleq \int_0^z w^{p-1} e^{-w}\, \mathrm{d}w$.

(b) An upper bound on the detection time T_d of an incipient fault can be found by solving the equation

$$q_1(T_d, T_0, \alpha) - q_2(T_d, T_0, \alpha) = \frac{2(p-1)!}{M}\bar{r}_I^{(i)}(T_d), \tag{38}$$

where $\bar{r}_I^{(i)}$ is given by

$$\bar{r}_I^{(i)}(t) = \frac{1}{(p-1)!}\bar{\chi}_I^{(i)}\gamma\big(p, \alpha t\big) + \bar{\varepsilon}_{w_I}^{(i)}. \tag{39}$$

(c) The upper bound T_d decreases monotonically as the value of α increases.

(d) In the case of abrupt faults, the upper bound on the detection time T_d increases as the order p of the filter increases.

The proof of Theorem 3 can be found in [48]. Part (b) of the above theorem establishes the mathematical equation whose solution gives an upper bound on the detection time. At this point, we must stress that, although we refer to the solution of the equation as the upper bound of the detection time (because of the requirement (32)), there are cases where the solution is the actual detection time. For instance, consider the case where the magnitude of the fault is $\big|\phi_I^{(i)}(x_I(t'), z_I(t'), u_I(t'))\big| = M, \quad \forall\, t' \in [T_0, t]$ and $M > 2(\bar{\chi}_I^{(i)} + \bar{\varepsilon}_{w_I}^{(i)})$. Then, the solution of (38) gives the actual detection time.

Part (c) of the theorem shows that by increasing the value of the pole α, the upper bound on the detection time (and sometimes the actual detection time as explained before) decreases. On the other hand, the value of α regulates the cutoff frequency of the filter where the damping begins, so the pole location has an inherent trade-off between noise damping and fault detection speed.

Part (d) of the theorem states that in the case of abrupt faults, the upper bound on the detection time increases as the order p of the filter increases. Although the proof is for the case of abrupt faults, the same behavior is observed in the case of incipient faults as well. An obvious downside of higher order filtering is the possible increased detection time. There is also a qualitative explanation for Part (d), as it has necessarily to do with the phase lag introduced by the filter which increases with p. Simply put, by increasing p results in increased phase lag or delay between the input and output signals of the filter and since the detectability of a fault relies on the filtered signals, the detection time increases according to the delay incurred.

Remark 1 Prior to the occurrence of a fault, the residual differs from zero due to the effect of the filtered noise and filtered modeling uncertainty as indicated by (15). When a fault occurs, the residual is permanently contaminated by the filtered fault function as shown in the proof of Theorem 1. In general, the location of the poles simply affects the effectiveness of the noise dampening. To make things more clear, consider Theorems 2 and 3 which rely on the special case of the filter $H_p(s)$ given in (30). Theorem 2, states that in the case of a fault (abrupt or incipient), which satisfies the conditions given in the Theorem then the fault is guaranteed to be detected. Note that this is irrespective of the location of the filters' poles. In fact, as shown in Theorem 3, having faster poles results in a smaller upper bound on the detection time or even smaller actual detection time. In conclusion, the location of the poles does not limit the duration of the residual activation when a fault occurs, but instead the residual is permanently affected by the filtered fault function. Therefore, the location of the poles has an inherent trade-off between noise damping and fault detection speed.

Simulation results showing the effectiveness of the illustrated techniques can be found in [48].

4 The Cyber-Physical Networked Architecture

In this section, we present a cyber-physical networked fault detection architecture based on [14]. Let us note that the approach for distributed fault diagnosis of nonlinear uncertain large-scale systems that we have previously described is based on some underlying assumptions that may restrict its applicability, namely:

1. global synchronization: subsystems, sensors, and LFDs were assumed to share the same clock and sampling frequency;
2. perfect information exchange: it was assumed that information exchanged between LFDs and communicated from the system to the LFDs is without any error nor delay and it is immediately available at any point of the diagnosis system.

In several realistic contexts, (1) and (2) may not hold, and as a consequence, (i) some faults may become undetectable due to the fact that LFDs make detection decisions based on outdated information; (ii) delays in information exchange may cause longer detection times; (iii) the lack of accurate and timely information may cause false alarms.

In order to address these issues and the more complex nature of real CPS systems, we now consider a more comprehensive framework, where the previously proposed filtering design to reduce measurement noise is adapted in the current formulation in discrete time.

The proposed distributed fault detection architecture is made of three layers: the system layer, the sensor layer and the diagnosis layer. In Fig. 2, this layout was shown in a pictorial way. These three layers are briefly described next.

The *system layer* refers to the large-scale system to be monitored. It is described by the continuous-time state equations for each subsystem Eq. (1) and the output Eq. (2).

The *sensor layer* consists of the available sensors taking measurements $m_I^{(i)}(t)$ in continuous-time (see (5)) and sampling and sending such measurements to the I-th LFD at time instants $t_{sI}^{(i)}$ that are not necessarily equally spaced in time. As we do not assume that the measurements delivered by the sensors are synchronized with each other, each measurement is labeled with a Time Stamp (TS) [94] to indicate the time instant $t_{sI}^{(i)}$ at which the measurements are taken by sensor $S_I^{(i)}$ in the time coordinate t.

The communication between the sensors and the LFDs is achieved through the *first level communication network* (see Fig. 2). This network can introduce delays and packet losses, for instance because of collision between different sensors trying to communicate at the same time. Therefore, measurements communicated from the sensors to LFDs may be received at any time instant.

The *Diagnosis layer* consists of the previously introduced LFDs providing a distributed fault diagnosis procedure. The structure of each LFD is shown in Fig. 4. As previously mentioned, each LFD receives the measurements from specific sensors with the aim to provide local fault diagnosis decisions. The LFDs operate in a discrete-time synchronous time frame $k \in \mathbb{Z}$ which turns out to be more convenient for handling any communications delays, as will be seen in the next sections. For the sake of simplicity, the sampling time of the discrete-time frame is assumed to be unitary and the reference time is common, that is, the origin of the discrete-time axis is the same as that of the continuous-time axis. Therefore, the operation of the LFDs is based on the local discrete-time models, which are the discrete-time version of local models (1):

$$\begin{aligned} x_I(k+1) = f_I(x_I(k), u_I(k)) + g_I(x_I(k), z_I(k), u_I(k)) + \eta_I(x_I(k), z_I(k), u_I(k)) \\ + \beta_I(k-k_0)\phi_I(x_I(k), z_I(k), u_I(k)) \,, \end{aligned} \tag{40}$$

where ϕ_I describes the local discretized fault effects, occurring at some discrete-time k_0 (that is, $\beta_I(k-k_0)\phi_I(x_I(k), z_I(k), u_I(k)) = 0, k < k_0$). Each LFD exchanges information with neighboring LFDs by means of the *second level communication network* (see right side of Figs. 2 and 4). As we will see in the following, the exchanged information consists in the re-synchronized interconnection variables v_J. In Fig. 4, an example of a two LFDs architecture is presented to provide more insight into the structure of the proposed scheme.

In summary, two different and not reliable communication networks are considered in this work: the first level communication network allows each LFD to communicate with its local sensors and the second level communication network allows the communication between different LFDs for detection purposes. Both these communication networks may be subject to delays and packet losses. Given the different nature of the networks (the first is local, while the second is connecting different subsystems, which may be geographically apart), in the next section we provide two different strategies to manage communication issues: a re-synchronization method for the first level communication network and a delay compensation strategy for the second level communication network.

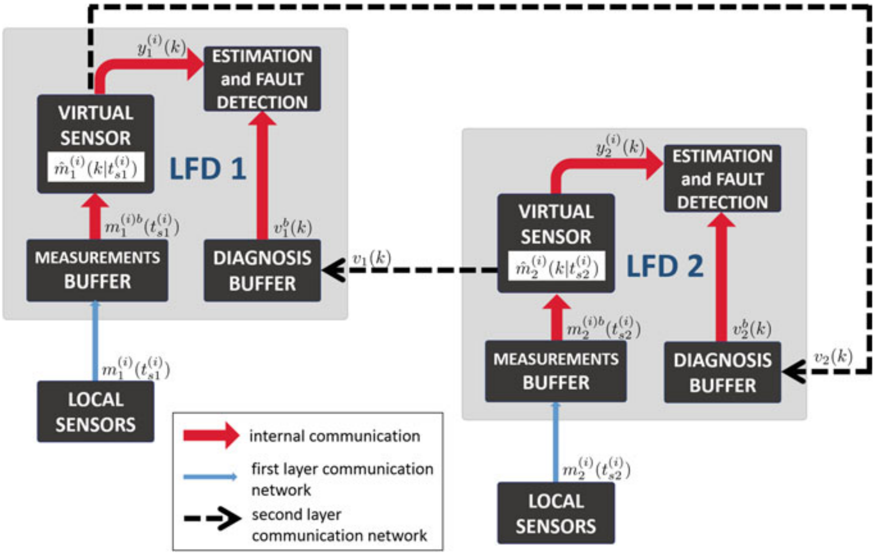

$y_1^{(i)}(k)$
ESTIMATION and FAULT DETECTION
VIRTUAL SENSOR
$\hat{m}_1^{(i)}(k|t_{s1}^{(i)})$
LFD 1
$m_1^{(i)b}(t_{s1}^{(i)})$
$v_1^b(k)$
MEASUREMENTS BUFFER
DIAGNOSIS BUFFER
$v_1(k)$
$m_1^{(i)}(t_{s1}^{(i)})$
LOCAL SENSORS
$y_2^{(i)}(k)$
ESTIMATION and FAULT DETECTION
VIRTUAL SENSOR
$\hat{m}_2^{(i)}(k|t_{s2}^{(i)})$
LFD 2
$m_2^{(i)b}(t_{s2}^{(i)})$
$v_2^b(k)$
MEASUREMENTS BUFFER
DIAGNOSIS BUFFER
$v_2(k)$
$m_2^{(i)}(t_{s2}^{(i)})$
LOCAL SENSORS
internal communication
first layer communication network
second layer communication network

Fig. 4 An example of a two LFDs architecture. The internal structure of each LFD is shown (similarly as in [14]), composed of two buffers (the measurements buffer and the diagnosis buffer) to collect the information received, respectively, by the local sensors and neighboring LFDs, the Virtual Sensor (processing the received measurements), and the Fault Detection unit, responsible for the monitoring analysis. The communicated information between LFDs is represented

4.1 Re-Synchronization at Diagnosis Level

Let us consider a state variable $x_I^{(i)}(t)$; as mentioned before, at time $t = t_{sI}^{(i)}$ the sensor $S_I^{(i)}$ takes the measurement $m_I^{(i)}(t_{sI}^{(i)})$ and sends it to the I-th LFD with a time stamp $t_{sI}^{(i)}$. The I-th diagnoser receives the measurement sent by $S_I^{(i)}$ at time $t_{aI}^{(i)} > t_{sI}^{(i)}$. Since the LFDs run the distributed fault diagnosis algorithm with respect to a discrete-time framework associated with an integer k (see (40)), an online re-synchronization procedure has to be carried out at the diagnosis level. Moreover, the possible time-varying delays and packet losses introduced by the communication networks between the local sensors and the corresponding LFDs have to be addressed since they may affect the fault diagnosis decision. Note that, the classical discrete-time FD architecture assumes that quantities sampled at exactly time k are used to compute quantities related to time $k + 1$. Unfortunately, the LFDs may receive measurements associated with time instants different from k, because of transmission delays and because of the arbitrary sampling time instants of the sensors. The availability of the time stamp $t_{sI}^{(i)}$ enables each LFD to implement a set of *local virtual sensors* by which the re-synchronization of the measurements received at the Diagnosis level is implemented. We assume that sensors and diagnosers share the same clock at the local level.[1]

Specifically, each LFD collects the most recent sensors measurements in a buffer and computes a projection $\hat{m}_I^{(i)}(k|t_{sI}^{(i)})$ of these latest available measurements $m_I^{(i)}(t_{sI}^{(i)})$, $i = 1, \dots, n_I$, to the discrete-time instant[2] $k \geq t_{aI}^{(i)} > t_{sI}^{(i)}$, by integrating the local nominal model on the time interval $[t_{sI}^{(i)}, k]$.

Remark 2 Let us note that measurements may be related to and could be received also before time $k - 1$, without any assumption on the delay length, thus allowing the possibility of measurement packet losses. Moreover, thanks to the use of the time stamps and the buffers, "out-of-sequence" packets can be managed. The same measurement could be used by the virtual sensor more than once to obtain more than one projections related to different discrete-time instants.

The projected measurement $\hat{m}_I^{(i)}(k|t_{sI}^{(i)})$ can be computed by noticing that, *under healthy mode of behavior*, the local nominal model (1) for the state component i at any time $t > t_{sI}^{(i)}$ can be rewritten as

$$x_I^{(i)}(t) = x_I^{(i)}(t_{sI}^{(i)}) + \int_{t_{sI}^{(i)}}^{t} [f_I^{(i)}(x_I(\tau), u_I(\tau)) + g_I^{(i)}(x_I(\tau), z_I(\tau), u_I(\tau)) \\ + \eta_I^{(i)}(x_I(\tau), z_I(\tau), u_I(\tau))]d\tau \,.$$

Hence, the LFD implements a *virtual sensor* that generates an estimate of the measurement at discrete-time k given by

$$\begin{aligned} \hat{m}_I^{(i)}(k|t_{sI}^{(i)}) &= m_I^{(i)}(t_{sI}^{(i)}) \\ &\quad + \int_{t_{sI}^{(i)}}^{k} [f_I^{(i)}(\hat{m}_I(\tau|t_{sI}^{(i)}), u_I(\tau)) + g_I^{(i)}(\hat{m}_I(\tau|t_{sI}^{(i)}), \hat{m}_{zI}(\tau|t_{sI}^{(i)}), u_I(\tau)) \\ &\quad + \hat{\eta}_I^{(i)}(\hat{m}_I(\tau|t_{sI}^{(i)}), \hat{m}_{zI}(\tau|t_{sI}^{(i)}), u_I(\tau))]d\tau \,, \end{aligned} \tag{41}$$

where $\hat{\eta}_I$ characterizes an adaptive approximator designed to learn the unknown modeling uncertainty function η_I [27] and $\hat{m}_{zI}$ are the projections of the measured interconnection variables m_{zI}. An example enhancing the re-synchronization procedure for one LFD monitoring a subsystem with three state variables is illustrated in Fig. 5.

[1] As example, this could be obtained in accordance with the IEEE 1588-2002 standard ("Standard for a Precision Clock Synchronization Protocol for Networked Measurement and Control Systems"), where each diagnoser can be selected as a synchronization master for the sensors that communicate with it.

[2] Recall that the sampling time of the diagnosers is supposed to be unitary for simplicity.

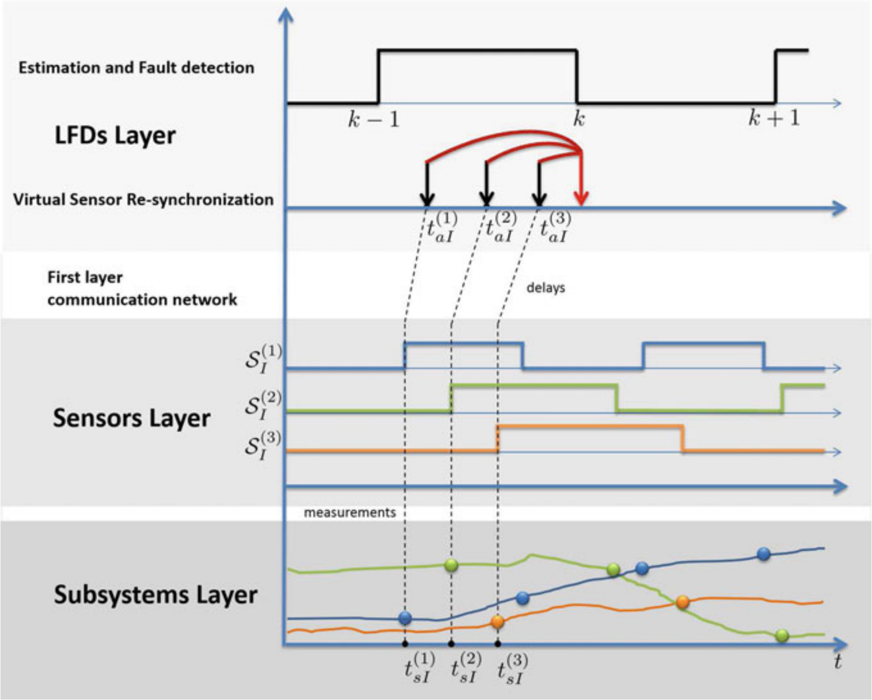


Fig. 5 The re-synchronization procedure [14] needed to manage delays and packet losses in the communication networks between each LFD and its local sensors. A single LFD is considered whose local model depends on three variables, which are measured by three different sensors. The clock signals of each layer involved are shown

Remark 3 It is worth noting that the discrete-time index $k \in \mathbb{Z}$ represents kind of a "virtual Time Stamp" (vTS) computed by the LFDs after the re-synchronization task and communicated in the second level communication network between LFDs. This will be exploited in Sect. 4.2.

Remark 4 Although in (41), for analysis purposes, $\hat{\eta}_I$ represents the output of a continuous-time adaptive approximator, for implementation reasons, a suitable discrete-time approximator will be used, designed as explained in Sect. 4.4.

The above-described projection and re-synchronization procedure gives rise to an additional source of measurement uncertainty: the *virtual measurement error*, which is defined as

$$\xi_I^{(i)}(k) \triangleq \hat{m}_I^{(i)}(k|t_{sI}^{(i)}) - x_I^{(i)}(k).$$

For the sake of analysis, it is worth noting that, due to synchronization and measurement noise, the virtual measurement error is given by

$$
\begin{aligned}
\xi_I^{(i)}(k) &= m_I^{(i)}(t_{sI}^{(i)}) - x_I^{(i)}(t_{sI}^{(i)}) \\
&\quad + \int_{t_{sI}^{(i)}}^{k} [\Delta_{synch} f_I^{(i)}(\tau) + \Delta_{synch} g_I^{(i)}(\tau) + \Delta_{synch} \eta_I^{(i)}(\tau)] d\tau \\
&= w_I^{(i)}(t_{sI}^{(i)}) + \int_{t_{sI}^{(i)}}^{k} [\Delta_{synch} f_I^{(i)}(\tau) + \Delta_{synch} g_I^{(i)}(\tau) + \Delta_{synch} \eta_I^{(i)}(\tau)] d\tau \,,
\end{aligned}
\tag{42}
$$

where

$$
\Delta_{synch} f_I^{(i)}(\tau) \triangleq f_I^{(i)}(\hat{m}_I(\tau|t_{sI}^{(i)}), u_I(\tau)) - f_I^{(i)}(x_I(\tau), u_I(\tau)) \,,
$$

$$
\Delta_{synch} g_I^{(i)}(\tau) \triangleq g_I^{(i)}(\hat{m}_I(\tau|t_{sI}^{(i)}), \hat{m}_{zI}(\tau|t_{sI}^{(i)}), u_I(\tau)) - g_I^{(i)}(x_I(\tau), z_I(\tau), u_I(\tau)) \,,
$$

and

$$
\Delta_{synch} \eta_I^{(i)}(\tau) \triangleq \hat{\eta}_I^{(i)}(\hat{m}_I(\tau|t_{sI}^{(i)}), \hat{m}_{zI}(\tau|t_{sI}^{(i)}), u_I(\tau)) - \eta_I^{(i)}(x_I(\tau), z_I(\tau), u_I(\tau)) \,.
$$

For notational convenience, we now collect the projected measurements $\hat{m}_I^{(i)}(k|t_{sI}^{(i)})$ in a vector, which, in the following, we denote as $y_I(k)$, with k being its vTS:

$$
y_I(k) = \text{col} \left\{ \hat{m}_I^{(i)}(k|t_{sI}^{(i)}), i = 1, \ldots, n_I \right\} \,.
$$

Therefore, it is as if the virtual sensor implemented by the LFDs takes uncertain local measurements y_I of the state x_I, according to

$$
y_I(k) = x_I(k) + \xi_I(k),
$$

where ξ_I is the unknown virtual measurement error (42). Moreover, in place of the interconnection variables z_I, only the vector

$$
v_I(k) = z_I(k) + \varsigma_I(k)
$$

is available for diagnosis, as it is possible to see in Fig. 6, where ς_I is composed by the components of ξ_J affecting the relevant components of y_J (as before, J refers to a neighboring subsystem). For simplicity, we assume here that the control signal u_I is available to the diagnoser without any delays or other uncertainty.

The virtual measuring errors ξ_I and ς_I are unstructured and unknown. For each $i = 1, \ldots, n_I$ and $j = 1, \ldots, \bar{n}_I$, it is possible to compute a bound for their components using (42):

$$
\left| \xi_I^{(i)}(k) \right| \leq \bar{\xi}_I^{(i)}(k), \qquad \left| \varsigma_I^{(j)}(k) \right| \leq \bar{\varsigma}_I^{(j)}(k),
$$

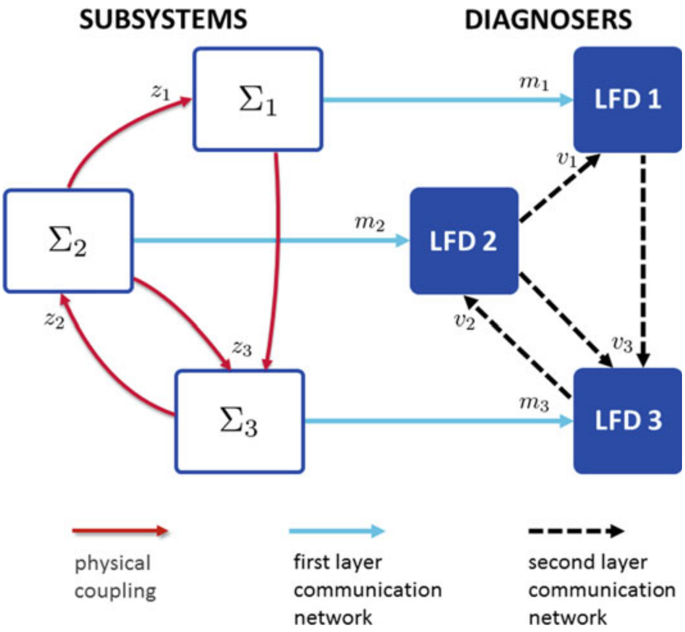


Fig. 6 An example of the multi-layer fault detection architecture. The interconnection variables z_I and the corresponding projected measurements v_I communicated among the diagnosers

where

$$\bar{\xi}_I^{(i)}(k) = \bar{w}_I^{(i)}(t_{sI}^{(i)}) + \int_{t_{sI}^{(i)}}^{k} \bar{\Delta}_{synch} f_I^{(i)}(\tau) + \bar{\Delta}_{synch} g_I^{(i)}(\tau) + \bar{\Delta}_{synch} \eta_I^{(i)}(\tau) d\tau \quad (43)$$

is a positive function, $\bar{w}_I^{(i)}$ is the one defined in Assumption 3,

$$\bar{\Delta}_{synch} f_I^{(i)}(\tau) = \max_{x_I \in \mathscr{R}^{n_I}} \left| f_I^{(i)}(\hat{m}_I(\tau), u_I(\tau)) - f_I^{(i)}(x_I(\tau), u_I(\tau)) \right|,$$

$$\bar{\Delta}_{synch} g_I^{(i)}(\tau) = \max_{x_I \in \mathscr{R}^{n_I}, z_I \in \mathscr{R}^{\bar{n}_I}} \left| g_I^{(i)}(\hat{m}_I(\tau), \hat{m}_{zI}(\tau), u_I(\tau)) - g_I^{(i)}(x_I(\tau), z_I(\tau), u_I(\tau)) \right|,$$

remembering that the sets $\mathscr{R}^{n_I}$, $\mathscr{R}^{\bar{n}_I}$ are the domain of the state and interconnection variables, respectively, and $\bar{\Delta}_{synch} \eta_I^{(i)}(\tau)$ can be computed in an analogous way as in (65) (see Sect. 4.6). The bound $\bar{\varsigma}_I$ is computed with the same procedure by the neighboring subsystems. In the next section, the fault diagnosis procedure is presented.

