2018

Sparsity-Promoting Optimal Controller Design

MirSaleh Bahavarnia
Lehigh University

Sparsity-Promoting Optimal Controller Design

by

MirSaleh Bahavarnia

A Dissertation

Presented to the Graduate and Research Committee

of Lehigh University

in Candidacy for the Degree of

Doctor of Philosophy

in

Mechanical Engineering

Lehigh University

May 2018

Approved and recommended for acceptance as a dissertation in partial fulfillment of the requirements for the degree of Doctor of Philosophy.

MirSaleh Bahavarnia

Sparsity-Promoting Optimal Controller Design

Date

Prof. Nader Motee, Dissertation Director, Chair

Accepted Date

Committee Members

Prof. Eugenio Schuster

Prof. Subhrajit Bhattacharya

Prof. Frank E. Curtis

To My Beloved Mother (Mehrangiz), Father (MirAliakbar), and Sisters (Nasrin and Neda).

Acknowledgements

First of all, I would like to kindly thank my advisor, Prof. Nader Motee for his inspiring supervisions. Without his fruitful advices, I would not be able to reach insightful understanding of my research interests. He helped me to come out of my comfort zone and search for novel ideas to develop new control theoretic efficient algorithms to reach our fundamental objectives. Definitely, my research path was full of difficulties as others. However, I am proud to say that he greatly supported me through such a path to overcome those difficulties. Particularly, I should be grateful for suggesting the amazing research topic by him. Honestly, I really enjoyed throughout my Ph.D. research activities in such an amazing research area.

I am pleased to express my gratitude to the rest of my Ph.D. committee members including Prof. Eugenio Schuster, Prof. Subhrajit Bhattacharya, and Prof. Frank E. Curtis for their extremely kind supports along my general exam, Ph.D. proposal defense, and Ph.D. dissertation defense. I have learnt so many control concepts from Prof. Schuster through my Ph.D. studies. Prof. Bhattacharya's motivating comments helped me to strongly deal with research problems. I have been lucky to attend Prof. Curtis's nonlinear programming class and learn so many effective tools in nonlinear optimization from him. Regarding the academic writing/presentation skills, I should be thankful to Ms. Rita DiFiore-Czipczer and Ms. Teresa Cusumano. I am also thankful to the Department of Mechanical Engineering and Mechanics kind staff including Ms. JoAnn Casciano, Ms. Jennifer Smith, Ms. Barbara McGuire, Mr.

Muhammed Naazer, and Ms. Allison Marsteller. I am deeply grateful to Prof. Gary Harlow, Prof. John Coulter, and Prof. Donald Rockwell for their kind supports. Let me thank Ms. Bonnie Beidleman, Ms. Jeanne Ma, Ms. Clara Buie, and Ms. Olga Scarpero, for their kind helps at OISS. Additionally, I appreciate my Lehigh University teachers' kind efforts which facilitated expansion of my knowledge in various fields.

My deep appreciation goes to Prof. Mohammad Saleh Tavazoei who was my B.Sc. advisor and collaborator at Sharif University of Technology. He was the first person who made the control theory interesting for me via his insightful taught control theory courses. Also, I had a chance to start my academic research and publish my first journal paper, first conference papers, and first book chapter along with him. In addition, I should thank Prof. Mohammad Mobed and Prof. Nasser Sadati who taught me the rest of wonderful fundamental control theory courses. Also, I am grateful to all the other teachers from my B.Sc. university, high school, guidance school, and primary school who made me familiar with their fantastic taught courses.

Let me take a moment to thank the rest of my collaborators including Dr. Reza Arastoo, Dr. Mayuresh V. Kothare, Mr. Hossein K. Mousavi, Dr. Paulo Tabuada, and Dr. Christoforos Somarakis. Among those, I have been in touch with Dr. Arastoo from 2013 and we have had a great collaborative publications from 2015 till now. I had a chance to collaborate with Mr. Mousavi in one of his specific research areas. He has provided me with interesting ideas through our collaborations. I should deeply appreciate Dr. Tabuada's comprehensive comments on one of my important research items. Without his fruitful comments, we would not be able to do such a solid research work. My kind thanks go to Dr. Somarakis who presented two of our research papers in CDC 2017. He has provided me with his philosophical thoughts in doing research.

I should thank the group of my friends who have supported me along my graduate life at Lehigh University: Dr. Milad Siami, Ms. Rozhin Hajian, Dr. Alireza Yektamaram, Dr. Vahid Gholizadeh, Dr. Sadegh Bolouki, Ms. Shokoufeh Elahi,

Mr. Yousef Jalali, Dr. Omid Ahmadi, Prof. Arash Naraghi, Ms. Maral Adeli, Mr. Nasser Heydari, Ms. Elnaz Rasti, Mr. Yaser Ghaedsharaf, Ms. Shima Dezfulian, Mr. Mohammad Shahabsafa, Ms. Sajedeh Yazdanparast, Dr. Ebrahim Tahmasebi, Dr. Forough Mahmoudabadi, Mr. Afshin Oroojlooy, Ms. Somayeh Khakpash, Mr. Evan Eckersley, Mr. Daniel Beadle, Ms. Maryam Daviran, Ms. Shabnam Ghiasvand, Mr. Mohammad Javad Asadi, Mr. Ali Almasi, Mr. Daniel Loikits, Mr. Kuanchen Xiong, Mr. Nasser Vahedi, Mr. Vahid Rahmanian, Mr. Alireza Famili, Mr. Majid Jahani, Mr. Mohammad Pirhooshyaran, Mr. Soheil Sadeghi, Mr. Pedram Yousefian, Mr. Saleh Teymouri, Mr. Arash Amini, and Mr. Milad Habibi.

It is a great honor for me to thank my artist friends who helped me to tolerate the difficulties of Ph.D. studying away from my home country via their awesome artistic works: Mr. Mohammad Haghighi, Mr. Mahdi Ghorbaniantabrizi, Mr. Mostafa Saffarian, Mr. Nariman Hodjati, Mr. Mostafa Jafari, Mr. Abbas Jamshidi, Mr. Behzad Shaghaghi, Ms. Masoumeh Radjabi, Mr. Shayan Keshavarz, Mr. Alireza Shahmohammadi, Mr. Omid Sayareh, Mr. Faraz Minooei, Ms. Sepideh Raissadat, Mr. Iman Vaziri, Mr. Amir Khajehpour, Ms. Hanna Davarmanesh, Dr. Reza Rezaiesarlak, Mr. Sirvan Manhoobi, Mr. Behfar Bahadoran, and Ms. Shabnam Ahangar.

I should also thank some of my old friends from my either high school or B.Sc. degree university: Mr. Salar Jarhan, Mr. Reza Salemmilani, Mr. Saeed Saeedmonir, Mr. Yahya Alamdarimilani, Dr. Sadra Sadraddini, Dr. Alborz Alavian, Ms. Hadis Mohammadi, Mr. Salar Fattahi, Mr. Hesam Mohammadi, Ms. Sepideh Hasanmoghadam, Mr. Reza Khodaeimehr, Mr. Amirhossein Taghvaei, and Dr. Armin Zare.

Contents

List of Tables

List of Figures

Abstract

Considering the class of linear time-invariant (LTI) systems and utilizing the various mathematical tools, for diverse scenarios, we design sparsity-promoting feedback controllers while attaining a reasonable performance loss. Diverse scenarios can be classified as follows: (i) feedback controller sparsification subject to attain a similar frequency behavior for the case without/with parametric uncertainty (Chapters 2 and 4) (ii) improvement on sparsity in time domain in addition to sparsity promotion in feedback controller (Chapters 3 and 8) (iii) sparse feedback controller design for uncertain time-delay systems (Chapter 5) (iv) row-column (r, c)-sparse feedback controller design (Chapter 6) (v) feedback controller sparsification for large-scale systems (Chapters 7 and 9). Sparsity promotion in feedback controller is done via several techniques including ℓ_1-relaxation, a notion of non-fragility, and quasi-norms. Sparsity improvement in time domain is obtained via periodic time-triggered and self-triggered control. In Chapters 2, 3, 4, 5, and 6, the non-convexity arisen by Lyapunov stability condition is handled utilizing the bi-linear rank penalty technique. In Chapters 7 and 9, stability is provided by means of continuity of maximum real part of eigenvalue of the closed-loop system. In Chapter 8, stability is imposed by a performance-based condition which consists of a quadratic cost-to-go and a Lyapunov function.

Chapter 1

Introduction

1.1 Literature Review

The area of sparsity-promoting control systems has been growing rapidly in the past decade and it has been applied to various real-world applications such as formation control of autonomous vehicles, frequency synchronization in wide area control of power networks, transportation networks, mobile wireless networks, only to name a few. In several important applications, the centralized control methodologies are unable to be applied due to the lack of access to global information in subsystem level throughout the network. Such a design constraint has motivated researchers to investigate the possibility of designing near-optimal sparse feedback controllers for large-scale dynamical networks [1–35].

To generally solve the sparsity-promoting control problems, diverse methods have been proposed. In [1, 6], the authors propose an ADMM-based primal-dual iterative approach which also takes advantage of conjugate gradient method. In [36], a projection-based method is developed in which some bounds on the optimal value of the sparsity-constrained problem is presented. Motivated by linearization idea, utilizing the sequential convex programming has led to the methods proposed by

[18, 20].

All the methods mentioned so far (except the [18]), have been proposed for the continuous-time setup. However, in addition, some methods have been presented by considering the discrete-time setup which propose sparsity-promoting state feedback gain controllers [13, 18, 37–40]. In [38, 39], a decentralized state feedback controller is presented based on convex relaxations where utilizing a graph theoretic proof, an upper bound on rank of the relaxed SDP solution is derived and if such a rank is equal to 1, then the globally optimal solution can be reconstructed from the relaxed SDP solution. However, the solution obtained by such a relaxation-based method is decentralized and is not presented for general sparse controllers. But, it is spanning both finite and infinite horizon discrete-time sparsity-promoting LQR problems.

The methods presented in [13, 22] are basically classified as (convex optimization)-based methods. Another recently proposed method recast the sparsity-promoting optimal control problem as a rank-constrained optimization problem, then tries to take advantage of ADMM and Singular Value Decomposition (SVD) to deal with such a rank-constrained optimization problem [21]. Authors in [27, 28, 32, 35], instead of utilizing ADMM, have utilized bi-linear rank penalty technique to tackle the rank-constrained optimization problem. The solution proposed by such a technique, satisfies a rank inequality which features the sub-optimality of the proposed solution. Newly, in [33], the authors have novelly proposed a non-fragility based method for sparsification of large-scale feedback controllers. Also, another large-scale feedback controller sparsification is proposed by [41] which is built upon minimization of quasi-norms.

One of the newly-investigated systems in sparsity-promoting control is spatially decaying systems [7, 14, 30, 42]. In such research works, a large class of spatially decaying systems is classified where their quadratically-optimal feedback controllers inherit spatial decay property from the dynamics of the underlying system. Moreover,

a method based on q-Banach algebras is proposed where sparsity and spatial localization features of spatially decaying systems can be studied when q is chosen sufficiently small or sufficiently large. For the class of spatially distributed systems, due to specific properties of such systems, truncation-based theoretical sparsity-promoting optimal control designs are provided by [30, 42]. Obviously, for a general class of systems, it is not possible to derive and design theoretical sparsity-promoting controllers.

The concept of sparsity is not limited to the space of state feedback controllers. In other words, the concept of sparsity can be considered in the time domain as well. To be more specific, time-triggered, self-triggered, or event-triggered methods can be seen as methods in which sparsity in time is promoted in the sense that number of triggering times is supposed to be as few as possible. To address the research works in such an area, we suggest to take a look at [26, 34, 43–65].

1.2 Main Contributions

1.2.1 Dense Output Feedback Controller Sparsification while Preserving its Frequency Characteristics

The dense output feedback controller sparsification is investigated while the frequency characteristics of designed closed-loop system remains similar to that of the system controlled with dense one. Considering a well-performing pre-designed dense controller and utilizing the concept of $\mathcal{H}_2/\mathcal{H}_\infty$ control, a rank constrained optimization problem is developed which has the capability of transferring such a dense controller to a sparse one while preserving the frequency characteristics with reasonable tolerance. In our proposed method, sparsification leads to less number of communication links and $\mathcal{H}_2/\mathcal{H}_\infty$ minimization guarantees the preservation of frequency characteristics. Finally, the effectiveness of our proposed method is evaluated by testing it on synchronous generators with sparse interconnection topology, network with unstable

nodes, and mass-spring system.

1.2.2 Periodic Time-Triggered Sparse Linear Quadratic Controller Design

The periodic time-triggered sparse Linear Quadratic Controller (LQC) design is investigated for the class of Linear Time Invariant (LTI) systems. Given a time period and keeping the control input fixed during such a time period, an optimization problem is formulated in which the objective function consists of a quadratic performance term along with an ℓ_0-regularization term. Recasting such an optimization problem as a rank-constrained optimization problem and utilizing the weighted ℓ_1-relaxation enable us to apply so-called bi-linear rank penalty technique to design periodic time-triggered sparse LQC. Employing the various test cases and running our proposed algorithm for different values of time period, performance/sparsity trade-off curves are visualized which suggest a helpful criterion to choose the time period in a way that the desired balance between controller sparsity and rate of periodic triggering is made.

1.2.3 Feedback Controller Sparsification Under Parametric Uncertainties

We consider the problem of output feedback controller sparsification for systems with parametric uncertainties. The performance of a centralized controller deteriorates as a result of the sparsification process. We develop an optimization scheme that minimizes this deterioration, while promoting sparsity pattern of the feedback gain. In order to improve temporal proximity of an existing closed-loop system and its sparsified counterpart, we also incorporate an additional constraint into the problem

formulation so as to bound the variation in the system output pre and post sparsification. We also show that the resulting non-convex optimization problem can equivalently be reformulated into a rank-constrained optimization problem. We then formulate a minimization problem along with an algorithm to obtain a sub-optimal solution via the bi-linear rank penalty technique. Finally, a sub-optimal sparse controller design for IEEE 39-bus New England power network is utilized to showcase the effectiveness of our proposed method.

1.2.4 Sparse Memoryless LQR Design for Uncertain Linear Time-Delay Systems

The sparse memoryless LQR design problem is formulated for uncertain linear time-delay systems. In such a problem, the goal is to minimize a quadratic cost supplemented by sparsity-promoting term (weighted-ℓ_1 in our case) subject to stability of closed-loop system under norm-bounded uncertainty. It is shown that such an optimization problem can be reformulated as a rank-constrained optimization problem which consists of convex constraints except one rank constraint. Utilizing the bi-linear rank penalty technique, the sparse memoryless LQR is designed. Numerous numerical results depict that there exists a trade-off between time-delay and sparsification quality. In addition, the larger time-delay, the poorer performance-sparsity trade-off is observed.

1.2.5 Row-Column Sparse Linear Quadratic Controller Design via Bi-Linear Rank Penalty Technique and Non-Fragility Notion

We consider the problem of row-column sparse linear quadratic controller (LQC) design. An optimization problem is formulated in which the quadratic performance

loss is minimized subject to satisfaction of $m + n$ sparsity constraints to obtain the row-column (r, c)-sparse LQC design where m and n refer to the number of inputs and states, respectively and r/c represent the maximum allowed density level for each row/column of controller. It is expressed that the obtained non-convex optimization problem can equivalently be reformulated as a rank-constrained problem with $m+n+1$ rank constraints. After applying the non-fragility notion provided by [33] to such a rank-constrained problem, bi-linear rank penalty technique is deployed to find a sub-optimal row-column (r, c)-sparse LQC design which fulfills the rank constraint with desired tolerance. At last, to verify our proposed algorithm, given a randomly generated system, a sub-optimal row-column (r, c)-sparse LQC design is proposed and subsequently, the fundamental trade-off between r/c and quadratic performance loss is visualized.

1.2.6 State Feedback Controller Sparsification via Non-Fragility Notion

We introduce a notion of non-fragility for a state feedback controller which stabilizes a linear time-invariant (LTI) system. The lower and upper bounds on such an introduced non-fragility are derived. On the basis of such derived bounds on non-fragility, a sparsification procedure is developed to obtain sparsified state feedback controllers out of a given stabilizing state feedback controller. Investigating the extensive numerical simulations, it is observed that the proposed method is capable of being applied to large-scale systems consisting of thousands of states. As further, as illustrated via case studies, the (non-fragilty)-based sparsification procedure can outperform a well-respected existing method in the literature, in terms of sparsity-performance trade-off behavior. Also, considering a set of sparse stabilizing state feedback controllers and applying the (non-fragilty)-based sparsification procedure, a trade-off between upper

bound on non-fragility and sparsity level of such state feedback controllers is visualized. Moreover, two greedy algorithms are proposed to obtain a set of sparse state feedback controllers out of a given stabilizing state feedback controller.

1.2.7 Improving Sparsity in Time and Space via Self-Triggered Sparse Optimal Controllers

The optimal control of linear time-invariant (LTI) systems via self-triggered sparse optimal control (SSOC) laws is considered. The control objective is to design an optimal control law which stabilizes the LTI system for all initial conditions, requires less sensing, minimizes communication requirements among the subsystems, minimizes the number of active actuators, and provides guaranteed closed-loop performance bounds. To achieve such control objectives, a sequence of ℓ_0-regularized linear-quadratic optimal control problems is formulated, wherein the objective is to optimize a cost function which involves three terms: one for maximizing the inter-execution time, another for minimizing the number of nonzero elements of the state feedback gain, and the last for minimizing the number of active actuators. Deriving the lower bounds on inter-execution times, we propose a scheme to solve this problem. Such a scheme consists of two main levels: (i) A nonlinear optimization is solved for inter-execution time while the feedback gain is kept fixed. (ii) An ℓ_1-relaxed semi-definite program (SDP) is solved for feedback gain while the inter-execution time is kept fixed. We show that the proposed SSOC laws are feasible and results in a stabilizing sequence of sparse optimal controllers. Additionally, we prove that the performance of the resulting closed-loop system does not exceed a pre-specified performance bound. Due to numerical verification of our proposed method on spatially distributed systems, the sparsity in time and space is improved compared to the periodic time-triggered LQR design. Moreover, a tradeoff between pre-specified performance bound and sparsity in time/space is observed. Furthermore, the effect

of spatially decaying rate on sparsification process is visualized.

1.2.8 Feedback Controller Sparsification via Quasi-Norms

We utilize the $q \in (0, 1)$ quasi-norms to sparsify a given well-performing feedback controller which stabilizes a linear time-invariant (LTI) system. To achieve such a goal, we firstly formulate an unconstrained optimization problem which incorporates two terms: (i) The Frobenius norm of difference of the given feedback controller and the one to be designed; (ii) The $q \in (0, 1)$ quasi-norm of the feedback controller to be designed. The former term heuristically features the closed-loop stability and the latter term promotes the sparsity. Next, obtaining an analytic threshold for the sparsity-promoting parameter, the analytic solution of the formulated unconstrained optimization problem is expressed which is basically the designed sparse feedback controller. Throughout the numerical simulations, it is observed that in some cases, our proposed method can outperform the well-known truncation operator which appears in cardinality minimization problems. In other words, sometimes, the $q \in (0, 1)$ quasi-norms can be more effective than ℓ_0 sparsity measure. As another observation, when q decreases, the sparsity-performance balance is significantly improved. Furthermore, our proposed method is interestingly capable of being applied to the large-scale systems with thousands of states.

1.3 Subject-Based Classification of Chapters

1.3.1 Spatial Sparsity

Chapters 2, 3, 4, 5, 6, 7, 8, and 9.

1.3.2 Temporal Sparsity

Chapters 3 and 8.

1.3.3 Regulation

Chapters 3, 5, 6, and 8.

1.3.4 Disturbance Attenuation

Chapters 2, 4, 7, and 9.

1.3.5 Similar Frequency Behavior

Chapters 2 and 4.

1.3.6 Quadratic Performance Loss Minimization

Chapters 2, 3, 4, 5, 6, and 8.

1.3.7 Design Under Parametric Uncertainty

Chapters 4 and 5.

1.3.8 Design for Time-Delay Systems

Chapter 5.

1.3.9 Feedback Sparsification

Chapters 2, 4, 7, 8, and 9.

1.3.10 Sparse Feedback Design

Chapters 3, 5, and 6.

1.3.11 State Feedback

Chapters 2, 3, 4, 5, 6, 7, 8, and 9.

1.3.12 Output Feedback

Chapters 2, 4, and 9.

1.3.13 Bi-Linear Rank Penalty Technique

Chapters 2, 3, 4, 5, and 6.

1.3.14 Row-Column Sparsity

Chapter 6.

1.3.15 Non-Fragility

Chapters 6 and 7.

1.3.16 Quasi-Norm Minimization

Chapter 9.

1.3.17 ℓ_1 Relaxation

Chapters 2 , 3, 4, 5, and 8.

1.3.18 $\mathcal{H}_2$ Minimization

Chapters 2, 3, 4, 5, and 6.

1.3.19 $\mathcal{H}_\infty$ Minimization

Chapters 2 and 4.

1.3.20 Convex Optimization Techniques

Chapters 2, 3, 4, 5, 6, and 8.

1.3.21 Non-Convex Optimization Techniques

Chapters 7, 8, and 9.

Chapter 2

Dense Output Feedback Controller Sparsification while Preserving its Frequency Characteristics

2.1 Introduction

Although numerous works have been done in the area of distributed controller design [42, 66] , the capability of efficiently solving the general problem via a systematic approach is far away from the desirable point. For some classes of systems such as spatially invariant systems and spatially decaying systems useful results on the structure of the solution space have been derived [42, 67]. Furthermore, several other design frameworks, each with their specific imperfections, have also been proposed to design sparse/structured controllers for the continuous/discrete time linear time invariant systems both in time and frequency domain [2, 18, 19, 21, 27, 28, 39, 40, 68].

The common approach in the synthesis of distributed controllers is to minimize performance loss subject to stability guarantee and minimization of number of needed communication links among controller nodes. Unlike such an approach and similar to

methodology presented by [21, 27, 40], our proposed framework in this chapter is based on the assumption that there already exists a well-performing dense controller such as conventional centralized LQR method. We aim to synthesize a sparse controller which preserves the performance characteristics as much as possible close to that of the pre-assumed dense controller. Such a performance preservation is achieved via adopting the concepts from mixed $\mathcal{H}_2/\mathcal{H}_\infty$ control [69–71] to not only obtain minimum gap in the frequency characteristics of the closed-loop transfer functions, but also consider the difference between the characteristics of the control signals generated by both dense and sparse controllers.