4.2 The Distributed Fault Detection Methodology

For fault detection purposes, each LFD communicates with neighboring LFDs. It is assumed that the inter-LFD communication is carried over a packet-switched network, which we call the *second level communication network*, possibly subject to packet delays and losses. In order to manage delays in this network, the data packets are Time Stamped, with the virtual Time Stamp, which contains the time instant the virtual measurements are referred to. In this layer, we assume to have perfect clock synchronization between the LFDs. In this way, all the devices of the monitoring architecture can share the same clock, that is, they know the reference time, and the use of Time Stamps can be valid.

Furthermore, we propose to provide each LFD with a buffer to collect the variables sent by neighbors. In the following, we denote with the superscript "b" the most recent value of a variable (or of a communicated function value) in the corresponding buffer of a given LFD; for example, v_I^b denotes the most recent value of the measured interconnection vector v_I contained in the buffer of the I-th LFD, while $[f_I(\cdot)]^b$ denotes the most recent value of the function $[f_I(\cdot)]$ in the buffer.

Each LFD computes a nonlinear adaptive estimate $\tilde{x}_I$ of the associated monitored subsystem state x_I. The local estimator, called *Fault Detection Approximation Estimator* (FDAE), is based on the local discrete-time nominal model (Eq. (40)). Similarly to what done in the first part of this chapter (Sect. 3), to dampen the effect of the virtual measurement error $\xi_I(k)$, each measured variable $y_I^{(i)} = x_I^{(i)} + \xi_I^{(i)}$ is filtered by $H(z)$, where $H(z)$ is a p-th order, asymptotically stable filter (poles lie inside the open unit disc $|z| = 1$) with proper transfer function

$$H(z) = \frac{d_0 + d_1 z^{-1} + d_2 z^{-2} + \ldots + d_p z^{-p}}{1 + c_1 z^{-1} + \ldots + c_p z^{-p}}. \tag{44}$$

Generally, each measured variable $y_I^{(i)}(k)$ can be filtered by a different filter but, without loss of generality, we consider $H(z)$ to be the same for all the output variables, in order to simplify notation and presentation. In addition, note that the form of $H(z)$ allows both IIR and FIR types of digital filters. The filter $H(z)$ can be written as $H(z) = zH_p(z)$ where $H_p(z)$ is the strictly proper transfer function

$$H_p(z) = \frac{d_0 z^{-1} + d_1 z^{-2} + d_2 z^{-3} + \ldots + d_p z^{-(p+1)}}{1 + c_1 z^{-1} + \ldots + c_p z^{-p}}. \tag{45}$$

Note that, the filter $H_p(z)$ is also asymptotically stable since it comprises of the same poles as $H(z)$ with an additional pole at $z = 0$ (inside $|z| = 1$). Since the filters $H(z)$ and $H_p(z)$ (with impulse responses $h(t)$ and $h_p(t)$, respectively) are asymptotically stable, they are also BIBO stable. Therefore, for bounded virtual measurement error $\xi_I(k)$, the filtered virtual measurement error[3] $\Xi_I(k) \triangleq H(z)\,[\xi_I(k)]$ is bounded as

[3] For notational convenience, we use the shorthand $H(z)\,[\xi(k)]$ to denote $\mathscr{Z}^{-1}\,\{H(z)\Xi(z)\}$.

follows:

$$\left|\Xi_I^{(i)}(k)\right| \leq \bar{\Xi}_I^{(i)}(k) \quad i = 1, \ldots, n_I \tag{46}$$

where $\bar{\Xi}_I^{(i)}$ are bounding functions that can be computed as $\bar{\Xi}_I^{(i)} \triangleq \bar{H}(z)[\bar{\xi}_I^{(i)}]$, being $\bar{H}(z)$ a filter with impulse response $\bar{h}(k)$ that satisfies $|h(k)| \leq \bar{h}(k)$ and using Eq. (43). The selection of suitable filters $\bar{H}(z)$ can be made by utilizing the methods indicated in Sect. 4.7. Note that we denote with capital letters the filtered signals.

4.3 Fault Detection Estimation and Residual Generation

In this subsection, we present a method for computing the local state estimate $\tilde{x}_I$ for fault detection purposes. The local estimation $\tilde{x}_I^{(i)}$ is given by

$$\begin{aligned}\tilde{x}_I^{(i)}(k+1) = f_I^{(i)}(y_I(k), u_I(k)) + g_I^{(i)}(y_I(k), v_I^b(k), u_I(k))\\ + \hat{\eta}_I^{(i)}(y_I(k), v_I^b(k), u_I(k), \hat{\vartheta}_I(k)),\end{aligned} \tag{47}$$

with initial condition $\tilde{x}_I^{(i)}(0) = y_I^{(i)}(0)$, where $\hat{\eta}_I$ is the output of an adaptive approximator designed in Sect. 4.4 to learn the unknown modeling uncertainty function η_I, $\hat{\vartheta}_I \in \hat{\Theta}_I$ denotes its adjustable parameters vector and t_b is the virtual time stamp of the most recent information received v_I^b in the buffer at time k.

The local estimation residual error $r_I(k)$ is defined as

$$r_I(k) \triangleq Y_I(k) - \widehat{Y}_I(k), \tag{48}$$

where we obtain the filtered output $Y_I(k)$ by locally filtering the measurement output signal $y_I(k)$

$$Y_I(k) \triangleq H(z)\left[y_I(k)\right], \tag{49}$$

and the output estimates as

$$\widehat{Y}_I(k) \triangleq H(z)\left[\tilde{x}_I(k)\right]. \tag{50}$$

The residual constitutes the basis of the fault detection scheme. It can be compared, component by component, to a suitable adaptive detection threshold $\bar{r}_I \in \mathbb{R}^{n_I}$, thus generating a local fault decision attesting the status of the subsystem: healthy or faulty. A fault in the overall system is said to be detected when $|r_I^{(i)}(k)| > \bar{r}_I^{(i)}(k)$, for at least one component i in any I-th LFD.

We now analyze the filtered measurements and estimates:

$$\begin{aligned}Y_I(k) &= H(z)\left[y_I(k)\right] = H(z)\left[x_I(k) + \xi_I(k)\right]\\ &= H_p(z)\left[z\left[x_I(k)\right]\right] + \Xi_I(k).\end{aligned} \tag{51}$$

In the absence of any faults (i.e., $\phi_I\big(x_I(k), z_I(k), u_I(k)\big) = 0$), (51) becomes

$$\begin{aligned} Y_I(k) &= H_p(z)\big[x_I(k+1) + z\big[x_I(0)\delta(k)\big]\big] + \Xi_I(k) \\ &= H_p(z)\big[f_I\big(x_I(k), u_I(k)\big) + g_I\big(x_I(k), z_I(k), u_I(k)\big) \\ &\quad + \eta_I\big(x_I(k), z_I(k), u_I(k)\big)\big] + h(k)x_I(0) + \Xi_I(k), \end{aligned} \tag{52}$$

where $\delta(k)$ denotes the discrete-time unit-impulse sequence.

The filtered output estimation model for Y_I, denoted by $\widehat{Y}_I$, can be analyzed from the estimate provided by (47) as follows:

$$\begin{aligned} \widehat{Y}_I^{(i)}(k) =& H_p(z)\Bigg[f_I^{(i)}\big(y_I(k), u_I(k)\big) + g_I^{(i)}\big(y_I(k), v_I^b(k), u_I(k)\big) \\ &+ \hat{\eta}_I^{(i)}\big(y_I(k), v_I^b(k), u_I(k), \hat{\vartheta}_I(k)\big)\Bigg] + h(k)y_I^{(i)}(0). \end{aligned} \tag{53}$$

Therefore, the residual (48) is readily computable from (49) and (50). The residual is analyzed in Sect. 4.6 to obtain a suitable adaptive detection threshold. Now, we design the adaptive approximator $\hat{\eta}_I$, needed to compute the state estimate (47) and hence (50).

4.4 Learning of the Modeling Uncertainty

Reducing the modeling uncertainty enables improved detection thresholds which, in turn, results in better detection capabilities. In this subsection, we consider the design of a nonlinear adaptive approximator, exploiting the variables available in the local buffers in each LFD to manage communication delays (the details of the delay compensation strategy are given in Sect. 4.5). The structure of the linear in the parameters nonlinear multivariable approximator is not dealt with in this chapter (nonlinear approximation schemes like neural networks, fuzzy logic networks, wavelet networks, spline functions, polynomials, etc., can be used).

As shown later on in this subsection, adaptation of the parameters $\hat{\vartheta}_I$ of the approximator is achieved through the design of a dynamic state estimator which takes on the form:

$$\hat{x}_I^{(i)}(k+1) = \lambda(\hat{x}_I^{(i)}(k) - y_I^{(i)}(k)) + f_I^{(i)}(y_I, u_I) + g_I^{(i)}(y_I, v_I^b, u_I) + \hat{\eta}_I^{(i)}(y_I, v_I^b, u_I, \hat{\vartheta}_I), \tag{54}$$

where $0 < \lambda < 1$ is a design parameter. Let us introduce the estimation error

$$\varepsilon_I(k) \triangleq y_I(k) - \hat{x}_I(k)$$

We compute the i-th state estimation error component as follows:

$$\begin{aligned}\varepsilon_I^{(i)}(k+1) &= y_I^{(i)}(k+1) - \hat{x}_I^{(i)}(k+1)\\ &= \lambda\varepsilon_I^{(i)} + \Delta f_I^{(i)} + \Delta g_I^{(i)} + \Delta\eta_I^{(i)} - \lambda\xi_I^{(i)} + \lambda\xi_I^{(i)}(k) + \xi_I^{(i)}(k+1)\,,\end{aligned} \tag{55}$$

where

$$\Delta f_I^{(i)} \triangleq f_I^{(i)}(x_I, u_I) - f_I^{(i)}(y_I, u_I)\,,$$

$$\Delta g_I^{(i)} \triangleq g_I^{(i)}(x_I, z_I, u_I) - g_I^{(i)}(y_I, v_I^b, u_I)\,,$$

and

$$\Delta\eta_I^{(i)} \triangleq \eta_I^{(i)}(x_I, z_I, u_I) - \hat{\eta}_I^{(i)}(y_I, v_I^b, u_I, \hat{\vartheta}_I)\,.$$

From this equation, the following learning law can be derived using Lyapunov stability techniques (see [107]) for every I:

$$\hat{\vartheta}_I(k+1) = P_{\hat{\Theta}_I}\left[\hat{\vartheta}_I(k) + \gamma_I L_I^\top[\varepsilon_I(k+1) - \lambda\varepsilon_I(k)\right]\,, \tag{56}$$

where $L_I^\top = \partial\hat{\eta}_I/\partial\hat{\vartheta}_I$ is the gradient matrix of the online approximator with respect to its adjustable parameters and $\gamma_I = \mu_I/\rho_I + \left\|L_I^\top\right\|_F^2$, with $P_{\hat{\Theta}_I}$ being a projection operator restricting $\hat{\vartheta}_I$ within $\hat{\Theta}_I$ [68], $\|\cdot\|_F$ denotes the Frobenius norm and $\rho_I > 0$, $0 < \mu_I < 2$ are design constants that guarantee the stability of the learning law [68].

4.5 Delay Compensation Strategy

Next, we analyze the properties of the Fault Detection estimator introduced in Sect. 4.3, where the filtered measurements are used; in particular, we explain how the estimator manages delays and packet losses in the second-level communication network between diagnosers.

In order to compute (47) and (54), the generic J-th diagnoser communicates to the neighboring LFDs the current values of the variables v_I. It is worth noting that this information exchange between diagnosers can be affected by time-varying delays and packet losses and hence a compensation strategy has to be devised. The delay compensation strategy is derived without any assumption on the delay length, thus eventually dealing with the problem of packet losses and "out-of-sequence" packets. We assume that the communication network between diagnosers is designed so to avoid pathological scenarios, such as, for example, a situation in which the communication delay is always larger than the sampling time. It is important to note that a re-synchronization strategy like the one used in the first level communication networks cannot be used in this case, since here we consider data exchanged between different LFDs, and each LFD, of course, does not know the model of neighboring subsystems.

As in [12], thanks to the use of the virtual Time Stamps, the most recent measurements and information are considered. When a data packet arrives, its virtual Time Stamp v_{TS} is compared to t_b, which is the virtual Time Stamp of the information already in the buffer. If $v_{\mathrm{TS}} > t_b$, then the novel data packet takes its place in the buffer and $t_b \leftarrow v_{\mathrm{TS}}$. At time t_c, with $k < t_c < k+1$, each LFD computes the estimates for the time instant $k+1$ using information referred to time k. A variable in the buffer is up to date if $t_b = k$. Should a delay or a packet loss occur in the second level communication network, we proceed as follows. If some of the interconnection variables are not up to date, that is $t_b < k$, then the learning of the modeling uncertainty function η_I (56) is temporarily paused. Anyway, not up to date interconnection variables are used to compute the local value of the interconnection function in the state estimators (47) and (54), but this error is taken into account in the computation of the detection threshold, as will be seen in the following subsection.

4.6 Detection Threshold

In order to define an appropriate threshold for the detection of faults, we now analyze the dynamics of the output estimation error when the system is under healthy mode of behavior. Since, from (52) we have

$$\begin{aligned} Y_I^{(i)}(k) = H_p(z)\big[f_I^{(i)}\big(x_I(k), u_I(k)\big) + g_I^{(i)}\big(x_I(k), z_I(k), u_I(k)\big) \\ + \eta_I^{(i)}\big(x_I(k), z_I(k), u_I(k)\big)\big] + h(k)x_I^{(i)}(0) + \Xi_I^{(i)}(k), \end{aligned} \tag{57}$$

we are able to compute the residual defined in (48) by using (53) and (57):

$$r_I^{(i)}(k) = \left[\chi_I^{(i)}(k)\right]^b - \xi_I^{(i)}(0)h(k) + \Xi_I^{(i)}(k)\,, \tag{58}$$

where the total uncertainty term $\chi_I^{(i)}(k)$ is defined as

$$\chi_I^{(i)}(k) \triangleq H_p(z)\big[\Delta f_I^{(i)}(k) + \Delta g_I^{(i)}(k) + \Delta \eta_I^{(i)}(k)\big]. \tag{59}$$

The function error $\Delta\eta_I$ can be computed as the sum of four different terms:

$$\Delta\eta_I = L_I\tilde{\vartheta}_I + \upsilon_I + \Delta\hat{\eta}_I + \Delta\eta_I^{\tau}\,. \tag{60}$$

The first term takes into account the error due to the parameters' estimation. This error can be characterized by introducing an *optimal weight vector* [98] $\hat{\vartheta}_I^*$ as follows:

$$\hat{\vartheta}_I^* \triangleq \arg\min_{\hat{\vartheta}_I} \sup_{x_I, z_I, u_I} \left\| \eta_I(x_I, z_I, u_I) - \hat{\eta}_I(x_I, z_I, u_I, \hat{\vartheta}_I \right\|, \tag{61}$$

with $\hat{\vartheta}_I, x_I, z_I, u_I$ taking values in their respective domains, and by defining the parameter estimation error

$$\tilde{\vartheta}_I \triangleq \hat{\vartheta}_I^* - \hat{\vartheta}_I .$$

The second term in (60) is the so-called *Minimum Functional Approximation Error* υ_I, which describes the least possible approximation error that can be obtained at time k if $\hat{\vartheta}_I$ were optimally chosen:

$$\upsilon_I(k) \triangleq \eta_I(x_I, z_I, u_I) - \hat{\eta}_I(x_I, z_I, u_I, \hat{\vartheta}_I^*) .$$

Then, a term representing the error caused by the use of the uncertain measurements instead of the actual values of the state variables is defined:

$$\Delta\hat{\eta}_I \triangleq \hat{\eta}_I(x_I, z_I, u_I, \hat{\vartheta}_I) - \hat{\eta}_I(y_I, v_I, u_I, \hat{\vartheta}_I) .$$

Finally, the estimation error due to the use of delayed measurements is taken into account by

$$\Delta\eta_I^{\tau} \triangleq \hat{\eta}_I(y_I, v_I, u_I, \hat{\vartheta}_I) - \hat{\eta}_I(y_I, v_I^b, u_I, \hat{\vartheta}_I)$$

where v_I is the current measured variable and v_I^b is the value in the buffer, which is “old” in the presence of delays. Clearly, $\Delta\eta_I^{\tau} = 0$ when up to date measurements are used (in this case, $v_I^b = v_I$).

Using (60), the total uncertainty term $\chi_I^{(i)}(k)$ in (59) can be rewritten as

$$\chi_I^{(i)}(k) \triangleq H_p(z)\big[\Delta f_I^{(i)}(k) + \Delta g_I^{(i)}(k) + L_I^{(i)}\tilde{\vartheta}_I(k) + \upsilon_I^{(i)}(k) \\ + \Delta\hat{\eta}_I^{(i)}(k) + \Delta\eta_I^{\tau(i)}(k)\big], \tag{62}$$

where $L_I^{(s_I)}$ indicates the s_I-th line of the matrix L_I. Using the triangle inequality, (58) satisfies:

$$\begin{aligned} \left| r_I^{(i)}(k) \right| &\le \left| \left[\chi_I^{(i)}(k) \right]^b \right| + \left| \xi_I^{(i)}(0) h(k) \right| + \left| \Xi_I^{(i)}(k) \right| \\ &\le \left[\left| \chi_I^{(i)}(k) \right| \right]^b + \bar{\xi}_I^{(i)}(0) \left| h(k) \right| + \bar{\Xi}_I^{(i)}(k). \end{aligned} \tag{63}$$

From (62) and using again the triangle inequality, we can obtain

$$\left| \chi_I^{(i)}(k) \right| \le \left| H_p(z)\big[\Delta f_I^{(i)}(k) + \Delta g_I^{(i)}(k) + \Delta\eta_I^{(i)}(k)\big] \right|$$

$$\begin{aligned}
&\leq \sum_{n=0}^{k} \left|h_p(k-n)\right| \Big| \Delta f_I^{(i)}(n) + \Delta g_I^{(i)}(n) + L_I^{(i)} \tilde{\vartheta}_I(n) + \upsilon_I^{(i)}(n) \\
&\quad + \Delta \hat{\eta}_I^{(i)}(n) + \Delta \eta_I^{\tau(i)}(n) \Big| \\
&\leq \bar{\chi}_I^{(i)}(k) \triangleq \bar{H}_p(z) \big[\bar{\Delta} f_I^{(i)}(k) + \bar{\Delta} g_I^{(i)}(k) + \bar{\Delta} \eta_I^{(i)}(k) \big], \qquad (64)
\end{aligned}$$

where $\bar{H}_p(z)$ is the transfer function with impulse response that satisfies $\left|h_p(k)\right| \leq \bar{h}_p(k)$ (more details for the selection of $\bar{H}_p(z)$ are given in Sect. 4.7),

$$\bar{\Delta} f_I^{(i)}(k) \triangleq \max_{|\xi_I| \leq \bar{\xi}_I} \left\{ \left| \Delta f_I^{(i)}(k) \right| \right\},$$

$$\bar{\Delta} g_I^{(i)}(k) \triangleq \max_{|\xi_I| \leq \bar{\xi}_I(k)} \max_{|\varsigma_I| \leq \bar{\varsigma}_I(k)} \left\{ \left| \Delta g_I^{(i)}(k) \right| \right\}$$

and

$$\begin{aligned}
\bar{\Delta} \eta_I^{(i)}(k) \triangleq \left\| L_I^{(i)} \right\| \kappa_I(\hat{\vartheta}_I) + \bar{\upsilon}_I^{(i)}(k) + \max_{|\xi_I| \leq \bar{\xi}_I(k)} \max_{|\varsigma_I| \leq \bar{\varsigma}_I(k)} \left| \Delta \hat{\eta}_I^{(i)}(k) \right| \\
+ \max_{v_I \in \mathscr{R}^v} \left| \hat{\eta}_I^{(i)}(y_I, v_I, u_I, \hat{\vartheta}_I) - \hat{\eta}_I^{(i)}(y_I, v_I^b(t_b), u_I, \hat{\vartheta}_I) \right|,
\end{aligned} \qquad (65)$$

with $\bar{\upsilon}_I$ denoting a bound to the minimum functional approximation error, the function κ_I being such that $\kappa_I(\hat{\vartheta}_I) \geq \left\| \tilde{\vartheta}_I \right\|$ and $\mathscr{R}^{v_I} \subset \mathbb{R}^{\bar{\eta}_I}$, where this last term represents a local domain of the interconnection variable and is communicated by the neighboring LFDs at $k = 0$. It is important to remark that $\mathscr{R}^{v_I}$ coincides with the domain $\mathscr{D}_{z_I}$ for subsystem I. Thanks to the way the threshold is designed from (63), it is straightforward that it guarantees the absence of false alarms, since the residual *prior to the fault occurrence* always satisfies

$$\left| r_I^{(i)}(k) \right| \leq \bar{r}_I^{(i)}(k),$$

where the detection threshold $\bar{r}_I^{(i)}$ is defined as

$$\bar{r}_I^{(i)}(k) \triangleq \left[\bar{\chi}_I^{(i)}(k) \right]^b + \bar{\xi}_I^{(i)}(0) \left| h(k) \right| + \bar{\Xi}_I^{(i)}(k). \qquad (66)$$

Remark 5 Notice that, even in the case of a conservative bound $\bar{\xi}_I^{(i)}$, the second term $\bar{\xi}_I^{(i)} \left| h(k) \right|$ affects the detection threshold only during the initial portion of the transient (the impulse response $h(k)$ of the filter $H(z)$ decays exponentially). Moreover, the term $\bar{\Xi}_I^{(i)}$ in (65) takes into account the uncertainty due to the delays in the communication network between LFDs. This term is instrumental to ensure the absence of false alarms caused by these communication delays.

Remark 6 The terms $\bar{\xi}_I(k)$ and $\bar{\varsigma}_I(k)$ are computed by the LFDs at each time step after the re-synchronization task (see (43)) and are available to compute the fault detection threshold.

Remark 7 Admittedly, the bounds used in (64) and (65) give rise to conservative thresholds but have the advantage of guaranteeing the absence of false-positive alarms and of being easily computable requiring a small amount of data to be exchanged between the LFDs. In the presence of a priori knowledge on the process to be monitored, tighter bound could be devised. For example, Lipschitz conditions on the local models could be easily exploited to devise tighter detection thresholds.

4.7 Selection of Filter $\bar{H}_p(z)$

A practical issue that requires consideration is the selection of the filter $\bar{H}_p(z)$ whose impulse response must satisfy $|h_p(t)| \leq \bar{h}_p(t)$ as stated before. In the case where the impulse response $h_p(t)$ is nonnegative, the selection $\bar{H}_p(z) = H_p(z)$ is trivial. Sufficient conditions for nonnegative impulse response for a class of discrete-time transfer functions are given in [60]. In the following, we present two methods for choosing $\bar{H}_p(z)$, one considering $H(z)$ as a digital IIR filter and the other one as a FIR filter.

First, we consider the case where $H(z)$ is an IIR filter. Due to the way $H_p(z)$ was defined, $H_p(z)$ is strictly proper and asymptotically stable. Hence, the impulse response $h_p(k)$ satisfies $|h_p(k)| \leq \kappa\lambda^k$ for all $k \in \mathbb{N}$, for some $\kappa > 0$ and $\lambda \in [0, 1)$. Since $|h_p(k)| \leq \bar{h}_p(k)$ must hold, the impulse response $\bar{h}_p(k)$ can be selected as $\bar{h}_p(k) = \kappa\lambda^k$ and thus $\bar{H}_p(z) = \frac{\kappa}{1-\lambda z^{-1}}$.

Now, let us consider the case in which $H(z)$ is a FIR filter. FIR filters have several advantages, as they are inherently stable and can easily be designed to be linear phase which corresponds to uniform delay at all frequencies. Let $H(z)$ be a p-th order FIR filter given by $H(z) = \sum_{n=0}^{p} d_n z^{-n}$. Therefore, $H_p(z) = z^{-1}H(z) = \sum_{n=0}^{p} d_n z^{-(n+1)}$ and $\bar{h}_p(k)$ can be selected as $\bar{h}_p(k) = |h_p(k)|$ which leads to the FIR filter $\bar{H}_p(z) = \sum_{n=0}^{p} |d_n| z^{-(n+1)}$.

4.8 The Local Fault Detection Algorithm

Now, all the elements needed to implement the fault detection scheme are available. For the sake of clarity, the implementation of the local fault detection methodology is sketched in the following Algorithm 1. Extensive simulation results showing the effectiveness of the presented approach can be found in [14].

Algorithm 1 Fault detection algorithm for the I-th LFD

```
Learning = ON
Initialize the estimate $\hat{x}_I(0) = y_I(0)$
Initialize the estimate $\tilde{x}_I(0) = y_I(0)$
Compute the estimate $\hat{x}_I(1)$ (Eq. (54))
Compute the estimate $\tilde{x}_I(1)$ (Eq. (47))
Set $k = 1$
while A fault is not detected do
  Measurements $y_I(k)$ are acquired
  Compute $\varepsilon_I(k) = y_I(k) - \hat{x}_I(k)$ (for learning)
  Compute $Y_I(k)$ (Eq. (49)), $\widehat{Y}_I(k)$ (Eq. (50))
  Compute the residual $r_I(k) = Y_I(k) - \widehat{Y}_I(k)$
  Information from neighbors is acquired
  Compute the threshold $\bar{r}_I(k)$ (Eq. (66))
  Compare $|r_I(k)|$ with $\bar{r}_I(k)$
  if $|r_I(k)| > \bar{r}_I(k)$ then
    A fault is detected
    Learning = OFF
  end if
  if Some components $i$ of $v_I(k)$ are not received then
    Learning = OFF
  else
    Learning = ON
    $v_I^{b(i)}(k) = v_I^{(i)}(k)$
  end if
  if Learning = ON then
    Update $\hat{\vartheta}_I(k)$ (Eq. (56))
  else
    $\hat{\vartheta}_I(k) = \hat{\vartheta}_I(k-1)$
  end if
  Compute the novel estimate $\hat{x}_I(k+1)$ (Eq. (54))
  Compute the novel estimate $\tilde{x}_I(k+1)$ (Eq. (47))
  $k = k + 1$
end while
```

4.9 Detectability Conditions

In this subsection, we address some sufficient conditions for detectability of faults by the proposed distributed networked fault detection scheme, thus considering the behavior of the fault detection algorithm in the case of a faulty system. We assume that at an unknown time k_0 a fault ϕ_I occurs. The fault detectability analysis constitutes a theoretical result that characterizes quantitatively (and implicitly) the class of faults detectable by the proposed scheme.

Theorem 4 (Fault Detectability) *A fault in the I-th subsystem occurring at time $k = k_0$ is detectable at a certain time $k = k_d$ if the fault function $\phi_I^{(i)}(x_I(k), z_I(k), u_I(k))$*

satisfies the following inequality for some $i = 1, \ldots, n_I$*:*

$$\left|\sum_{n=k_0}^{k_d} h_p(k-n)\phi_I^{(i)}\big(x_I(n), z_I(n), u_I(n)\big)\right| > 2\bar{r}_I^{(i)}(k_d). \tag{67}$$

Proof After fault occurrence, that is for $k > k_0$, Eq. (58) becomes

$$\begin{aligned} r_I^{(i)}(k) &= \chi_I^{(i)}(k)^b + H_p(z)\big[\phi_I^{(i)}\big(x_I(k), z_I(k), u_I(k)\big)\big] - \xi_I^{(i)}(0)h(k) + \Xi_I^{(i)}(k) \\ &= \chi_I^{(i)}(k)^b - \xi_I^{(i)}(0)h(k) + \Xi_I^{(i)}(k) + H_p(z)\big[\phi_I^{(i)}\big(x_I(k), z_I(k), u_I(k)\big)\big]. \end{aligned} \tag{68}$$

Using the triangle inequality, from (68) we can write

$$\begin{aligned} \left|r_I^{(i)}(k)\right| \geq &- \left|\chi_I^{(i)}(k)^b\right| - \left|\xi_I^{(i)}(0)h(k)\right| - \left|\Xi_I^{(i)}(k)\right| \\ &+ \left|H_p(z)\big[\phi_I^{(i)}\big(x_I(k), z_I(k), u_I(k)\big)\big]\right| \end{aligned} \tag{69}$$

and by using a similar procedure as in the derivation of (66), (69) becomes

$$\left|r_I^{(i)}(k)\right| \geq -\bar{r}_I^{(i)}(k) + \left|H_p(z)\big[\phi_I^{(i)}\big(x_I(k), z_I(k), u_I(k)\big)\big]\right|. \tag{70}$$

For fault detection at time $k = k_d$, the inequality $|r_I^{(i)}(k_d)| > \bar{r}_I^{(i)}(k_d)$ must hold for some $i = 1, \ldots, n_I$, so the final fault detectability condition is obtained:

$$\left|H_p(z)\big[\phi_I^{(i)}(x_I(k_d), z_I(k_d), u_I(k_d))\big]\right| > 2\bar{r}_I^{(i)}(k_d).$$

This can be rewritten in the summation form (67) of the Theorem. □

This theorem provides a sufficient condition for the implicit characterization of a class of faults that can be detected by the proposed fault detection scheme. Let us note that the detectability condition represents the minimum cumulative magnitude of the fault that can be detected under a specific trajectory of the system. It is possible to study this condition off line for representative trajectories of the system.

4.10 Identification of the Faulty Subsystem

In the next section, we consider the fault diagnosis problem. More specifically, we illustrate an approach for the adaptive learning of the local fault function after fault detection. Before developing the adaptive approximation procedure, we present an important remark.

A fundamental question regarding fault detectability is whether the fault that occurs in subsystem Σ_J is detectable not only by the LFD $\mathscr{F}_J$, but also by the LFD $\mathscr{F}_I$ of the neighboring subsystem Σ_I, whose state is influenced by Σ_J dynamics.

It can be shown (the interested reader can refer to [52]), that the proposed fault detection scheme guarantees that, a process fault $\phi_J(\cdot)$ occurring in subsystem Σ_J which affects Σ_I, can only be detected by its corresponding LFD $\mathscr{F}_J$ and not by the LFD $\mathscr{F}_I$. This result is essentially the implication of using the measurements of the state and interconnection variables in the estimation model given by (11). Qualitatively, this can be explained as follows. When a process fault occurs in Σ_J, the fault affects its states which in turn affect other subsystems through the interconnection variables. So, the states of Σ_J are "contaminated" by the process fault and the measurements of these states also contain the process fault effects. Therefore, a subsystem Σ_I that is affected by Σ_J, is affected by the process fault that occurred in Σ_J through the interconnection variables z_I and the detection LFD $\mathscr{F}_I$ makes use of the measurements v_I which are also "contaminated" by the same fault. Hence, the effect of the process fault that occurred in Σ_J, is "canceled out" in the LFD $\mathscr{F}_I$ and it is unable detect the fault. Hence, a process fault occurring in subsystem Σ_J is detectable only by its respective detection LFD $\mathscr{F}_J$ and not by any other LFD $\mathscr{F}_I$. This is a very important result because when a fault is detected in a subsystem, at the same time the faulty subsystem is identified, and further fault isolation/identification methods can be used targeting only the particular faulty subsystem.

5 Fault Diagnosis - Learning the Fault Function

After a fault is detected by the LFD $\mathscr{F}_I$ at time T_d, the fault isolation task is initiated to identify the type of fault occurring in the faulty subsystem Σ_I. In order to do this, various approaches can be used, and two of them are discussed in the sequel.

5.1 Generalized Observer Scheme

A fault isolation logic can be implemented based on a Generalized Observer Scheme (GOS, see [33, 65]). As in [31], it is assumed that each subsystem knows a *local fault set* $\mathscr{O}_I$, collecting all the $N_{\mathscr{O}_I}$ possible fault functions: $\phi_I^l(x_I, z_I, u_I)$, $l \in \{1, \ldots, N_{\mathscr{O}_I}\}$. Once a fault is detected at time T_d in the I-th subsystem, the respective LFD $\mathscr{F}_I$ activates $N_{\mathscr{O}_I}$ estimators, where each filter is sensitive to a specific fault: the generic l-th fault isolation estimator of the I-th LFD is matched to the corresponding fault function ϕ_I^l, belonging to the local fault set $\mathscr{O}_I$. Each l-th estimator provides a local state estimate $\hat{x}_I^l$ of the local state x_I affected by the l-th fault:

$$\hat{x}_I^{l(i)}(k+1) = \lambda(\hat{x}_I^{l(i)}(k) - y_I^{(i)}(k)) + f_I^{(i)}(y_I, u_I) + g_I^{(i)}(y_I, v_I^b, u_I) \\ + \hat{\eta}_I^{(i)}(y_I, v_I^b, u_I, \hat{\vartheta}_I(T_d)) + \phi_I^{l(i)}(y_I, v_I^b, u_I), \tag{71}$$

where the learning of the modeling uncertainty has been stopped at time T_d in order not to learn the fault effect. The difference between the estimate $\hat{x}_I^l$ and the resynchronized measurements y_I, after filtering, consists of the fault isolation estimation residual $r_I^{\,l} \triangleq Y_I - \hat{Y}_I^l$, where $\hat{Y}_I^l \triangleq H(z)[\hat{x}_I^l(k)]$. This residual is compared, component by component, to some properly designed isolation thresholds $\bar{r}_I^{\,l}$ so that if the j-th fault (in the fault set $\mathcal{O}_I$) has occurred, then it is guaranteed that

$$|r_I^{\,j(i)}(k)| \le \bar{r}_I^{\,j(i)}(k) \quad \forall k > T_d, i = 1, \ldots, n_I. \tag{72}$$

The isolation thresholds are defined similarly as the detection threshold in (66), modifying $\bar{\chi}_I^{(i)}(k)$ adding the following term:

$$\bar{\Delta}\phi_I^{l(i)}(k) \triangleq \max_{|\xi_I| \le \bar{\xi}_I(k)} \max_{|\varsigma_I| \le \bar{\varsigma}_I(k)} \left\{ \left| \Delta\phi_I^{l(i)}(k) \right| \right\},$$

being $\Delta\phi_I^{l(i)}(k) = \phi_I^{l(i)}(x_I, z_I, u_I) - \phi_I^{l(i)}(y_I, v_I^b, u_I)$.

If a residual crosses its corresponding threshold, then we can exclude the occurrence of the considered l-th fault. Therefore, if we are able to exclude all the faults but one, then we can say that the fault is isolated.

5.2 Learning the Fault Function

In the case that the fault functions are not known a priori, we can use a different approach based on the adaptive learning of the fault function. According to the approximation model (54) introduced in Sect. 4.4 for learning the modeling uncertainty, when a fault is detected in the I-th subsystem, then the approximation model starts to learn the combined effect of the modeling uncertainty and the fault function. Assuming that the detection time T_d is sufficiently long, so that the modeling uncertainty is learned, its estimation is given by $\hat{\eta}_I(y_I(k), v_I^b(k), u_I(k), \hat{\vartheta}_I(T_d))$. Therefore, by allowing a sufficiently long learning period T_L after the fault detection, the approximator $\hat{\eta}_I$ learns the combined effect of the modeling uncertainty and the fault function as $\hat{\eta}_I(y_I(k), v_I^b(k), u_I(k), \hat{\vartheta}_I(T_d + T_L))$ for $k > T_d + T_L$. Therefore, the estimated fault function is given by $\hat{\phi}_I(k) = \hat{\eta}_I(y_I(k), v_I^b(k), u_I(k), \hat{\vartheta}_I(T_d + T_L)) - \hat{\eta}_I(y_I(k), v_I^b(k), u_I(k), \hat{\vartheta}_I(T_d))$, $k > T_d + T_L$. Note that, the fault could be incipient and still be developing at the end of the learning period, so the designer may let the learning process to continue. In this case, the estimated fault function is given by $\hat{\phi}_I(k) = \hat{\eta}_I(y_I(k), v_I^b(k), u_I(k), \hat{\vartheta}_I(k)) - \hat{\eta}_I(y_I(k), v_I^b(k), u_I(k), \hat{\vartheta}_I(T_d))$, $k > T_d + T_L$. The estimated fault function can then be used for fault accommodation purposes in order to guarantee the stability of the faulty system. For more information regarding this approach for learning the fault function, the interested reader can refer to [53].

6 Concluding Remarks

This chapter has reviewed a distributed fault diagnosis framework specifically designed for uncertain networked nonlinear large-scale systems concerning various sources of uncertainty, namely modeling uncertainty, measurement noise, and network-related uncertainties.

In order to deal with the presence of measurement noise, a filtering scheme has been presented by integrating a general class of filters into the design of the residual and threshold signals in a way that takes advantage of the filtering noise suppression properties. Essentially, filtering dampens the effect of measurement noise in a certain frequency range allowing to set tighter detection thresholds and thus enhancing fault detectability. The main implications of the filtering scheme is rigorously investigated providing insights on the impact of the filters' poles and on the fault detection time.

The modeling uncertainties are also taken into account by means of an adaptive learning technique.

Furthermore, the chapter addressed the need for integration between the different levels composing CPS systems, by proposing a comprehensive architecture, where all parts of complex distributed systems are considered: the physical environment, the sensor level, the diagnosers layer and the communication networks. By adapting and incorporating the devised filtering scheme into the overall framework, a distributed fault diagnosis approach has been designed for distributed uncertain nonlinear large-scale systems to specifically address the issues emerging when considering networked diagnosis systems, such as the presence of delays and packet dropouts in the communication networks that degrade performance and could be a source of instability, misdetection, and false alarms. Multi-rate systems, where the measurements may not be synchronous, were also considered. Under the stated assumptions, the proposed architecture guarantees the absence of false-positive alarms.

Finally, some information was provided regarding the actions that can be taken after the detection of a fault in order to isolate the potential fault by identifying its location and magnitude, or even learning the fault function. Based on this information, actions can be taken in order to alleviate the fault effects and safeguard the system operation.

Modern, complex, interconnected systems can be prone to various sources of faults due to the increased complexity or even malicious attacks which can be considered as a "type" of fault. As a result, comprehensive fault diagnosis schemes need to be devised by considering the recent technological challenges, and this chapter has reviewed an integrated methodology which represents a step in that direction.

References

1. K. Adjallah, D. Maquin, and J. Ragot, “Nonlinear observer-based fault detection,” in *IEEE Conference on Control Applications*, no. 3, 1994, pp. 1115–1120.
2. A. Ashari, R. Nikoukhah, and S. Campbell, “Active robust fault detection in closed-loop systems: Quadratic optimization approach,” *IEEE Transactions on Automatic Control*, vol. 57, no. 10, pp. 2532–2544, 2012.
3. A. Ashari, R. Nikoukhah, and S. Campbell, “Effects of feedback on active fault detection,” *Automatica*, vol. 48, no. 5, pp. 866–872, 2012.
4. K. Baheti and H. Gill, “Cyber-physical Systems,” in *The Impact of Control Technology*, T. Samad and A. M. Annaswamy, Eds. IEEE Control Systems Society, 2011, pp. 161–166. [Online]. Available: http://ieeecss.org/general/impact-control-technology
5. R. Beard, “Failure accomodation in linear systems through self–reorganization,” *Technical Report MTV-71-1, Man Vehicle Laboratory, MIT, Cambridge, MA*, 1971.
6. F. Blanchini, D. Casagrande, G. Giordano, S. Miani, S. Olaru, and V. Reppa, “Active fault isolation: A duality-based approach via convex programming,” *SIAM Journal on Control and Optimization*, vol. 55, no. 3, pp. 1619–1640, 2017.
7. M. Blanke, M. Kinnaert, J. Lunze, and M. Staroswiecki, *Diagnosis and Fault Tolerant Control*. Berlin: Springer, 2003.
8. M. Blanke, M. Kinnaert, J. Lunze, and M. Staroswiecki, “Distributed fault diagnosis and fault-tolerant control,” in *Diagnosis and Fault-Tolerant Control*. Springer, 2016, pp. 467–518.
9. M. Blanke, M. Kinnaert, J. Lunze, and M. Staroswiecki, *Diagnosis and Fault-Tolerant Control*, 2nd ed. Springer Verlag, 2010.
10. S. Bodenburg and J. Lunze, “Plug-and-play reconfiguration of locally interconnected systems with limited model information,” *IFAC-PapersOnLine*, vol. 48, no. 22, pp. 20–27, 2015.
11. F. Boem, R. Carli, M. Farina, G. Ferrari-Trecate, and T. Parisini, “Scalable monitoring of interconnected stochastic systems,” in *2016 IEEE 55th Conference on Decision and Control (CDC)*, 2016, pp. 1285–1290.
12. F. Boem, R. Ferrari, T. Parisini, and M. Polycarpou, “Distributed fault detection for uncertain nonlinear systems: a network delay compensation strategy,” in *Proc. 2013 American Control Conference*, 2013.
13. F. Boem, S. Riverso, G. Ferrari-Trecate, and T. Parisini, “Plug-and-play fault detection and isolation for large-scale nonlinear systems with stochastic uncertainties,” *IEEE Transactions on Automatic Control (In press)*, 2018.
14. F. Boem, R. M. Ferrari, C. Keliris, T. Parisini, and M. M. Polycarpou, “A distributed networked approach for fault detection of large-scale systems,” *IEEE Transactions on Automatic Control*, vol. 62, no. 1, pp. 18–33, 2017.
15. F. Boem, R. M. Ferrari, and T. Parisini, “Distributed Fault Detection and Isolation of Continuous-Time Nonlinear Systems,” *European Journal of Control*, vol. 5-6, pp. 603–620, 2011.
16. F. Boem, R. M. Ferrari, T. Parisini, and M. M. Polycarpou, “Distributed fault diagnosis for continuous-time nonlinear systems: The input–output case,” *Annual Reviews in Control*, vol. 37, no. 1, pp. 163–169, 2013.
17. A. A. Cardenas, S. Amin, Z.-S. Lin, Y.-L. Huang, C.-Y. Huang, and S. Sastry, “Attacks against process control systems: risk assessment, detection, and response,” in *Proceedings of the 6th ACM Symposium on Information, Computer and Communications Security*, ser. ASIACCS ’11 New York, NY, USA: ACM, 2011, pp. 355–366.
18. J. Chen and R. J. Patton, *Robust Model-Based Fault Diagnosis for Dynamic Systems*. Kluwer Academic Publishers, Norwell, MA, USA, 1999.
19. P. Cheng, L. Shi, and B. Sinopoli, “Guest editorial special issue on secure control of cyber-physical systems,” *IEEE Transactions on Control of Network Systems*, vol. 4, no. 1, pp. 1–3, 2017.
20. S. Cheong and I. Manchester, “Input design for discrimination between classes of lti models,” *Automatica*, vol. 53, pp. 103–110, 2015.