In particular, firstly, it is assumed a well-performing pre-designed output feedback controller is given and then considering the bounds on difference of output and control input signals and their corresponding well-performing pre-designed signals, (i.e., $\mathcal{H}_\infty$-norm of difference between closed-loop constructed by our proposed output feedback controller and the one constructed by such a well-performing pre-designed output feedback controller), respectively, the $\mathcal{H}_2$-norm of difference between closed-loop constructed by our proposed output feedback controller and the one constructed by such a well-performing pre-designed output feedback controller is minimized while minimizing the number of communication links of our proposed output feedback controller.

It is shown that our proposed synthesis framework can equivalently be reformulated as a fixed rank-constrained optimization where all non-convexities are collected into a rank constraint.

This chapter is organized as follows: After expressing our used mathematical notations in Section 2.2, Section 2.3 is dedicated to formulate the problem which is supposed to be solved. In Section 2.4, it is explained how our formulated problem can equivalently be reformulated as an optimization problem consisting of several linear matrix inequalities and a rank constraint. Section 2.5 provides some visions to our

chosen sparsity measure and the bi-linear rank penalty technique algorithm which is chosen to come up with the corresponding rank-constrained optimization problem. Our sparsification method is verified via various numerical simulations presented in Section 2.6. Finally, Section 2.7 reveals some discussions and conclusions.

2.2 Mathematical Notations

Throughout the chapter, the following notations are adopted: The space of n by m matrices with real elements is indicated by $\mathbb{R}^{n\times m}$. The n by n identity matrix is denoted by I_n. Operators $\mathbf{Tr}(.)$ and $\mathbf{rank}(.)$ denote the trace and rank of the matrix operands. The transpose operator is denoted by $(.)^T$. The matrix element-wise product, i.e., Hadamard product is represented by $\circ$. A matrix is said to be Hurwitz if all its eigenvalues lie in the open left half of the complex plane. $\|.\|_0$ represents the cardinality of a vector/matrix, while $\|.\|_1$, $\|.\|_2$, and $\|.\|_F$ denote ℓ_1, ℓ_2, and Frobenius norm operators, respectively. Also, the norm $\|.\|_{\mathcal{L}_2(\mathbb{R}^n)}$ is defined by

$$\|x\|^2_{\mathcal{L}_2(\mathbb{R}^n)} := \int_0^\infty {\|x(t)\|_2}^2 \, dt.$$

A real symmetric matrix is said to be positive definite (semi-definite) if all its eigenvalues are positive (non-negative). $\mathbb{S}^n_{++}$ ($\mathbb{S}^n_{+}$) denotes the space of positive definite (positive semi-definite) real symmetric matrices, and the notation $X \succ Y$ ($X \succeq Y$) means $X - Y \in \mathbb{S}^n_{++}$ ($X - Y \in \mathbb{S}^n_{+}$). The i^{th} largest singular value of a matrix is denoted by $\sigma_i(.)$.

2.3 Problem Formulation

Let a linear time invariant (LTI) continuous-time system be given by its state space realization

$$\begin{cases} \dot{x}(t) = Ax(t) + B_1 u(t) + B_2 d(t) \\ y(t) = C_1 x(t) \end{cases},$$

where $A \in \mathbb{R}^{n\times n}$, $B_1 \in \mathbb{R}^{n\times m}$, $B_2 \in \mathbb{R}^{n\times r}$, and $C_1 \in \mathbb{R}^{p\times n}$. It is assumed that the pair (A, B_1) is controllable and (A, C_1) is detectable.

Our goal is to design a static feedback controller

$$u(t) = KC_2 x(t), \quad K \in \mathcal{K}, \quad C_2 \in \mathbb{R}^{q\times n},$$

which achieves minimum performance difference comparing to a reference well-performing pre-designed dense controller, namely $\hat{K}$, while minimizing the number of non-zero elements of the controller matrix. We, also, desire that controller to be contained in a set of admissible feedback gains with previously specified structure, denoted by $\mathcal{K}$. In this chapter, we just consider the case where the set $\mathcal{K}$ is convex, since it reduces the complexity of the problem, and, more importantly, it covers a wide range of practical constraints on the controller that should be considered in the synthesis of the controller. For instance, in some real-world applications, it is not practically feasible to construct a link between special nodes; such limitations are translated to the convex constraints for which the corresponding element of the controller matrix should be equal to zero which can be applied via element-wise Hadamard product. Other practical limitations such as upper bounds on the elements of the controller matrix, (e.g., technological impositions), can also be addressed by convex constraints on controller matrix K.

Additionally, it is preferred the energy level of the input/output signals, generated

by the designed sparse controller, to be in the close neighbourhood of that of the input/output, produced by the original dense controller when an input signal $d(t)$ with bounded energy is applied to the closed-loop plant. Representing the closed-loop systems controlled by the controllers K and $\hat{K}$ by the state space realizations $\mathcal{S}$ and $\hat{\mathcal{S}}$, respectively, the search for sparse controller K can be formulated as the following optimization problem:

$$\begin{aligned}
\underset{K,\epsilon_0,\epsilon_1}{\text{minimize}} \quad & \epsilon_0 + \lambda_1\epsilon_1 + \lambda_2\|K\|_0 && \text{(2.1a)}\\
\text{subject to:} \quad & K \in \mathcal{K}, && \text{(2.1b)}\\
& A + B_1KC_2 : \text{ Hurwitz}, && \text{(2.1c)}\\
& \|\mathcal{S} - \hat{\mathcal{S}}\|_{\mathcal{H}_2}^2 \leq \epsilon_0, && \text{(2.1d)}\\
& \|y_{\mathcal{S}} - y_{\hat{\mathcal{S}}}\|_{\mathcal{L}_2(\mathbb{R}^p)} < \epsilon_1\|d\|_{\mathcal{L}_2(\mathbb{R}^r)}, && \text{(2.1e)}
\end{aligned}$$

where $\|.\|_{\mathcal{H}_2}$ and $\|.\|_{\mathcal{H}_\infty}$ are $\mathcal{H}_2$ and $\mathcal{H}_\infty$ norms, respectively, and λ_1 and λ_2 are regularization parameters. Moreover, positive constants ϵ_0 and ϵ_1 are upper bounds on $\mathcal{H}_2$ and $\mathcal{H}_\infty$ norms of difference system $\mathcal{S} - \hat{\mathcal{S}}$.

Remark 1. *In case of $C_2 = C_1 \neq I$, the controller K would be an output feedback controller.*

It is worth noting that the term appeared on left hand side of inequality (2.1d) can be simplified into the $\mathcal{H}_2$ norm squared of an augmented system, namely $\bar{\mathcal{S}}$, constructed by the following state space realization matrices:

$$\bar{A} = \begin{bmatrix} A + B_1KC_2 & 0 \\ 0 & A + B_1\hat{K}C_2 \end{bmatrix},\ \bar{B} = \begin{bmatrix} B_2^T & B_2^T \end{bmatrix}^T,\ \bar{C} = \begin{bmatrix} C_1 & -C_1 \end{bmatrix}.$$

Furthermore, the constraint (2.1e) can equivalently be cast by enforcing bounds on the $\mathcal{H}_\infty$ norm of the closed-loop transfer functions from $d(t)$ to $y(t)$. Then, problem

(2.1) can be reformulated as follows:

$$\begin{aligned} \underset{K,\epsilon_0,\epsilon_1}{\text{minimize}} \quad & \epsilon_0 + \lambda_1\epsilon_1 + \lambda_2\|K\|_0 \\ \text{subject to:} \quad & K \in \mathcal{K}, \\ & A + B_1KC_2 : \text{ Hurwitz}, \\ & \|\bar{C}(sI - \bar{A})^{-1}\bar{B}\|_{\mathcal{H}_2}^2 \leq \epsilon_0, \\ & \|\bar{C}(sI - \bar{A})^{-1}\bar{B}\|_{\mathcal{H}_\infty} < \epsilon_1. \end{aligned} \tag{2.2}$$

In problem (2.2), the terms ϵ_0 and ϵ_1 in the objective function capture the gap between the frequency response of the systems in terms of $\mathcal{H}_2$ and $\mathcal{H}_\infty$ norms, respectively. Hence, it makes it possible to identify another stable network with sparser communication structure and approximately the same frequency characteristics. Unlike the design of sparse LQR controllers, introduced by [68], taking such an approach in the design of the controllers with sparse structures has the capability of exploiting the deliverable merits in various controller synthesis strategies.

Next, it is described how to formulate problem (2.2) as an optimization problem with linear/bi-linear matrix inequality/equality constraints. Then, it is shown how all nonlinear constraints can be summarized in a fixed rank constraint.

2.4 Fixed Rank Optimization Reformulation

In this section, some lemmas are expressed which help us cast the constraints of the optimization problem as a fixed rank constraint along with some linear matrix inequalities.

Lemma 1 ([21])**.** *Assuming $\mathcal{P}$ is a stable Linear Time Invariant system with realization matrices $(\mathcal{A}, \mathcal{B}, \mathcal{C})$, where $\mathcal{A} \in \mathbb{R}^{n\times n}$, $\mathcal{B} \in \mathbb{R}^{n\times m}$, and $\mathcal{C} \in \mathbb{R}^{p\times n}$, and the pair $(\mathcal{A}, \mathcal{B})$ is controllable; then, $\|\mathcal{P}\|_{\mathcal{H}_2}^2 \leq \gamma$ if and only if there exists a positive definite*

matrix $\mathcal{X} \succ 0$ *such that*

$$\mathbf{Tr}(\mathcal{C}\mathcal{X}\mathcal{C}^T) \leq \gamma,\ \mathcal{Y} + \mathcal{Y}^T + \mathcal{B}\mathcal{B}^T \preceq 0,\ \mathbf{rank} \begin{bmatrix} \mathcal{X} & \mathcal{Y} \\ I_n & \mathcal{A}^T \end{bmatrix} = n.$$

The previous lemma helps cast the $\mathcal{H}_2$-optimal sparsification problem as a rank-constrained optimization problem where all nonlinear constraints are lumped into a fixed rank constraint. Several solving algorithms have been proposed to efficiently solve rank-constrained optimization problems [72–74]. Hence, we aim to make such algorithms applicable in solving our problem by collecting various forms of non-convex/combinatorial constraints into a rank constraint.

Similar to the rank-constrained reformulation of the $\mathcal{H}_2$ problem, it is proved that the $\mathcal{H}_\infty$ constraints of the problem (2.2) can also be cast as a finite set of rank-constrained LMI's. Next lemma helps us to accommodate the $\mathcal{H}_\infty$ constraints in the framework of rank-constrained optimizations.

Lemma 2 ([21])**.** *Given* $\mathcal{P}$ *is a Linear Time Invariant system with realization matrices* $(\mathcal{A}, \mathcal{B}, \mathcal{C})$*, where* $\mathcal{A} \in \mathbb{R}^{n\times n}$*,* $\mathcal{B} \in \mathbb{R}^{n\times m}$*, and* $\mathcal{C} \in \mathbb{R}^{p\times n}$*, the matrix* $\mathcal{A}$ *is Hurwitz, and* $\|\mathcal{P}\|_{\mathcal{H}_\infty} < \gamma$ *if and only if there exists a positive definite matrix* $\mathcal{X} \succ 0$ *such that*

$$\begin{bmatrix} \mathcal{Y} + \mathcal{Y}^T + \mathcal{C}^T\mathcal{C} & \mathcal{X}\mathcal{B} \\ \mathcal{B}^T\mathcal{X} & -\gamma^2 I_m \end{bmatrix} \prec 0,\ \mathbf{rank} \begin{bmatrix} \mathcal{X} & \mathcal{Y} \\ I_n & \mathcal{A} \end{bmatrix} = n.$$

Consequently, we can reformulate the problem (2.2) as a rank-constrained problem, as explained in the following.

Theorem 3 ([21]). *The $\mathcal{H}_2/\mathcal{H}_\infty$ problem (2.2) is equivalent to the following rank-constrained optimization problem:*

$$\underset{K,\epsilon_0,\epsilon_1,\Phi}{\textit{minimize}} \quad \epsilon_0 + \lambda_1\epsilon_1 + \lambda_2\|K\|_0 \tag{2.3a}$$

$$\textit{subject to:} \quad K \in \mathcal{K}, \tag{2.3b}$$

$$X_i \succ 0, \quad i = 1, 2, \tag{2.3c}$$

$$\begin{bmatrix} M_1 & X_1\bar{C}^T & \bar{B} \\ \bar{C}X_1 & -\epsilon_1 I_p & 0 \\ \bar{B}^T & 0 & -\epsilon_1 I_r \end{bmatrix} \prec 0, \tag{2.3d}$$

$$M_2 + \bar{B}\bar{B}^T \preceq 0, \tag{2.3e}$$

$$\mathbf{Tr}(\bar{C}X_2\bar{C}^T) \leq \epsilon_0, \tag{2.3f}$$

$$\mathbf{rank}(\Phi) = 2n, \tag{2.3g}$$

where

$$\Phi = \begin{bmatrix} I & \bar{A}^T \\ X_1 & Y_1 \\ X_2 & Y_2 \end{bmatrix}, \tag{2.4a}$$

$$M_1 = X_1A_o^T + Y_1B_K^T + A_oX_1 + B_K{Y_1}^T, \tag{2.4b}$$

$$M_2 = X_2A_o^T + Y_2B_K^T + A_oX_2 + B_K{Y_2}^T, \tag{2.4c}$$

$$A_o = \begin{bmatrix} A & 0 \\ 0 & A + B_1\hat{K}C_2 \end{bmatrix}, \tag{2.4d}$$

$$B_K = \begin{bmatrix} B_1^T & 0 \end{bmatrix}^T, \tag{2.4e}$$

$$C_K = \begin{bmatrix} C_2 & 0 \end{bmatrix}. \tag{2.4f}$$

2.5 The Choice of the Sparsity Measure and a Tractable Design Protocol

2.5.1 The Choice of the Sparsity Measure

There are quite a number of sparsity measures of mostly used in diverse areas of science. Among the functions used to measure the sparsity of matrices, ℓ_1 norm and its weighted version, as convex relaxations of the ℓ_0 measure, [75] and the references within, are definitely the most common ones and have been utilized in numerous applications [18, 68]. Non-convex surrogates for the cardinality function, such as ℓ_q measure for $q \in (0, 1)$, have also received an increasing attention in the literature, recently [7, 14]. However, since adopting weighted ℓ_1 norm in optimization problems does not cause numerical issues, which usually occur in ℓ_q and ℓ_0 measure minimization problems due to their non-convex and combinatorial natures, respectively, we choose to employ weighted ℓ_1, as the measure of the sparsity of the controller matrix in the current work.

2.5.2 Bi-Linear Rank Penalty Technique

The choice of weighted ℓ_1 norm notably reduces the complexity of our problem, since the norm is a convex function and, as a result, the only arising non-convexity in problem (2.3) becomes the rank constraint (2.3g). However, the existence of the rank constraint still makes our optimization problem computationally expensive. Although an efficient systematic algorithm to solve rank-constrained problem has not been developed yet, there exists a set of optimization protocols which have the capability of solving special types of rank-constrained optimization problems by achieving sub-optimal solutions. In [21], authors have proposed to utilize the Alternating Direction Method of Multipliers (ADMM), originally developed in 1970, to solve a

rank-constrained optimization problem. The method has been proved to be useful in determining the optimal solution of large-scale optimization problems [76]; however, its convergence has not been proved for non-convex problems.

In this chapter, instead of ADMM, we will take advantage of bi-linear rank penalty technique used by [27, 28]. Before presenting the rank penalty technique, it is important to note that the rank constraint in the optimization problem (2.3g), can equivalently be replaced by $\mathbf{rank}(\Psi) = 2n$, where Ψ is symmetric square matrix and constructed as follows:

$$\Psi = \begin{bmatrix} X_2 & Y_2 & X_1 & I \\ Y_2^T & - & Y_1^T & KC_K \\ X_1 & Y_1 & - & - \\ I & (KC_K)^T & - & - \end{bmatrix}, \tag{2.5}$$

where the elements with no specific significance are depicted by "-". As discussed in [21, 27, 28], the rank constraint on the matrix Ψ can be relaxed by replacing it with a positive semi-definite constraint, i.e., $\Psi \succeq 0$, due to the assumption $X_1 \in \mathbb{S}^{2n}_{++}$.

The next corollary is obtained immediately.

Corollary 4. *The problem (2.3) can be cast as the following problem:*

$$\begin{aligned}
\underset{K,\epsilon_0,\epsilon_1,\Psi}{\textit{minimize}} \quad & \epsilon_0 + \lambda_1\epsilon_1 + \lambda_2\|K\|_0 && \text{(2.6a)}\\
\textit{subject to:} \quad & K \in \mathcal{K}, && \text{(2.6b)}\\
& X_i \succ 0, \quad i = 1, 2, && \text{(2.6c)}\\
& \begin{bmatrix} M_1 & X_1\bar{C}^T & \bar{B} \\ \bar{C}X_1 & -\epsilon_1 I_p & 0 \\ \bar{B}^T & 0 & -\epsilon_1 I_r \end{bmatrix} \prec 0, && \text{(2.6d)}\\
& M_2 + \bar{B}\bar{B}^T \preceq 0, && \text{(2.6e)}\\
& \mathbf{Tr}(\bar{C}X_2\bar{C}^T) \leq \epsilon_0, && \text{(2.6f)}\\
& \mathbf{rank}(\Psi) = 2n. && \text{(2.6g)}
\end{aligned}$$

As for the sparsity-promoting term of the objective function, since the ℓ_0 measure is an integer-valued function, utilizing it in our formulation brings the complications of combinatorial optimization. In order to reduce the complexity of sparse vector/matrix recovery problems, the ℓ_1 norm and its weighted version are utilized which are most useful convex surrogates of the ℓ_0 measure. Then, we will have

$$\begin{aligned}
\underset{K,\epsilon_0,\epsilon_1,\Psi}{\text{minimize}} \quad & \epsilon_0 + \lambda_1\epsilon_1 + \lambda_2\|W \circ K\|_1 && \text{(2.7)}\\
\text{subject to:} \quad & (2.6b) - (2.6g), \\
& (2.4b) - (2.4f).
\end{aligned}$$

where the weight matrix $W = [W_{ij}] \in \mathbb{R}^{m\times q}$ is element-wise positive and chosen according to the objectives of the problem.

The convex relaxation of the sparsity-promoting term in the cost function of (2.7) leaves us with an optimization problem in which non-convexity only arises in the

form of a rank constraint, i.e., $\mathbf{rank}(\Psi) = 2n$. It is known that existence of the rank constraint still causes our optimization problem to become NP-hard. Therefore, we propose a technique, which is built upon the method proposed in [27, 28], to solve the rank constraint optimization problem. Fundamentally, this method is based on substituting the rank constraint on the symmetric matrix Ψ with a positive semi-definite constraint while introducing extra convex constraints along with a bi-linear term to the cost function. Since the resulting optimization is all convex except for the auxiliary bi-linear term in the objective function, it can iteratively be solved [77, 78].

Definition 1. *For a given $\epsilon > 0$ and matrix X, we say that rank of X is k with tolerance ϵ, and it is denoted by $\mathbf{rank}(X;\epsilon)$, if exactly k singular values of X are larger than or equal to ϵ.*

Theorem 5 ([27])**.** *Let us consider the rank-constrained optimization problem (2.7) and define the following auxiliary optimization problem:*

$$\begin{aligned} \underset{K,\epsilon_0,\epsilon_1,\Psi,Y}{\text{minimize}} \quad & \epsilon_0 + \lambda_1\epsilon_1 + \lambda_2\|W \circ K\|_1 + \nu\mathbf{Tr}(Y\Psi) \\ \text{subject to:} \quad & (2.6b) - (2.6f), \\ & (2.4b) - (2.4f), \\ & 0 \preceq Y \preceq I_{6n+m}, \\ & \mathbf{Tr}(Y) = 4n + m, \\ & \Psi \succeq 0, \end{aligned} \tag{2.8}$$

in which $\lambda_1, \lambda_2, \nu > 0$ and the element-wise positive matrix W are some given design parameters. If problem (2.7) is feasible, then there exists a constant $\eta > 0$ for which the optimal solution Ψ from solving (2.8) satisfies

$$\mathbf{rank}(\Psi; \eta\nu^{-1}) \;\leq\; 2n, \tag{2.9}$$

i.e., rank of Ψ is less than or equal to $2n$ with tolerance threshold $\eta\nu^{-1}$ according to Definition 1.

Remark 2. *It should be reminded that according to (2.5) and the specific structure of matrix Ψ, it is always true that* $\mathbf{rank}(\Psi) \geq 2n$. *Keeping this in mind and considering the inequality (2.9), under the following condition:*

$$\sigma_{2n}\big(\Psi^*(\nu)\big) \geq \eta\nu^{-1}, \tag{2.10}$$

the rank equality constraint with tolerance $\eta\nu^{-1}$ gets satisfied, i.e., $\mathbf{rank}(\Psi, \eta\nu^{-1}) = 2n$. *In fact, the condition (2.10) is equivalent to the inequality* $\mathbf{rank}(\Psi; \eta\nu^{-1}) \geq 2n$.

As a result of the previous theorem, we can now solve the optimization problem (2.8) for an appropriately-chosen parameter ν to obtain a sub-optimal solution to the problem (2.7). For the sake of simplicity in our notations, the stack of all optimization variables excluding variable Y is denoted by Z. The optimization problem (2.8) can be rewritten as

$$\begin{aligned} &\underset{Z,Y}{\text{minimize}} \quad \mathcal{F}(Z,Y) \\ &\text{subject to:} \quad Z \in \mathcal{C}_z, \quad Y \in \mathcal{C}_y, \end{aligned}$$

where $\mathcal{C}_z$ is the convex set defined by the constraints (2.6b)-(2.6f), (2.4b)-(2.4f), along with $\Psi \succeq 0$, and the convex set $\mathcal{C}_y$ is generated by $\mathbf{Tr}(Y) = 4n + m$ and $0 \preceq Y \preceq I_{6n+m}$. Note that $\mathcal{F}(Z,Y)$ represents the bi-linear objective function in the minimization problem (2.8). The above reformulation allows us to carry out this problem by iteratively optimizing the objective function for Z and Y. As a result, the main steps of this iterative method can be divided into two sub-problems

1. Z-minimization step,

2. Y-minimization step.

As both Z-minimization and Y-minimization steps are convex optimizations, they can be performed in a computationally efficient manner. Furthermore, for the Y-minimization, there also exists analytic solution, stated in the next theorem.

Theorem 6 ([27]). *The optimal solution to the* Y-minimization *step is given by*

$$Y^* = I_{6n+m} - \sum_{i=1}^{2n} u_i {u_i}^T, \tag{2.11}$$

where vectors u_i *for* $i = 1, \ldots, 2n$ *are the singular vectors corresponding to the* $2n$ *larger singular values of* Ψ.

2.5.3 Summary of The Approximation Algorithm

We utilize the following sequence of iterations to obtain the minimizer of the constrained problem (2.8). First, we solve the Z-minimization and Y-minimization subproblems:

$$Z^{(k+1)} = \arg\underset{Z \in \mathcal{C}_z}{\text{minimize}} \;\; \mathcal{F}(Z, Y^{(k)}), \tag{2.12}$$

$$Y^{(k+1)} = I_{6n+m} - \sum_{i=1}^{2n} u_i^{(k+1)} {u_i^{(k+1)}}^T, \tag{2.13}$$

where $\Psi^{(k+1)} = \sum_{i=1}^{6n+m} \sigma_i^{(k+1)} u_i^{(k+1)} {u_i^{(k+1)}}^T$ is the singular value decomposition of $\Psi^{(k+1)}$. The stopping criterion is established by $\varepsilon^{(k+1)} \leq \varepsilon^*$, where ε^* is the given desired precision, with the following update law:

$$\varepsilon^{(k+1)} = \frac{\|K^{(k+1)} - K^{(k)}\|_2}{\|K^{(k+1)}\|_2}. \tag{2.14}$$

In the last step of the algorithm, we truncate negligible elements, e.g., those smaller than 5×10^{-5}, of the resulting feedback gain K. The small enough elements show very weak couplings between the nodes in the information structure of the controller.

Algorithm 1: Solution to problem (2.8)

Inputs: A, B_1, B_2, C_1, C_2, Q, R, λ_1, λ_2, ν, $\mathcal{K}$, W, and ε^*.

1: *Initialization:*
   Set $Y^{(0)} = I_{6n+m}$, $\varepsilon^{(0)} > \varepsilon^*$, $K^{(0)} = 0_{m\times q}$ and $k = 0$.
2: **While** $\varepsilon^{(k)} > \varepsilon^*$ **Do**
3: Update $Z^{(k+1)}$ by solving (2.12),
4: Update $Y^{(k+1)}$ using the equation (2.13),
5: Update $\varepsilon^{(k+1)}$ using the equation (2.14),
6: $k \leftarrow k + 1$,
7: **End While**
8: Truncate K.

Output: K

A summary of our proposed algorithm is described in Algorithm 1.

Remark 3. *The choice of the weight matrix W plays an important role in the sparsity-promoting properties of our method. When a proper weight matrix is not accessible, the weighted ℓ_1 norm technique can also be employed to promote the sparse controller recovery. In this method, the weight assigned to each controller element is updated inversely proportional to the value of the corresponding matrix element recovered from the previous iteration, i.e.,*

$$W_{ij}^{(k+1)} = \frac{1}{|K_{ij}^{(k)}| + \xi}, \quad \forall i, j, \tag{2.15}$$

where the constant $\xi > 0$ which is chosen as a relatively small constant, is augmented to the denominator of the update law (2.15) to guarantee the stability of the algorithm, especially, when $K_{ij}^{(k)}$ turns out to be zero in the previous iteration [75]. It should be noted that, our simulation results are obtained by incorporating this update law into the first few iteration.

Remark 4. *It is remarkable that, in implementation phase of our algorithm, utilization of constraints*

$$\epsilon_0 \leq 0.01\rho_0^2 \|\hat{\mathcal{S}}\|_{\mathcal{H}_2}^2, \tag{2.16}$$

$$\epsilon_1 \leq 0.01\rho_1 \|\hat{\mathcal{S}}\|_{\mathcal{H}_\infty}, \tag{2.17}$$

will help us to find a better locally optimal solutions. Because, otherwise, by removing them, convex optimization solver would not be able to give sparse controller designs with higher quality in terms of performance/sparsity specifications. Also, due to some practical purposes, sometimes, it is not permitted to have a $\mathcal{H}_2/\mathcal{H}_\infty$ gap larger than a certain value which highlights the necessity of using constraints (2.16) and (2.17).

2.6 Numerical Simulations

The effectiveness of our proposed method is evaluated on three classes of dynamical systems:

1. Mass-spring system,
2. Synchronous generators with sparse interconnection topology,
3. Network with unstable nodes.

In this section, based on matrix C_2, each subsection is divided into two parts as expressed below:

1. $C_2 = I$ which specifies the case that a state feedback is designed.
2. $C_2 \neq I$ which corresponds to structured state feedback design which includes the class of output feedback designs.

The pre-designed well-performing feedback controller $\hat{K}$ is computed via method proposed by [79]. Such a chapter, given an upper bound on $\mathcal{H}_\infty$ of closed-loop system $\hat{\mathcal{S}}$, proposes an $\mathcal{H}_\infty$ output feedback controller $\hat{K}$.

Before proceeding to consider our test cases, following density/performance relative specifications are defined to compare the frequency characteristics of our proposed controller with respect to the pre-designed dense controller

$$\mathcal{R}_D = \frac{\|K\|_0}{\|\hat{K}\|_0},\ \mathcal{R}_2 = \frac{\|\mathcal{S} - \hat{\mathcal{S}}\|_{\mathcal{H}_2}}{\|\hat{\mathcal{S}}\|_{\mathcal{H}_2}},\ \mathcal{R}_\infty = \frac{\|\mathcal{S} - \hat{\mathcal{S}}\|_{\mathcal{H}_\infty}}{\|\hat{\mathcal{S}}\|_{\mathcal{H}_\infty}},\ \mathcal{R}_J = \frac{J(\mathcal{S}) - J(\hat{\mathcal{S}})}{J(\hat{\mathcal{S}})},$$

where $J(.)$ represents the quadratic cost for a closed-loop system. For $\mathcal{S}$, such a quantity can be computed for both cases $(C_2 = I)$, (i.e., state feedback) and $(C_2 \neq I)$, (i.e., structured state feedback).

When $(C_2 = I)$

$$J(\mathcal{S}) = \mathbf{Tr}(B_2 X B_2^T),$$

where X is the unique positive definite solution of the following Lyapunov equation:

$$(A + B_1 K C_2)^T X + X(A + B_1 K C_2) + Q + C_2^T K^T R K C_2 = 0,$$

and when $(C_2 \neq I)$

$$J(\mathcal{S}) = \mathbf{Tr}(B_2 X B_2^T),$$

where X is the unique positive definite solution of the following Riccati-like equation:

$$(A + B_1 K C_2)^T X + X(A + B_1 K C_2) + Q + C_2^T K^T R K C_2 + \frac{1}{\gamma^2} X B_2 B_2^T X = 0,$$

and γ is the $\mathcal{H}_\infty$ upper bound parameter which is introduced in [79].

In a similar way, $J(\hat{\mathcal{S}})$ is computed for both cases $(C_2 = I)$ and $(C_2 \neq I)$.

2.6.1 Mass-Spring System

State Feedback ($C_2 = I$)

For a mass-spring system with N masses on a line, assuming that p_i is the displacement of the i^{th} mass from its reference position, denoting the state variables by $x_1 := [p_1, \ldots, p_N]^T$ and $\dot{x}_2 := x_1$, the state space realization matrices are given by

$$A = \begin{bmatrix} 0 & I \\ T & 0 \end{bmatrix}, B = \begin{bmatrix} 0 \\ I \end{bmatrix},$$

where T is an $N \times N$ tri-diagonal Toeplitz matrix with -2 on its main diagonal and 1 on its first sub-diagonal and super-diagonal, and I and O are $N \times N$ identity and zero matrices, respectively [68]. State performance weight Q is set to I and control performance weight R is set to $10I$. The output matrix C_1 is assumed to be equal to $C_2 = I$.

For parameters shown in Table 2.1, the density-performance trade-off plots are depicted in Figure 2.1. As Figure 2.1 demonstrates, there is a trade-off between density level of our designed controller and the amount of relative performance loss. The larger value in sparsity-promoting coefficient λ_2, the sparser design we get, the more performance loss occurs. Due to plot shown in Figure 2.1(a), at the expense of about 5.4170 % $\mathcal{R}_J$ relative performance loss, 86 % of controller links has been removed which has led to fully-decentralized controller. It shows that our proposed method is reasonably effective. However, the effectiveness of our rank-penalty technique is somewhat decreased when sparsity-promoting regularization λ_2 enlarges. Since the objective function 2.8 is a linear combination of performance/sparsity terms and the rank bi-linear term, there is another trade-off (in addition to the trade-off existing between performance loss and density level) between quality of sparsification and exactness of the achieved controller design K. As an observational evidence to such

N	λ_1	λ_2	ν	ε^*
5	1	$\in [0.01, 10]$ with a log-scale	1000	0.001

Table 2.1: Parameters for mass-spring system.

an issue, according to Figures 2.1(a), 2.1(b), and 2.1(c), a significantly large slope is observed in three segments connecting two leftmost points of such plots. One possible solution to overcome such an issue may be increasing the ν to obtain less performance loss for large values of λ_2.

Structured State Feedback ($C_2 \neq I$)

In this part, the matrix C_2 is chosen as $C_2 = [0\ I]$. In fact, in such a case, due to the physical model of mass-spring system, the feedback is taken from velocities. It is noteworthy that in previous part, the feedback is taken from both positions and velocities which needs extra effort with respect to the one considered in current part. A question is arisen: why do not we consider the case $C_2 = [I\ 0]$, (i.e., taking feedback from just positions)? The answer is that for such a matrix C_2, the controllability matrix $[C_2B_2\ C_2AB_2 \dots C_2A^{n-1}B_2]$ gets equal to zero and consequently, is not full-row rank.

The output matrix C_1 is opted equal to C_2. Then, according to the remark 1, the K would be an output feedback controller. Utilizing the $\mathcal{H}_\infty$ output feedback design procedure proposed by [79] for $\mathcal{H}_\infty$ for an $\mathcal{H}_\infty$ upper bound of $\gamma = 10$, provides us an output feedback controller $\hat{K}$. Next, our sparsification method is run to get the output feedback controller K. Similar to plots depicted by Figures 2.1(a), 2.1(b), and 2.1(c) and considering 20 log-scale values in $[0.01, 10]$ for λ_2, we achieve plots shown in Figures 2.2(a), 2.2(b), and 2.2(c).

2.6.2 Synchronous Generators with Sparse Interconnection Topology

State Feedback ($C_2 = I$)

A power network consists of N_G synchronous generators with sparse interconnection topology are considered [14]. The generators are randomly and uniformly distributed in a box-shape region with dimensions 10×10 unit square. The rotor dynamics of generators for purely inductive lines and constant-current loads are given by the classic second-order Kuramoto model

$$M_i\ddot{\theta}_i(t) + D_i\dot{\theta}_i(t) = P_{G_i}(t) - \sum_{j=1}^{N_G} P_{ij} \sin\big(\theta_i(t) - \theta_j(t)\big),$$

for $i \in \mathbb{G} = \{1, \ldots, N\}$, where P_{G_i} is the effective power input to generator i and the coupling weight P_{ij} is the maximum power transferred between generators i and j which is given by $P_{ij} = E_i E_j |Y_{ij}|$. The constant E_i is the internal voltage of generator i. All angles are measured with respect to a 60 Hz rotating frame. The reduced complex admittance matrix with elements $|Y_{ij}|$ incorporates models of transmission lines and transformers connecting generators i and j. The spatial location of generator i is denoted by $z_i \in \mathbb{R}^{2\times 1}$. In order to construct a sample sparse power network, first we uniformly distribute N_G generators in the region. Then, we define the coupling structure of the network by imposing the following proximity rule: If $\|z_i - z_j\|_2 > \rho$, then $|Y_{ij}|$ is set to be equal to 0, otherwise, $|Y_{ij}|$ can be chosen as a nonzero number which is drawn from the uniform distribution $U(0, \mu)$ for some $\rho, \mu > 0$. The parameter ρ defines the proximity radius between the neighbors. The corresponding graph is shown by $\mathcal{G}$ and its incidence matrix by $B(\mathcal{G})$. The vector of all angles, angular velocities, and effective power inputs are represented by $\theta = (\theta_1, \ldots, \theta_{N_G})^T$, $\dot{\theta} = \omega = (\omega_1, \ldots, \omega_{N_G})^T$, and $P_G = (P_{G_1}, \ldots, P_{G_{N_G}})^T$, respectively. The centralized

optimal governor control problem is to find the vector of effective power inputs for generators to promote the steady-state security of the grid by improving the rotor angle profile, i.e., the goal is to minimize

$$J = \int_0^\infty \big(\theta(t)^T Q_\theta \theta(t) + \omega(t)^T Q_\omega \omega(t) + P_G(t)^T P_G(t)\big) dt,$$

where $Q_\theta = B(\mathcal{G})^T B(\mathcal{G})$ and $Q_\omega = \frac{1}{2} M$.

In order to visualize the spatial structure of the centralized optimal state feedback controller, the swing equation is linearized around the operating point $(\bar{\theta}, \bar{\dot{\theta}}) = (0, 0)$ by replacing the nonlinear coupling terms $\sin(\theta_i - \theta_j)$ by $\theta_i - \theta_j$. The linearized swing equations are given by

$$M\ddot{\theta} + D\dot{\theta} + L\theta = 0, \tag{2.18}$$

where $M = \mathbf{diag}(M_1, \ldots, M_{N_G})$, $D = \mathbf{diag}(D_1, \ldots, D_{N_G})$, and $L = [L_{ij}]_{(i,j)\in\mathbb{G}}$ is the Laplacian or admittance matrix with off-diagonal elements, (i.e., $i \neq j$)

$$L_{ij} = -P_{ij},$$

and diagonal elements

$$L_{ii} = \sum_{k=1, k\neq i}^{N_G} P_{ik}.$$

The centralized optimal state feedback control law for the linearized model (2.18) is given by

$$P_{G_i} = \sum_{j=1}^{N_G} [K \; C_2]_{ij} \begin{bmatrix} \theta_j \\ \\ \omega_j \end{bmatrix},$$

where $[K \; C_2]_{ij} \in \mathbb{R}^{1\times 2}$. The numerical simulations are done with parameters given in per unit system as follows: $D_i = M_i = E_i = 1$ for all $i = 1, \ldots, N_G$, $\rho = 7$, and

N	λ_1	λ_2	ν	ε^*	ρ	μ
10	1	0.1	1000	0.001	7	5

Table 2.2: Parameters for synchronous generators with sparse interconnection topology.

$\mu = 5$. In our simulations, those sparse network samples are selected for which

$$\left(\begin{bmatrix} 0 & I \\ -L & -D \end{bmatrix}, \begin{bmatrix} 0 \\ I \end{bmatrix}, \begin{bmatrix} Q_\theta^{\frac{1}{2}} & 0 \\ 0 & Q_\omega^{\frac{1}{2}} \end{bmatrix} \right),$$

is stabilizable and detectable. Matrices Q and R are set to $5I$ and I, respectively.

Remark 5. *Since our goal is synchronization of angles of generators and $A+B_1KC_2$ will have a pole at zero, we take advantage of linear algebraic trick used by [80] to compute R_J and K. To reach such a goal, the output matrix C_1 is selected as $C_1 = \begin{bmatrix} UU^T & 0 \\ 0 & I \end{bmatrix}$ where U is the $N_G \times (N_G - 1)$ matrix consisting of columns which construct an orthonormal basis orthogonal to span of vector of all ones.*

For parameters shown in Table 2.2, spatial distribution of 10 synchronous generators, the sparsity pattern of sparsified controller K, schatten 2-norm of $\mathcal{S}$ and $\hat{\mathcal{S}}$, and maximum/minimum singular values of $\mathcal{S}$ and $\hat{\mathcal{S}}$ are visualized in Figure 2.3.

For such a designed controller, $\mathcal{R}_D$, $\mathcal{R}_2$, $\mathcal{R}_\infty$, and $\mathcal{R}_J$, are 36 %, 5.2943 %, 9.0774 %, and 1.1338 %, respectively. It is worth noting that utilizing the constraints (2.16) and (2.17), has made a notable improvement in terms of obtaining a lower performance loss comparing to the numerical results within [21, 27, 28] in which improving constraints like (2.16) and (2.17) are not incorporated. Frequency plots provided by Figures 2.3(c) and 2.3(d), validate that two closed-loop systems, the one with pre-designed dense controller, i.e., $\hat{\mathcal{S}}$ and the one with our sparse controller design, i.e., $\mathcal{S}$, have reasonably similar frequency characteristics. In particular, for high frequencies, they act extremely similar.

Figures 2.4(a) and 2.4(b) show angles and angular velocities versus time. As it is observed angles of all generators are synchronized via our proposed controller K.

Structured State Feedback ($C_2 \neq I$)

Here, we consider the case that $C_2 = [0\ I]$. For parameters shown in Table 2.2, spatial distribution of 10 synchronous generators, the sparsity pattern of sparsified controller K, schatten 2-norm of $\mathcal{S}$ and $\hat{\mathcal{S}}$, and maximum/minimum singular values of $\mathcal{S}$ and $\hat{\mathcal{S}}$ are visualized in Figure 2.5.

For such a designed controller, $\mathcal{R}_D$, $\mathcal{R}_2$, $\mathcal{R}_\infty$, and $\mathcal{R}_J$ are 42 %, 5.9487 %, 13.1502 %, and 2.3250 %, respectively.

Figures 2.6(a) and 2.6(b) show angles and angular velocities versus time. As it is observed angles of all generators are synchronized via our proposed controller K.

2.6.3 Network with Unstable Nodes

State Feedback ($C_2 = I$)

In the following simulations, N nodes are randomly distributed (with a uniform distribution) in a region of area 10×10 unit square [42]. Each node is assumed to be a linear system which is coupled through its dynamics and the LQ cost functional to other subsystems. The aggregate dynamics of N linear subsystems can be described as

$$\dot{\psi}_k(t) = [A]_{kk}\psi_k(t) + \sum_{i=1, i\neq k}^{N} [A]_{ki}\psi_i(t) + [B]_{kk}u_k(t),$$

for all $k \in \mathbb{G} = \{1, \ldots, N\}$. It is assumed that for all $k \in \mathbb{G} = \{1, \ldots, N\}$ and $i \neq k$ we have

$$[A]_{kk} = \begin{bmatrix} 1 & 1 \\ 1 & 2 \end{bmatrix}, [B]_{kk} = \begin{bmatrix} 0 \\ 1 \end{bmatrix}, [A]_{ki} = \frac{1}{\mathcal{X}_\alpha(\mathrm{dis}(k,i))} \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}, [B]_{ki} = \begin{bmatrix} 0 \\ 0 \end{bmatrix},$$

N	λ_1	λ_2	ν	ε^*	α
15	1	1	1000	0.001	1

Table 2.3: Parameters for network with unstable nodes.

where $\mathcal{X}_\alpha$ is the coupling characteristic function which can be chosen as an exponentially decaying function, i.e., $\mathcal{X}_\alpha(d) = e^{-\alpha d}$ and $\text{dis}(k,i)$ denotes the Euclidean distance between locations of nodes k and i. The output matrix C_1 is opted as $C_1 = C_2 = I$. Both state performance weight Q and control performance weight R are set to identity.

The spatial visualization of network with 15 unstable nodes and its corresponding sparse controller design are depicted in Figure 2.7 where the corresponding selected parameters are shown in Table 2.3.

In such a case, $\mathcal{R}_D$, $\mathcal{R}_2$, $\mathcal{R}_\infty$, and $\mathcal{R}_J$ are 18.67 %, 16.2391 %, 27.7032 %, and 1.2317 %, respectively. Running the code developed by authors of [68] for our 15-node system, a sparse controller F and structured sparse controller F^{opt} with $\mathcal{R}_D = 18.67\%$ are obtained. The triple $(\mathcal{R}_2, \mathcal{R}_\infty, \mathcal{R}_J)$ for F and F^{opt} are $(15.7615\ \%, 37.7944\ \%, 0.66\ \%)$ and $(15.0428\ \%, 39.2530\ \%, 0.54\ \%)$, respectively. A simple comparison shows that our designed sparse K outperforms the ones proposed by [68] in terms of $\mathcal{H}_\infty$ performance criterion (36.43 % compared to F^{opt} and 41.69 % compared to F, respectively). Also, in the case of $\mathcal{H}_2$ performance criterion, although their method outperforms ours, the weakness of our designed controller compared to theirs are 3.03 % and 7.95 % for F and F^{opt}, respectively which are negligible. It must be added that such a result is expected. Because, in our proposed method, $\mathcal{H}_\infty$-norm is incorporated in addition to considering the $\mathcal{H}_2$-norm and minimized, while the method proposed by [68], only considers the $\mathcal{H}_2^2$ LQR cost with special performance output matrix $\begin{bmatrix} Q^{\frac{1}{2}} \\ -R^{\frac{1}{2}}F \end{bmatrix}$. However, in terms of $\mathcal{R}_J$ measure, the weakness

of our proposed method are 86.6212 % and 128.0926 % for F and F^{opt}, respectively. It is not far away from our expectations. Because, their method specifically aims to minimize the $\mathcal{R}_J$ measure.

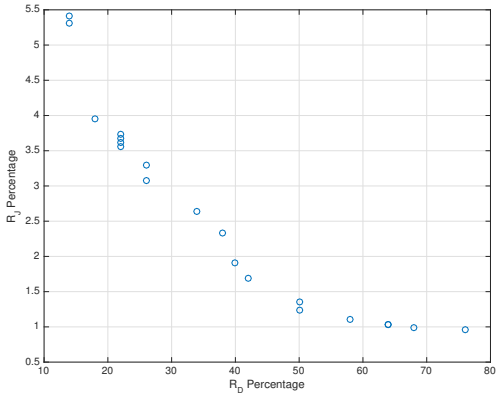
Structured State Feedback ($C_2 \neq I$)

Here, we consider the case that $C_2 = \begin{bmatrix} I & I \\ 0 & I \end{bmatrix}$. The output matrix C_1 is chosen equal to $C_2 = \begin{bmatrix} I & I \\ 0 & I \end{bmatrix}$, i.e., K would be an output feedback controller. Both state performance weight Q and control performance weight R are set to identity. The spatial visualization of network with 15 unstable nodes and its corresponding sparse output feedback controller design are depicted in Figure 2.8 where the corresponding selected parameters are shown in Table 2.3.

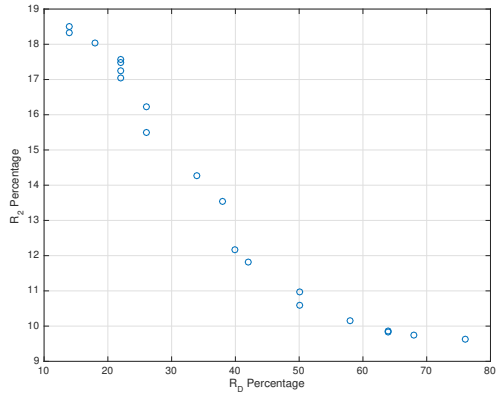
In such a case, $\mathcal{R}_D$, $\mathcal{R}_2$, $\mathcal{R}_\infty$, and $\mathcal{R}_J$ are 19.5556 %, 25.8923 %, 36.2158 %, and 3.3510 %, respectively.