21. L. Chiang, E. Russell, and R. Braatz, *Fault Detection and Diagnosis in Industrial Systems*. Springer-Verlag, London, 2001.
22. R. Clark, "Instrument fault detection," *IEEE Transactions on Aerospace and Electronic Systems*, no. 3, pp. 456–465, 1978.
23. M. Davoodi, N. Meskin, and K. Khorasani, "Simultaneous fault detection and consensus control design for a network of multi-agent systems," *Automatica*, vol. 66, pp. 185–194, 2016.
24. C. Desoer and M. Vidyasagar, *Feedback Systems: Input-Output Properties*, 1st ed. Academic Press, 1975.
25. P. Dorato, R. Tempo, and G. Muscato, "Bibliography on robust control," *Automatica*, vol. 29, no. 1, pp. 201–213, 1993.
26. F. Dorfler, F. Pasqualetti, and F. Bullo, "Distributed detection of cyber-physical attacks in power networks: A waveform relaxation approach," in *Communication, Control, and Computing (Allerton), 2011 49th Annual Allerton Conference on*, sept. 2011, pp. 1486–1491.
27. J. Farrell and M. M. Polycarpou, *Adaptive Approximation Based Control: Unifying Neural, Fuzzy, and Traditional Adaptive Approximation Approaches*. Hoboken, NJ: Wiley-Interscience, 2006.
28. J. Farrell, T. Berger, and B. Appleby, "Using learning techniques to accommodate unanticipated faults," vol. 13, pp. 40—49, 1993.
29. H. Ferdowsi, D. Raja, and S. Jagannathan, "A decentralized fault prognosis scheme for nonlinear interconnected discrete-time systems," in *American Control Conference*, 2012, pp. 5900–5905.
30. L. Ferranti, Y. Wan, and T. Keviczky, "Predictive flight control with active diagnosis and reconfiguration for actuator jamming." *IFAC-PapersOnLine*, vol. 48, no. 23, pp. 166–171, 2015.
31. R. M. Ferrari, T. Parisini, and M. M. Polycarpou, "Distributed fault detection and isolation of large-scale nonlinear systems: an adaptive approximation approach," *IEEE Transactions on Automatic Control*, vol. 57, no. 2, pp. 275–290, 2012.
32. E. Franco, R. Olfati-Saber, T. Parisini, and M. M. Polycarpou, "Distributed fault diagnosis using sensor networks and consensus-based filters," in *Decision and Control, 2006 45th IEEE Conference on*. IEEE, 2006, pp. 386–391.
33. P. Frank, "Fault diagnosis in dynamic systems using analytical and knowledge-based redundancy: A survey and some new results," *Automatica*, vol. 26, no. 3, pp. 459–474, 1990.
34. P. Frank and S. X. Ding, "Survey of robust residual generation and evaluation methods in observer-based fault detection systems," *Journal of Process Control*, no. 6, pp. 403–424, 1997.
35. Z. Gao, C. Cecati, and S. Ding, "A survey of fault diagnosis and fault-tolerant techniques –part i: Fault diagnosis with model-based and signal-based approaches," *IEEE Transactions on Industrial Electronics*, vol. 62, no. 6, pp. 3757–3767, 2015.
36. E. A. Garcia and P. Frank, "Deterministic nonlinear observer-based approaches to fault diagnosis: a survey," *Control Engineering Practice*, vol. 5, no. 5, pp. 663–670, 1997.
37. J. Gertler, "Survey of model-based failure detection and isolation in complex plants," *IEEE Control Systems Magazine*, vol. 8, no. 6, pp. 3–11, 1988.
38. J. Gertler, "Fault detection and isolation using parity relations," *Control Engineering Practice*, vol. 5, no. 5, pp. 653–661, May 1997.
39. J. Gertler, *Fault detection and diagnosis in engineering systems*, 1st ed. CRC Press, 1998.
40. V. Gupta and V. Puig, "Distributed fault diagnosis using minimal structurally over-determined sets: Application to a water distribution network," in *3rd Conference on Control and Fault-Tolerant Systems (SysTol)*. IEEE, 2016, pp. 811–818.
41. H. Hammouri, M. Kinnaert, and E. El Yaagoubi, "Observer-based approach to fault detection and isolation for nonlinear systems," *IEEE Transactions on Automatic Control*, vol. 44, no. 10, pp. 1879–1884, 1999.
42. F. Harirchi, S. Yong, E. Jacobsen, and N. Ozay, "Active model discrimination with applications to fraud detection in smart buildings," in *IFAC World Congress, Toulouse, France*, 2017.

43. I. Hwang, S. Kim, Y. Kim, and C. E. Seah, "A Survey of Fault Detection, Isolation, and Reconfiguration Methods," *IEEE Transactions on Control Systems Technology*, vol. 18, no. 3, pp. 636–653, May 2010.
44. R. Isermann, *Fault-Diagnosis Systems: An Introduction from Fault Detection to Fault Tolerance*. Springer-Verlag, 2006.
45. R. Isermann, "Process fault detection based on modeling and estimation methods - A survey," *Automatica*, vol. 20, no. 4, pp. 387–404, July 1984.
46. K. H. Johansson, G. J. Pappas, P. Tabuada, and C. J. Tomlin, "Guest editorial special issue on control of cyber-physical systems," *IEEE Transactions on Automatic Control*, vol. 59, no. 12, pp. 3120–3121, 2014.
47. H. Jones, "Failure detection in linear systems," Ph.D. Thesis, Dept. of Aero and Astro, MIT, Cambridge, MA, 1973.
48. C. Keliris, M. M. Polycarpou, and T. Parisini, "A Distributed Fault Detection Filtering Approach for a Class of Interconnected Continuous-Time Nonlinear Systems," *IEEE Transactions on Automatic Control*, vol. 58, no. 8, pp. 2032–2047, 2013.
49. C. Keliris, M. M. Polycarpou, and T. Parisini, "A Distributed Fault Detection Filtering Approach for a Class of Interconnected Input-Output Nonlinear Systems," in *Proc. of European Control Conference*, 2013, pp. 422–427.
50. C. Keliris, M. M. Polycarpou, and T. Parisini, "A Distributed Fault Diagnosis Approach Utilizing Adaptive Approximation for a Class of Interconnected Continuous-Time Nonlinear Systems," in *Proc. of Control and Decision Conference*, 2014, pp. 6536–6541.
51. C. Keliris, M. M. Polycarpou, and T. Parisini, "A Robust Nonlinear Observer-based Approach for Distributed Fault Detection of Input-Output Interconnected Systems," *Automatica*, vol. 53, no. 3, pp. 408–415, 2015.
52. C. Keliris, M. M. Polycarpou, and T. Parisini, "Distributed Fault Diagnosis for Process and Sensor Faults in a Class of Interconnected Input-Output Nonlinear Discrete-Time Systems," *International Journal of Control*, 2015.
53. C. Keliris, M. M. Polycarpou, and T. Parisini, "An Integrated Learning and Filtering Approach for Fault Diagnosis of a Class of Nonlinear Dynamical Systems," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 28, no. 4, pp. 988–1004, Apr 2017.
54. S. Klinkhieo and R. J. Patton, "A Two-Level Approach to Fault-Tolerant Control of Distributed Systems Based on the Sliding Mode," in *7th IFAC Symposium on Fault Detection, Supervision and Safety of Technical Processes, Barcelona, Spain*, 2009, pp. 1043–1048.
55. J. Lan and R. Patton, "Decentralized fault estimation and fault-tolerant control for large-scale interconnected systems: An integrated design approach," in *UKACC 11th International Conference on Control*. IEEE, 2016, pp. 1–6.
56. N. Léchevin and C. Rabbath, "Decentralized Detection of a Class of Non-Abrupt Faults With Application to Formations of Unmanned Airships," *IEEE Transactions on Control Systems Technology*, vol. 17, no. 2, pp. 484–493, 2009.
57. E. Lee, "Cyber physical systems: Design challenges," in *Object Oriented Real-Time Distributed Computing (ISORC), 2008 11th IEEE International Symposium on*, may 2008, pp. 363 –369.
58. E. A. Lee, "Cyber-physical systems - are computing foundations adequate?" in *Position Paper for NSF Workshop On Cyber-Physical Systems: Research Motivation, Techniques and Roadmap*, October 2006.
59. J. Lee, B. Bagheri, and H.-A. Kao, "A cyber-physical systems architecture for industry 4.0-based manufacturing systems," *Manufacturing Letters*, vol. 3, pp. 18–23, 2015.
60. Y. Liu and P. Bauer, "Sufficient conditions for non-negative impulse response of arbitrary-order systems," in *IEEE Asia Pacific Conference on Circuits and Systems*, 2008, pp. 1410–1413.
61. M. W. Maier, "Architecting principles for systems-of-systems," *Systems Engineering*, vol. 1, no. 4, pp. 267–284, 1998.
62. G. Marseglia and D. Raimondo, "Active fault diagnosis: A multi-parametric approach," *Automatica*, vol. 79, pp. 223–230, 2017.

63. F. Pasqualetti, F. Dörfler, and F. Bullo, “Attack detection and identification in cyber-physical systems – part i: Models and fundamental limitations,” *ArXiv e-prints*, Feb. 2012.
64. F. Pasqualetti, F. Dörfler, and F. Bullo, “Attack detection and identification in cyber-physical systems – part ii: Centralized and distributed monitor design,” *ArXiv e-prints*, Feb. 2012.
65. R. Patton, P. Frank, and D. Clark, *Fault Diagnosis in Dynamic Systems: Theory and Application*. Upper Saddle River, NJ, USA: Prentice Hall, 1989.
66. R. J. Patton, C. Kambhampati, A. Casavola, P. Zhang, S. X. Ding, and D. Sauter, “A generic strategy for fault-tolerance in control systems distributed over a network,” *European Journal of Control*, vol. 13, no. 2–3, pp. 280–296, 2007.
67. I. R. Petersen and R. Tempo, “Robust control of uncertain systems: Classical results and recent developments,” *Automatica*, vol. 50, no. 5, pp. 1315–1335, 2014.
68. M. M. Polycarpou, “On–line approximators for nonlinear system identification: a unified approach,” in *Control and Dynamic Systems: Neural Network Systems Techniques and Applications*, X. Leondes, Ed. New York: Academic, 1998, vol. 7, pp. 191–230.
69. M. M. Polycarpou and A. J. Helmicki, “Automated fault detection and accommodation: a learning systems approach,” *IEEE Transactions on Systems, Man, and Cybernetics*, vol. 25, no. 11, pp. 1447–1458, 1995.
70. M. M. Polycarpou and A. Trunov, “Learning approach to nonlinear fault diagnosis: detectability analysis,” *IEEE Transactions on Automatic Control*, vol. 45, no. 4, pp. 806–812, Apr. 2000.
71. I. Punčochář, J. Široky, and M.Šimandl, “Constrained active fault detection and control,” *IEEE Transactions on Automatic Control*, vol. 60, no. 1, pp. 253–258, 2015.
72. D. M. Raimondo, F. Boem, A. Gallo, and T. Parisini, “A decentralized fault-tolerant control scheme based on active fault diagnosis,” in *IEEE 55th Conference on Decision and Control*, 2016, pp. 2164–2169.
73. D. Raimondo, G. Marseglia, R. Braatz, and J. Scott, “Fault-tolerant model predictive control with active fault isolation,” in *Conference on Control and Fault-Tolerant Systems (SysTol)*. IEEE, 2013, pp. 444–449.
74. R. R. Rajkumar, I. Lee, L. Sha, and J. Stankovic, “Cyber-physical systems: the next computing revolution,” in *Proceedings of the 47th Design Automation Conference*, ser. DAC ’10. New York, NY, USA: ACM, 2010, pp. 731–736.
75. V. Reppa, P. Papadopoulos, M. M. Polycarpou, and C. G. Panayiotou, “A distributed architecture for hvac sensor fault detection and isolation,” *IEEE Transactions on Control Systems Technology*, vol. 23, no. 4, pp. 1323–1337, 2015.
76. V. Reppa, M. M. Polycarpou, and C. G. Panayiotou, “Decentralized isolation of multiple sensor faults in large-scale interconnected nonlinear systems,” *IEEE Transactions on Automatic Control*, vol. 60, no. 6, pp. 1582–1596, 2015.
77. V. Reppa, M. M. Polycarpou, and C. G. Panayiotou, “Distributed sensor fault diagnosis for a network of interconnected cyberphysical systems,” *IEEE Transactions on Control of Network Systems*, vol. 2, no. 1, pp. 11–23, 2015.
78. S. Riverso, F. Boem, G. Ferrari-Trecate, and T. Parisini, “Plug-and-play fault detection and control-reconfiguration for a class of nonlinear large-scale constrained systems,” *IEEE Transactions on Automatic Control*, vol. 61, no. 12, pp. 3963–3978, 2016.
79. T. Samad and T. Parisini, “Systems of Systems,” in *The Impact of Control Technology*, T. Samad and A. M. Annaswamy, Eds. IEEE Control Systems Society, 2011, pp. 175–183. [Online]. Available: http://ieeecss.org/general/impact-control-technology
80. T. Samad and T. Parisini, “Systems of systems,” *The Impact of Control Technology (T.Samad and A.Annaswamy, eds.)*, 2011. [Online]. Available: www.ieeecss.org
81. H. Sandberg, S. Amin, and K. H. Johansson, “Cyberphysical security in networked control systems: An introduction to the issue,” *IEEE Control Systems*, vol. 35, no. 1, pp. 20–23, Feb 2015.
82. J. Scott, R. Findeisen, R. Braatz, and D. Raimondo, “Input design for guaranteed fault diagnosis using zonotopes,” *Automatica*, vol. 50, no. 6, pp. 1580–1589, 2014.
83. L. Sha, S. Gopalakrishnan, X. Liu, and Q. Wang, “Cyber-physical systems: A new frontier,” in *Machine Learning in Cyber Trust*. Springer US, 2009, pp. 3–13.

84. I. Shames, A. M. Teixeira, H. Sandberg, and K. H. Johansson, "Distributed fault detection for interconnected second-order systems," *Automatica*, vol. 47, no. 12, pp. 2757–2764, 2011.
85. F. Shi and R. Patton, "Fault estimation and active fault tolerant control for linear parameter varying descriptor systems," *International Journal of Robust and Nonlinear Control*, vol. 25, no. 5, pp. 689–706, 2015.
86. M. Simandl and I. Puncochar, "Active fault detection and control: Unified formulation and optimal design," *Automatica*, vol. 45, no. 9, pp. 2052–2059, 2009.
87. J. Škach, I. Punčochář, and F. L. Lewis, "Optimal active fault diagnosis by temporal-difference learning," in *Decision and Control (CDC), 2016 IEEE 55th Conference on*. IEEE, 2016, pp. 2146–2151.
88. R. S. Smith, "Covert misappropriation of networked control systems: Presenting a feedback structure," *IEEE Control Systems*, vol. 35, no. 1, pp. 82–92, 2015.
89. S. Stankovic, N. Ilic, Z. Djurovic, M. Stankovic, and K. Johansson, "Consensus based overlapping decentralized fault detection and isolation," in *Conference on Control and Fault-Tolerant Systems (SysTol'10)*, 2010, pp. 570–575.
90. M. Staroswiecki and A. M. Amani, "Fault-tolerant control of distributed systems by information pattern reconfiguration," *International Journal of Adaptive Control and Signal Processing*, 2014.
91. M. Staroswiecki and A. Amani, "Fault-tolerant control of distributed systems by information pattern reconfiguration," *International Journal of Adaptive Control and Signal Processing*, vol. 29, no. 6, pp. 671–684, 2015.
92. S. Tabatabaeipour, "Active fault detection and isolation of discrete-time linear time-varying systems: a set-membership approach," *International Journal of Systems Science*, vol. 46, no. 11, pp. 1917–1933, 2015.
93. P. Tabuada, "Cyber physical systems: Position paper," in *NSF Workshop on Cyber-Physical Systems*, 2006.
94. P. L. Tang and C. de Silva, "Compensation for transmission delays in an ethernet-based control network using variable-horizon predictive control," *IEEE Transactions on Control Systems Technology*, vol. 14, no. 4, pp. 707 – 718, 2006.
95. A. Teixeira, H. Sandberg, and K. Johansson, "Networked control systems under cyber attacks with applications to power networks," in *American Control Conference (ACC), 2010*, 30 2010-july 2 2010, pp. 3690–3696.
96. A. Teixeira, I. Shames, H. Sandberg, and K. H. Johansson, "Distributed fault detection and isolation resilient to network model uncertainties," *IEEE Transactions on Cybernetics*, vol. 44, no. 11, pp. 2024–2037, Nov 2014.
97. R. Tempo, G. Calafiore, and F. Dabbene, *Randomized algorithms for analysis and control of uncertain systems: with applications*. Springer Science & Business Media, 2012.
98. A. Vemuri and M. M. Polycarpou, "On-line approximation methods for robust fault detection," *Proc. 13th IFAC World Congress*, vol. K, pp. 319–324, 1996.
99. V. Venkatasubramanian, R. Rengaswamy, S. Kavuri, and K. Yin, "A review of process fault detection and diagnosis:: Part III: Process history based methods," *Computers & Chemical Engineering*, vol. 27, no. 3, pp. 327–346, Mar. 2003.
100. V. Venkatasubramanian, R. Rengaswamy, K. Yin, and S. Kavuri, "A review of process fault detection and diagnosis Part I: Quantitative model-based methods," *Computers & Chemical Engineering*, vol. 27, no. 3, pp. 293–311, Mar. 2003.
101. V. Venkatasubramanian, R. Rengaswamy, and S. Kavuri, "A review of process fault detection and diagnosis:: Part II: Qualitative models and search strategies," *Computers & Chemical Engineering*, vol. 27, no. 3, pp. 313–326, 2003.
102. L. Wei, W. Gui, Y. Xie, and S. X. Ding, "Decentralized Fault Detection System Design for Large-Scale Interconnected Systems," in *7th IFAC symposium on Fault Detection, Supervision and Safety of Technical Processes, Barcelona, Spain*, 2009, pp. 816–821.
103. W. Wolf, "Cyber-physical systems," *Computer*, vol. 42, no. 3, pp. 88–89, march 2009.
104. F. Xu, S. Olaru, V. Puig, C. Ocampo-Martinez, and S.-I. Niculescu, "Sensor-fault tolerance using robust mpc with set-based state estimation and active fault isolation," *International Journal of Robust and Nonlinear Control*, 2016.

105. F. Xu, V. Puig, C. Ocampo-Martinez, and X. Wang, “Set-valued observer-based active fault-tolerant model predictive control,” *Optimal Control Applications and Methods*, 2016.
106. S. Zanero, “Cyber-physical systems,” *Computer*, vol. 50, no. 4, pp. 14–16, 2017.
107. X. Zhang, M. M. Polycarpou, and T. Parisini, “A robust detection and isolation scheme for abrupt and incipient faults in nonlinear systems,” vol. 47, no. 4, pp. 576–593, 2002.
108. X. Zhang, M. M. Polycarpou, and T. Parisini, “Decentralized fault detection for a class of large-scale nonlinear uncertain systems,” in *48h IEEE Conference on Decision and Control and 28th Chinese Control Conference*, 2009, pp. 6988–6993.
109. K. Zhou and J. C. Doyle, *Essentials of robust control*. Prentice hall, Upper Saddle River, NJ, 1998, vol. 104.
110. Y. Zhou, F. Boem, C. Fischione, and T. Parisini, “Distributed fault detection with sensor networks using pareto-optimal dynamic estimation method,” in *2016 European Control Conference (ECC)*, 2016, pp. 728–733.

Networked Quantum Systems

Ian R. Petersen

Abstract This chapter presents a survey of results in the area of networked quantum systems. The chapter includes background material on quantum linear system models and finite level quantum system models. Different forms of these models are discussed and the issue of physical realizability is addressed. Also, the Kalman decomposition for linear quantum systems is described. The use of optical linear quantum networks in the physical realization of quantum systems is discussed for both the passive and non-passive case.

1 Introduction

Roberto Tempo was a friend and colleague of mine for almost all of my career. During this time, we only wrote one paper together, which was the Automatica survey paper [57]. This paper surveyed the area of Robust Control, which was our area of great common interest. In this chapter, I have tried to follow the style of that survey paper but covering my other main area of interest, which is quantum networks. The study of networks in control systems was one of Roberto's important areas of research, although he never ventured into the area of quantum networks. This chapter presents a survey of recent results on the theory of quantum systems and quantum networks from a systems and control point of view. Ideas from systems and control theory, and in particular nonlinear filtering theory, have been applied to the quantum systems area since the early work of Belevkin; e.g., see [2–4]. Also, nonlinear and optimal control ideas have been applied to control problems for quantum systems since the early work of Tarn and Rabitz; e.g., see [25, 68, 70]. In addition, the theory of quantum stochastic differential equations (QSDEs) was developed by Hudson and Parthasarathy, which put stochastic Heisenberg picture models of quantum systems on a firm mathematical footing; e.g., see [26, 52]. This approach to the modelling of quantum systems often

I. R. Petersen (✉)
Research School of Engineering, Australian National University,
2601 Canberra, ACT, Australia
e-mail: i.r.petersen@gmail.com

© Springer Nature Switzerland AG 2018

T. Başar (ed.), *Uncertainty in Complex Networked Systems*,
Systems & Control: Foundations & Applications,
https://doi.org/10.1007/978-3-030-04630-9_17

leads to control theory which closely resembles the corresponding control theory for classical systems and is used for many of the results which are presented in this survey. Also, these developments led to an ongoing interest in the connections between control theory and quantum physics, including some experimental work such as the quantum optics experiments in the group of Mabuchi; e.g., see [1, 5, 9, 10, 23, 29, 30, 41–43, 58, 63].

Building on the work of Belevkin, the area of quantum filtering was one of the first areas of quantum systems theory in which classical control theory ideas made a major impact; e.g., see [6, 7]. This work together with the QSDE modelling approach of [26, 52] led naturally to the development of a theory of quantum networks; e.g., see [14–17, 19, 28]. Also, this theory led to quantum versions of some classical control problems such as the H^∞ control problem and the linear quadratic Gaussian (LQG) control problem; e.g., see [11, 27, 47].

An important idea to emerge from the use of QSDE models of quantum systems in the solution of problems of quantum H^∞ and LQG control was the notion of physical realizability presented in [27]. This idea arose in the consideration of coherent quantum control problems in which both the plant and controller are quantum systems. This is because methods which lead to the synthesis of a controller described in terms of a QSDE model need to ensure that this model can actually be implemented as a physical quantum system. The notion of physical realizability was further explored in the papers [12, 13, 31, 39, 59, 66]. In addition, a number of papers have considered the problem of constructing a quantum system using a network of optical components to implement any physically realizable quantum; e.g., see [20, 45, 46, 49, 50, 53, 67]. Some of these results are surveyed in this chapter. The chapter will also survey some results on the structure of linear quantum systems such as described in the papers [18, 22, 55, 69] and the book [48]. Additional results about the structure and properties of linear quantum networks which may be of interest to the readers, but which are not covered in this survey, can be found in the papers [21, 32, 33, 35–38, 56, 61, 62, 64].

The remainder of this chapter proceeds as follows. In Sect. 2, various classes of quantum system models are introduced. These models include general linear quantum systems described in terms of QSDEs, passive linear quantum systems, which are a special class of linear quantum systems which do not contain any energy generating active elements, and finite level quantum systems described in terms of QSDEs. This section also considers the property of physical realizability for these various classes of models as well as describing the Kalman decomposition for linear quantum systems. The section also presents Schrödinger picture models for finite level quantum systems in terms of master equations. Section 3 considers the problem of constructing a network of quantum optical components to physically implement a given linear quantum system. In Sect. 4, some conclusions are given.

2 Quantum System Models

2.1 Linear Quantum Systems

The linear quantum systems we will consider involve a collection of quantum harmonic oscillators. In the Heisenberg picture of quantum mechanics, the dynamics of these oscillators can be described in terms of the time evolution of operator variables such as the vector of *annihilation operators*

$$a = \begin{bmatrix} a_1 \\ a_2 \\ \vdots \\ a_n \end{bmatrix}. \tag{1}$$

Each quantum harmonic oscillator has a corresponding annihilation operator a_i, which is an operator on an underlying infinite dimensional Hilbert space $\mathscr{H}$. Also, each quantum harmonic oscillator is associated with a corresponding *creation operator* a_i^*, which is the adjoint of the annihilation operator a_i; e.g., see [16, 44, 48, 52, 55].