2.7 Conclusion

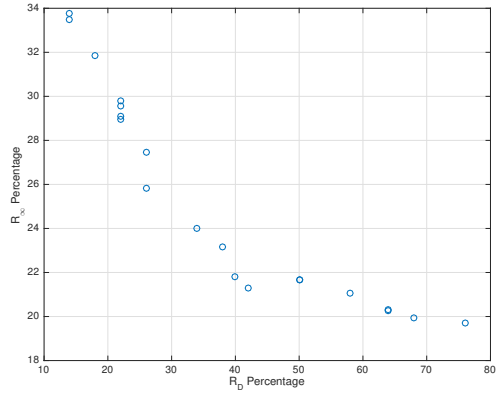
A new sparsification approach is developed to obtain optimal sparse controllers. Basically, an available pre-designed dense controller is altered towards a sparse controller, while heeding the performance deterioration caused by the sparsification process. By equivalently reformulating the problem into a fixed rank optimization problem and utilizing the bi-linear rank penalty technique, a method is achieved by which a sparse structured controller capable of exhibiting similar frequency and time characteristics of the pre-designed controller, in terms of $\mathcal{H}_2$ and $\mathcal{H}_\infty$ norms is proposed. Our method can also be modified to incorporate constraints on the control signal. Since our method takes advantage of SDP solvers, it is not applicable to large networks. A future work can be development of scalable sparsification methods.



(a)

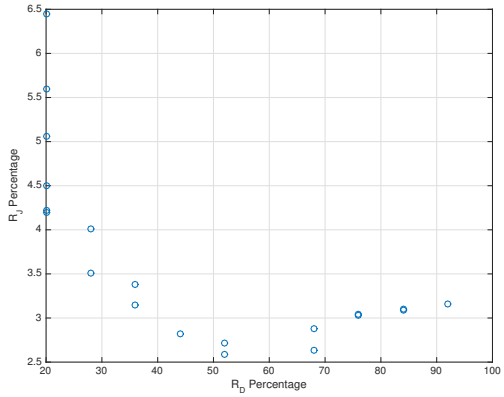


(b)

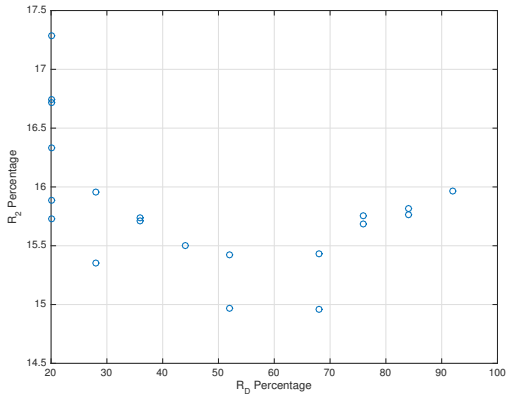


(c)

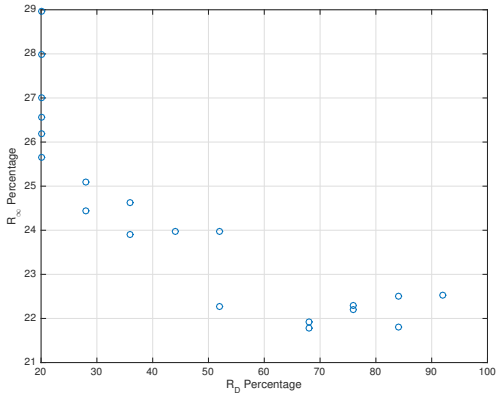
Figure 2.1: Density-Performance trade-off curves for a mass-spring system ($C_2 = I$) (a) $\mathcal{R}_J$ percentage versus $\mathcal{R}_D$ (b) $\mathcal{R}_2$ percentage versus $\mathcal{R}_D$ percentage (c) $\mathcal{R}_\infty$ percentage versus $\mathcal{R}_D$ percentage.



(a)



(b)



(c)

Figure 2.2: Density-Performance trade-off curves for a mass-spring system ($C_2 \neq I$) (a) $\mathcal{R}_J$ percentage versus $\mathcal{R}_D$ (b) $\mathcal{R}_2$ percentage versus $\mathcal{R}_D$ percentage (c) $\mathcal{R}_\infty$ percentage versus $\mathcal{R}_D$ percentage.

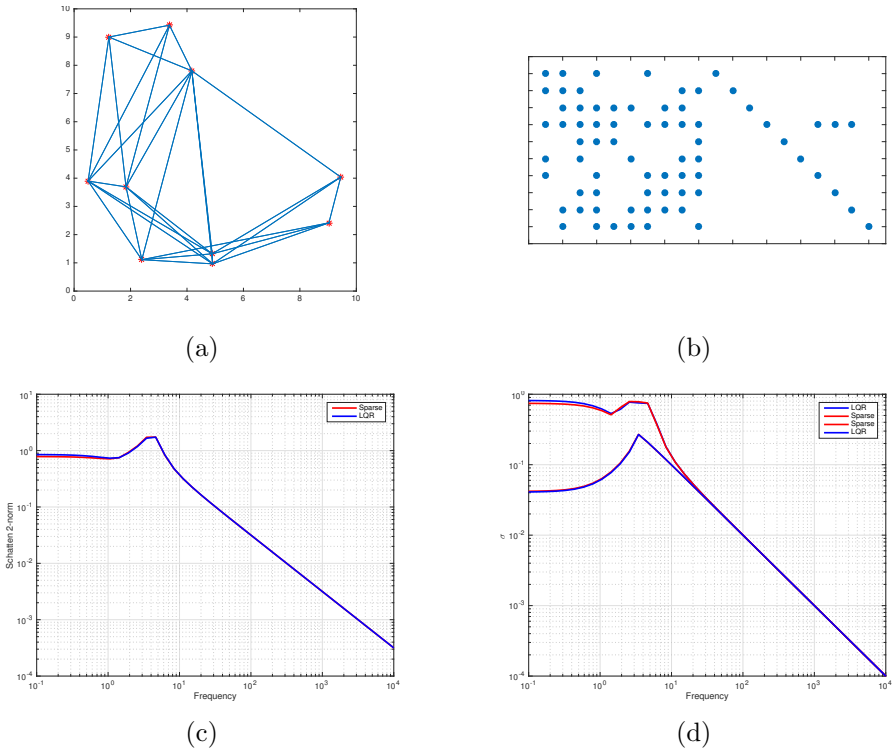
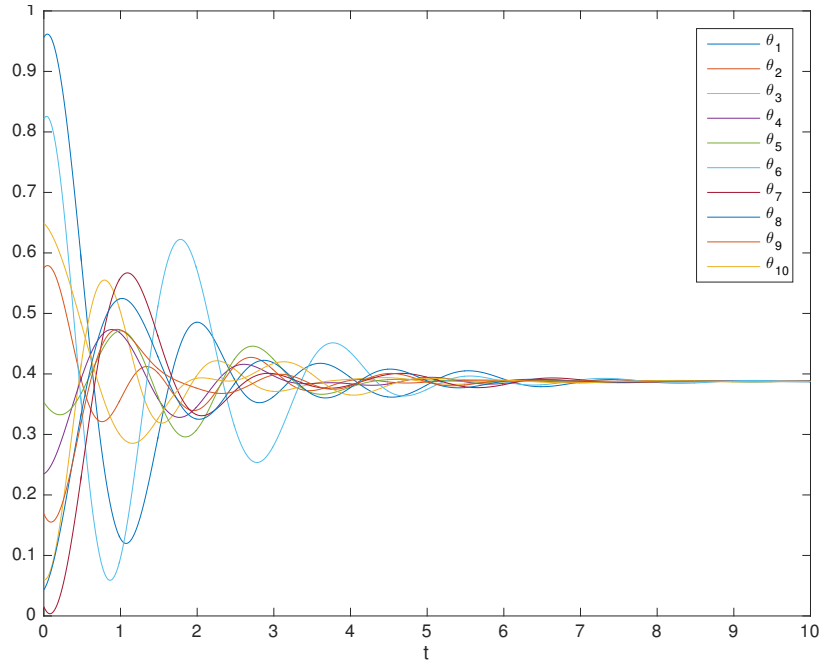
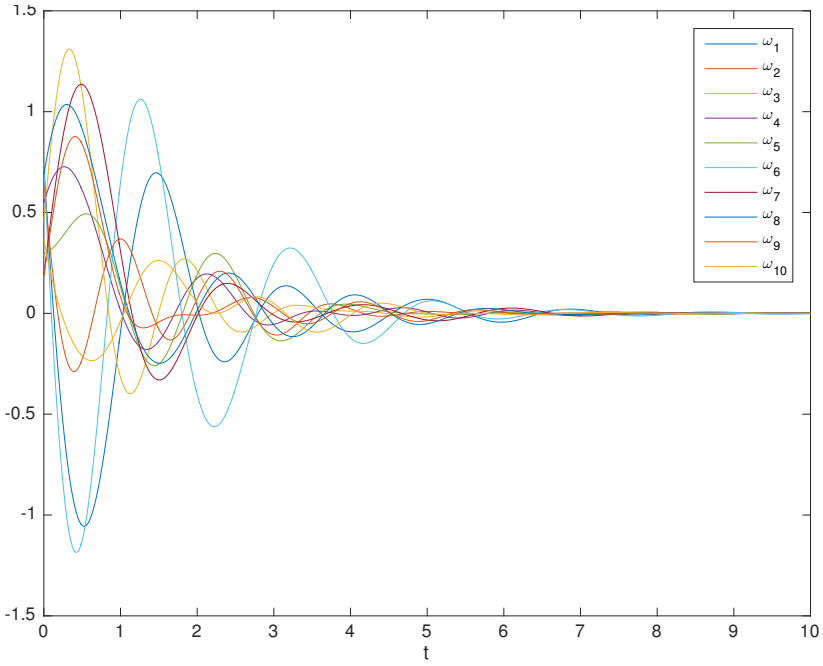


Figure 2.3: State feedback ($C_2 = I$): (a) Spatial distribution of sparse interconnection topology consisting of 10 synchronous generators. Blue solid lines represent the bi-directional links connecting synchronous generators specified by red $*$ (b) The sparsity pattern of sparsified controller K. Blue dots represent the non-zero elements (c) Schatten 2-norm of $\mathcal{S}$ (Red) and $\hat{\mathcal{S}}$ (Blue) (d) Maximum/minimum singular values of $\mathcal{S}$ (Red) and $\hat{\mathcal{S}}$ (Blue).



(a)



(b)

Figure 2.4: State feedback ($C_2 = I$): (a) Angles versus time (b) Angular velocities versus time.

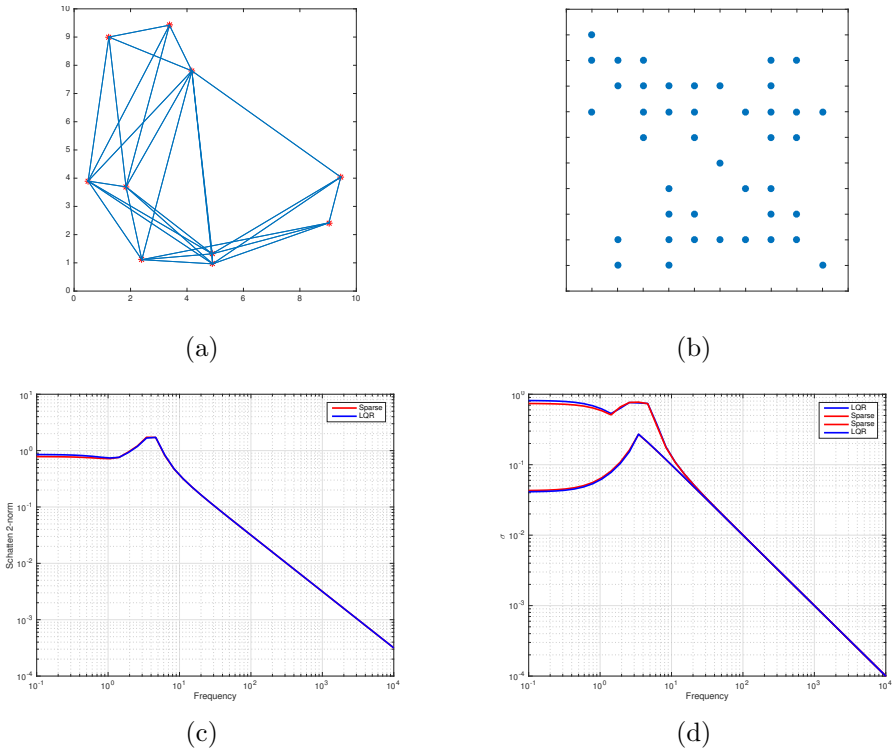
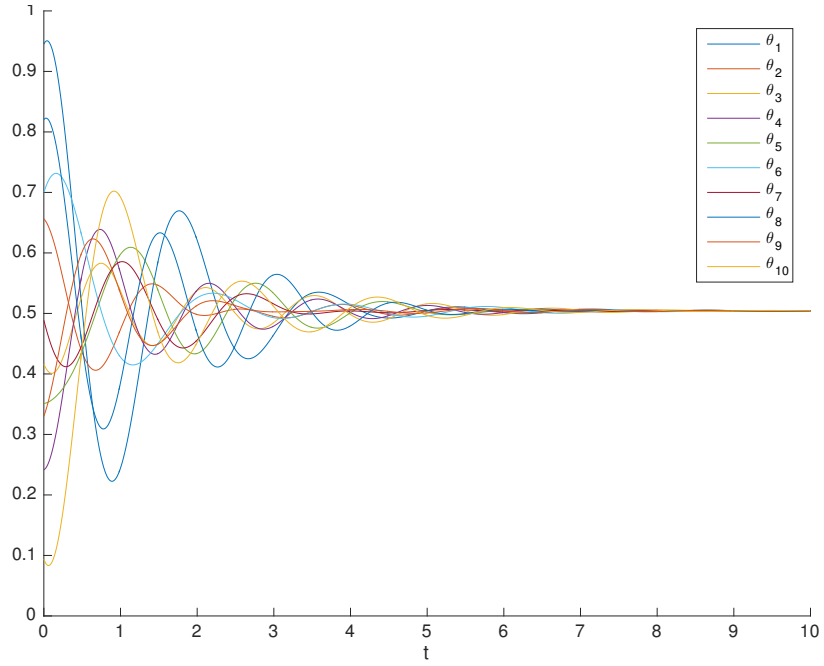
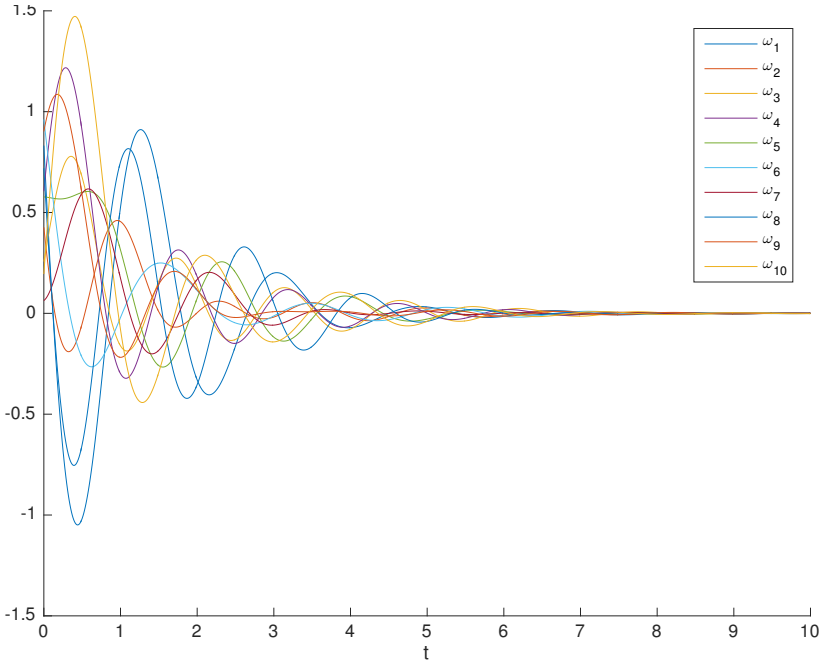


Figure 2.5: Structured state feedback ($C_2 \neq I$): (a) Spatial distribution of sparse interconnection topology consisting of 10 synchronous generators. Blue solid lines represent the bi-directional links connecting synchronous generators specified by red $*$ (b) The sparsity pattern of sparsified controller K. Blue dots represent the non-zero elements (c) Schatten 2-norm of $\mathcal{S}$ (Red) and $\hat{\mathcal{S}}$ (Blue) (d) Maximum/minimum singular values of $\mathcal{S}$ (Red) and $\hat{\mathcal{S}}$ (Blue).



(a)



(b)

Figure 2.6: Structured state feedback ($C_2 \neq I$): (a) Angles versus time (b) Angular velocities versus time.

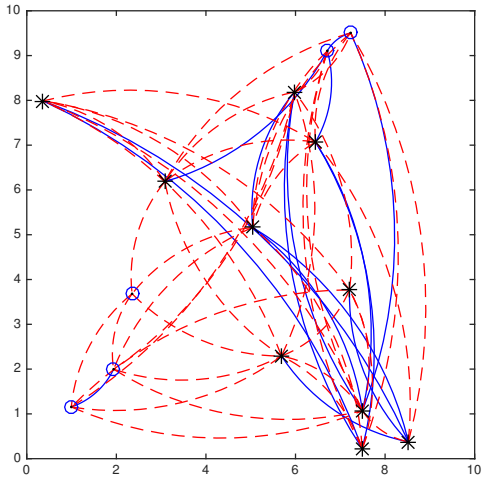


Figure 2.7: Spatial distribution of 15 unstable nodes and the sparsity visualization of sparsified controller K. Blue solid lines, red dashed lines, blue $\circ$, and black $*$, represent the bi-directional links, one-way links, self-loops, and no-self-loops, respectively.

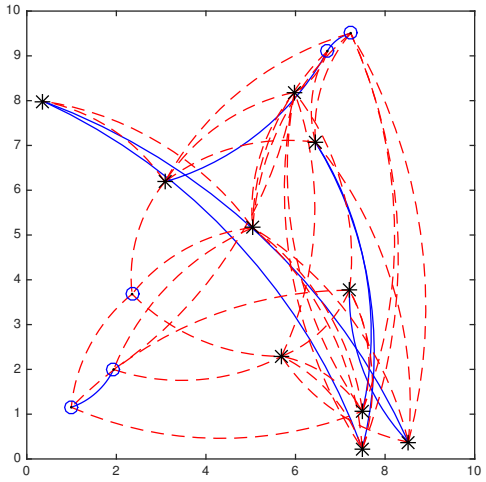


Figure 2.8: Spatial distribution of 15 unstable nodes and the sparsity visualization of sparsified output feedback controller K. Blue solid lines, red dashed lines, blue $\circ$, and black $*$, represent the bi-directional links, one-way links, self-loops, and no-self-loops, respectively.

Chapter 3

Periodic Time-Triggered Sparse Linear Quadratic Controller Design

3.1 Introduction

Sampled-Data control systems have been studied deeply from previous decades [55, 81, 82]. In such a control setup, the system to be controlled is in continuous time, however the controller is synthesized in a discrete manner [83]. Because, there is a great tendency to use discrete implementations in technological applications. One of the crucial topics in such a control systems is how to decrease the number of samplings which even it could be done in a non-uniform fashion. Equivalently, the long maximum allowable time interval is of desire in terms of sampling cost. Indeed, such an objective leads to *Temporal Sparsity-Promoting Optimal Control* [56] which deals with *sparsity* in *time* horizon.

The *sparsity* in the *space* of static feedback controllers is called *Spatial Sparsity-Promoting Optimal Control* [1]. In recent years, the spatial sparsity-promoting optimal control has been growing rapidly [1, 3, 7, 13, 14, 18, 19, 21, 23, 24, 27–29, 31, 38, 40, 42, 80, 84]. The fundamental objective of spatial sparsity-promoting

optimal control is to decrease the number of communication links between nodes preserving the guaranteed level of performance. Such a balance is obtained through some sparsity-promoting ℓ_1-regularized term which is supplemented to the performance loss term in objective function of the corresponding regularized optimization problem.

Considering the switching control strategies, some works have been done to achieve *Temporal Sparsity-Promoting Optimal Control* in sampled-data control framework. One of the significant methods to obtain such an aim is called *Self-triggered Control* [43]. The fundamental advantage of self-triggered control is that the control signal is kept fixed when there is no need to new update (sampling) [52, 56, 58, 59]. Indeed, some performance-preserving condition or Lyapunov-based stability condition (called self-triggering conditions) is checked in such a methodology and specifies whether a new sampling is necessary or not. The more detailed explanations about synthesizing of self-triggered control can be found in [56, 59]. In [56], it is highlighted that next update time is calculated based on current state information. Unlikely, in event-triggered control (another aperiodic control method) the previous states are also used to compute the activation time [44]. Likewise, in [62], assuming the finite sequence of interval lengths and their corresponding spatially-varying stabilizing controllers, a self-triggered method is proposed to find the switching rule on the basis of current state information. By *spatially-varying*, we mean different feedback gains in the space of feedback gains which are applied throughout the time intervals. In [85], the proposed self-triggering method, ensures $\mathcal{L}_2$ stability of the closed loop system and the average time period has an increasing behavior versus $\mathcal{L}_2$ gain. In [62], some similar but not exact relationship between $\mathcal{H}_{2,\infty}$ performance indices and the average time period is expressed.

Motivated by such results between performance indices and average time period,

we propose *periodic time-triggered sparse LQC design* via bi-linear rank penalty technique. Next, we define spatio-temporal sparsity criterion to evaluate the *spatial sparsity* and *temporal sparsity* at the same time. Simply, such a criterion is defined as a summation of density level of periodic time-triggered sparse LQC and non-negative multiplier of inverse of time period. Thus, for a fixed performance loss, the less this criterion, the more desirable design is obtained. It is known that traditional centralized LQR, takes so many samples and also uses too much battery life for communication between nodes. The spatio-temporal sparsity criterion enables us to make a decision based on our desired level of trade-off between complexity of the controller and performance loss. It is worth noting that current sparsity-promoting optimal control strategies choose continuous-time or discrete-time setup to derive sparse controllers while the latter group don't consider the sampling rate connecting the continuous-time and discrete-time setups. In this work, we have tried to utilize such an issue in our sparse control design.

This chapter is structured as follows: Mathematical notations are expressed in section 3.2. Section 3.3 is devoted to the statement of the problem to be solved. In section 3.4, periodic time-triggered sparse LQC design procedure is presented. Section 3.5 describes bi-linear rank penalty technique in a detailed way. Section 3.6 defining the spatio-temporal sparsity criterion, investigates various numerical simulations to visualize the relationship between spatio-temporal sparsity criterion and performance loss of periodic time-triggered sparse LQC. Finally, section 3.7 concludes the chapter with drawing some future insights.

3.2 Mathematical Notations

The set of real numbers, positive integer numbers, and non-negative integer numbers are denoted by $\mathbb{R}$, $\mathbb{N}$, and $\mathbb{Z}_+$, respectively. The set of real-valued $n \times 1$ vectors and set

of real-valued $m \times n$ matrices are represented by $\mathbb{R}^n$ and $\mathbb{R}^{m \times n}$, respectively. The positive semi-definiteness and positive definiteness are shown by $\succeq$ and $\succ$, respectively. The identity matrix is I as usual. The Euclidean norm of vector v is denoted by $\|v\|_2$. Matrix operators acting on some arbitrarily-chosen matrix M, are summarized in Table 3.1.

Symbol	Definition
$\|M\|_0$	Cardinality of matrix M, i.e., number of nonzero elements of matrix M
$\|M\|_1$	ℓ_1 norm of matrix M, i.e., $\sum_{i,j} \|M_{ij}\|$
$\|M\|_2$	Largest singular value of matrix M
$\|M\|_F$	Frobenius norm of matrix M, i.e., $\sqrt{\sum_{i,j} M_{ij}^2}$
$\mathbf{Tr}(M)$	Trace of matrix M, i.e., $\sum_i M_{ii}$
$\lambda_i(M)$	i^{th} largest eigenvalue of matrix M, i.e., for $j > k$, $\lambda_j \leq \lambda_k$
$\rho(M)$	Spectral radius of matrix M, i.e., $\max_i \|\lambda_i(M)\|$

Table 3.1: Matrix operators

The Hadamard product is shown by $\circ$. The normal distribution with zero mean and σ^2 variance is denoted by $\mathcal{N}(0, \sigma^2)$. The expectation of a random variable w is represented by $\mathbb{E}\{w\}$.

Definition 2. *For a given $\epsilon > 0$ and matrix X, we say that rank of X is k with tolerance ϵ, and it is denoted by* $\mathbf{rank}(X; \epsilon)$*, if exactly k singular values of X are larger than or equal to ϵ.*

3.3 Problem Formulation

We consider the class of linear time invariant (LTI) systems

$$\dot{x}(t) = Ax(t) + Bu(t),\ x(t_0) = x_0, \tag{3.1}$$

which is stabilized by

$$u(t) = F_k x(t_k), \quad \text{for} \quad t \in [t_k, t_{k+1}), \tag{3.2}$$

where $x \in \mathbb{R}^n$, $u \in \mathbb{R}^m$, $k \in \mathbb{Z}_+$, $t_0 = 0$ and x_0 is drawn from a standard normal distribution, (i.e., with zero mean and unit standard deviation).

The sequence of feedback controllers and triggering times are denoted by $\{F_k\}_{k=0}^{\infty}$ and $\{t_k\}_{k=0}^{\infty}$, respectively. The main assumptions about such spatio-temporal sequences are stated as follows.

Assumption 1. *The sequence of linear quadratic controllers $\{F_k\}_{k=0}^{\infty}$ is invariant, i.e., we are dealing with spatially-invariant case ($F_k = F$) for all $k \in \mathbb{Z}_+$.*

Assumption 2. *The sequence of triggering times $\{t_k\}_{k=0}^{\infty}$ is known and consists of equidistant values. In other words, we have a temporally-known periodic time setup ($t_k = k\delta$) for all $k \in \mathbb{Z}_+$ and some given positive δ which is called time period.*

Considering Assumptions 1 and 2, a spatially-invariant and temporally-known periodic time-triggered sparse LQC design is desired. In order to achieve such a goal, the Periodic Time-Triggered Sparse LQC Problem (**P1**) is defined as follows:

$$\begin{aligned} &\underset{F}{\text{minimize}}\ \mathbb{E}\left\{\int_0^{\infty} \big(x(t)^T Q x(t) + u(t)^T R u(t)\big) dt\right\} + \gamma \|F\|_0 \qquad (\mathbf{P1}) \\ &\text{subject to: (3.1) and (3.2)}, \\ &\qquad\qquad F : \text{stabilizing}, \end{aligned}$$

where the regularization parameter γ determines what amount of sparsification is needed. Also, $Q \succeq 0$ and $R \succ 0$ are state-weight and input-weight matrices, respectively. The quadratic terms appeared on objective function of (**P1**), represent the performance loss in (**P1**).

3.4 Periodic Time-Triggered Sparse LQC Design Procedure

Now, we try to simplify (**P1**) as much as possible. To reach such an aim, firstly, we remove the sparsity-promoting term $\gamma\|F\|_0$ from the objective function and provide some lemmas and propositions to get an equivalent form for the Periodic Time-Triggered LQC Problem (**P2**) which will be defined later. Then, finally, the sparsity-promoting term $\gamma\|W \circ F\|_1$ (the weighted ℓ_1-regularization term) is added to the corresponding performance loss term. The similar approach has been taken by authors of [18] in which an equivalent form of $\mathcal{H}_2$ problem is obtained and then sparsity-promoting term is augmented to the corresponding $\mathcal{H}_2$-squared term.

Lemma 7. *Solving the system (3.1) and (3.2) for time interval $[t_k, t_{k+1})$, the corresponding state $x(t)$ for all $k \in \mathbb{Z}_+$ is calculated as follows:*

$$x(t) = M(t - t_k)x(t_k),$$

where $M(\tau)$ is defined as follows:

$$M(\tau) := e^{A\tau}\big(I + Z(\tau)BF\big),\ Z(\tau) = \int_0^\tau e^{-A\xi}d\xi.$$

Proof. Proof is immediately resulted from solving a first order ordinary differential system. □

Lemma 8. *For all $t \in [t_k, t_{k+1})$ and $k \in \mathbb{Z}_+$, we have*

$$x(t) = M(t - t_k)M^k(\delta)x_0. \tag{3.3}$$

Proof. Applying the induction principle to Lemma 7, proof is quite straightforward. □

Lemma 9. *The quadratic part of the objective function in* (**P1**)*, i.e.,*

$$\mathbb{E}\left\{\int_0^\infty \big(x(t)^T Q x(t) + u(t)^T R u(t)\big)dt\right\},$$

reduces to the following form:

$$\mathbf{Tr}\Big(\sum_{k=0}^{\infty} M^k(\delta)^T Y(\delta) M^k(\delta)\Big), \tag{3.4}$$

where

$$Y(\delta) := \int_0^\delta M(\tau)^T Q M(\tau) d\tau + \delta F^T R F. \tag{3.5}$$

Proof. According to (3.3), breaking down the integral to sum of sub-integrals, cyclic property of **Tr**, commutative property of any pair of linear operators $\sum$, **Tr**, and $\mathbb{E}$,

we have

$$
\begin{aligned}
&\mathbb{E}\left\{\int_0^{\infty}\big(x(t)^T Q x(t)+u(t)^T R u(t)\big)dt\right\}\\
&=\mathbb{E}\left\{\sum_{k=0}^{\infty}\int_{t_k}^{t_{k+1}}\big(x(t)^T Q x(t)+u(t)^T R u(t)\big)dt\right\}\\
&=\mathbb{E}\Big\{\sum_{k=0}^{\infty}\mathbf{Tr}\Big(M^k(\delta)^T Y(\delta) M^k(\delta) x_0 x_0^T\Big)\Big\}\\
&=\mathbb{E}\Big\{\mathbf{Tr}\Big(\sum_{k=0}^{\infty}M^k(\delta)^T Y(\delta) M^k(\delta) x_0 x_0^T\Big)\Big\},\\
&=\mathbf{Tr}\Big(\mathbb{E}\Big\{\sum_{k=0}^{\infty}M^k(\delta)^T Y(\delta) M^k(\delta) x_0 x_0^T\Big\}\Big)\\
&=\mathbf{Tr}\Big(\sum_{k=0}^{\infty}M^k(\delta)^T Y(\delta) M^k(\delta)\mathbb{E}\{x_0 x_0^T\}\Big).
\end{aligned}
\tag{3.6}
$$

Since x_0 is drawn from a standard normal distribution, we have $\mathbb{E}\{x_0 x_0^T\}=I$. Hence, the formula (3.6) takes the following form:

$$
\mathbf{Tr}\Big(\sum_{k=0}^{\infty}M^k(\delta)^T Y(\delta) M^k(\delta)\Big).
$$

Thus, proof is completed. □

The expression (3.4) is not at desired simplicity level and still has some sort of complicated appearance. The following proposition is utilized to simplify (3.4).

Proposition 10. *The periodic time-triggered LQC is stabilizing if and only if*

$$
\rho\big(M(\delta)\big)<1,
$$

holds.

Proof. Let us assume that the periodic time-triggered LQC is stabilizing. Thus, since

we have $x(t_k) = M(\delta)^k x_0$, it yields that

$$\lim_{k\to\infty} \|x(t_k)\|_2 = 0. \tag{3.7}$$

According to (3.7), it is resulted that we must have $\rho\big(M(\delta)\big) < 1$.
Since all elements of $M(t-t_k)$ are continuous functions of $t-t_k$ and $t-t_k$ is bounded by 0 and δ, according to a mathematical fact, there exists a $\mathcal{M}$ for which we have $\max\left(\|M(t-t_k)\|_2\right) = \mathcal{M}$.

Now, let us assume that we have $\rho\big(M(\delta)\big) < 1$. Thus, (3.7) will be satisfied. Hence, for a given ϵ there exists a k_ϵ for which we have $\|x(t_k)\|_2 \leq \frac{\epsilon}{\mathcal{M}}$ for all $k \geq k_\epsilon$.

For any t chosen greater than t_{k_ϵ}, there exists an index l for which t lies between t_l and t_{l+1}. Now, we claim that $\|x(t)\|_2 \leq \epsilon$ can be achieved by considering the following inequalities:

$$\|x(t)\|_2 = \|M(t-t_l)x(t_l)\|_2 \leq \|M(t-t_l)\|_2 \|x(t_l)\|_2 \leq \mathcal{M}\|x(t_l)\|_2 \leq \epsilon.$$

Thus, $\|x(t)\|_2 \leq \epsilon$ will be satisfied for all $t \geq t_{k_\epsilon}$. Hence, the periodic time-triggered LQC is stabilizing. □

Thus, in the rest of the chapter, to deal with stability guarantee, we are allowed to consider the necessary and sufficient condition derived by Proposition 10 in our problem castings. The following assumption expresses such a consideration. Then, assuming the $\rho\big(M(\delta)\big) < 1$ and defining the

$$P(\delta) := \sum_{k=0}^{\infty} M^k(\delta)^T Y(\delta) M^k(\delta),$$

$P(\delta)$ will be the unique positive definite solution of the following discrete Lyapunov equation:

$$M(\delta)^T P(\delta) M(\delta) - P(\delta) + Y(\delta) = 0. \tag{3.8}$$

Remark 6. *From here, for the sake of the simplicity and space saving, the argument δ is dropped in our notations, if it is necessary.*

Now, using the stability criterion (3.8), (**P2**) is defined as follows:

$$\begin{aligned} &\underset{F,P}{\text{minimize}}\ \mathbf{Tr}(P) \\ &\text{subject to: } M^T P M - P + \int_0^\delta M(\tau)^T Q M(\tau) d\tau + \delta F^T R F = 0, \\ &\qquad\qquad P \succ 0. \end{aligned} \tag{\textbf{P2}}$$

As it is observed, in (**P2**), the constraint corresponding to the Lyapunov equation has some non-convexities. To deal with such non-convexities, we utilize the Schur complement and fixed-rank constraint reformulation. Taking an advantage of rank-constrained optimization, we propose a spatially-invariant and temporally-known periodic time-triggered sparse LQC design. The following lemma takes a crucial step toward the achieving simplest form for (**P2**).

Lemma 11. *The discrete Lyapunov equation (3.8) can be rewritten as follows:*

$$M(\delta)^T P(\delta) M(\delta) - P(\delta) + H_0(\delta) + F^T H_1(\delta)^T + H_1(\delta) F + F^T H_2(\delta) F = 0,$$

where

$$\begin{aligned} H_0(\tau) &= \int_0^\tau e^{A^T t} Q e^{At} dt,\ H_1(\tau) = \int_0^\tau e^{A^T t} Q e^{At} Z(t) B dt, \\ H_2(\tau) &= \int_0^\tau \big(e^{At} Z(t) B\big)^T Q \big(e^{At} Z(t) B\big) dt + \tau R. \end{aligned}$$

Proof. Note that by substituting the $M(\tau) = e^{A\tau}\big(I + Z(\tau)BF\big)$ in (3.5) and doing some simple multiplications, $Y(\delta)$ is expressed in terms of $H_0(\delta)$, $H_1(\delta)$, and $H_2(\delta)$. □

Lemma 12. *For all $\tau > 0$, matrix $H_2(\tau)$ is positive definite, i.e., $H_2(\tau) \succ 0$ in the cone of all positive definite matrices.*

Proof. In order to prove that the matrix $H_2(\zeta)$ is positive definite for all positive values of ζ, we consider an arbitrary vector $v \in \mathbb{R}^m$ $(v \neq 0)$. Then, we prove that $v^T H_2(\zeta) v$ is positive. Thus, we have

$$\begin{aligned} v^T H_2(\zeta) v &= v^T \Big(\int_0^\zeta \big(e^{A\tau} Z(\tau) B\big)^T Q \big(e^{A\tau} Z(\tau) B\big) d\tau + \zeta R \Big) v, \\ &= \int_0^\zeta \big(e^{A\tau} Z(\tau) B v\big)^T Q \big(e^{A\tau} Z(\tau) B v\big) d\tau + \zeta v^T R v. \end{aligned}$$

Since $Q \succeq 0$ and $R \succ 0$ hold, the term appeared inside the integral is non-negative and $\zeta v^T R v$ is positive, respectively. The definite integral of a non-negative function over some interval gives a non-negative value. Thus, $v^T H_2(\zeta) v > 0$ and proof is done. □

Lemma 13. *The optimization problem* (**P2**) *is equivalent to the following auxiliary optimization problem:*

$$\begin{aligned} \underset{F,P}{\textit{minimize}} \quad & \mathbf{Tr}(P) \\ \textit{subject to:} \quad & M^T P M - P + \int_0^\delta M(\tau)^T Q M(\tau) d\tau + \delta F^T R F \preceq 0, \\ & P \succ 0. \end{aligned}$$

Proof. Suppose that pairs $(\hat{F}, \hat{P})$ and (F^*, P^*) denote the corresponding optimal solutions of (**P2**) and the auxiliary optimization problem, respectively. Since $(\hat{F}, \hat{P})$ belongs to feasible set of the auxiliary optimization problem, then we have $\mathbf{Tr}(P^*) \leq \mathbf{Tr}(\hat{P})$. It is known that there exists a positive semi-definite matrix N for which we

have

$$M^{*T}P^*M^* - P^* + \int_0^\delta M^*(\tau)^T Q M^*(\tau) d\tau + \delta F^{*T} R F^* + N = 0.$$

In other words, we have

$$P^* = \sum_{k=0}^{\infty} M^{*k^T} (\int_0^\delta M^*(\tau)^T Q M^*(\tau) d\tau + \delta F^{*T} R F^* + N) M^{*k}.$$

Defining the $\tilde{P}$ as follows:

$$\tilde{P} = \sum_{k=0}^{\infty} M^{*k^T} (\int_0^\delta M^*(\tau)^T Q M^*(\tau) d\tau + \delta F^{*T} R F^*) M^{*k},$$

it implies that $\mathbf{Tr}(P^*) \geq \mathbf{Tr}(\tilde{P})$. We know that $\tilde{P}$ satisfies the following equation:

$$M^{*T}\tilde{P}M^* - \tilde{P} + \int_0^\delta M^*(\tau)^T Q M^*(\tau) d\tau + \delta F^{*T} R F^* = 0.$$

Thus, the pair belongs to feasible set of (**P2**). Then, we can conclude that $\mathbf{Tr}(\tilde{P}) \geq \mathbf{Tr}(\hat{P})$. Thus, $\mathbf{Tr}(P^*) \geq \mathbf{Tr}(\hat{P})$ and consequently $\mathbf{Tr}(P^*) = \mathbf{Tr}(\hat{P})$ are resulted and proof is done. □

Now, considering Lemmas 11, 12, and 13, we can state the following proposition.

Proposition 14. *The optimization problem* (**P2**) *can equivalently be reformulated as*

the following rank-constrained problem:

$$\begin{aligned} &\underset{P,F,K}{\textit{minimize}}\ \mathbf{Tr}(P) \qquad\qquad (\mathbf{P3}) \\ &\textit{subject to:} \begin{bmatrix} K & 0 & M \\ 0 & H_2^{-1} & F \\ M^T & F^T & P - H_0 - F^T H_1^T - H_1 F \end{bmatrix} \succeq 0, \\ &\qquad P \succ 0,\ \mathbf{rank}(\begin{bmatrix} K & I \\ I & P \end{bmatrix}) = n. \end{aligned}$$

Proof. In order to prove such a proposition, a series of equivalent statements is expressed as follows:

$$\begin{aligned} \underset{F,P}{\text{minimize}} \quad & \mathbf{Tr}(P) \\ \text{subject to:} \quad & M^T P M - P + \int_0^\delta M(\tau)^T Q M(\tau) d\tau + \delta F^T R F = 0, \\ & P \succ 0, \end{aligned}$$

$$\begin{aligned} \underset{F,P}{\text{minimize}} \quad & \mathbf{Tr}(P) \\ \text{subject to:} \quad & M^T P M - P + \int_0^\delta M(\tau)^T Q M(\tau) d\tau + \delta F^T R F \preceq 0, \\ & P \succ 0, \end{aligned}$$

$$\begin{aligned} \underset{F,P}{\text{minimize}} \quad & \mathbf{Tr}(P) \\ \text{subject to:} \quad & M^T P M - P + H_0 + F^T H_1^T + H_1 F + F^T H_2 F \preceq 0, \\ & P \succ 0, \end{aligned}$$

$$\begin{aligned} \underset{F,P}{\text{minimize}} \quad & \mathbf{Tr}(P) \\ \text{subject to:} \quad & \begin{bmatrix} M \\ F \end{bmatrix}^T \begin{bmatrix} P & 0 \\ 0 & H_2 \end{bmatrix} \begin{bmatrix} M \\ F \end{bmatrix} \preceq P - H_0 - F^T H_1^T - H_1 F, \\ & P \succ 0, \end{aligned}$$

$$\begin{aligned} \underset{F,P,K}{\text{minimize}} \quad & \mathbf{Tr}(P) \\ \text{subject to:} \quad & \begin{bmatrix} K & 0 & M \\ 0 & H_2^{-1} & F \\ M^T & F^T & P - H_0 - F^T H_1^T - H_1 F \end{bmatrix} \succeq 0, \\ & P \succ 0,\ K = P^{-1}. \end{aligned}$$

The equivalences up to here can be proved by using Lemmas 11, 12, and 13 and also applying the Schur complement. Due to a linear algebraic fact, $K = P^{-1}$ can be cast as $\mathbf{rank}(\begin{bmatrix} K & I \\ I & P \end{bmatrix}) = n$. Then, we get

$$\begin{aligned} \underset{P,F,K}{\text{minimize}} \quad & \mathbf{Tr}(P) \\ \text{subject to:} \quad & \begin{bmatrix} K & 0 & M \\ 0 & H_2^{-1} & F \\ M^T & F^T & P - H_0 - F^T H_1^T - H_1 F \end{bmatrix} \succeq 0, \\ & P \succ 0,\ \mathbf{rank}(\begin{bmatrix} K & I \\ I & P \end{bmatrix}) = n. \end{aligned}$$

□

Remark 7. *It is known that dealing with ℓ_0 sparsity measure is generally an NP-hard problem. Thus, to make numerical computations tractable, we will substitute*

the ℓ_0 sparsity measure with weighted ℓ_1 norm which has been shown to be effective in sparsification problems [1], [18],[21]. Then, we reach to the following regularized problem:

$$\underset{P,F,K}{\textit{minimize}}\ \mathbf{Tr}(P)+\gamma\|W\circ F\|_1 \qquad (\mathbf{P4})$$

$$\textit{subject to:}\ \begin{bmatrix} K & 0 & M \\ 0 & H_2^{-1} & F \\ M^T & F^T & P-H_0-F^TH_1^T-H_1F \end{bmatrix}\succeq 0,$$

$$P\succ 0,\ \mathbf{rank}(\begin{bmatrix} K & I \\ I & P \end{bmatrix})=n,$$

in which γ and the element-wise positive matrix $W=[W_{ij}]\in\mathbb{R}^{m\times n}$ are some given design parameters.

3.5 Bi-linear Rank Penalty Technique

The terms in optimization problem (**P4**) are all convex, except the rank constraint. This section is devoted to highlight our approach in dealing with this non-convex term.

It is known that presence of the rank constraint causes our optimization problem to become NP-hard. Therefore, we propose a technique, which is built upon the method proposed in [27, 28], to solve the rank-constrained optimization problem. Basically, this method is based on substituting the rank constraint on the symmetric matrix $\begin{bmatrix} K & I \\ I & P \end{bmatrix}$ with a positive semi-definite constraint while introducing extra convex constraints along with a bi-linear term to the cost function. Since the resulting optimization is all convex except for the auxiliary bi-linear term in the objective function, it can iteratively be solved [77, 78].

Theorem 15 ([27]). *Let us consider the rank-constrained optimization problem* (**P4**) *and define the following auxiliary optimization problem:*

$$\underset{P,F,K,G}{\textit{minimize}}\ \mathbf{Tr}(P)+\gamma\|W\circ F\|_1+\nu\mathbf{Tr}(G\begin{bmatrix}K & I\\ I & P\end{bmatrix}) \tag{\textbf{P5}}$$

$$\textit{subject to: }\begin{bmatrix}Y & 0 & M\\ 0 & H_2^{-1} & F\\ M^T & F^T & P-H_0-F^TH_1^T-H_1F\end{bmatrix}\succeq 0,$$

$$P\succ 0,\ \begin{bmatrix}K & I\\ I & P\end{bmatrix}\succeq 0,\ 0\preceq G\preceq I,\ \mathbf{Tr}(G)=n,$$

in which $\nu>0$ *is a penalty parameter. If* (**P4**) *is feasible, then there exists a constant* $\eta>0$ *for which the optimal solution* $\begin{bmatrix}K & I\\ I & P\end{bmatrix}$ *obtained from solving* (**P5**)*, satisfies*

$$\mathbf{rank}(\begin{bmatrix}K & I\\ I & P\end{bmatrix};\eta\nu^{-1})\leq n,$$

i.e., rank of $\begin{bmatrix}K & I\\ I & P\end{bmatrix}$ *is less than or equal to* n *with tolerance threshold* $\eta\nu^{-1}$ *according to Definition 2. In addition,* η *is greater than or equal to optimal value of* (**P4**).