The operators a_i and a_i^* do not commute but rather satisfy the following canonical commutation relations:

$$[a_i, a_j^*] = a_i a_j^* - a_j^* a_i = \delta_{ij}. \tag{2}$$

Here δ_{ij} denotes the Kronecker delta multiplied by the identity operator. Also,

$$[a_i, a_j] = 0, \; [a_i^*, a_j^*] = 0. \tag{3}$$

In addition, we use the notation

$$a^\# = \begin{bmatrix} a_1^* \\ a_2^* \\ \vdots \\ a_n^* \end{bmatrix},$$

$a^T = \begin{bmatrix} a_1 & a_2 & \dots & a_n \end{bmatrix}$, and $a^\dagger = \left(a^\#\right)^T$. Also, we often use the ’doubled-up’ notation

$$\breve{a} = \begin{bmatrix} a \\ a^\# \end{bmatrix}.$$

Using the above notation, the commutation relations can be written as

$$\left[\begin{bmatrix} a \\ a^{\#} \end{bmatrix}, \begin{bmatrix} a \\ a^{\#} \end{bmatrix}^{\dagger}\right] = \begin{bmatrix} a \\ a^{\#} \end{bmatrix}\begin{bmatrix} a \\ a^{\#} \end{bmatrix}^{\dagger} - \left(\begin{bmatrix} a \\ a^{\#} \end{bmatrix}^{\#}\begin{bmatrix} a \\ a^{\#} \end{bmatrix}^{T}\right)^{T}$$
$$= \begin{bmatrix} I & 0 \\ 0 & -I \end{bmatrix}. \tag{4}$$

In an open linear quantum system, the quantum harmonic oscillators being considered are coupled to a collection of quantum fields representing the environment of the quantum system or the system's interaction with a driving laser. These fields are modelled by corresponding field annihilation operators $\mathscr{A}_1(t), \mathscr{A}_2(t), \ldots, \mathscr{A}_m(t)$, which are on corresponding Fock spaces $\mathscr{F}_i$. Also, the adjoints of these field annihilation operators are corresponding field creation operators $\mathscr{A}_1^*(t), \mathscr{A}_2^*(t), \ldots, \mathscr{A}_m^*(t)$. The field operators have corresponding forward differentials

$$d\mathscr{A}_j(t) = \mathscr{A}_j(t+dt) - \mathscr{A}_j(t)$$

and

$$d\mathscr{A}_j^*(t) = \mathscr{A}_j^*(t+dt) - \mathscr{A}_j^*(t),$$

which satisfy the Ito relations

$$\begin{aligned} d\mathscr{A}_j(t)d\mathscr{A}_k(t)^* &= \delta_{jk}dt; \\ d\mathscr{A}_j^*(t)d\mathscr{A}_k(t) &= 0; \\ d\mathscr{A}_j(t)d\mathscr{A}_k(t) &= 0; \\ d\mathscr{A}_j^*(t)d\mathscr{A}_k^*(t) &= 0; \end{aligned}$$

e.g., see [7, 26, 48, 51, 52]. The field annihilation operators define corresponding quantum stochastic processes and we often use the vector notation

$$\mathscr{A}(t) = \begin{bmatrix} \mathscr{A}_1(t) \\ \mathscr{A}_2(t) \\ \vdots \\ \mathscr{A}_m(t) \end{bmatrix}.$$

Also, we use the following vector notation for the corresponding field creation operators

$$\mathscr{A}^{\#}(t) = \begin{bmatrix} \mathscr{A}_1^*(t) \\ \mathscr{A}_2^*(t) \\ \vdots \\ \mathscr{A}_m^*(t) \end{bmatrix}.$$

The dynamics of a linear quantum system are determined by the Hamiltonian, which is a self-adjoint operator on the underlying Hilbert space $\mathscr{H}$ and the coupling operators, which are also operators on $\mathscr{H}$ but not necessarily self-adjoint. For a linear quantum system, the Hamiltonian operator is of the form

$$H = \frac{1}{2}\left[a^\dagger \; a^T\right] M \begin{bmatrix} a \\ a^\# \end{bmatrix} \tag{5}$$

where $M \in \mathbb{C}^{2n \times 2n}$ is a Hermitian matrix of the form

$$M = \begin{bmatrix} M_1 & M_2 \\ M_2^\# & M_1^\# \end{bmatrix} \tag{6}$$

such that $M_1 = M_1^\dagger$, $M_2 = M_2^T$. Here, we use the notation $M^\dagger$ to denote the complex conjugate transpose of the complex matrix M. Also, we use the notation M^T to denote the transpose of the complex matrix M, and we use the notation $M^\#$ to denote the complex conjugate of the complex matrix M.

For an open linear quantum system, we also specify a vector of coupling operators of the form

$$L = \left[N_1 \quad N_2\right] \begin{bmatrix} a \\ a^\# \end{bmatrix} \tag{7}$$

where $N_1 \in \mathbb{C}^{m \times n}$ and $N_2 \in \mathbb{C}^{m \times n}$. Also, we write

$$\begin{bmatrix} L \\ L^\# \end{bmatrix} = N \begin{bmatrix} a \\ a^\# \end{bmatrix} = \begin{bmatrix} N_1 & N_2 \\ N_2^\# & N_1^\# \end{bmatrix} \begin{bmatrix} a \\ a^\# \end{bmatrix}. \tag{8}$$

In addition, the input–output dynamics of an open linear quantum system are dependent on a unitary scattering matrix $S \in \mathbb{C}^{n \times n}$. These quantities define the (S, L, H) parameters of an open linear quantum system; e.g., see [16, 28].

In the Heisenberg picture of quantum mechanics, the dynamics of a linear quantum system can be described in terms of a set of QSDEs. For an open linear quantum system with (S, L, H) parameters defined as above, the corresponding QSDEs are given as follows:

$$\begin{aligned} \begin{bmatrix} da(t) \\ da(t)^\# \end{bmatrix} &= F \begin{bmatrix} a(t) \\ a(t)^\# \end{bmatrix} dt + G \begin{bmatrix} d\mathscr{A}(t) \\ d\mathscr{A}(t)^\# \end{bmatrix}; \\ \begin{bmatrix} d\mathscr{A}^{out}(t) \\ d\mathscr{A}^{out}(t)^\# \end{bmatrix} &= H \begin{bmatrix} a(t) \\ a(t)^\# \end{bmatrix} dt + K \begin{bmatrix} d\mathscr{A}(t) \\ d\mathscr{A}(t)^\# \end{bmatrix} \end{aligned} \tag{9}$$

where

$$\begin{aligned} F &= \begin{bmatrix} F_1 & F_2 \\ F_2^\# & F_1^\# \end{bmatrix}; \quad G = \begin{bmatrix} G_1 & G_2 \\ G_2^\# & G_1^\# \end{bmatrix}; \\ H &= \begin{bmatrix} H_1 & H_2 \\ H_2^\# & H_1^\# \end{bmatrix}; \quad K = \begin{bmatrix} K_1 & K_2 \\ K_2^\# & K_1^\# \end{bmatrix}. \end{aligned} \tag{10}$$

and

$$\begin{aligned}
F &= -iJM - \frac{1}{2}JN^\dagger JN;\\
G &= -JN^\dagger \begin{bmatrix} S & 0 \\ 0 & -S^\# \end{bmatrix};\\
H &= N;\\
K &= \begin{bmatrix} S & 0 \\ 0 & S^\# \end{bmatrix};
\end{aligned} \tag{11}$$

e.g, see [16, 48, 55]. Here,

$$J = \begin{bmatrix} I & 0 \\ 0 & -I \end{bmatrix}. \tag{12}$$

Using the fact that S is unitary, it follows that

$$KJK^\dagger = J. \tag{13}$$

Also,

$$KK^\dagger = I. \tag{14}$$

2.2 Passive Linear Quantum Systems

An important class of linear quantum systems is those in which the dynamics can be described purely in terms of QSDEs involving only annihilation operators. This class of linear quantum systems corresponds to those systems containing only passive elements such as cavities, beamsplitters and phase shifters; e.g., see [39, 40, 42, 48, 53, 55]. This class of linear quantum systems corresponds to the case in which $M_2 = 0$ and $N_2 = 0$. This leads to QSDEs of the form

$$\begin{aligned}
da(t) &= Fa(t)dt + Gd\mathscr{A}(t);\\
d\mathscr{A}^{out}(t) &= Ha(t)dt + Kd\mathscr{A}(t)
\end{aligned} \tag{15}$$

where

$$\begin{aligned}
F &= -iM_1 - \frac{1}{2}N_1^\dagger N_1;\\
G &= -N_1^\dagger S;\\
H &= N_1;\\
K &= S.
\end{aligned} \tag{16}$$

In this case, the commutation relations (4) are replaced by the relations

$$[a, a^\dagger] = I.$$

2.3 Position and Momentum Operator Linear Quantum Systems

In the QSDE description of a general linear quantum system (9), the matrices F, G, H, K are in general complex matrices of the form (10). However, it is possible to introduce a change of variables so that the corresponding QSDE description involves only real matrices; e.g., see [27, 47, 48, 51, 55, 59]. In this case, the system variables as well as the field variables are given in terms of position and momentum operators instead of annihilation and creation operators. That is, we introduce the following change of variables:

$$\begin{aligned} x &= \begin{bmatrix} q \\ p \end{bmatrix} = \Phi \begin{bmatrix} a \\ a^{\#} \end{bmatrix}; \\ u &= \begin{bmatrix} \mathscr{Q}(t) \\ \mathscr{P}(t) \end{bmatrix} = \Phi \begin{bmatrix} \mathscr{A}(t) \\ \mathscr{A}(t)^{\#} \end{bmatrix}; \\ y &= \begin{bmatrix} \mathscr{Q}^{out}(t) \\ \mathscr{P}^{out}(t) \end{bmatrix} = \Phi \begin{bmatrix} \mathscr{A}^{out}(t) \\ \mathscr{A}^{out}(t)^{\#} \end{bmatrix} \end{aligned} \tag{17}$$

where the matrices Φ have the form

$$\Phi = \begin{bmatrix} I & I \\ -iI & iI \end{bmatrix}. \tag{18}$$

In this description, q is a vector of the self-adjoint position operators and p is a vector of self-adjoint momentum operators. Also, $\mathscr{Q}(t)$ is a vector of the self-adjoint field position operators and $\mathscr{P}(t)$ is a vector of self-adjoint field momentum operators.

When these transformations are applied to the QSDEs (9), QSDEs of the following form are obtained:

$$\begin{aligned} \begin{bmatrix} dq(t) \\ dp(t) \end{bmatrix} &= A \begin{bmatrix} p(t) \\ q(t) \end{bmatrix} dt + B \begin{bmatrix} d\mathscr{Q}(t) \\ d\mathscr{P}(t) \end{bmatrix}; \\ \begin{bmatrix} d\mathscr{Q}^{out}(t) \\ d\mathscr{P}^{out}(t) \end{bmatrix} &= C \begin{bmatrix} q(t) \\ p(t) \end{bmatrix} dt + D \begin{bmatrix} d\mathscr{P}(t) \\ d\mathscr{Q}(t) \end{bmatrix}, \end{aligned} \tag{19}$$

where

$$\begin{aligned} A &= \Phi F \Phi^{-1}; \\ B &= \Phi G \Phi^{-1}; \\ C &= \Phi H \Phi^{-1}; \\ D &= \Phi K \Phi^{-1} \end{aligned} \tag{20}$$

These matrices are all real. Also, it follows from (3) that

$$\left[\begin{bmatrix} q \\ p \end{bmatrix}, \begin{bmatrix} q \\ p \end{bmatrix}^{\dagger}\right] = \Xi \tag{21}$$

where

$$\Xi = \Phi J \Phi^{\dagger}, \tag{22}$$

which is a Hermitian matrix. In addition, we can write

$$\Xi = \Phi J \Phi^{\dagger} = 2i \begin{bmatrix} 0 & I \\ -I & 0 \end{bmatrix} = 2i\mathbb{J} \tag{23}$$

where

$$\mathbb{J} = \begin{bmatrix} 0 & I \\ -I & 0 \end{bmatrix}. \tag{24}$$

In this position and momentum operator description of a linear quantum system, it is often more convenient to write the Hamiltonian and coupling operators in terms of the vectors q and p rather than the vectors a and $a^{\#}$. Hence, applying the transformations (17) to Eq. (5), we obtain

$$H = \frac{1}{2}\begin{bmatrix} q^T & p^T \end{bmatrix} R \begin{bmatrix} q \\ p \end{bmatrix}, \quad \begin{bmatrix} L \\ L^{\#} \end{bmatrix} = V \begin{bmatrix} q \\ p \end{bmatrix}$$

where

$$R = \left(\Phi^{\dagger}\right)^{-1} M \Phi^{-1}, \quad V = N\Phi^{-1}. \tag{25}$$

Note that the matrix R is real and symmetric. However, the matrix V may be complex.

Also applying the transformations (17) to the Eq. (7), we obtain

$$\begin{bmatrix} L + L^{\#} \\ \frac{L-L^{\#}}{i} \end{bmatrix} = \Phi \begin{bmatrix} L \\ L^{\#} \end{bmatrix} = \Phi V \begin{bmatrix} q \\ p \end{bmatrix} = W \begin{bmatrix} q \\ p \end{bmatrix}$$

where

$$W = \Phi V = \Phi N \Phi^{-1},$$

which is a real matrix. This equation can be written as

$$\begin{bmatrix} \Re(L) \\ \Im(L) \end{bmatrix} = \frac{1}{2} W \begin{bmatrix} q \\ p \end{bmatrix}.$$

In terms of the matrices R, S and W, the matrices A, B, C, D can now be written

$$A = 2\mathbb{J}R + \frac{1}{2}\mathbb{J}W^T\mathbb{J}W;$$

$$\begin{aligned} B &= \mathbb{J}W^T\mathbb{J}D; \\ C &= W; \\ D &= \frac{1}{2}\begin{bmatrix} S+S^\# & i\left(S-S^\#\right) \\ i\left(S-S^\#\right) & S+S^\# \end{bmatrix}. \end{aligned} \tag{26}$$

It follows from this that

$$DD^T = I \tag{27}$$

and

$$D\mathbb{J}D^T = \mathbb{J}. \tag{28}$$

2.4 Physical Realizability of Linear Quantum Systems

We now consider conditions under which a set of annihilation-creation QSDEs of the form (9) in fact corresponds to a linear quantum system with an (S, L, H) description of the form (5), (7). In this case, the QSDEs (9) are said to be physically realizable; e.g., see [24, 27, 39, 40, 47, 48, 53–55, 59]. This notion is formalized in the following definition.

Definition 1 Annihilation-creation QSDEs of the form (9), (10) are said to be *physically realizable* if there exist complex matrices $M = M^\dagger$, N, S such that $S^\dagger S = I$, and (6), (8) and (11) are satisfied.

The following theorem gives necessary and sufficient conditions for physical realizability in this case.

Theorem 1 ([55, 59]) *The annihilation-creation QSDEs (9) are physically realizable if and only if the following equations are satisfied:*

$$\begin{aligned} &FJ + JF^\dagger + GJG^\dagger = 0; \\ &G = -JH^\dagger JK; \\ &KJK^\dagger = J; \\ &KK^\dagger = I. \end{aligned} \tag{29}$$

Corresponding to the annihilation-creation QSDEs (9) is the transfer function matrix

$$\Gamma(s) = H\left(sI - F\right)^{-1}G + K. \tag{30}$$

Definition 2 A transfer function matrix $\Gamma(s)$ is said to be *annihilation-creation physically realizable* if it has a minimal state space realization (30) such that the corresponding annihilation-creation QSDEs (9) are physically realizable.

The following theorem gives a necessary and sufficient condition for a transfer function matrix to be annihilation-creation physically realizable.

Theorem 2 (See [31, 55, 59]) *The transfer function matrix (30) is annihilation-creation physically realizable if and only if the following conditions are satisfied:*

(i)

$$\Gamma(s) J \Gamma^{\sim}(s) = J$$

for all $s \in \mathbb{C}_+$.

(ii)

$$\Gamma(\infty)\Gamma(\infty)^{\dagger} = I; \tag{31}$$

Here, $\Gamma^{\sim}(s) \overset{\Delta}{=} \Gamma(-s^*)^{\dagger}$ *and* $\mathbb{C}_+$ *denotes the set* $\{s \in \mathbb{C} : \Re[s] \geq 0\}$.

We now consider the physical realizability of passive linear quantum systems of the form (15).

Definition 3 Passive QSDEs of the form (15) are said to be *physically realizable* if there exist complex matrices $M_1 = M_1^{\dagger}$, N_1, and S such that $S^{\dagger} S = I$ and (16) is satisfied.

The following theorem gives necessary and sufficient conditions for physical realizability in the passive case.

Theorem 3 (See [39, 54, 55]) *The passive QSDEs (15) are physically realizable if and only if the following equations are satisfied:*

$$\begin{aligned} &F + F^{\dagger} + GG^{\dagger} = 0; \\ &G = -H^{\dagger} K; \\ &K^{\dagger} K = I. \end{aligned} \tag{32}$$

Corresponding to the passive QSDEs (15) is the transfer function matrix

$$\Gamma(s) = H\,(sI - F)^{-1}\,G + K. \tag{33}$$

Definition 4 A transfer function matrix $\Gamma(s)$ is said to be *passive physically realizable* if it has a minimal state space realization (33) such that the corresponding passive QSDEs (15) are physically realizable.

The following theorem gives a necessary and sufficient condition for a transfer function matrix to be passive physically realizable.

Theorem 4 (See [39, 54, 55]) *A transfer function matrix $\Gamma(s)$ is passive physically realizable if and only if the following conditions are satisfied:*

(i) *All of the poles of $\Gamma(s)$ have strictly negative real parts;*

(ii)

$$\Gamma(i\omega)^{\dagger}\Gamma(i\omega) = I$$

for all $\omega \in \mathbb{R}$.

We now consider the physical realizability of position-momentum linear quantum systems of the form (19).

Definition 5 Position-momentum QSDEs of the form (19) are said to be *position-momentum physically realizable* if there exist real matrices R, W and D such that $R = R^T$, $D\mathbb{J}D^T = \mathbb{J}$, $DD^T = I$, and (26) is satisfied.

The following theorem gives necessary and sufficient conditions for physical realizability in the position-momentum case.

Theorem 5 (See [27, 47, 55]) *The position-momentum QSDEs (19) are physically realizable if and only if the following equations are satisfied:*

$$\begin{aligned} &A\mathbb{J} + \mathbb{J}A^T + B\mathbb{J}B^T = 0;\\ &B = \mathbb{J}C^T\mathbb{J}D;\\ &D\mathbb{J}D^T = \mathbb{J};\\ &DD^T = I. \end{aligned} \tag{34}$$

Corresponding to the position-momentum QSDEs (19) is the transfer function matrix

$$\Upsilon(s) = C(sI - A)^{-1}B + D. \tag{35}$$

Definition 6 A transfer function matrix $\Upsilon(s)$ is said to be *position-momentum physically realizable* if it has a minimal state space realization (35) such that the corresponding position-momentum QSDEs (19) are physically realizable.

The following theorem gives a necessary and sufficient condition for a transfer function matrix to be position-momentum physically realizable.

Theorem 6 (See [31, 55]) *A transfer function matrix $\Upsilon(s)$ is position-momentum physically realizable if and only if the following conditions are satisfied:*

(i) *All of the poles of $\Upsilon(s)$ have strictly negative real parts;*

(ii)

$$\Upsilon(s)\mathbb{J}\Upsilon^{\sim}(s) = \mathbb{J}$$

for all $s \in \mathbb{C}_+$;

(iii)

$$\Upsilon(\infty)\Upsilon(\infty)^{\dagger} = I.$$

2.5 The Kalman Decomposition for Linear Quantum Systems

In this subsection, we consider a Kalman decomposition for quantum linear systems; see [18, 22, 69]. We first consider physically realizable annihilation-creation quantum linear systems of the form (9), (11). The Kalman decomposition for the system (9), (11) involves decomposing the system into three subsystems via a state space transformation of the form

$$\begin{bmatrix} \breve{\boldsymbol{a}}_h \\ \breve{\boldsymbol{a}}_{co} \\ \breve{\boldsymbol{a}}_{\bar{c}\bar{o}} \end{bmatrix} = T^\dagger \breve{\boldsymbol{a}}$$

such that the commutation relations (4) are satisfied by each subsystem in the transformed system. That is, each subsystem in the transformed system is required to be a physically realizable quantum system. In order to achieve this, we require that the transformation matrix T satisfies the following blockwise Bogoliubov condition:

$$T^\dagger J_n T = \begin{bmatrix} J_{n_3} & 0 & 0 \\ 0 & J_{n_1} & 0 \\ 0 & 0 & J_{n_2} \end{bmatrix}.$$

Here the notation J_k refers to a $(2k) \times (2k)$ matrix of the form (12).

Theorem 7 (See [69]) *There exists a unitary and blockwise Bogoliubov coordinate transformation*

$$\begin{bmatrix} \breve{\boldsymbol{a}}_h \\ \breve{\boldsymbol{a}}_{co} \\ \breve{\boldsymbol{a}}_{\bar{c}\bar{o}} \end{bmatrix} = T^\dagger \breve{\boldsymbol{a}} \tag{36}$$

which transforms the physically realizable linear quantum system (9), (11) into the form

$$\begin{bmatrix} \dot{\breve{\boldsymbol{a}}}_h(t) \\ \dot{\breve{\boldsymbol{a}}}_{co}(t) \\ \dot{\breve{\boldsymbol{a}}}_{\bar{c}\bar{o}}(t) \end{bmatrix} = \bar{\mathscr{A}} \begin{bmatrix} \breve{\boldsymbol{a}}_h(t) \\ \breve{\boldsymbol{a}}_{co}(t) \\ \breve{\boldsymbol{a}}_{\bar{c}\bar{o}}(t) \end{bmatrix} + \bar{\mathscr{B}} \breve{\boldsymbol{b}}(t); \tag{37}$$

$$\breve{\boldsymbol{b}}_{\mathrm{out}}(t) = \bar{\mathscr{C}} \begin{bmatrix} \breve{\boldsymbol{a}}_h(t) \\ \breve{\boldsymbol{a}}_{co}(t) \\ \breve{\boldsymbol{a}}_{\bar{c}\bar{o}}(t) \end{bmatrix} + \breve{\boldsymbol{b}}(t), \tag{38}$$

where

$$\bar{\mathscr{A}} \triangleq T^\dagger \mathscr{A} T = \begin{bmatrix} \mathscr{A}_h & \mathscr{A}_{12} & \mathscr{A}_{13} \\ \mathscr{A}_{21} & \mathscr{A}_{co} & 0 \\ \mathscr{A}_{31} & 0 & \mathscr{A}_{\bar{c}\bar{o}} \end{bmatrix};$$

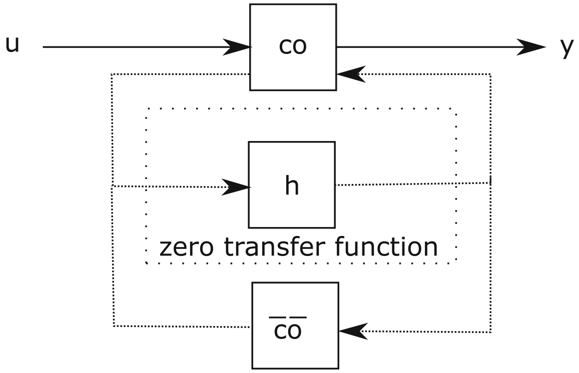


Fig. 1 Block diagram corresponding to annihilation-creation Kalman decomposition

$$\bar{\mathscr{B}} \triangleq T^\dagger \mathscr{B} = \begin{bmatrix} \mathscr{B}_h \\ \mathscr{B}_{co} \\ 0 \end{bmatrix};$$
$$\bar{\mathscr{C}} \triangleq \mathscr{C} T = \begin{bmatrix} \mathscr{C}_h & \mathscr{C}_{co} & 0 \end{bmatrix} \tag{39}$$

and

$$\begin{bmatrix} \mathscr{A}_{21} \\ \mathscr{A}_{31} \end{bmatrix} (sI - \mathscr{A}_h)^{-1} \begin{bmatrix} \mathscr{A}_{12} & \mathscr{A}_{13} \end{bmatrix} = 0. \tag{40}$$

Here the pair $(\mathscr{A}_{co}, \mathscr{B}_{co})$ is controllable and the pair $(\mathscr{A}_{co}, \mathscr{C}_{co})$ is observable.