According to the specific structure of matrix $\begin{bmatrix}K & I\\ I & P\end{bmatrix}$, the inequality

$$\mathbf{rank}(\begin{bmatrix}K & I\\ I & P\end{bmatrix})\geq n,$$

holds. As a consequence of Theorem 15, we can now solve the optimization problem

(**P5**) for an appropriately-chosen parameter ν to get a sub-optimal solution to (**P4**). For the sake of simplicity in our notations, the stack of all optimization variables excluding variable G is denoted by H. The optimization problem (**P5**) can be rewritten as:

$$\begin{aligned} &\underset{H,G}{\text{minimize}}\ \mathcal{F}(H,G) \\ &\text{subject to: } H \in \mathcal{C}_h, \quad G \in \mathcal{C}_g, \end{aligned}$$

where $\mathcal{C}_h$ is the convex set defined by constraints of (**P5**) which do not contain G, and the convex set $\mathcal{C}_g$ is generated by those constraints of (**P5**) which consist of G. Note that $\mathcal{F}(H,G)$ represents the bi-linear objective function in (**P5**). The previously mentioned reformulation enables us to implement this problem by iteratively optimizing the objective function for H and G. As a result, the main steps of this iterative method can be divided into two sub-problems: H-minimization and G-minimization problems. Since both H-minimization and G-minimization steps are convex optimization problems, they can be run in a computationally efficient manner. However, for the G-minimization, there also exists an analytic solution which is expressed in the next proposition.

Proposition 16 ([27])**.** *The optimal solution to the* G-minimization *step is given by*

$$G^* = I - \sum_{i=1}^{n} u_i {u_i}^T,$$

where vectors u_i *for* $i = 1, \ldots, n$ *are the singular vectors corresponding to the* n *largest singular values of* $\begin{bmatrix} K & I \\ I & P \end{bmatrix}$.

The following sequence of iterations is utilized to get the minimizer of (**P5**). First,

we solve the H-minimization and G-minimization sub-problems

$$H^{(k+1)} = \arg\underset{H\in\mathcal{C}_h}{\text{minimize}} \;\; \mathcal{F}(H, G^{(k)}), \tag{3.9}$$

$$G^{(k+1)} = I - \sum_{i=1}^{n} u_i^{(k+1)} u_i^{(k+1)^T}, \tag{3.10}$$

where $\begin{bmatrix} K & I \\ I & P \end{bmatrix}^{(k+1)} = \sum_{i=1}^{2n} \sigma_i^{(k+1)} u_i^{(k+1)} u_i^{(k+1)^T}$ is the corresponding singular value decomposition. The stopping criterion is applied via $\varepsilon^{(k+1)} \leq \varepsilon^*$, where ε^* is the pre-specified precision, with the following update rule:

$$\varepsilon^{(k+1)} = \frac{\|F^{(k+1)} - F^{(k)}\|_2}{\|F^{(k+1)}\|_2}. \tag{3.11}$$

In the last step of the algorithm, we truncate negligible elements, e.g., those smaller than 5×10^{-5}, of the resulting feedback controller F. The sufficiently small elements show weak couplings between the nodes in the information structure of the controller. A summary of our proposed algorithm is described in Algorithm 1.

Algorithm 1: Solution to (**P5**)

Inputs: A, B, Q, R, γ, ν, W, and ε^*.

1: *Initialization:*
   Set $G^{(0)} = I$, $\varepsilon^{(0)} > \varepsilon^*$, $F^{(0)} = 0_{m\times n}$ and $k = 0$.
2: **While** $\varepsilon^{(k)} > \varepsilon^*$ **Do**
3: Update $H^{(k+1)}$ by solving (3.9),
4: Update $G^{(k+1)}$ using the (3.10),
5: Update $\varepsilon^{(k+1)}$ using the (3.11),
6: $k \leftarrow k + 1$,
7: **End While**
8: Truncate F.

Output: F

Remark 8. *The choice of the weight matrix W plays a significant role in the sparsity-promoting properties of our method. When a proper weight matrix is not accessible,*

the weighted ℓ_1 norm technique can also be utilized to promote the sparsity of feedback controller. In this method, the weight assigned to each controller element is updated inversely proportional to the value of the corresponding matrix element obtained from the previous iteration, i.e.,

$$W_{ij}^{(k+1)} = \frac{1}{|F_{ij}^{(k)}| + \xi}, \quad \forall i, j, \tag{3.12}$$

where the constant $\xi > 0$ is opted as a relatively small constant and added to the denominator of the update rule (3.12) to guarantee the stability of the algorithm, specifically, when $F_{ij}^{(k)}$ turns out to be zero in the previous iteration [75]. It is noteworthy that, our simulation results are obtained via utilization of this update rule to the first few iterations.

Remark 9. *By putting a pre-specified upper bound on performance loss, the balance between performance loss and sparsity level can be obtained in a better way. Because, an additional regularization term has been added to the objective function to deal with rank constraint. This fact can be helpful, when we are allowed to have at most a certain level of performance loss.*

3.6 Numerical Simulations

In this section which takes an advantage of sub-optimally solving of (**P5**), for each given time period, the regularization parameter γ takes logarithmically-scaled values in some pre-specified interval. Then, for two different values of δ, solving the (**P5**), the corresponding values of $\|F\|_0$, and performance loss are computed. Finally, defining the spatio-temporal sparsity criterion

$$\text{STSC} := \|F\|_0 + \frac{c}{\delta},$$

and comparing to the traditional centralized LQR, the relationship between such a criterion and performance loss is visualized. The parameter $c \geq 0$ balances the relationship between spatial sparsity and temporal sparsity. We will set it equal to 1 in our numerical simulations. However, based on extra information, it can be chosen accordingly.

In order to depict such a relationship between spatio-temporal sparsity criterion and performance loss, we consider three cases: (i) IEEE 39-Bus Power Network (ii) Randomly-Generated Systems (iii) Spatially-Decaying Systems.

3.6.1 IEEE 39-Bus Power Network

Here, we consider the IEEE 39-bus test case which its model has been depicted in Figure 3.1. Such a power network consists of 10 generators. The model which we take an advantage of is the linearized swing equation model which is used in [86]. Assuming the $Q = I$, $R = 0.1I$, $\nu = 5000$, $\gamma \in [10^{-3}, 10^{-1}]$, and $\varepsilon^* = 10^{-2}$, Figure 3.2

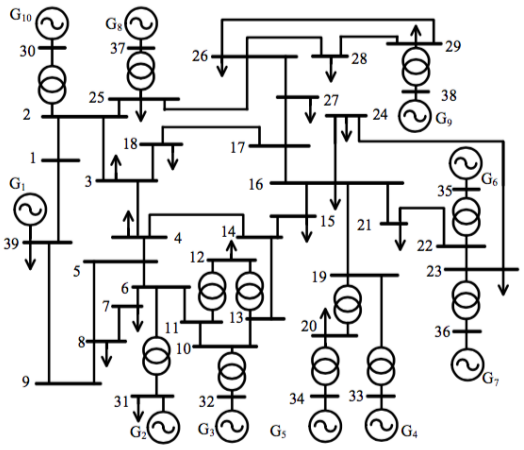


Figure 3.1: IEEE 39-bus power system model

depicts the performance loss versus spatio-temporal sparsity criterion for $\delta = 0.1$ and $\delta = 0.2$. Also, the sparsity pattern of designed controller for $\delta = 0.2$ and $\gamma = 10^{-1}$ has been shown in Figure 3.3.

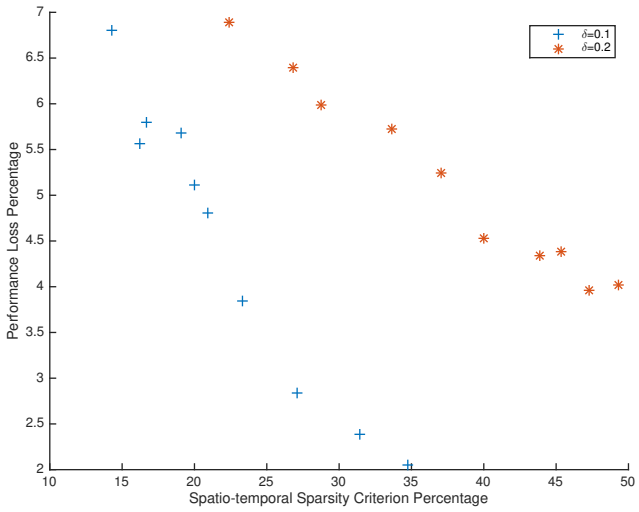


Figure 3.2: Performance loss percentage versus spatio-temporal sparsity criterion percentage for $\delta = 0.1$ and $\delta = 0.2$ (IEEE 39-bus power network).

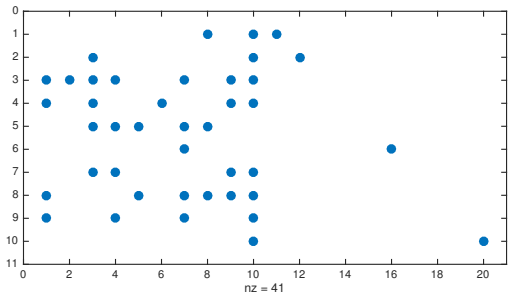


Figure 3.3: Sparsity pattern of designed controller for $\delta = 0.2$ and $\gamma = 10^{-1}$ (IEEE 39-bus power network). The corresponding performance loss percentage is equal to 6.8862 %. Blue dots represent the non-zero elements.

3.6.2 Randomly-Generated Systems

Let us consider an 10×10 randomly-generated system. In other words, suppose that the matrix A is defined as follows:

$$A = \text{randn}(10),$$

where randn(10) is a MATLAB command which produces an 10×10 normally-distributed randomly-generated matrix. Also, let us assume that other parameters

are as $B = \text{randn}(10)$, $Q = I$, $R = 0.1I$, $\nu = 5000$, $\gamma \in [10^{-7}, 10^{-5}]$, and $\varepsilon^* = 10^{-2}$. The performance loss versus spatio-temporal sparsity criterion for $\delta = 0.1$ and $\delta = 0.2$ has been visualized in Figure 3.4. Also, the sparsity pattern of designed controller for

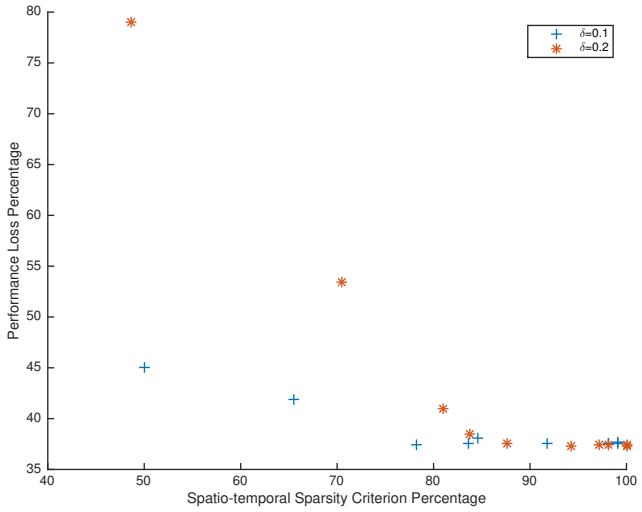


Figure 3.4: Performance loss percentage versus spatio-temporal sparsity criterion percentage for $\delta = 0.1$ and $\delta = 0.2$ (10×10 randomly-generated system).

$\delta = 0.2$ and $\gamma = 10^{-5}$ has been shown in Figure 3.5.

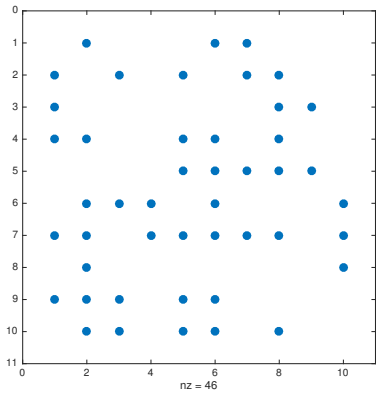


Figure 3.5: Sparsity pattern of designed controller for $\delta = 0.2$ and $\gamma = 10^{-5}$ (10×10 randomly-generated system). The corresponding performance loss percentage is equal to 79.0340 %. Blue dots represent the non-zero elements.

3.6.3 Spatially-Decaying Systems

The ij^{th} element of spatially-decaying system is defined as follows:

$$A_{ij} = \mathcal{A}_{ij} e^{-\alpha |i-j|^{\beta}},$$

where $\mathcal{A}_{ij}$ is a normally-distributed random variable with zero mean and unit variance, i.e., it belongs to $\mathcal{N}(0,1)$, α determines the band-width of matrix A and β specifies the rate of spatially-decaying in such a system.

Let us consider a 10×10 spatially-decaying system with $\alpha_A = 1$ and $\beta_A = 0.75$. Assuming the $\alpha_B = 2$, $\beta_B = 0.5$, $Q = I$, $R = 0.1I$, $\nu = 5000$, $\gamma \in [10^{-7}, 10^{-5}]$, and $\varepsilon^* = 10^{-2}$, the performance loss versus spatio-temporal sparsity criterion for $\delta = 0.3$ and $\delta = 0.6$ has been visualized in Figure 3.6. Also, the sparsity pattern of designed

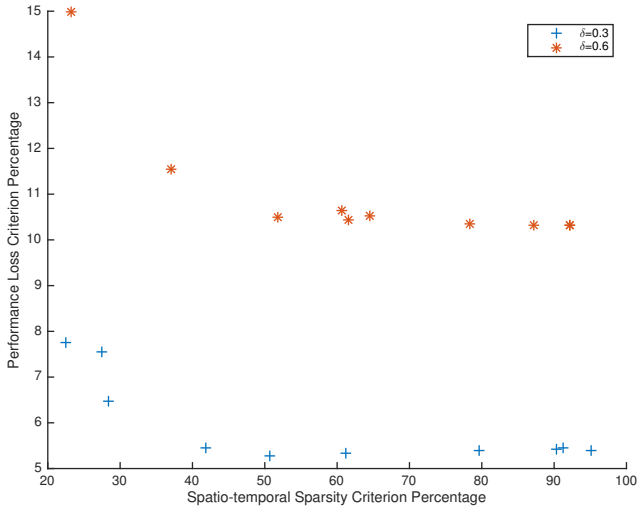


Figure 3.6: Performance loss percentage versus spatio-temporal sparsity criterion percentage for $\delta = 0.3$ and $\delta = 0.6$ (10×10 spatially-decaying system).

controller for $\delta = 0.6$ and $\gamma = 10^{-5}$ has been shown in Figure 3.7.

As it is observed in Figures 3.2, 3.4, and 3.6, there is a trade-off between performance loss and spatio-temporal sparsity criterion. One of the benefits of such a trade-off is that by prefixing the specified amount of performance loss, we can check

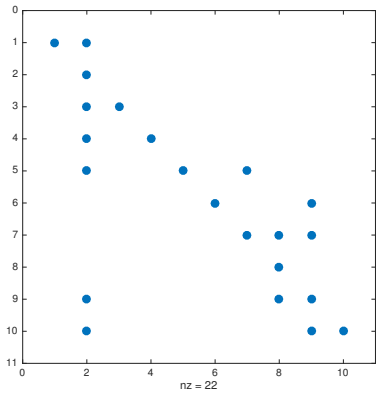


Figure 3.7: Sparsity pattern of designed controller for $\delta = 0.6$ and $\gamma = 0.00001$ (10×10 spatially-decaying system). The corresponding performance loss percentage is equal to 14.9894 %. Blue dots represent the non-zero elements.

which value of δ leads to less spatio-temporal sparsity criterion.

3.7 Conclusion

In this chapter, we present a combination of switched control methods and sparse control: spatially-invariant and temporally-known periodic time-triggered sparse LQC design. Defining the spatio-temporal sparsity criterion, a trade-off between such a criterion and performance loss is observed. Visualizing the performance/sparsity trade-off curves for different values of time period, suggests a methodological way to choose time period, when we are given a pre-specified level of performance loss. A future work can be the co-design approach to design the controller at each step of inter-execution time calculation in either periodic or aperiodic time setups. In other words, spatially-invariant and temporally-unknown periodic time-triggered sparse LQC design or spatially-varying and temporally-unknown aperiodic time-triggered sparse LQC design.

Chapter 4

Feedback Controller Sparsification Under Parametric Uncertainties

4.1 Introduction

It has been known that optimal controller design under controller structural constraints is a challenging problem. Nonetheless, numerous studies have been carried out to either propose controller design frameworks or reveal inherent structural properties of controllers for special classes of systems [3, 19, 42, 67, 87, 88].

Another concern in the design of large-scale control systems is the number of communication links between the subsystems which poses major issues especially when establishing links between nodes is very costly. Sparsifying the controller gain leads to fewer information pathways as well as fewer controller sensors and actuators. As a result, the design of controller gains with minimum number of non-zero elements can mitigate the communication overflow issues emergent in large interconnected systems. In the sparsity-promoting control problem, the ultimate objective is to minimize the number of feedback links without losing much performance. This is achieved by incorporating additional functions into the optimization cost function to

penalize the number of communication links. The problem has been addressed by a number of researchers, who opted for various techniques to tackle the inherently non-convex problem [6, 9, 10, 15, 20, 68].

In [15], the authors proposed a novel framework in which all non-convexities are lumped into a rank constraint further enabling it to address output feedback problems with norm constraints on the input/output signals. In some recent papers an unconventional approach to synthesize near optimal sparse controllers has been adopted. This proposed sparse controller design framework is founded based on the assumption that a pre-designed well-performing controller is available and the ultimate goal is to obtain a sparse feedback controller approximating the attributes and qualities of the original well-preforming controller [21, 27, 40, 89]. In this chapter, we extend the work published in [21] by introducing parametric time-varying uncertainty to the open loop system. We then show that by utilizing the results from $\mathcal{H}_2$ and $\mathcal{H}_\infty$ control [90, 91], this novel approach to robust controller design can equivalently be reformulated into a rank-constrained optimization where all non-convexities are collected into the fixed rank constraint. The next notable improvement in the current chapter is that we replaced the ADMM algorithm [21] with our novel algorithm, which employs a bi-linear optimization to reach the sub-optimal solution of the rank-constrained optimization problem. We also show that our optimization parameters can be tuned such that the rank of the optimal solution of our proposed minimization satisfies the constraint with arbitrary tolerance. We, then, use the proposed procedure to study the controller spairsification problem in a power network model.

This chapter is structured as follows: Section 4.2 provides key definitions and notations used throughout the chapter. In Section 4.3, we formally state the problem we aim to solve. In Sections 4.4 and 4.5, we elaborate how our problem can equivalently be reformulated into an optimization problem constrained to several linear matrix inequalities and a fixed rank constraint. Section 4.6 provides insight into our

proposed algorithm and states several related results. The results of our numerical simulations are presented in Section 4.7. Finally, we end with concluding remarks in Section 4.8.

4.2 Mathematical Notations

Throughout this chapter, matrices are customarily referred to with upper-case letters. The vectors, on the other hand, are symbolized by lower-case letters with components denoted by the same letter using subscripts. A unit vector with its i^{th} element equal to one is denoted by e_i. The set of real numbers is denoted by $\mathbb{R}$. The space of n by m matrices with real elements is indicated by $\mathbb{R}^{n\times m}$. The set of real matrices with non-negative (positive) elements is represented by $\mathbb{R}_{+}^{n\times m}$ ($\mathbb{R}_{++}^{n\times m}$). The n by n identity matrix is denoted by I_n. The vector of singular values of matrix X is denoted by $\sigma(X)$. The element-wise product of two matrices, i.e., Hadamard product, is represented by $\circ$. If $X = [X_{ij}]$, then the matrix $|X|$ is the element-wise absolute value of X, i.e., $|X| = [|X_{ij}|]$. The number of non-zero elements of a matrix is denoted by $\|.\|_0$ while $\|.\|_1$ denotes ℓ_1 norm, $\|.\|_2$ denotes the maximum singular value, and the $\mathcal{L}_2$-norm is defined by

$$\|x\|_{\mathcal{L}_2(\mathbb{R}^n)} := \Big(\int_0^\infty \|x(t)\|_2^2 \, dt \Big)^{1/2}.$$

Whenever it is not confusing, we use $\mathcal{L}_2$ instead of $\mathcal{L}_2(\mathbb{R}^n)$. $\mathbf{Tr}(.)$ and $\mathbf{rank}(.)$ denote the trace and rank of the matrix operands, respectively. The operator $\mathbf{diag}(.)$ constructs block diagonal matrix from input arguments.

Definition 3. *For a given $\epsilon > 0$ and matrix X, we say that rank of X is k with tolerance ϵ, and it is denoted by $\mathbf{rank}(X;\epsilon)$, if exactly k singular values of X are larger than or equal to ϵ.*

A matrix is said to be Hurwitz if all of its eigenvalues lie within the open left half of the complex plane. A real symmetric matrix is said to be positive definite (semi-definite) if all of its eigenvalues are positive (non-negative). $\mathbb{S}^n_{++}$ ($\mathbb{S}^n_{+}$) denotes the space of positive definite (positive semi-definite) real symmetric matrices, and the notation $X \succeq Y$ ($X \succ Y$) means $X - Y \in \mathbb{S}^n_{+}$ ($X - Y \in \mathbb{S}^n_{++}$).

Remark 10. *For simplicity of our notations, we will use a new notation in statements of theorems, where we use an asterisk '*' to represent the upper triangular sub-blocks of symmetric matrices. Moreover, in the occasions when the optimal solutions of the optimization problems in these theorems do not depend on some of the sub-blocks of matrices, we use a dash '-' to represent such sub-blocks with no apparent utilization in the problem.*

4.3 Problem Formulation

4.3.1 LTI Systems with Parametric Uncertainties

The focus of this chapter is on the following class of uncertain linear time-invariant (LTI) systems that are defined by the state space realization[1]

$$\begin{aligned} \dot{x}(t) &= [A + \Delta_A]x(t) + [B_1 + \Delta_{B_1}]u(t) + B_2 d(t), \\ y(t) &= Cx(t), \end{aligned} \tag{4.1}$$

where $x(t) \in \mathbb{R}^n$ is the state vector, $u(t) \in \mathbb{R}^m$ is the control input and $d(t) \in \mathbb{R}^p$ represents the exogenous disturbance input. We assume that the matrices $A \in \mathbb{R}^{n\times n}$, $B_1 \in \mathbb{R}^{n\times m}$, $B_2 \in \mathbb{R}^{n\times p}$, and $C \in \mathbb{R}^{q\times n}$ are constant real matrices describing the dynamics of the nominal system, whereas Δ_A and Δ_{B_1} represent the parameter uncertainties of the matrices A and B_1, respectively. In this chapter, we consider a

[1] It is assumed that the pair (A, B_1) is controllable and (A, C) is detectable.

special uncertainty structure expressed by

$$\begin{bmatrix} \Delta_A & \Delta_{B_1} \end{bmatrix} = D\Delta \begin{bmatrix} E_A & E_{B_1} \end{bmatrix}, \tag{4.2}$$

where D, E_A and E_{B_1} are known constant real matrices with appropriate dimensions, which characterize the structure of the uncertainties, while Δ is an unknown i by j real matrix which is constrained by

$$\Delta^T \Delta \preceq \rho^2 I_j. \tag{4.3}$$

This class of uncertain linear systems was initially reported by Petersen in papers [92, 93] and later thoroughly addressed by Khargonekar *et* al. [94].

4.3.2 Controller Sparsification via $\mathcal{H}_p$ Approximations

Suppose that a pre-designed well-performing controller, namely $\hat{K}$, is readily available and the nominal system controlled by such a controller, represented by $\hat{\mathcal{S}}$, has all the desired characteristics. The objective is to synthesize a constant gain output feedback controller of the form

$$u(t) = Ky(t), \quad K \in \mathcal{K}, \tag{4.4}$$

with minimum number of non-zero elements, while minimizing the performance deterioration from that of the closed-loop system $\hat{\mathcal{S}}$ under parametric uncertainties. In (4.4), $\mathcal{K}$ denotes a set of admissible feedback gains which holds desirable properties such as pre-defined communication layout.

Assumption 3. *It is assumed that the set $\mathcal{K}$ is convex.*

It should be emphasized that this assumption does not offer any premise on characterization of the set of all stabilizable output feedback controllers. In our follow-up

discussions, we will show that if our proposed optimal control design is feasible, then the resulting output feedback controller will be stabilizing and satisfy the structural constraint $K \in \mathcal{K}$.

There are numerous applications associated with such convexly constrained controller design, such as power grids or multi-UAV systems. It is sometimes practically infeasible to establish some specific communication links between particular nodes due to the nodes distant locations or security issues in networks. There are also cases where the attenuation/amplification in certain feedback paths is upper bounded, due to technological shortcomings. Such restrictions are addressed by forcing the corresponding controller elements to be contained in a convex set.

Our goal is to solve the following ℓ_0-regularized optimal control problem to compute a sparse output feedback controller under parametric uncertainties:

$$\underset{K,\varepsilon_y,\varepsilon_{\mathcal{S}}}{\text{minimize}}\ \varepsilon_{\mathcal{S}} + \lambda_1\varepsilon_y + \lambda_2\|K\|_0 \tag{4.5a}$$

$$\text{subject to: } K \in \mathcal{K}, \tag{4.5b}$$

$$\mathcal{S} : \text{Stable}, \tag{4.5c}$$

$$\|y_{\mathcal{S}} - y_{\hat{\mathcal{S}}}\|_{\mathcal{L}_2} < \varepsilon_y\|d\|_{\mathcal{L}_2}, \tag{4.5d}$$

$$\|\mathcal{S} - \hat{\mathcal{S}}\|^2_{\mathcal{H}_2} \leq \varepsilon_{\mathcal{S}}, \tag{4.5e}$$

in which $\|.\|_{\mathcal{H}_2}$ is the well-known $\mathcal{H}_2$ norm. Nominal closed-loop system $\hat{\mathcal{S}}$ is a previously designed desired optimal closed-loop system with output signal $y_{\hat{\mathcal{S}}}$ and $\mathcal{S}$ is the resulting system by closing the loop using sparse feedback controller K. The output signal of $\mathcal{S}$ is denoted by $y_{\mathcal{S}}$. In order to promote sparsity of feedback gain matrix K, the ℓ_0 measure of K, which is denoted by $\|K\|_0$, has been added to the cost function. Two design parameters λ_1 and λ_2 are introduced to achieve desired trade-off between performance loss and sparsity.

In the optimal control problem (4.5), constraint (4.5e) is included to ensure that

the nominal closed-loop system $\hat{\mathcal{S}}$ is well-approximated by a closed-loop system controlled by a sparse controller K. To enhance temporal features of our approximation, we also incorporate another requirement into our design scheme, characterized by constraint (4.5d). This constraint guarantees that the energy level of the difference between the output signals of the two closed-loop systems remains under a pre-specified level ϵ_y when both closed-loop systems are excited by a disturbance input d with unit norm.

The goal of this chapter is to study the effect of parametric uncertainties on the best achievable levels of sparsity. However, finding the optimal solution of the problem (4.5) is inherently NP-hard; see our discussion in Section 4.5. In Section 4.6, we will propose a tractable approximation algorithm to solve this problem. The following sections discuss the equivalent problem reformulation exploited in numerically solving our optimization problem.

4.4 Equivalent Reformulation

The first two terms in the cost function of the optimization problem (4.5) can be simplified into the $\mathcal{H}_2/\mathcal{H}_\infty$ norms of an augmented system, namely $\bar{\mathcal{S}}$, constructed by the following state space realization matrices:

$$\bar{A} = \mathbf{diag}(\bar{A}_{11}, A + B_1\hat{K}C),\ \bar{B} = \begin{bmatrix} B_2^T & B_2^T \end{bmatrix}^T,\ \bar{C} = \begin{bmatrix} C & -C \end{bmatrix}, \tag{4.6}$$

where $\bar{A}_{11} = [A+\Delta_A]+[B_1+\Delta_{B_1}]KC$. As it can be seen, the system $\bar{\mathcal{S}}$ represents the difference between the nominal system controlled by the pre-designed controller and the uncertain system, stabilized by closing its feedback loop using a sparse controller. Hence, we can re-formulate our problem into the $\mathcal{H}_2/\mathcal{H}_\infty$ norm minimization of the

augmented system as follows:

$$\begin{aligned} \underset{K,\varepsilon_y,\varepsilon_{\mathcal{S}}}{\text{minimize}} \ \underset{\Delta_A,\Delta_{B_1}}{\text{maximize}} \ & \varepsilon_{\mathcal{S}} + \lambda_1\varepsilon_y + \lambda_2\|K\|_0 \\ \text{subject to: } & K \in \mathcal{K}, \\ & \bar{A}_{11} \text{ Hurwitz}, \\ & \|\bar{C}(sI-\bar{A})^{-1}\bar{B}\|_{\mathcal{H}_\infty} < \varepsilon_y, \\ & \|\bar{C}(sI-\bar{A})^{-1}\bar{B}\|_{\mathcal{H}_2}^2 \leq \varepsilon_{\mathcal{S}}. \end{aligned} \tag{4.7}$$

In problem (4.7), the attempt is to minimize the worst case gap between the frequency response of the systems in terms of a weighted sum of the $\mathcal{H}_2$ and $\mathcal{H}_\infty$ norms. Therefore, unlike the design schemes introduced in [15, 68], the approach proposed in this chapter allows us to exploit the advantages offered by other controller design schemes in the sparse controller design. In the next section, we show that the optimization problem (4.7) includes bi-linear matrix inequality constraints mainly due to the existence of the Lyapunov stability conditions. Here, we intend to employ the idea of lumping all nonlinear constraints into a rank-constrained problem, proposed in [15], to rewrite problem as a rank-constrained optimization. Based on the obtained reformulation, it is possible to either develop heuristics to sub-optimally solve the problem or provide necessary and sufficient conditions for the feasibility of the points with particular desired costs.

4.5 Fixed Rank Optimization Reformulation

The approach adopted in this chapter is based on solving the problem of sparse controller approximation via rank-constrained optimization. Hence, we start by stating the main lemmas which helps us cast the constraints of the optimization problem as rank-constrained linear matrix inequalities.

Lemma 17 ([15]). *Let $\mathcal{U} \in \mathbb{R}^{n\times n}$, $\mathcal{V} \in \mathbb{R}^{n\times m}$, $\mathcal{W} \in \mathbb{R}^{m\times m}$, and $\mathcal{Y} \in \mathbb{R}^{m\times n}$, with $\mathcal{U} \succ 0$. Then,* $\mathbf{rank}(\mathcal{M}) = n$ *if and only if $\mathcal{W} = \mathcal{Y}\mathcal{U}\mathcal{Y}^T$, $\mathcal{V}^T = \mathcal{Y}\mathcal{U}$, and $\mathcal{Z} = \mathcal{U}^{-1}$ where*

$$\mathcal{M} = \begin{bmatrix} \mathcal{U} & \mathcal{V} & I_n \\ \mathcal{V}^T & \mathcal{W} & \mathcal{Y} \\ I_n & \mathcal{Y}^T & \mathcal{Z} \end{bmatrix}.$$

The above lemma can be utilized to collect almost all non-convex terms of the optimization problems in one and only one constraint in the form of a rank constraint. There are a number of algorithms proposed to solve rank-constrained optimization problems [72–74, 95, 96]. In this manuscript, we aim to render such algorithms applicable in solving our inherently nonlinear controller sparsification problem by collecting various forms of non-convex/combinatorial constraints into a fixed rank constraint.

As a first step, we show how the $\mathcal{H}_2$ norm of an uncertain system can be formulated by rank-constrained linear matrix inequalities.

Lemma 18 ([27]). *Given a strictly proper uncertain linear system $\mathcal{P}$ with state space realization $(\mathcal{A} + \Delta_{\mathcal{A}}, \mathcal{B}, \mathcal{C})$, where $\mathcal{A} \in \mathbb{R}^{n\times n}$, $\mathcal{B} \in \mathbb{R}^{n\times m}$, $\mathcal{C} \in \mathbb{R}^{q\times n}$, $\Delta_{\mathcal{A}} = \mathcal{D}\Delta\mathcal{E}$ and $\Delta^T\Delta \preceq \rho^2 I_j$, then $\mathcal{P}$ is stable and $\|\mathcal{P}\|_{\mathcal{H}_2}^2 \leq \gamma$ if and only if there exists a positive*

definite matrix $\mathcal{X} \succ 0$ *and a positive scalar* ε *such that*

$$\begin{aligned}
\mathbf{Tr}(\mathcal{C}\mathcal{X}\mathcal{C}^T) &\leq \gamma, \\
\begin{bmatrix} \mathcal{Y}_1 + \mathcal{Y}_1^T + \mathcal{B}\mathcal{B}^T + \varepsilon\rho\mathcal{D}\mathcal{D}^T & \sqrt{\rho}\mathcal{Y}_2 \\ \sqrt{\rho}\mathcal{Y}_2^T & -\varepsilon I_j \end{bmatrix} &\prec 0, \\
\mathbf{rank} \begin{bmatrix} \mathcal{X} & * & * & * \\ \mathcal{Y}_1^T & - & * & * \\ \mathcal{Y}_2^T & - & - & * \\ I_n & \mathcal{A}^T & \mathcal{E}^T & - \end{bmatrix} &= n.
\end{aligned}$$

Similar to Lemma 18, which paves the way in casting the $\mathcal{H}_2$ norm term in our optimal controller sparsification problem, as a rank-constrained optimization problem, the $\mathcal{H}_\infty$ norm term of problem (4.7) can also be equivalently represented with a set of rank-constrained linear matrix inequalities. In the next lemma, we prove such equivalence, which later helps in accommodating the whole problem of controller sparsification under parametric uncertainties into the framework of rank-constrained optimization.

Lemma 19 ([27])**.** *Suppose a strictly proper uncertain LTI plant* $\mathcal{P}$*, represented in the state space triplet* $(\mathcal{A}+\Delta_\mathcal{A}, \mathcal{B}, \mathcal{C})$*, where* $\mathcal{A} \in \mathbb{R}^{n\times n}$*,* $\mathcal{B} \in \mathbb{R}^{n\times m}$*,* $\mathcal{C} \in \mathbb{R}^{q\times n}$*,* $\Delta_\mathcal{A} = \mathcal{D}\Delta\mathcal{E}$ *and* $\Delta^T\Delta \preceq \rho^2 I_j$*, then the system is stable with* $\mathcal{H}_\infty$ *norm less than* γ *if and only if*

there exists a positive definite matrix $\mathcal{X} \succ 0$ *and a positive scalar* $\varepsilon > 0$ *satisfying*

$$\begin{bmatrix} \mathcal{Y}_1 + \mathcal{Y}_1^T + \varepsilon\rho\mathcal{D}\mathcal{D}^T & * & * & * \\ \mathcal{B}^T & -\gamma I_m & * & * \\ (\mathcal{C}\mathcal{X}) & 0 & -\gamma I_q & * \\ \sqrt{\rho}\mathcal{Y}_2^T & 0 & 0 & -\varepsilon I_j \end{bmatrix} \prec 0,$$

$$\mathbf{rank} \begin{bmatrix} \mathcal{X} & * & * & * \\ \mathcal{Y}_1^T & - & * & * \\ \mathcal{Y}_2^T & - & - & * \\ I_n & \mathcal{A}^T & \mathcal{E}^T & - \end{bmatrix} = n.$$

Consequently, we can reformulate the problem (4.7) into a rank-constrained problem, as described in the sequel.

Theorem 20 ([27])**.** *The optimization problem (4.7) is equivalent to the following*

rank-constrained optimization problem:

$$\underset{K,\varepsilon_y,\varepsilon_{\mathcal{S}}}{\text{minimize}}\ \varepsilon_{\mathcal{S}} + \lambda_1\varepsilon_y + \lambda_2\|K\|_0 \tag{4.8}$$

$$\text{subject to: } K \in \mathcal{K},$$

$$X_r \succ 0, \quad r = 1, 2,$$

$$\varepsilon_r > 0, \quad r = 1, 2,$$

$$\mathbf{Tr}(\bar{C}X_1\bar{C}^T) \leq \varepsilon_{\mathcal{S}},$$

$$\begin{bmatrix} P_1 + \bar{B}\bar{B}^T & * \\ \sqrt{\rho}Y_2^T & -\varepsilon_1 I_j \end{bmatrix} \prec 0,$$

$$\begin{bmatrix} P_2 & * & * & * \\ \bar{B}^T & -\varepsilon_y I_p & * & * \\ (\bar{C}X_2) & 0 & -\varepsilon_y I_q & * \\ \sqrt{\rho}Y_4^T & 0 & 0 & -\varepsilon_2 I_j \end{bmatrix} \prec 0,$$

$$\mathbf{rank}(M_1) = 2n,$$

where

$$P_r = Y_{2r-1} + Y_{2r-1}^T + \varepsilon_r \rho \bar{D}\bar{D}^T, \qquad r = 1, 2,$$

$$M_1 = \begin{bmatrix} X_1 & * & * & * & * \\ Y_1^T & - & * & * & * \\ Y_2^T & - & - & * & * \\ X_2 & Y_3 & Y_4 & - & * \\ I_{2n} & A_{cl}^T & E_{cl}^T & - & - \end{bmatrix},$$

$$A_{cl} = \mathbf{diag}(A + B_1 K C, A + B_1 \hat{K} C) \in \mathbb{R}^{2n \times 2n},$$

$$\bar{D} = \begin{bmatrix} D^T & 0 \end{bmatrix}^T \in \mathbb{R}^{2n \times i},$$

$$E_{cl} = \begin{bmatrix} E_A + E_{B_1} K C & 0 \end{bmatrix} \in \mathbb{R}^{j \times 2n}.$$

The next corollary is now immediate.

Corollary 21. *The optimization problem (4.7) can equivalently be cast as the following rank-constrained optimization problem:*

$$\underset{K,\varepsilon_y,\varepsilon_{\mathcal{S}}}{\text{minimize}}\ \varepsilon_{\mathcal{S}} + \lambda_1\varepsilon_y + \lambda_2\|K\|_0 \tag{4.9a}$$

$$\text{subject to: } K \in \mathcal{K}, \tag{4.9b}$$

$$X_r \succ 0, \quad r = 1, 2, \tag{4.9c}$$

$$\varepsilon_r > 0, \quad r = 1, 2, \tag{4.9d}$$

$$\mathbf{Tr}(\bar{C}X_1\bar{C}^T) \leq \varepsilon_{\mathcal{S}}, \tag{4.9e}$$

$$\begin{bmatrix} Q_1 + \bar{B}\bar{B}^T + \varepsilon_1\rho\bar{D}\bar{D}^T & * \\ \sqrt{\rho}R_1 & -\varepsilon_1 I_j \end{bmatrix} \prec 0, \tag{4.9f}$$

$$\begin{bmatrix} Q_2 + \varepsilon_2\rho\bar{D}\bar{D}^T & * & * & * \\ \bar{B}^T & -\varepsilon_y I_p & * & * \\ (\bar{C}X_2) & 0 & -\varepsilon_y I_q & * \\ \sqrt{\rho}R_2 & 0 & 0 & -\varepsilon_2 I_j \end{bmatrix} \prec 0, \tag{4.9g}$$

$$\mathbf{rank}(M_2) = 2n, \tag{4.9h}$$

where

$$Q_r = X_r A_o^T + A_o X_r + Y_r B_K^T + B_K^T Y_r^T, \quad r = 1, 2, \tag{4.10a}$$

$$R_r = E_o X_r + E_{B_1} Y_r^T, \qquad r = 1, 2, \tag{4.10b}$$

$$M_2 = \begin{bmatrix} X_1 & * & * & * \\ Y_1^T & - & * & * \\ X_2 & Y_2 & - & * \\ I_{2n} & (KC_K)^T & - & - \end{bmatrix}, \tag{4.10c}$$

$$A_o = \mathbf{diag}(A, A + B_1 \hat{K} C) \in \mathbb{R}^{2n \times 2n}, \tag{4.10d}$$

$$\bar{D} = \begin{bmatrix} D^T & 0 \end{bmatrix}^T \in \mathbb{R}^{2n \times i}, \tag{4.10e}$$

$$E_o = \begin{bmatrix} E_A & 0 \end{bmatrix} \in \mathbb{R}^{j \times 2n}, \tag{4.10f}$$

$$C_K = \begin{bmatrix} C & 0 \end{bmatrix} \in \mathbb{R}^{q \times 2n}, \tag{4.10g}$$

$$B_K = \begin{bmatrix} B_1^T & 0 \end{bmatrix}^T \in \mathbb{R}^{2n \times m}. \tag{4.10h}$$

4.6 A Tractable Approximation Algorithm for Computing Sparse Feedback Controllers

Although both optimizations (4.8) and (4.9) can be utilized to solve our controller sparsification problem, we choose to only implement the one formulated in (4.9). The terms in our optimization problem are all convex except the sparsity-promoting term in the cost function and the rank constraint. This section intends to shed light on our approach in dealing with these two non-convex and combinatorial terms.

As for the sparsity-promoting term of the objective function, since the ℓ_0 norm is an integer-valued function, utilizing it in our formulation introduces the complications of combinatorial optimization. In order to reduce the complexity of sparse

vector/matrix recovery problems, we employ the ℓ_1 norm and its weighted versions. This is because convex surrogates of the ℓ_0 norm are among the most common functions used to measure the sparsity and have been utilized in diverse applications [68, 97]. Therefore, we have

$$\begin{aligned} &\underset{K,\varepsilon_y,\varepsilon_{\mathcal{S}}}{\text{minimize}}\ \varepsilon_{\mathcal{S}} + \lambda_1\varepsilon_y + \lambda_2\|W \circ K\|_1 \\ &\text{subject to: } (4.9b) - (4.9h), \\ &\qquad\qquad (4.10a) - (4.10h), \end{aligned} \tag{4.11}$$

where the weight matrix $W = [W_{ij}] \in \mathbb{R}^{m\times q}$ is element-wise positive and chosen according to the objectives of the problem.

The convex relaxation of the sparsity-promoting term in the cost function of (4.11) leaves us with an optimization problem in which non-convexity only arises in the form of a rank constraint, i.e., $\mathbf{rank}(M_2) = 2n$. It is known that presence of the rank constraint still causes our optimization problem to become NP-hard. Therefore, we propose a technique, which is built upon the method studied in [28], to solve the rank constraint optimization problem. In a nutshell, this method is based on substituting the rank constraint on the symmetric matrix M_2 with a positive semidefinite constraint while introducing extra convex constraints along with a bi-linear term to the cost function. Since the resulting optimization is all convex except for the auxiliary bi-linear term in the objective function, it can iteratively be solved [77, 78].

Theorem 22 ([27])**.** *Let us consider the rank-constrained optimization problem (4.11)*

Algorithm 1: Solution to problem (4.12)

Inputs: A, B_1, B_2, C, Q, R, λ_1, λ_2, ν, $\mathcal{K}$, W, ρ, and ε^*.

1: *Initialization:*
   Set $Y^{(0)} = I_{6n+m}$, $\varepsilon^{(0)} > \varepsilon^*$, $K^{(0)} = 0_{m\times q}$ and $k = 0$.
2: **While** $\varepsilon^{(k)} > \varepsilon^*$ **Do**
3: Update $Z^{(k+1)}$ by solving (4.14),
4: Update $Y^{(k+1)}$ using the equation (4.15),
5: Update $\varepsilon^{(k+1)}$ using the equation (4.16),
6: $k \leftarrow k + 1$,
7: **End While**
8: Truncate K.

Output: K

and define the following auxiliary optimization problem:

$$\begin{aligned}
&\underset{Y,K,\varepsilon_y,\varepsilon_{\mathcal{S}}}{\text{minimize}}\ \varepsilon_{\mathcal{S}} + \lambda_1\varepsilon_y + \lambda_2\|W \circ K\|_1 + \nu\mathbf{Tr}(YM_2) \qquad (4.12)\\
&\text{subject to: } (4.9b) - (4.9g),\\
&\qquad\qquad (4.10a) - (4.10h),\\
&\qquad\qquad 0 \preceq Y \preceq I_{6n+m},\\
&\qquad\qquad \mathbf{Tr}(Y) = 4n + m,\\
&\qquad\qquad M_2 \succeq 0,
\end{aligned}$$

in which $\lambda_1, \lambda_2, \nu > 0$ and the element-wise positive matrix W are some given design parameters. If problem (4.11) is feasible, then there exists a constant $\eta > 0$ for which the optimal solution M_2 from solving (4.12) *satisfies*

$$\mathbf{rank}(M_2; \eta\nu^{-1}) \ \leq\ 2n,$$

i.e., rank of M_2 is less than or equal to $2n$ with tolerance threshold $\eta\nu^{-1}$ according to Definition 3.