A block diagram for the system (37)–(39) is given in Fig. 1.

We now consider the Kalman decomposition for physically realizable passive linear quantum systems of the form (15), (16). The Kalman decomposition for the system (15), (16) involves decomposing the system into two subsystems via a state space transformation of the form

$$\begin{bmatrix} \boldsymbol{a}_{\bar{c}\bar{o}} \\ \boldsymbol{a}_{co} \end{bmatrix} = T^\dagger \boldsymbol{a}$$

such that the commutation relations (4) are satisfied by each subsystem in the transformed system. That is, each subsystem in the transformed system is required to be a physically realizable quantum system. In order to achieve this, we require that the transformation matrix T is unitary.

Theorem 8 (See [69]) *There exists a unitary coordinate transformation*

$$\begin{bmatrix} \boldsymbol{a}_{\bar{c}\bar{o}} \\ \boldsymbol{a}_{co} \end{bmatrix} = T^\dagger \boldsymbol{a}$$

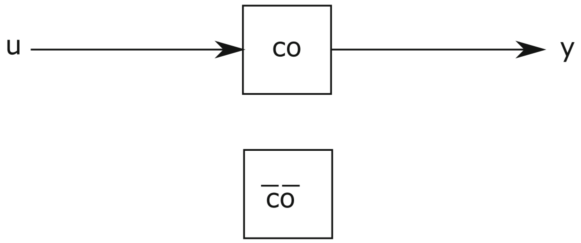


Fig. 2 Block diagram corresponding to passive Kalman decomposition

which transforms the physically realizable passive linear quantum system (15), (16) into the form

$$\begin{bmatrix} \dot{\boldsymbol{a}}_{\bar{c}\bar{o}}(t) \\ \dot{\boldsymbol{a}}_{co}(t) \end{bmatrix} = \bar{\mathscr{A}} \begin{bmatrix} \boldsymbol{a}_{\bar{c}\bar{o}}(t) \\ \boldsymbol{a}_{co}(t) \end{bmatrix} + \bar{\mathscr{B}} \boldsymbol{b}(t); \tag{41}$$

$$\boldsymbol{b}_{\text{out}}(t) = \bar{\mathscr{C}} \begin{bmatrix} \boldsymbol{a}_{\bar{c}\bar{o}}(t) \\ \boldsymbol{a}_{co}(t) \end{bmatrix} + \boldsymbol{b}(t), \tag{42}$$

where

$$\begin{aligned} \bar{\mathscr{A}} &\triangleq T^{\dagger} \mathscr{A} T = \begin{bmatrix} \mathscr{A}_{\bar{c}\bar{o}} & 0 \\ 0 & \mathscr{A}_{co} \end{bmatrix}; \\ \bar{\mathscr{B}} &\triangleq T^{\dagger} \mathscr{B} = \begin{bmatrix} 0 \\ \mathscr{B}_{co} \end{bmatrix}; \\ \bar{\mathscr{C}} &\triangleq \mathscr{C} T = \begin{bmatrix} 0 \; \mathscr{C}_{co} \end{bmatrix}. \end{aligned} \tag{43}$$

Here the pair $(\mathscr{A}_{co}, \mathscr{B}_{co})$ *is controllable and the pair* $(\mathscr{A}_{co}, \mathscr{C}_{co'})$ *is observable.*

A block diagram for the system (41)–(43) is given in Fig. 2.

We now consider the Kalman decomposition for physically realizable position-momentum linear quantum systems of the form (19), (26). The Kalman decomposition for the system (19), (26) involves decomposing the system into four subsystems via a state space transformation of the form

$$\begin{bmatrix} \boldsymbol{q}_h \\ \boldsymbol{p}_h \\ \hline \boldsymbol{x}_{co} \\ \hline \boldsymbol{x}_{\bar{c}\bar{o}} \end{bmatrix} = S^{\top} \boldsymbol{x}.$$

Each of these four subsystems corresponds to a subsystem in the standard classical Kalman decomposition. However, in the quantum case, the subsystems corresponding to the variables $\boldsymbol{q}_h$ and $\boldsymbol{p}_h$ will always be of the same dimension and we will group these two subsystems together by defining a vector of variables $\boldsymbol{x}_h = \begin{bmatrix} \boldsymbol{q}_h \\ \boldsymbol{p}_h \end{bmatrix}$ so that the transformed system can be regarded as involving three subsystems:

$$\begin{bmatrix} \boldsymbol{x}_h \\ \boldsymbol{x}_{co} \\ \boldsymbol{x}_{\bar{c}\bar{o}} \end{bmatrix} = S^\top \boldsymbol{x}.$$

Then, we require the transformation matrix to be constructed so that the commutation relations (3) are satisfied by each subsystem in the transformed system. That is, each subsystem in the transformed system is required to be a physically realizable position-momentum quantum system. In order to achieve this, we require that the transformation matrix S satisfies the following blockwise symplectic condition:

$$S^\top \mathbb{J}_n S = \operatorname{diag}\left(\mathbb{J}_{n_3}, \mathbb{J}_{n_1}, \mathbb{J}_{n_2}\right). \tag{44}$$

Here the notation $\mathbb{J}_k$ refers to a $(2k) \times (2k)$ matrix of the form (24).

Theorem 9 (See [69]) *There exists a real orthogonal and blockwise symplectic coordinate transformation*

$$\left[\begin{array}{c} \boldsymbol{q}_h \\ \boldsymbol{p}_h \\ \hline \boldsymbol{x}_{co} \\ \hline \boldsymbol{x}_{\bar{c}\bar{o}} \end{array}\right] = S^\top \boldsymbol{x} \tag{45}$$

which transforms the physically realizable position-momentum linear quantum system (19), (26) into the form

$$\left[\begin{array}{c} \dot{\boldsymbol{q}}_h(t) \\ \dot{\boldsymbol{p}}_h(t) \\ \hline \dot{\boldsymbol{x}}_{co}(t) \\ \hline \dot{\boldsymbol{x}}_{\bar{c}\bar{o}}(t) \end{array}\right] = \bar{A} \left[\begin{array}{c} \boldsymbol{q}_h(t) \\ \boldsymbol{p}_h(t) \\ \hline \boldsymbol{x}_{co}(t) \\ \hline \boldsymbol{x}_{\bar{c}\bar{o}}(t) \end{array}\right] + \bar{B}\boldsymbol{u}(t); \tag{46}$$

$$\boldsymbol{y}(t) = \bar{C} \left[\begin{array}{c} \boldsymbol{q}_h(t) \\ \boldsymbol{p}_h(t) \\ \hline \boldsymbol{x}_{co}(t) \\ \hline \boldsymbol{x}_{\bar{c}\bar{o}}(t) \end{array}\right] + \boldsymbol{u}(t), \tag{47}$$

where matrices $\bar{A}$, $\bar{B}$, $\bar{C}$ are of the form

$$\bar{A} = \left[\begin{array}{cc|c|c} A_h^{11} & A_h^{12} & A_{12} & A_{13} \\ 0 & A_h^{22} & 0 & 0 \\ \hline 0 & A_{21} & A_{co} & 0 \\ \hline 0 & A_{31} & 0 & A_{\bar{c}\bar{o}} \end{array}\right];$$

$$\bar{B} = \left[\begin{array}{c} B_h \\ 0 \\ \hline B_{co} \\ \hline 0 \end{array}\right];$$

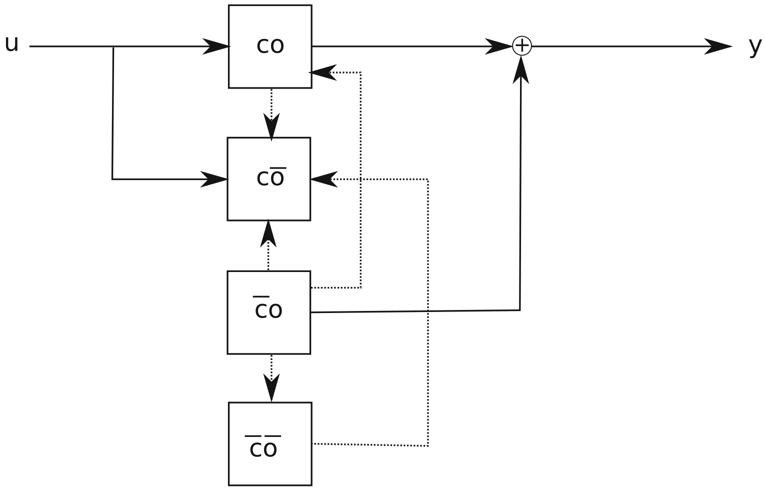


Fig. 3 Block diagram corresponding to position-momentum Kalman decomposition

$$\bar{C} = \left[\, 0 \; C_h \middle| C_{co} \middle| 0 \,\right]. \tag{48}$$

After a rearrangement, the system (46)–(47) becomes

$$\begin{bmatrix} \dot{\boldsymbol{q}}_h(t) \\ \dot{\boldsymbol{x}}_{co}(t) \\ \dot{\boldsymbol{x}}_{c\bar{o}}(t) \\ \dot{\boldsymbol{p}}_h(t) \end{bmatrix} = \begin{bmatrix} A_h^{11} & A_{12} & A_{13} & A_h^{12} \\ 0 & A_{co} & 0 & A_{21} \\ 0 & 0 & A_{c\bar{o}} & A_{31} \\ 0 & 0 & 0 & A_h^{22} \end{bmatrix} \begin{bmatrix} \boldsymbol{q}_h(t) \\ \boldsymbol{x}_{co}(t) \\ \boldsymbol{x}_{c\bar{o}}(t) \\ \boldsymbol{p}_h(t) \end{bmatrix} + \begin{bmatrix} B_h \\ B_{co} \\ 0 \\ 0 \end{bmatrix} \boldsymbol{u}(t); \tag{49}$$

$$\boldsymbol{y}(t) = [0 \; C_{co} \; 0 \; C_h] \begin{bmatrix} \boldsymbol{q}_h(t) \\ \boldsymbol{x}_{co}(t) \\ \boldsymbol{x}_{c\bar{o}}(t) \\ \boldsymbol{p}_h(t) \end{bmatrix} + \boldsymbol{u}(t). \tag{50}$$

Here the pairs (A_h^{11}, B_h) and (A_{co}, B_{co}) are controllable and the pairs (A_{co}, C_{co}) and (A_h^{22}, C_h) are observable.

A block diagram for the system (46)–(48) is given in Fig. 3.

2.6 Finite Level Quantum Systems in the Heisenberg Picture

In the Heisenberg picture model of an open finite level quantum system, the model consists of bilinear QSDEs where the system variables are initialized at the *generators of $SU(n)$* for an n-level quantum system; see [13]. These generators are known as the *generalized Gell-Mann matrices*. The generators are labelled $\{I, \lambda_1, \ldots, \lambda_s\}$, where $s = n^2 - 1$. These matrices satisfy $\mathrm{Tr}(\lambda_i \lambda_j) = 2\delta_{ij}$, where δ_{ij} denotes the Kronecker delta. They also satisfy the following commutation and anti-commutation relations:

$$[\lambda_i, \lambda_j] = 2\boldsymbol{i} \sum_{k=1}^{s} f_{ijk} \lambda_k;$$

$$\{\lambda_i, \lambda_j\} = \frac{4}{n}\delta_{ij} + 2\sum_{k=1}^{s} d_{ijk}\lambda_k.$$

Then, the product $\lambda_i \lambda_j$ can be computed as

$$\lambda_i \lambda_j = \frac{1}{2}\left([\lambda_i, \lambda_j] + \{\lambda_i, \lambda_j\}\right) = \frac{2}{n}\delta_{ij} + \sum_{k=1}^{s}\left(\boldsymbol{i} f_{ijk} + d_{ijk}\right)\lambda_k,$$

where the real completely antisymmetric tensor f_{ijk} and the real completely symmetric tensor d_{ijk} are the *structure constants* of $SU(n)$. These tensors satisfy

$$f_{ilm} f_{mjk} + f_{jlm} f_{imk} + f_{klm} f_{ijm} = 0; \tag{51a}$$

$$f_{ilm} d_{mjk} + f_{jlm} d_{imk} + f_{klm} d_{ijm} = 0; \tag{51b}$$

$$\sum_{k=1}^{s} f_{ilk} f_{mjk} = \frac{2}{n}\left(\delta_{im}\delta_{lj} - \delta_{ij}\delta_{lm}\right) + \sum_{k=1}^{s}\left(d_{imk} d_{ljk} - d_{ijk} d_{lmk}\right); \tag{51c}$$

$$\sum_{m,k=1}^{s} f_{imk} f_{jmk} = n\delta_{ij}. \tag{51d}$$

We define the matrices F_i, $D_i \in \mathbb{R}^{s\times s}$, $i \in \{1, \ldots, s\}$, such that their (j, k) components are $(F_i)_{jk} = f_{ijk}$ and $(D_i)_{jk} = d_{ijk}$. The identities (51a)–(51c) imply [13]

$$[F_i, F_j] = -\sum_{k}^{s} f_{ijk} F_k; \tag{52a}$$

$$[F_i, D_j] = -\sum_{k}^{s} f_{ijk} D_k; \tag{52b}$$

$$F_i D_j + F_j D_i = \sum_{k}^{s} d_{ijk} F_k; \tag{52c}$$

$$D_i F_j + D_j F_i = \sum_{k}^{s} d_{ijk} F_k; \tag{52d}$$

$$\left(D_i D_j - F_j F_i\right)_{ml} = \sum_k^s d_{ijk}(D_k)_{ml} + \frac{2}{n}\left(\delta_{ij}\delta_{ml} - \delta_{im}\delta_{jl}\right). \tag{52e}$$

Definition 7 Let $\beta \in \mathbb{C}^s$. The linear mappings $\Theta^-, \Theta^+ : \mathbb{C}^s \to \mathbb{C}^{s\times s}$ are defined as

$$\Theta^-(\beta) = \left(F_1^T\beta, \cdots, F_s^T\beta\right) = \begin{pmatrix} \beta^T F_1^T \\ \vdots \\ \beta^T F_s^T \end{pmatrix}; \tag{53a}$$

$$\Theta^+(\beta) = \left(D_1^T\beta, \cdots, D_s^T\beta\right) = \begin{pmatrix} \beta^T D_1^T \\ \vdots \\ \beta^T D_s^T \end{pmatrix}. \tag{53b}$$

It follows from the properties of the f and d-tensors that $\Theta^-(\beta)$ and $\Theta^+(\beta)$ are antisymmetric and symmetric, respectively. For an s-dimensional row vector β, then we use the notation $\Theta^-(\beta) = \Theta^-(\beta^T)$ and $\Theta^+(\beta) = \Theta^+(\beta^T)$. Also, we consider the *stacking operator* vec : $\mathbb{C}^{m\times n} \to \mathbb{C}^{mn}$ which acts on a matrix to create a column vector by stacking its columns on top of one another. The stacking operator vec has the following property:

$$\text{vec}(ABC) = (C^T \otimes A)\,\text{vec}(B) \tag{54}$$

for $A \in \mathbb{C}^{n\times m}$, $B \in \mathbb{C}^{m\times l}$ and $C \in \mathbb{C}^{l\times r}$ where $n, m, l, r \in \mathbb{N}$, and $\otimes$ denotes the Kronecker product. The matrices $\Theta^-(\beta)$ and $\Theta^+(\beta)$ are such that

$$\text{vec}(\Theta^-(\beta)) = \begin{pmatrix} \Theta_1^-(\beta) \\ \vdots \\ \Theta_s^-(\beta) \end{pmatrix} = F\beta,$$

and

$$\text{vec}(\Theta^+(\beta)) = \begin{pmatrix} \Theta_1^+(\beta) \\ \vdots \\ \Theta_s^+(\beta) \end{pmatrix} = D\beta,$$

where $\Theta_i^-(\beta) = F_i^T\beta$,

$$F = (F_1, \cdots, F_s)^T, \tag{55}$$

$\Theta_i^+(\beta) = D_i\beta$,

$$D = (D_1, \cdots, D_s)^T \tag{56}$$

and $\beta \in \mathbb{C}^s$. Using (51d), F satisfies

$$F^T F = nI. \tag{57}$$

The following lemma gives some of the properties of $\Theta^-(\cdot)$ and $\Theta^+(\cdot)$.

Lemma 1 (See [13]) *Let $\beta, \gamma \in \mathbb{C}^s$ be given. Then, the mappings $\Theta^-(\cdot)$ and $\Theta^+(\cdot)$ satisfy*

$$\Theta^-(\beta)\gamma = -\Theta^-(\gamma)\beta; \tag{58a}$$

$$\Theta^+(\beta)\gamma = \Theta^+(\gamma)\beta; \tag{58b}$$

$$\Theta^-(\beta)\beta = 0; \tag{58c}$$

$$\Theta^-\left(\Theta^-(\beta)\gamma\right) = [\Theta^-(\beta), \Theta^-(\gamma)]; \tag{58d}$$

$$\Theta^-\left(\Theta^+(\beta)\gamma\right) = \Theta^-(\beta)\Theta^+(\gamma) + \Theta^-(\gamma)\Theta^+(\beta); \tag{58e}$$

$$\Theta^+\left(\Theta^-(\beta)\gamma\right) = [\Theta^+(\beta), \Theta^-(\gamma)] = [\Theta^-(\beta), \Theta^+(\gamma)]; \tag{58f}$$

$$\Theta^+\left(\Theta^+(\beta)\gamma\right) = \Theta^+(\beta)\Theta^+(\gamma) - \Theta^-(\gamma)\Theta^-(\beta) - \frac{2}{n}\left(\beta^T\gamma I - \beta\gamma^T\right). \tag{58g}$$

Now we define a vector of *spin operators* X whose components are the system variables evolving in $SU(n)$ for a finite level open quantum system:

$$X = \begin{pmatrix} X_1 \\ \vdots \\ X_s \end{pmatrix}. \tag{59}$$

Here $X_1, \ldots, X_s$ are spanned by the generalized Gell-Mann matrices. The initial value of the system variables can be set to $X(0) = (\lambda_1^T, \ldots, \lambda_s^T)^T$ with $\lambda_1, \ldots, \lambda_s$ being the generators of $SU(n)$.

Using this notation, the commutation and anti-commutation relations can be expressed as

$$[X, X^T] = 2\boldsymbol{i}\Theta^-(X); \tag{60a}$$

$$\{X, X^T\} = \frac{4}{n}I + 2\Theta^+(X). \tag{60b}$$

Here, $\{X, Y^T\} \triangleq XY^T + (YX^T)^T$. Therefore,

$$XX^T = \frac{1}{2}\left([X, X^T] + \{X, X^T\}\right) = \frac{2}{n}I + \boldsymbol{i}\Theta^-(X) + \Theta^+(X). \tag{61}$$

As in the case of linear quantum systems, the dynamics of an open finite level quantum system are specified by a system Hamiltonian and vector of coupling operators. Without significant loss of generality, we will restrict attention to the class of linear Hamiltonians

$$H = \alpha X \tag{62}$$

with $\alpha^T \in \mathbb{R}^s$, and also consider the class of vector coupling operators of the form

$$L = \Gamma X \tag{63}$$

with $\Gamma \in \mathbb{C}^{m\times s}$. Here we assume that the scattering matrix S is equal to the identity:

$$S = I. \tag{64}$$

These values of S, L, H then form an (S, L, H) description of a finite level open quantum system. The Heisenberg picture description of a finite level open quantum system interacting with m fields is then described by a set of bilinear QSDEs of the form

$$dX = A_0\,dt + AX\,dt + (BX, \cdots, B_{1m}X, B_{21}X, \cdots, B_{2m}X)\begin{bmatrix} d\mathscr{Q}(t) \\ d\mathscr{P}(t) \end{bmatrix}; \tag{65}$$

$$\begin{bmatrix} d\mathscr{Q}^{out}(t) \\ d\mathscr{P}^{out}(t) \end{bmatrix} = \begin{pmatrix} C_1 \\ C_2 \end{pmatrix} X\,dt + \begin{bmatrix} d\mathscr{Q}(t) \\ d\mathscr{P}(t) \end{bmatrix} \tag{66}$$

where $\begin{bmatrix} d\mathscr{Q}(t) \\ d\mathscr{P}(t) \end{bmatrix}$ defines the field quadrature variables as in (17). Here, $A_0 \in \mathbb{R}^s$, $A, B_{1k}, B_{2k} \in \mathbb{R}^{s\times s}$ and $C_1, C_2 \in \mathbb{R}^{m\times s}$ for $k = 1, \dots, m$. These real vectors and matrices are given by the following equations in terms of the system Hamiltonian and coupling operators:

$$A_0 = \frac{4\boldsymbol{i}}{n}\sum_{k=1}^{m} \Theta^-(\Gamma_k^\#)\Gamma_k^T; \tag{67a}$$

$$A = -2\Theta^-(\alpha) + \sum_{k=1}^{m}(R_k - \boldsymbol{i}\,Q_k); \tag{67b}$$

$$B_{1k} = \Theta^-\left(\boldsymbol{i}(\Gamma_k^\# - \Gamma_k)\right); \tag{67c}$$

$$B_{2k} = -\Theta^-(\Gamma_k + \Gamma_k^\#); \tag{67d}$$

$$C_1 = \Gamma + \Gamma^\#; \tag{67e}$$

$$C_2 = \boldsymbol{i}\left(\Gamma^\# - \Gamma\right) \tag{67f}$$

where

$$R_k \triangleq \Theta^-(\Gamma_k)\Theta^-(\Gamma_k^\#) + \Theta^-(\Gamma_k^\#)\Theta^-(\Gamma_k);$$
$$Q_k \triangleq \Theta^-(\Gamma_k)\Theta^+(\Gamma_k^\#) - \Theta^-(\Gamma_k^\#)\Theta^+(\Gamma_k).$$

2.7 Physical Realizability of Finite Level Quantum Systems

We now consider conditions under which a set of bilinear QSDEs of the form (65), (66) in fact corresponds to a finite level quantum system with an (S, L, H) description of the form (62), (63), (64). In this case, the bilinear QSDEs (65), (66) are said to be physically realizable. This notion is formalized in the following definition.

Definition 8 Bilinear QSDEs of the form (65), (66) are said to be *physically realizable* if there exist a vector α and a complex matrix Γ such that equations (67) are satisfied.

The following theorem gives necessary and sufficient conditions for physical realizability in this case.

Theorem 10 (See [13]) *The bilinear QSDEs (65), (66) are physically realizable if and only if the following conditions are satisfied:*

$$A_0 = \frac{1}{n}\sum_{k=1}^{n_w}(B_{1k} + \boldsymbol{i}B_{2k})\left((C_1)_k + \boldsymbol{i}(C_2)_k\right)^{\dagger}; \tag{68a}$$

$$B_{1k} = \Theta^{-}((C_2)_k); \tag{68b}$$

$$B_{2k} = -\Theta^{-}((C_1)_k); \tag{68c}$$

$$A + A^T + \sum_{i,k=1}^{2,n_w} B_{ik}{B_{ik}}^T = \frac{n}{2}\Theta^{+}(A_0). \tag{68d}$$

In this case, the coupling matrix is given by

$$\Gamma = \frac{1}{2}(C_1 + \boldsymbol{i}C_2), \tag{69}$$

and the vector α defining the system Hamiltonian, is given by

$$\alpha = \frac{1}{4n}\,\mathrm{vec}\left(A^T - A + \frac{1}{2}\sum_{k=1}^{n_w}\left(\{B_{1k}, \Theta^{+}((C_1)_k)\} + \{B_{2k}, \Theta^{+}((C_2)_k)\}\right)\right)^T F \tag{70}$$

where F is defined as in (55).

2.8 Finite Level Quantum Systems in the Schrödinger Picture

We now consider Schrödinger picture models of finite level quantum systems. In these models, we consider the evolution of the quantum state which is described by a positive density matrix ρ. The density ρ is an operator on the underlying finite-dimensional complex Hilbert space satisfying $\mathrm{tr}(\rho) = 1$. The dynamics of $\rho(t)$ are described by a Markovian master equation of the form

$$\dot{\rho}(t) = -\boldsymbol{i}[H, \rho(t)] + \sum_{k=1}^{m} \left(L_k \rho(t) L_k^\dagger - \frac{1}{2} L_k^\dagger L_k \rho(t) - \frac{1}{2} \rho(t) L_k^\dagger L_k \right) \qquad (71)$$

where H is the system Hamiltonian operator and

$$L = \begin{bmatrix} L_1 \\ L_2 \\ \vdots \\ L_m \end{bmatrix}$$

is the coupling operator vector for the system.

3 Linear Quantum Networks in the Realization of Quantum Systems

In this section, we consider various linear quantum networks which can be used in the implementation of physically realizable quantum linear systems and transfer function matrices. We begin with the case of passive physically realizable transfer function matrices which can be realized by a cascade of optical cavities and phase shifters.

3.1 Cascade Network Realization of Passive Physically Realizable Transfer Function Matrices

In this subsection, we consider the problem of constructing a network of quantum optical components such that this network has a transfer function matrix which is the same as a given passive physically realizable transfer function matrix $\varGamma(s)$ of the form (33). In this case, the network will consist of a cascade connection of optical ring cavities and phase shifters.

A multi-mirror ring cavity is a passive optical component consisting of a collection of partially reflecting mirrors arranged as shown in Fig. 4. Such an optical ring cavity is a passive optical device and can be described by a set of passive QSDEs of the form (15) as follows:

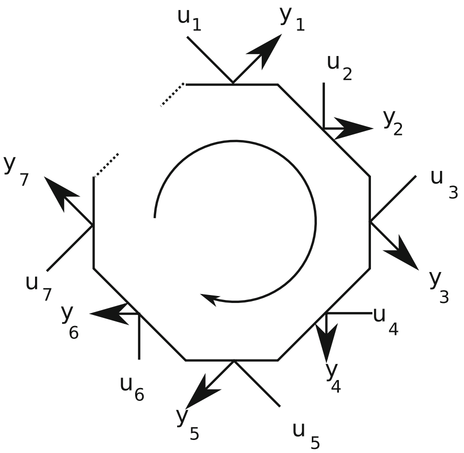


Fig. 4 Schematic representation of an m mirror optical cavity

$$
\begin{aligned}
da &= \left(-\frac{\gamma}{2} + \imath \Delta\right) a dt - \sum_{i=1}^{m} \sqrt{\kappa_i} du_i; \\
dy_i &= \sqrt{\kappa_i} a dt + du_i, \quad i = 1, 2, \ldots, m.
\end{aligned} \tag{72}
$$

Here, $\gamma = \sum_{i=1}^{m} \kappa_i$. The quantities $\kappa_i \geq 0$, $i = 1, 2, \ldots, m$ are the *coupling coefficients* which correspond to the reflectivities of the partially reflecting mirrors which make up the cavity. Also, the quantity $\Delta \in \mathbb{R}$ corresponds to the detuning between the resonant frequency of the cavity and the frequency of the coherent fields applied to the cavity.

We now generalize this m-input m-output linear quantum system by introducing phase shifters on each input and output channel as shown in the block diagram in Fig. 5. In practice, these phase shifts would be introduced by simply adjusting the optical path length in the corresponding optical channel. This leads to a passive quantum system which can be described by a set of QSDEs of the form (15) as follows:

$$
\begin{aligned}
da &= \left(-\frac{\gamma}{2} + \imath \Delta\right) a dt - \sum_{i=1}^{m} \sqrt{\kappa_i} e^{-\imath \theta_i} du_i; \\
dy_i &= \sqrt{\kappa_i} e^{\imath \theta_i} a dt + du_i, \quad i = 1, 2, \ldots, m.
\end{aligned} \tag{73}
$$

Thus, the phase shifter cavity system can be described by QSDEs of the form:

$$
da = p a dt - h^{\dagger} du; \quad dy = h a dt + du \tag{74}
$$

where

$$
p + p^{*} = -\gamma = -\sum_{i=1}^{m} \kappa_i = -h^{\dagger} h. \tag{75}
$$

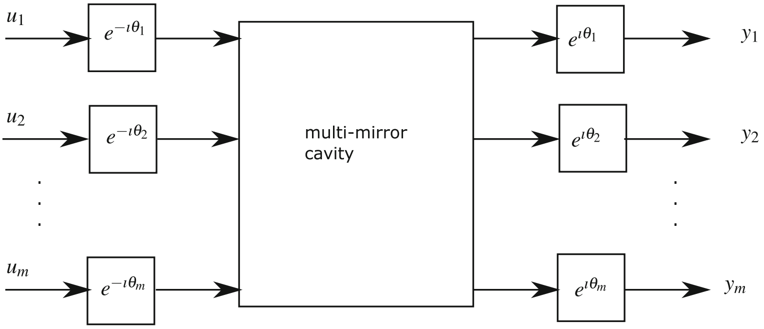


Fig. 5 Block diagram of an *m* channel optical cavity with phase shifters

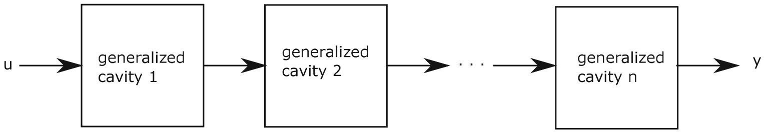


Fig. 6 Block diagram of a cascade of *n* generalized cavities

Here $p = -\gamma/2 + \imath\Delta$,

$$h = \begin{bmatrix} \sqrt{\kappa_1}e^{\imath\theta_1} \\ \sqrt{\kappa_2}e^{\imath\theta_2} \\ \vdots \\ \sqrt{\kappa_m}e^{\imath\theta_m} \end{bmatrix}, du = \begin{bmatrix} du_1 \\ du_2 \\ \vdots \\ du_m \end{bmatrix}, \; dy = \begin{bmatrix} dy_1 \\ dy_2 \\ \vdots \\ dy_m \end{bmatrix}.$$

It is straightforward to verify that these QSDEs are physically realizable according to Definition 3. Such as system is referred to as a generalized cavity. We now consider a linear quantum network consisting of a cascade of n such systems as shown in the block diagram in Fig. 6.

If each of the generalized cavities in the linear quantum network is described by the QSDEs

$$\begin{aligned} da_k &= p_k a_k dt - H_k^\dagger du_k; \\ dy_k &= H_k a_k dt + du_k; \qquad (76) \\ p_k + p_k^* &= -H_k^\dagger H_k, \qquad (77) \end{aligned}$$

then the total cascade network can be described in terms of passive QSDEs of the form (15) where

$$F_1 = \begin{bmatrix} p_1 & 0 & \dots & 0 \\ -H_2^\dagger H_1 & p_2 & & \\ \vdots & & \ddots & \vdots \\ & & & 0 \\ -H_n^\dagger H_1 & \dots & -H_n^\dagger H_{n-1} & p_n \end{bmatrix}, G_1 = -\begin{bmatrix} H_1^\dagger \\ H_2^\dagger \\ \vdots \\ H_n^\dagger \end{bmatrix},$$
$$H_1 = \left[H_1 \; H_2 \dots H_n \right], J = I. \tag{78}$$

The main result of this subsection shows that almost all passive physically realizable transfer function matrices of the form (33) can be realized via such a cascade linear quantum network. Indeed, the class of such transfer function matrices for which this physical realization exists are those passive physically realizable transfer function matrices with distinct poles. Such transfer function matrices have a state space realization in modal canonical form as follows:

$$\begin{aligned} d\tilde{a}(t) &= \tilde{F}\tilde{a}(t)dt + \tilde{G}du(t); \\ dy(t) &= \tilde{H}\tilde{a}(t)dt + du(t) \end{aligned} \tag{79}$$

where

$$\tilde{F} = \begin{bmatrix} p_1 & 0 & \dots & 0 \\ 0 & p_2 & & \vdots \\ \vdots & & \ddots & 0 \\ 0 & \dots & 0 & p_n \end{bmatrix}; \tilde{G} = \begin{bmatrix} \tilde{G}_1 \\ \tilde{G}_2 \\ \vdots \\ \tilde{G}_n \end{bmatrix};$$
$$\tilde{H} = \left[\tilde{H}_1 \; \tilde{H}_2 \dots \tilde{H}_n \right]. \tag{80}$$

In order to construct a cascade cavity physical realization for such a passive physically realizable transfer function matrix, we will apply the following algorithm.

Step 1: Begin with a minimal modal canonical form realization (79), (80) of the lossless bounded real transfer function $K(s)$.

Step 2: Let

$$\begin{aligned} \bar{H}_n &= \tilde{H}_n, \alpha_n = -\frac{\bar{H}_n^\dagger \bar{H}_n}{p_n + p_n^*}, \\ H_n &= \frac{\bar{H}_n}{\sqrt{\alpha_n}}, \; t(n,n) = \frac{1}{\sqrt{\alpha_n}}. \end{aligned} \tag{81}$$

Step 3: Calculate the quantities $H_{n-1}, H_{n-2}, \dots, H_1, \alpha_{n-1}, \alpha_{n-2}, \dots, \alpha_1, t(i, j)$, for $j = n-1, n-2, \dots, 1$ and $i \geq j$. These values are calculated using the following recursive formulas starting with the values determined in Step 2 for $i = n$:

$$\bar{H}_i = \left[I + \sum_{j=i+1}^{n} \frac{\tilde{H}_j}{p_j - p_i} \sum_{k=i+1}^{j} t(j,k) H_k^\dagger \right]^{-1} \tilde{H}_i; \tag{82}$$

$$\alpha_i = -\frac{\bar{H}_i^\dagger \bar{H}_i}{p_i + p_i^*}, \; H_i = \frac{\bar{H}_i}{\sqrt{\alpha_i}}, \tag{83}$$

$$t(k,i) = \frac{1}{p_i - p_k} \sum_{j=i+1}^{k} t(k,j) H_j^\dagger H_i \text{ for } k = i+1, \ldots, n, \tag{84}$$

$$t(i,i) = \frac{1}{\sqrt{\alpha_i}}. \tag{85}$$

Step 4: . Set $t(k,i) = 0$ for $k < i$ and define a transformation matrix T whose (i,j)th element is $t(i,j)$.

Then, we have the following result.

Theorem 11 (See [53]) *Consider an $m \times m$ lossless bounded real complex transfer function matrix $K(s)$ with a minimal modal canonical form quantum realization (79), (80) such that the eigenvalues of the matrix $\tilde{F}$ are all distinct and that all of the matrix inverses exist in Eq. (82) when the above algorithm is applied to the system (79), (80). Then, the vectors $H_1, H_2, \ldots, H_n$ defined in the above algorithm together with the eigenvalues $p_1, p_2, \ldots, p_n$ define an equivalent cascade quantum realization (15), (78) for the transfer function matrix $K(s)$. Furthermore, this system is such that the condition (77) is satisfied for all k. Moreover, the matrices $\{F, G, H, I\}$ defining this cascade quantum realization are related to the matrices $\{\tilde{F}, \tilde{G}, \tilde{H}, I\}$ defining the modal quantum realization (79), (80) according to the formulas:*

$$\tilde{F} = TFT^{-1}, \; \tilde{G} = TG, \; \tilde{H} = HT^{-1} \tag{86}$$

where the matrix T is defined in the above algorithm.

3.2 Cascade Network Realization for Generic Physically Realizable Transfer Function Matrices

In this subsection, we consider a general physically realizable position-momentum quantum linear system of the form (19), (26). We wish to construct a change of variables $\tilde{x} = Tx$ on the vector of system variables x defined in (17) such that

$$\tilde{x} = \begin{bmatrix} \tilde{x}_1 \\ \tilde{x}_2 \\ \vdots \\ \tilde{x}_{\frac{n}{2}} \end{bmatrix}$$

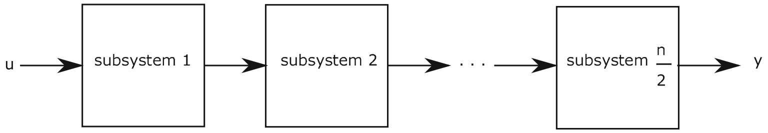


Fig. 7 Block diagram of a pure cascade of $\frac{n}{2}$ subsystems

where

$$\tilde{x}_i = \begin{bmatrix} \tilde{q}_i \\ \tilde{p}_i \end{bmatrix}$$

for $i = 1, 2, \dots, \frac{n}{2}$. Furthermore, it is required that the transformed system can be expressed as a pure cascade of its single mode subsystems corresponding to each sub-vector $\tilde{x}_i$ as illustrated in Fig. 7. In addition, it is required that the commutation relations (21) be preserved for each of the subsystems of the transformed quantum system. This restricts the class of transformation matrices T which can be considered. To specify this class of transformation matrices, we first write $T = \tilde{T}\Pi$ where Π is a permutation matrix such that

$$\Pi x = \Pi \begin{bmatrix} q \\ p \end{bmatrix} = \begin{bmatrix} q_1 \\ p_1 \\ q_2 \\ p_2 \\ \vdots \\ q_{\frac{n}{2}} \\ q_{\frac{n}{2}} \end{bmatrix}.$$

Then, the matrix $\tilde{T}$ is required to be a symplectic matrix. That is $\tilde{T}$ satisfies $\tilde{T}\mathbb{J}_n\tilde{T}^T = \mathbb{J}_n$ where $\mathbb{J}_n$ is the block diagonal matrix $\mathbb{J}_n = \mathrm{Diag}(\mathbb{J}_2, \mathbb{J}_2, \dots, \mathbb{J}_2)$ and $\mathbb{J}_2 = \begin{bmatrix} 0 & 1 \\ -1 & 0 \end{bmatrix}$.

Definition 9 A square matrix M of even dimension is said to be 2×2 block lower triangular if it has the form

$$M = \begin{bmatrix} M_{11} & 0 & 0 & \dots & 0 \\ M_{21} & M_{22} & 0 & \dots & 0 \\ \vdots & \ddots & \ddots & \ddots & \vdots \\ M_{n1} & M_{n2} & \dots & \dots & M_{nn} \end{bmatrix}$$

where each matrix M_{ij} is a 2×2 matrix. Furthermore, M is said to be 2×2 block upper triangular if M^T is 2×2 block lower triangular.

The algorithm for constructing the desired transformation matrix $\tilde{T}$ involves the symplectic QR transformation of [45].

Lemma 2 (See [45]) *Let V be a real inevitable matrix of even dimension n with linearly independent columns $v_1, v_2, \ldots, v_n$. Let $M_i = \left[\, v_{2i-1} \ v_{2i} \,\right]$ for $i = 1, 2, \ldots, \frac{n}{2}$. Also, let $\tilde{M}_1 = M_1$, $\tilde{M}_2 = \left[\, M_1 \ M_2 \,\right]$, …, $\tilde{M}_{\frac{n}{2}} = \left[\, M_1 \ M_2 \ \ldots \ M_{\frac{n}{2}} \,\right]$. Assume that $\tilde{N}_i = \tilde{M}_i^T \mathbb{J}_n \tilde{M}_i = \mathbb{J}_n$ is full rank for $i = 1, 2, \ldots, \frac{n}{2} - 1$. Then, the matrix V has a QR decomposition $V = SY$ where S is symplectic and Y is 2×2 block upper triangular. Furthermore, the matrix S can be constructed recursively.*

This lemma is used in [45] to prove the following result on a Symplectic Schur Decomposition.

Lemma 3 (See [45]) *Let A be a real matrix of even dimension n. Then there exits a symplectic matrix S and a 2×2 block lower triangular matrix U such that A has the symplectic Schur deposition $A = S^{-1}US$ if there exists an invertable matrix $\tilde{V}$ with linearly independent columns $\tilde{v}_1, \tilde{v}_2, \ldots, \tilde{v}_n$ such that the following conditions are satisfied.*

1. *The matrix $\tilde{V}^{-1}A\tilde{V}$ is 2×2 block upper triangular and in real Jordan canonical form.*
2. *The matrices $\tilde{N}_1, \tilde{N}_2, \ldots \tilde{N}_{\frac{n}{2}-1}$ given by*

$$\begin{aligned}
\tilde{N}_1 &= \left[\, \tilde{v}_1 \ \tilde{v}_2 \,\right]^T \mathbb{J}_n \left[\, \tilde{v}_1 \ \tilde{v}_2 \,\right]; \\
\tilde{N}_2 &= \left[\, \tilde{v}_1 \ \tilde{v}_2 \ \tilde{v}_3 \ \tilde{v}_4 \,\right]^T \mathbb{J}_n \left[\, \tilde{v}_1 \ \tilde{v}_2 \ \tilde{v}_3 \ \tilde{v}_4 \,\right]; \\
&\ \vdots \\
\tilde{N}_{\frac{n}{2}-1} &= \left[\, \tilde{v}_1 \ \tilde{v}_2 \ \ldots \ \tilde{v}_{n-2} \,\right]^T \mathbb{J}_n \left[\, \tilde{v}_1 \ \tilde{v}_2 \ \ldots \ \tilde{v}_{n-2} \,\right]
\end{aligned}$$

are all full rank.

This lemma enables the following result to be established.

Theorem 12 (See [45]) *Consider a physically realizable position-momentum quantum linear system of the form (19), (26). Suppose there exists a matrix $\tilde{V}$ associated with the matrix A for this system satisfying the conditions of Lemma 3. Then there exists a symplectic matrix S such that the transformed system matrix SAS^{-1} is 2×2 block lower triangular. This implies that the corresponding transformed system is physically realizable with a pure cascade realization and that the corresponding transfer function matrix $\Upsilon(s)$ in (35) is physically realizable with a pure cascade realization.*

This theorem leads to the following corollary.

Corollary 1 (See [45]) *Almost all physically realizable position-momentum quantum linear systems of the form (19), (26) have a pure cascade realization.*

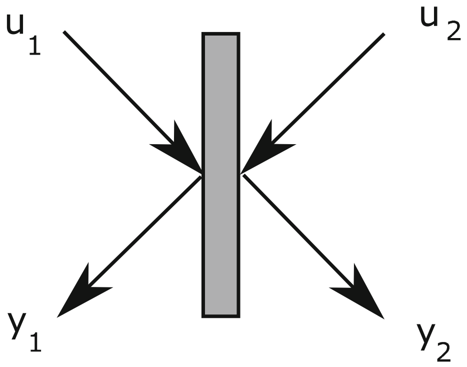


Fig. 8 Schematic diagram of a beamsplitter

3.3 Quantum Network Realization for General Physically Realizable Transfer Function Matrices

In this subsection, we consider the optical realization of general physically realizable quantum systems and transfer function matrices, including those which may not be realizable using a pure cascade. This requires a more complex class of quantum networks involving the use of feedback; see [20]. Here, we consider a physically realizable annihilation-creation quantum system of the form (9), (11) with transfer function matrix $\Gamma(s)$ (30). In addition to the passive cavities considered in Sect. 3.1, we will consider beamsplitters, static squeezers, and generalized cavities allowing for both passive and active channels.

A beamsplitter is a passive optical device consisting of a single partially reflecting mirror as illustrated in Fig. 8. A beamsplitter has two input channels and two output channels and is described by the equations

$$\begin{bmatrix} y_1 \\ y_2 \end{bmatrix} = R \begin{bmatrix} u_1 \\ u_2 \end{bmatrix}$$

where

$$R = \begin{bmatrix} e^{\imath \frac{\phi+\psi}{2}} \cos\frac{\theta}{2} & e^{\imath \frac{\phi-\psi}{2}} \sin\frac{\theta}{2} \\ -e^{\imath \frac{\phi-\psi}{2}} \sin\frac{\theta}{2} & e^{-\imath \frac{\phi+\psi}{2}} \cos\frac{\theta}{2} \end{bmatrix}.$$

Here, $\theta \in [0, 2\Pi]$ is the mixing angle of the beamsplitter. The angles ϕ and ψ correspond to phase shifts on the input and output channels of the beamsplitter, respectively. A matrix R of this form corresponds to a general 2×2 unitary matrix with unit determinant.

A static squeezer is an active optical device with one input channel and one output channel. It is described by the equations

$$\begin{bmatrix} y \\ y^* \end{bmatrix} = R \begin{bmatrix} u \\ u^* \end{bmatrix}$$

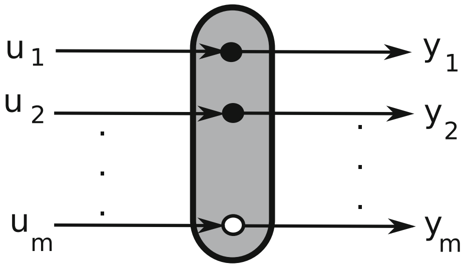


Fig. 9 Schematic diagram of a generalized active–passive cavity. The black dots represent purely active channels and the white dots represent purely passive channels. In the case that all of the channels are vector channels of the same dimension, this diagram is also used to represent a collection of generalized active–passive cavities in parallel

where

$$R = \begin{bmatrix} e^{\imath(\phi+\psi)}\cosh x & e^{\imath(\phi-\psi)}\sinh x \\ e^{\imath(\phi-\psi)}\sinh x & e^{-\imath\phi+\psi)}\cosh x \end{bmatrix}$$

where $x \in \mathbb{R}$ is the squeezing parameter and the angles ϕ and ψ correspond to phase shifts on the input and output channels of the squeezer, respectively.

Using the Shale decomposition of a Bogoliubov matrix [8, 16, 20, 34, 60], it turns out that any Bogoliubov static transformation

$$\begin{bmatrix} y \\ y^{\#} \end{bmatrix} = R \begin{bmatrix} u \\ u^{\#} \end{bmatrix}$$

can be constructed exclusively from phase shifters, beam splitters and squeezers.

Also, we consider a generalization of the multichannel cavities described by the QSDEs (73) to allow for active as well as passive input–output channels. These generalized cavities are described by the QSDEs

$$\begin{aligned} da &= \left(-\frac{\gamma}{2} + \imath\Delta\right) a dt - \sum_{i=1}^{m} \sqrt{\kappa_i} e^{-\imath\theta_i} du_i + \sum_{i=1}^{m} \sqrt{g_i} e^{\imath\phi_i} du_i^{*}; \\ dy_i &= \sqrt{\kappa_i} e^{\imath\theta_i} a dt + \sqrt{\kappa_i} e^{\imath\phi_i} a^{\#} dt + du_i, \quad i = 1, 2, \ldots, m. \end{aligned} \tag{87}$$

Here, $\gamma = \sum_{i=1}^{m} (\kappa_i - g_i)$. We will restrict attention to the case in which a channel is either purely passive corresponding to $g_i = 0$ or purely active corresponding to $\kappa_i = 0$. Such a generalized cavity is represented schematically as shown in Fig. 9.

We now present a canonical form for doubled-up complex matrices; see [20].

Theorem 13 (See [20]) *Consider a complex matrix N with the doubled-up form $N = \begin{bmatrix} N_1 & N_2 \\ N_2^{\#} & N_1^{\#} \end{bmatrix}$. Assume that all of the eigenvalues of the matrix $JN^{\dagger}JN$ are semi-simple and Ker$[JN^{\dagger}JN] =$ Ker$[N]$. Then there exist Bogoliubov matrices V, W*

and a doubled-up matrix $\hat{N} = \begin{bmatrix} \hat{N}_1 & \hat{N}_2 \\ \hat{N}_2^{\#} & \hat{N}_1^{\#} \end{bmatrix}$ *such that* $N = V\hat{N}JW^{\dagger}J$*. Here* $\hat{N}_1$ *and* $\hat{N}_2$ *are complex diagonal matrices whose diagonal elements are constructed from the eigenvalues of the matrix* $JN^{\dagger}JN$*.*

Applying this decomposition to the matrix N in a linear quantum system with (S, L, H) parameters defined as in (6), (8), the following result can be obtained. This result enables the realization of a general annihilation-creation physically realizable transfer function matrix; see [20].

Theorem 14 (See [20]) *Consider an open linear quantum system with* (S, L, H) *parameters defined as in (5), (8), QSDE description as in (9), (11) and transfer function matrix (30):*

$$\varGamma(s) = \left[I - N \left(sI + \imath JM + \frac{1}{2} JN^{\dagger}JN \right)^{-1} JN^{\dagger} \right] \begin{bmatrix} S & 0 \\ 0 & -S^{\#} \end{bmatrix}.$$

Furthermore, assume that the matrix N *in (8) satisfies the assumptions of Theorem 13 and let* $N = V\hat{N}JW^{\dagger}J$ *be the corresponding decomposition of this matrix. Then* $\varGamma(s)$ *can be factored as* $\varGamma(s) = V\hat{\varGamma}(s)JV^{\dagger}J \begin{bmatrix} S & 0 \\ 0 & -S^{\#} \end{bmatrix}$ *where*

$$\hat{\varGamma}(s) = I - N \left(sI + \imath J\hat{M} + \frac{1}{2} J\hat{N}^{\dagger}J\hat{N} \right)^{-1} J\hat{N}^{\dagger}$$

and $\hat{M} = W^{\dagger}MW$*. Furthermore,* $\hat{\varGamma}(s)$ *corresponds to an open linear quantum system with* (S, L, H) *parameters defined by* $S = I$*,*

$$H = \frac{1}{2} \begin{bmatrix} a^{\dagger} & a^{T} \end{bmatrix} \hat{M} \begin{bmatrix} a \\ a^{\#} \end{bmatrix},$$

and

$$\begin{bmatrix} L \\ L^{\#} \end{bmatrix} = \hat{N} \begin{bmatrix} a \\ a^{\#} \end{bmatrix}.$$

This open linear quantum system corresponds to QSDEs of the following form:

$$\begin{aligned} \begin{bmatrix} da(t) \\ da(t)^{\#} \end{bmatrix} &= -\left(\imath J\bar{M} + \frac{1}{2} J\bar{N}^{\dagger}J\bar{N} + \frac{1}{2} J\hat{N}^{\dagger}J\hat{N} \right) \begin{bmatrix} a(t) \\ a(t)^{\#} \end{bmatrix} dt \\ &\quad - J\bar{N}^{\dagger} \begin{bmatrix} d\mathscr{A}_{int}(t) \\ d\mathscr{A}_{int}(t)^{\#} \end{bmatrix} - J\hat{N}^{\dagger} \begin{bmatrix} d\mathscr{A}(t) \\ d\mathscr{A}(t)^{\#} \end{bmatrix}; \\ \begin{bmatrix} d\mathscr{A}^{out}(t) \\ d\mathscr{A}^{out}(t)^{\#} \end{bmatrix} &= \hat{N} \begin{bmatrix} a(t) \\ a(t)^{\#} \end{bmatrix} dt + \begin{bmatrix} d\mathscr{A}(t) \\ d\mathscr{A}(t)^{\#} \end{bmatrix}; \end{aligned}$$

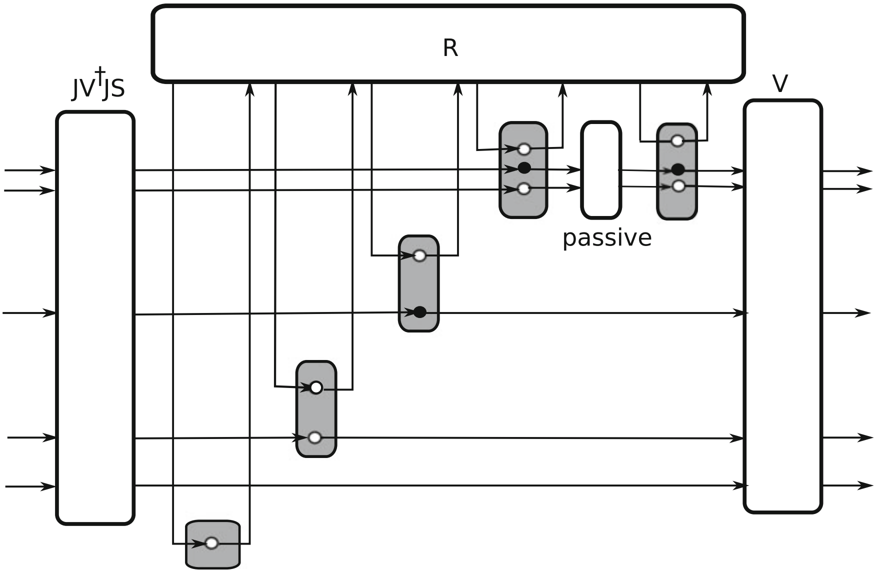


Fig. 10 Quantum optical realization of a general physically realizable transfer function matrix. In this diagram, the generalized active–passive cavity blocks represent collections of cavities in parallel

$$\begin{bmatrix} d\mathscr{A}^{out}_{int}(t) \\ d\mathscr{A}^{out}_{int}(t)^{\#} \end{bmatrix} = \bar{N} \begin{bmatrix} a(t) \\ a(t)^{\#} \end{bmatrix} dt + \begin{bmatrix} d\mathscr{A}_{int}(t) \\ d\mathscr{A}_{int}(t)^{\#} \end{bmatrix};$$
$$\begin{bmatrix} d\mathscr{A}_{int}(t) \\ d\mathscr{A}_{int}(t)^{\#} \end{bmatrix} = R \begin{bmatrix} d\mathscr{A}^{out}_{int}(t) \\ d\mathscr{A}^{out}_{int}(t)^{\#} \end{bmatrix}.$$

Here $\bar{N}$ is an arbitrary positive definite diagonal matrix, $\bar{M} = diag(D, D) + E + E^T$, D is an arbitrary positive definite diagonal matrix, and E is a matrix determined by the eigenvalues of $JN^{\dagger}JN$. Also, we define $X = 2\imath \left(J\bar{N}^{\dagger}J\right)^{-1}\left(J\hat{M} - J\bar{M}\right)\bar{N}^{-1}$. Then the feedback gain is defined by $R = (X - I)(X + I)^{-1}$. The structure of these matrices means that $\Gamma(s)$ can be realized by the quantum optical feedback network shown in Fig. 10.

Remark 1 The above theorem shows that for a general class of annihilation-creation physically realizable transfer function matrices, they can be implemented as a quantum optical network consisting of beamsplitters, phase shifters and squeezers, single channel passive optical cavities, two channel passive optical cavities, two channel optical cavities with one purely passive channel and one purely active channel and three channel optical cavities with two purely passive channels and one purely active channel. In practice, the optical channels containing purely active channels would need to be approximated by a combination of passive cavities and dynamic squeezers; e.g., see [16, 65].

4 Conclusions

In this chapter, we have considered the modelling and realization of quantum networks from a control theory point of view. The chapter has introduced various classes of quantum system models, particularly quantum linear systems. The properties of these models have then been studied by surveying a range of recent results. In particular, results on the property of physical realizability have been considered. This property characterizes when a given model corresponds to a physical quantum system obeying the laws of quantum mechanics. The chapter also surveyed results on the structure of physical quantum systems and the realization of quantum system models as quantum optical networks, with given structures.

References

1. M.A. Armen, J.K. Au, J.K. Stockton, A.C. Doherty, and H. Mabuchi. Adaptive homodyne measurement of optical phase. *Physical Review Letters*, 89(13), 2002. 133602.
2. V. P. Belavkin. Quantum stochastic calculus and quantum nonlinear filtering. *Journal of Multivariate Analysis*, 42:171–201, 1992.
3. V.P. Belavkin. On the theory of controlling observable quantum systems. *Automation and Remote Control*, 42(2):178–188, 1983.
4. V.P. Belavkin. Quantum stochastic calculus and quantum nonlinear filtering. *J. Multivariate Analysis*, 42:171–201, 1992.
5. L. Bouten, J. Stockton, A. Silberfarb, and H. Mabuchi. Scattering of polarized laser light by an atomic gas in free space: A quantum stochastic differential equation approach. *Physical Review A*, 75(025211), 2007.
6. L. Bouten, R. van Handel, and M. R. James. A discrete invitation to quantum filtering and feedback control. *SIAM Review*, 51(2):239–316, 2009.
7. L. Bouten, R. van Handel, and M.R. James. An introduction to quantum filtering. *SIAM J. Control and Optimization*, 46(6):2199–2241, 2007.
8. S. L. Braunstein. Squeezing as an irreducible resource. *PHYSICAL REVIEW A*, 71:055801, 2005.
9. A.C. Doherty, J.C. Doyle, H. Mabuchi, K. Jacobs, and S. Habib. Robust control in the quantum domain. In *Proceedings of the 39th IEEE Conference on Decision and Control*, page 949, Sydney, NSW, Australia, December 2000.
10. A.C. Doherty, S. Habib, K. Jacobs, H. Mabuchi, and S.M. Tan. Quantum feedback control and classical control theory. *Physical Review A*, 62, 2000. 012105.
11. A.C. Doherty and K. Jacobs. Feedback-control of quantum systems using continuous state-estimation. *Physical Review A*, 60:2700–2711, 1999.
12. M. F. Emzir, I. R Petersen, and M. Woolley. On physical realizability of nonlinear quantum stochastic differential equations. *Automatica*, 2018. To appear, accepted 5 April 2018.
13. L. A. Duffaut Espinosa, Z. Miao, I. R. Petersen, V. Ugrinovskii, and M. R. James. Physical realizability and preservation of commutation and anticommutation relations for n-level quantum systems. *SIAM Journal on Control and Optimization*, 54(2):632–661, 2016.
14. J. Gough and M. James. Quantum feedback networks: Hamiltonian formulation. *Communications in Mathematical Physics*, 287:1109–1132, 2009.
15. J. Gough and M. R. James. The series product and its application to quantum feedforward and feedback networks. *IEEE Transactions on Automatic Control*, 54(11):2530–2544, 2009.
16. J. E. Gough, M. R. James, and H. I. Nurdin. Squeezing components in linear quantum feedback networks. *Physical Review A*, 81:023804, 2010.

17. J. E. Gough and S. Wildfeuer. Enhancement of field squeezing using coherent feedback. *Physical Review A*, 80:042107, 2009.
18. J. E. Gough and G. Zhang. On realization theory of quantum linear systems. *Automatica*, 59:139–151, 2015.
19. J.E. Gough, R. Gohm, and M. Yanagisawa. Linear quantum feedback networks. *Physical Review A*, 78:062104, 2008.
20. S. Grivopoulos and I. R. Petersen. Linear quantum system transfer function realization using static networks for i/o processing and feedback. *SIAM Journal on Control and Optimization*, 55(5):3349–3369, 2017.
21. S. Grivopoulos and I. R. Petersen. Bilinear Hamiltonian interactions between linear quantum systems via feedback. *Automatica*, 89:103–110, 2018.
22. S. Grivopoulos, G. Zhang, I. R. Petersen, and J. E. Gough. The Kalman decomposition for linear quantum stochastic systems. In *Proceedings of the 2017 American Control Conference*, Seattle, WA, May 2017.
23. R. Hamerly and H. Mabuchi. Advantages of coherent feedback for cooling quantum oscillators. *Physical Review Letters*, 109:173602, 2012.
24. H. G. Harno and I. R. Petersen. Synthesis of linear coherent quantum control systems using a differential evolution algorithm. *IEEE Transactions on Automatic Control*, 60(3):799–805, 2015.
25. G.M. Huang, T.J. Tarn, and J.W. Clark. On the controllability of quantum-mechanical systems. *Journal of Mathematical Physics*, 24(11):2608–2618, 1983.
26. R.L. Hudson and K.R. Parthasarathy. Quantum Ito's formula and stochastic evolution. *Communications in Mathematical Physics*, 93:301–323, 1984.
27. M. R. James, H. I. Nurdin, and I. R. Petersen. H^∞ control of linear quantum stochastic systems. *IEEE Transactions on Automatic Control*, 53(8):1787–1803, 2008.
28. M.R. James and J.E. Gough. Quantum dissipative systems and feedback control design by interconnection. *IEEE Transactions on Automatic Control*, 55(8):1806 –1821, 2010.
29. J. Kerckhoff, L. Bouten, A. Silberfarb, and H. Mabuchi. Physical model of continuous two-qubit parity measurement in a cavity-QED network. *Physical Review A*, 79(2):024305, 2009.
30. J. Kerckhoff, H. I. Nurdin, D. S. Pavlichin, and H. Mabuchi. Designing quantum memories with embedded control: Photonic circuits for autonomous quantum error correction. *Phys. Rev. Lett.*, 105:040502, 2010.
31. A. Khodaparastsichani and I. R. Petersen. A modified frequency domain condition for the physical realizability of linear quantum stochastic systems. *IEEE Transactions on Automatic Control*, 63(1):277–282, 2018.
32. A. Khodaparastsichani, I. R. Petersen, and I. G. Vladimirov. Covariance dynamics and entanglement in translation invariant linear quantum stochastic networks. In *Proceedings of the 54th IEEE Conference on Decision and Control*, Osaka, Japan, December 2015.
33. A. Khodaparastsichani, I. Vladimirov, and I. R. Petersen. Decentralised coherent quantum control design for translation invariant linear quantum stochastic networks with direct coupling. In *Proceedings of 2015 Australian Control Conference*, Gold Coast, Australia, November 2015.
34. U Leonhardt and A Neumaier. Explicit effective hamiltonians for general linear quantum-optical networks. *Journal of Optics B: Quantum and Semiclassical Optics*, 6:L1–L4, 2004.
35. S. Ma, M. Woolley, and I. R. Petersen. Linear quantum systems with diagonal passive Hamiltonian and a single dissipative channel. *Systems & Control Letters*, 99:64–71, 2017.
36. S. Ma, M. Woolley, I. R. Petersen, and N. Yamamoto. Pure Gaussian quantum states from passive Hamiltonians and an active local dissipative process. In *55th IEEE Conference on Decision and Control*, Las Vegas, NV, December 2016.
37. S. Ma, M. Woolley, I. R. Petersen, and N. Yamamoto. Pure Gaussian states from quantum harmonic oscillator chains with a single local dissipative process. *Journal of Physics A*, 50(13):135301, 2017.
38. S. Ma, M. Woolley, I. R. Petersen, and N. Yamamoto. Cascade and locally dissipative realizations of linear quantum systems for pure gaussian state covariance assignment. *Automatica*, 2018. To appear, accepted 4 Nov 2017.

39. A. I. Maalouf and I. R. Petersen. Bounded real properties for a class of linear complex quantum systems. *IEEE Transactions on Automatic Control*, 56(4):786 – 801, 2011.
40. A. I. Maalouf and I. R. Petersen. Coherent H^∞ control for a class of linear complex quantum systems. *IEEE Transactions on Automatic Control*, 56(2):309–319, 2011.
41. H. Mabuchi. Experiments in real-time quantum feedback. In *Proceedings of the 41st IEEE Conference on Decision and Control*, pages 450–451, Las Vegas, Nevada, USA, December 2002.
42. H. Mabuchi. Coherent-feedback quantum control with a dynamic compensator. *Physical Review A*, 78:032323, 2008.
43. H. Mabuchi and N. Khaneja. Principles and applications of control in quantum systems. *International Journal of Robust and Nonlinear Control*, 15:647–667, 2005.
44. P. A. Meyer. *Quantum Probability for Probabilists*. Springer-Verlag, Berlin, second edition, 1995.
45. H. I. Nurdin, S. Grivopoulos, and I. R. Petersen. The transfer function of generic linear quantum stochastic systems has a pure cascade realization. *Automatica*, 69:324–333, 2016.
46. H. I. Nurdin, M. R. James, and A. C. Doherty. Network synthesis of linear dynamical quantum stochastic systems. *SIAM Journal on Control and Optimization*, 48(4):2686–2718, 2009.
47. H. I. Nurdin, M. R. James, and I. R. Petersen. Coherent quantum LQG control. *Automatica*, 45(8):1837–1846, 2009.
48. H. I. Nurdin and N. Yamamoto. *Linear Dynamical Quantum Systems: Analysis, Synthesis, and Control*. Springer, Berlin, 2017.
49. H.I. Nurdin. On synthesis of linear quantum stochastic systems by pure cascading. *IEEE Transactions on Automatic Control*, 55(10):2439 –2444, 2010.
50. H.I. Nurdin. Synthesis of linear quantum stochastic systems via quantum feedback networks. *IEEE Transactions on Automatic Control*, 55(4):1008 –1013, 2010.
51. H.I. Nurdin, I. R. Petersen, and M. R. James. On the infeasibility of entanglement generation in Gaussian quantum systems via classical control. *IEEE Transactions on Automatic Control*, 57(1):198–203, 2012. arXiv:1107.3174.
52. K.R. Parthasarathy. *An Introduction to Quantum Stochastic Calculus*. Birkhauser, Berlin, 1992.
53. I. R. Petersen. Cascade cavity realization for a class of complex transfer functions arising in coherent quantum feedback control. *Automatica*, 47(8):1757–1763, 2011.
54. I. R. Petersen. Singular perturbation approximations for a class of linear quantum systems. *IEEE Transactions on Automatic Control*, 58(1):193–198, 2013. arXiv:1107.5605.
55. I. R. Petersen. Quantum linear systems theory. *Open Automation and Control Systems Journal*, 8:67–93, 2016.
56. I. R. Petersen. Time averaged consensus in a direct coupled coherent quantum observer network. *Journal of Control Theory and Technology*, 15(3):163–176, 2017.
57. I. R. Petersen and R. Tempo. Robust control of uncertain systems: Classical results and recent developments. *Automatica*, 50:1315–1335, 2014.
58. G. Sarma, A. Silberfarb, and H. Mabuchi. Quantum stochastic calculus approach to modeling double-pass atom-field coupling. *Physical Review A*, 78:025801, 2008.
59. A. J. Shaiju and I. R. Petersen. A frequency domain condition for the physical realizability of linear quantum systems. *IEEE Transactions on Automatic Control*, 57(8):2033 – 2044, 2012.
60. D. Shale. Linear symmetries of free boson fields. *Transactions of the American Mathematical Society*, 103:149–167, 1962.
61. G. Shi, D. Dong, I. R. Petersen, and K. H. Johansson. Reaching a quantum consensus: Master equations that generate symmetrization and synchronization. *IEEE Transactions on Automatic Control*, 61(2):374–387, 2016.
62. G. Shi, S. Fu, and I. R. Petersen. Consensus of quantum networks with directed interactions: Fixed and switching structures. *IEEE Transactions on Automatic Control*, 62(4):2014–2019, 2017.
63. R. van Handel, J. K. Stockton, and H. Mabuchi. Modelling and feedback control design for quantum state preparation. *Journal of Optics B: Quantum and Semiclassical Optics*, 7(10):S179, 2005.

64. I. Vladimirov and I. R. Petersen. Physical realizability and mean square performance of translation invariant networks of interacting linear quantum stochastic systems. In *Proceedings of the 21th International Symposium on Mathematical Theory of Networks and Systems*, Groningen, The Netherlands, 2014.
65. S. L. Vuglar and I. R. Petersen. Singular perturbation approximations for general linear quantum systems. In *Proceedings of the 2012 Australian Control Conference*, Sydney, Australia, November 2012. arXiv:1208.6155.
66. S. L. Vuglar and I. R. Petersen. Quantum noises, physical realizability and coherent quantum feedback control. *IEEE Transactions on Automatic Control*, 62(2):998–1003, 2017.
67. S. Wang, H. I. Nurdin, G. Zhang, and M. R. James. Quantum optical realization of classical linear stochastic systems. *Automatica*, 49(10):3090 – 3096, 2013.
68. W.S. Warren, H. Rabitz, and M. Dahleh. Coherent control of quantum dynamics: The dream is alive. *Science*, 259:1581–1589, 1993.
69. G. Zhang, S. Grivopoulos, I. R. Petersen, and J. E. Gough. The Kalman decomposition for linear quantum systems. *IEEE Transactions on Automatic Control*, 63(2):331–346, 2018.
70. H. Zhang and H. Rabitz. Robust optimal control of quantum molecular systems in the presence of disturbances and uncertainties. *Physical Review A*, 49:2241–2254, 1994.