We should remind that according to (4.10c) and the specific structure of matrix

M_2 it is always true that $\mathbf{rank}(M_2) \geq 2n$. As a result of the previous theorem, we can now solve the optimization problem (4.12) for an appropriately-chosen parameter ν to obtain a sub-optimal solution to the problem (4.11). For the simplicity of our notations, the letter Z is used to denote the stack of all optimization variables excluding variable Y. The optimization problem (4.12) can be rewritten as follows:

$$\begin{aligned} &\underset{Z,Y}{\text{minimize}}\ \mathcal{F}(Z,Y) \\ &\text{subject to: } Z \in \mathcal{C}_z, \quad Y \in \mathcal{C}_y, \end{aligned}$$

where $\mathcal{C}_z$ is the convex set defined by the constraints (4.9b)-(4.9g), (4.10a)-(4.10h), along with $M_2 \succeq 0$, and the convex set $\mathcal{C}_y$ is generated by $\mathbf{Tr}(Y) = 4n + m$ and $0 \preceq Y \preceq I_{6n+m}$. Needless to say that $\mathcal{F}(Z,Y)$ represents the bi-linear objective function in the minimization problem (4.12). The above reformulation allows us to carry out this problem by iteratively optimizing the objective function for Z and Y. As a result, the main steps of this iterative method can be divided into two sub-problems, Z-minimization and Y-minimization problems. As both Z-minimization and Y-minimization steps are convex optimizations, they can be performed in a computationally efficient manner. However, for the Y-minimization, there also exists an analytic solution, stated in the next theorem.

Theorem 23 ([27])**.** *The optimal solution to the* Y-minimization *step is given by*

$$Y^* = I_{6n+m} - \sum_{i=1}^{2n} u_i {u_i}^T, \tag{4.13}$$

where vectors u_i *for* $i = 1, \ldots, 2n$ *are the singular vectors corresponding to the* $2n$ *larger singular values of* M_2*.*

4.6.1 Summary of the Approximation Algorithm

We utilize the following sequence of iterations to obtain the minimizer of the constrained problem (4.12). First, we solve the Z-minimization and Y-minimization subproblems

$$Z^{(k+1)} = \arg\underset{Z\in\mathcal{C}_z}{\text{minimize}} \;\; \mathcal{F}(Z, Y^{(k)}), \tag{4.14}$$

$$Y^{(k+1)} = I_{6n+m} - \sum_{i=1}^{2n} u_i^{(k+1)} u_i^{(k+1)^T}, \tag{4.15}$$

where $M_2^{(k+1)} = \sum_{i=1}^{6n+m} \sigma_i^{(k+1)} u_i^{(k+1)} u_i^{(k+1)^T}$ is the singular value decomposition of $M_2^{(k+1)}$. The stopping criterion is established by $\varepsilon^{(k+1)} \leq \varepsilon^*$, where ε^* is the given desired precision, with the following update law:

$$\varepsilon^{(k+1)} = \frac{\|K^{(k+1)} - K^{(k)}\|_2}{\|K^{(k+1)}\|_2}. \tag{4.16}$$

In the last step of the algorithm, we truncate negligible elements of the resulting feedback gain K, e.g., those smaller than 5×10^{-5},. These small elements show very weak couplings between the nodes in the information structure of the controller. A summary of our proposed algorithm is described in Algorithm 1.

Remark 11. *The choice of the weight matrix W plays an important role in the sparsity-promoting properties of our method. When a proper weight matrix is not accessible, the weighted ℓ_1 norm technique can also be employed to enhance the sparse controller recovery. In this method, the weight assigned to each controller element is updated inversely proportional to the value of the corresponding matrix element recovered from the previous iteration, i.e.,*

$$W_{ij}^{(k+1)} = \frac{1}{|K_{ij}^{(k)}| + \xi}, \quad \forall i, j, \tag{4.17}$$

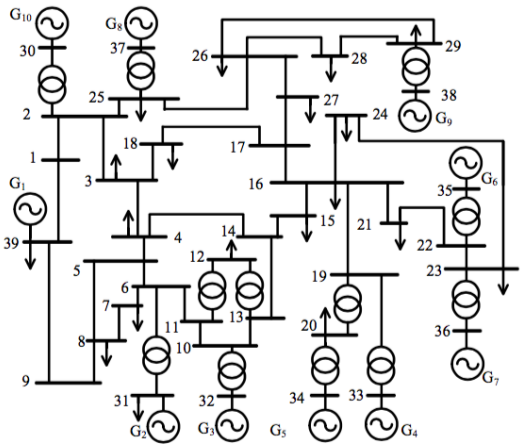


Figure 4.1: IEEE 39-bus power system model

where the constant $\xi > 0$ which is chosen as a relatively small constant, is augmented to the denominator of the update law (4.17) to guarantee the stability of the algorithm, especially, when $K_{ij}^{(k)}$ turns out to be zero in the previous iteration [75]. It should be noted that, our simulation results are obtained by incorporating this update law into the first few iterations.

4.7 Numerical Simulations

In this section, we examine our proposed method by utilizing the IEEE 39-Bus New England power system which consists of $N_G = 10$ synchronous generators. Specifically, we take advantage of the state-space model provided by [86], which is a linearized

state-space model of swing equations is characterized by (4.1), where

$$x = \begin{bmatrix} \theta^T & \omega^T \end{bmatrix}^T,\ A = \begin{bmatrix} 0 & I \\ -\tilde{M}^{-1}L & -\tilde{M}^{-1}\tilde{D} \end{bmatrix},\ B_1 = \begin{bmatrix} 0 \\ \tilde{M}^{-1} \end{bmatrix},\ B_2 = B_1,$$

$$\tilde{M} = \mathbf{diag}(\tilde{M}_1, \cdots, \tilde{M}_{N_G}),\ \tilde{D} = \mathbf{diag}(\tilde{D}_1, \cdots, \tilde{D}_{N_G}),\ u = Kx,\ K = \begin{bmatrix} K_\theta & K_\omega \end{bmatrix},$$

$$\omega = \dot{\theta}.$$

The Laplacian or admittance matrix L satisfies the following equations:

$$l_{ij} = -b_{ij}^{Kron},\ l_{ii} = \sum_{k=1, k\neq i}^{N_G} b_{ik}^{Kron},$$

where B^{Kron} is the susceptance matrix of the corresponding Kron reduced admittance matrix.

The power network utilized in our simulation is depicted in Figure 4.1, and its parameters, in per unit system, are presented in Table 4.1.

We define the following performance metrics which quantify the deviation in $\mathcal{H}_2$ and $\mathcal{H}_\infty$ norms casued by the sparsification process. They also allow for comparison of the sparsification performance in the absence and presence of uncertainty on the system matrices.

$$\mathcal{R}_2 = \frac{\|\mathcal{S} - \hat{\mathcal{S}}\|_{\mathcal{H}_2}}{\|\hat{\mathcal{S}}\|_{\mathcal{H}_2}}, \tag{4.18}$$

$$\mathcal{R}_\infty = \frac{\|\mathcal{S} - \hat{\mathcal{S}}\|_{\mathcal{H}_\infty}}{\|\hat{\mathcal{S}}\|_{\mathcal{H}_\infty}}. \tag{4.19}$$

Now, we assume that the susceptance corresponding to the link between two randomly-chosen nodes i_1 and i_2 is affected by an uncertainty of the form

$$\rho = \rho_{rel}(b_{i_1 i_2}^{Kron}),$$

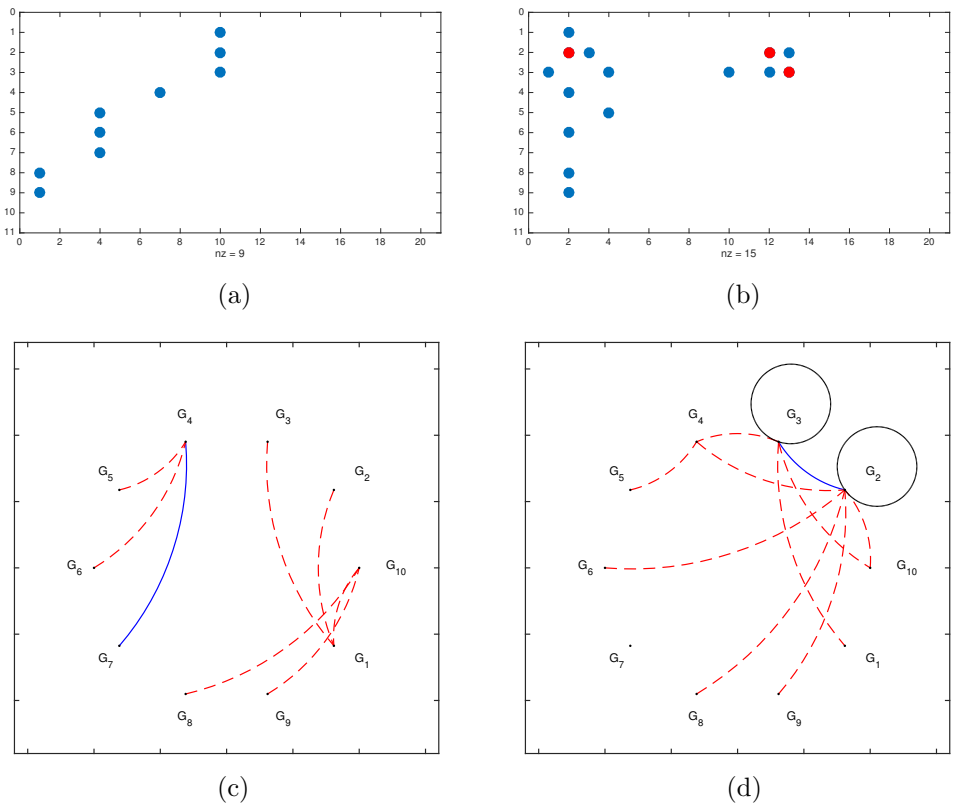


Figure 4.2: (a) Sparsity pattern of K for $\rho_{rel} = 0\%$; Blue and red bullets are used to depict diagonal and off-diagonal elements of K, respectively (b) Sparsity pattern of K for $\rho_{rel} = 30\%$ $\big((i_1, i_2) = (2, 3)\big)$. Blue and red dots represent the off-diagonal and diagonal non-zero elements, respectively. (c) Sparsity graph of $|K_\theta| + |K_\omega|$ for $\rho_{rel} = 0\%$; Blue solid lines, red dashed lines, and black self-loops are used to depict doubly-connected, singly-connected, and self-connected edges of $|K_\theta| + |K_\omega|$, respectively (d) Sparsity graph of $|K_\theta| + |K_\omega|$ for $\rho_{rel} = 30\%$ $\big((i_1, i_2) = (2, 3)\big)$.

where ρ_{rel} is called the relative uncertainty and b_{ij}^{Kron} is assumed to take non-zero values. In order to relax this assumption, the uncertainty and relative uncertainty will have to be defined in a different way, e.g.,

$$\rho = \rho_{rel} \min \left\{ \sum_{k=1, k\neq i_1}^{N_G} b_{i_1 k}^{Kron}, \sum_{k=1, k\neq i_2}^{N_G} b_{i_2 k}^{Kron} \right\}.$$

To study the effect of adding uncertainty to the link between generators i_1 and i_2,

the matrices D, E_A, and E_{B_1} are chosen as follows:

$$D = -\begin{bmatrix} 0 & 0 \\ 0 & \tilde{M}^{-1} \end{bmatrix}(e_{i_1+N_G} - e_{i_2+N_G}),\ E_A = e_{i_1+N_G}^T - e_{i_2+N_G}^T,\ E_{B_1} = 0.$$

Assuming $C = I$, $Q = I$, $R = 10I$, $\lambda_1 = 0.5$, $\lambda_2 = 0.1$, $\nu = 100$, $\xi = 10^{-6}$, and $\varepsilon^* = 10^{-2}$, we randomly choose two generators, $i_1 = 2$ and $i_2 = 3$, and consider the uncertainty cases $\rho_{rel} \in \{0\ \%, 30\ \%\}$. The results of the static state feedback controller design using our method are presented in Table 4.2. According to this table, an increase in uncertainty increases $\mathcal{R}_2$ and $\mathcal{R}_\infty$, and worsens the sparsification of the controller.

Figures 4.2(a) and 4.2(b) visualize the corresponding sparsity patterns for both cases $\rho_{rel} = 0$ % and $\rho_{rel} = 30$ %, respectively, and Figures 4.2(c) and 4.2(d) visualize the corresponding sparsity graphs for both cases $\rho_{rel} = 0$ % and $\rho_{rel} = 30$ %, respectively. It should be noted that elements K_{22}, K_{23}, $K_{2(12)}$, $K_{2(13)}$, $K_{3(12)}$, and $K_{3(13)}$ take non-zero values after applying the 30 % relative uncertainty. The interpretation is that, since uncertainty causes interference to the link between two randomly-chosen generators, the construction of communication links between such generators is vital.

Furthermore, additional plots are presented in Figure 4.6 to show the similarity of the frequency behavior of the sparsely-controlled system to that of the LQR-controlled system. The upper left sub-figure, i.e., Figure 4.6(a), depicts the largest and smallest singular values of $\mathcal{S}$ and $\hat{\mathcal{S}}$ for the case of $\rho_{rel} = 0$ %. It can be seen that the smallest singular values of the systems match for almost the whole frequency range and largest singular values achieve the same values for higher frequencies. Similar plots for the case of uncertain system with $\rho_{rel} = 30$ % are depicted in Figure 4.6(b). The plots depict that the deviation of the maximum singular value, caused by increasing the magnitude of the uncertainties, is much larger compared to the deviation of the minimum singular value. Also, the plots of Schatten 2-norm of the systems $\mathcal{S}$ and $\hat{\mathcal{S}}$,

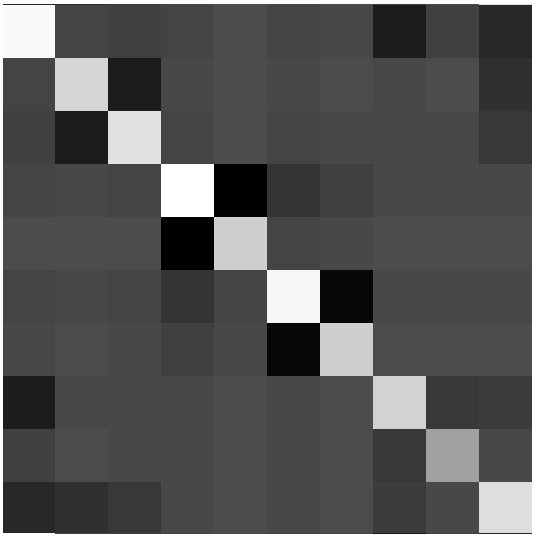


Figure 4.3: Gray scale pattern of susceptance of all links of power network, i.e., L.

are depicted in lower sub-figures of 4.6 for both cases, i.e., $\rho_{rel} = 0$ % and $\rho_{rel} = 30$ %. It is noteworthy that in neither of the cases, does the sparsification process seem to affect the higher frequency content of the closed loop systems. This is desirable, since the controller sparsification will not be amplifying the harmonics in power grids, which are the main cause of power quality degradation.

In order to verify the relationship between the magnitude of the susceptance of each link, visualized in Figure 4.3, and the density level of the corresponding elements in the controller design, we consider all cases with the $\rho_{rel} = 30$ % uncertainty on one link at a time, which results in 45 cases. We then, compute $f(K_\theta)$ and $f(K_\omega)$, the sub-blocks of the controller matrix K, in which the matrix-valued function $f(X) = [f(X)_{ij}]$ is defined as

$$f(X)_{ij} = \begin{cases} \|X_{ii}\|_0 + \|X_{ij}\|_0 + \|X_{ji}\|_0 + \|X_{jj}\|_0 & \text{if } i \neq j, \\ 0 & \text{otherwise.} \end{cases}$$

$f(K_\theta)$ and $f(K_\omega)$ are used to visualize the number of controller links, necessary to be added to the generators connected with the uncertain link. Figures 4.4(a) and 4.4(b)

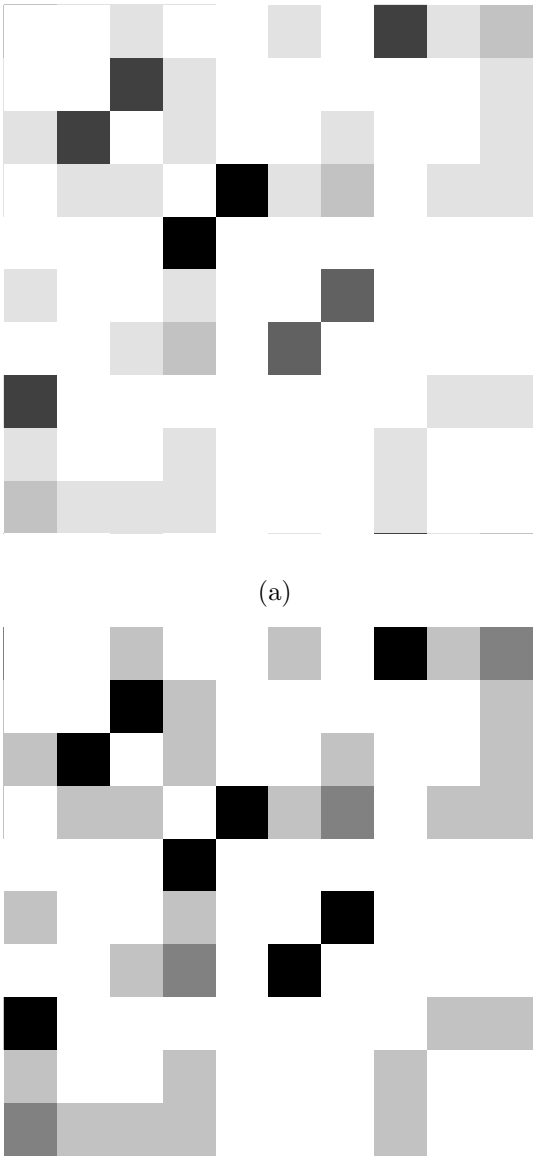
show this visualization.

As depicted in Figs. 4.4(a) and 4.4(b), in the case of links with higher susceptance, more communication links in controller design need to be established. This can be interpreted as the effective uncertainty of each link being proportional to the susceptance of that link. Therefore, an increase in susceptance of a link magnifies the uncertainty of that link, which results in establishment of more links in the designed controller to compensate for the fragility of the network on that link. This leads to similar patterns in Figures 4.3, 4.4(a), and 4.4(b).

We furthermore showcase the effect of increasing the relative uncertainty of the network links on the cardinality of their corresponding controller elements for two randomly chosen links, connecting generator 4 to generator 5 and generators 2 to 3. As seen in Figures 4.5(a) and 4.5(b), the increase of relative uncertainty, leads to construction of more communication links between two corresponding generators in the designed controller gain.

4.8 Conclusion

We have proposed a new approach for the design of optimal sparse controllers under parametric uncertainties. This method is developed based on altering an available previously designed controller towards a sparse controller, while heeding the performance deterioration caused by the process sparsification as well as the parameter uncertainties in the system. We have achieved our goal through formulating an optimization problem which seeks a sparse structured controller capable of exhibiting similar frequency and time characteristics of the previously designed controller, in terms of $\mathcal{H}_2$ and $\mathcal{H}_\infty$ norms. By equivalently reformulating the problem into a fixed rank optimization problem, we propose to utilize the bi-linear rank penalizing technique, modified to include weighted ℓ_1 norm minimization, as a computationally tractable



(a)

(b)

Figure 4.4: (a) Gray scale pattern of $f(K_\theta)+f(K_\omega)$ for $\rho_{rel} = 30$ % (b) Gray scale pattern of $f(|K_\theta| + |K_\omega|)$ for $\rho_{rel} = 30$ %.

algorithm to sub-optimally solve our problem. As our results are very promising, especially when the optimization parameters are finely tuned, we are considering a thorough study of the effects of parameter selection, with a focus on the weight on the bi-linear term, on the performance deterioration caused by the sparsification process. As another future research direction, our method can easily be modified to study

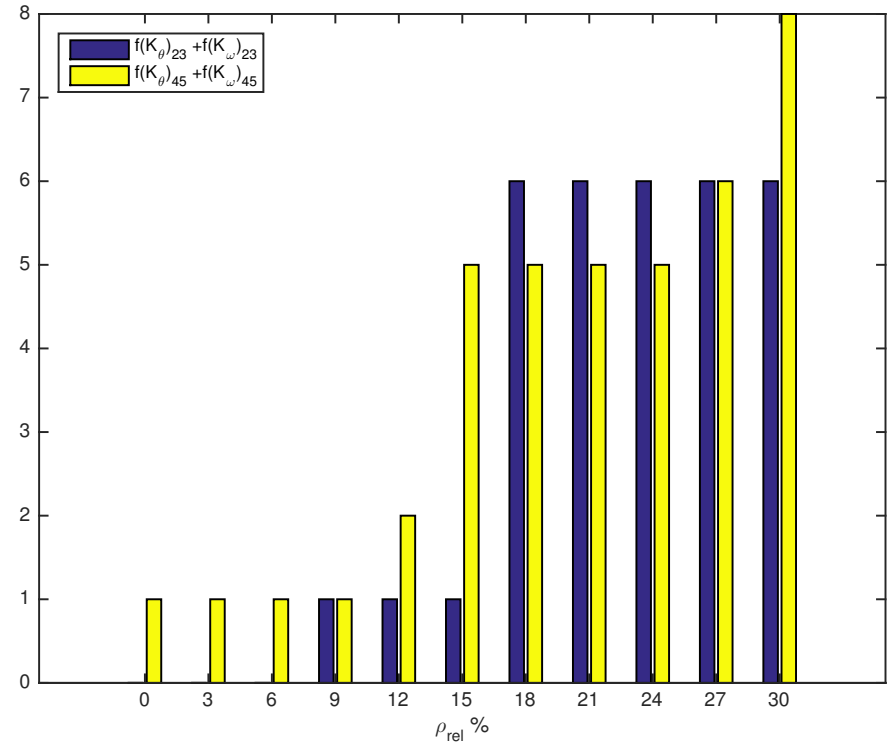
the effect of the structure and magnitude of the uncertainties on the robustness of the closed loop systems as well as the sparsity level of the controller. An important application of this study is the analysis of the robustness of networks, such as power grids, against possible attacks on the critical nodes.

Bus	Generator	$\tilde{M}_i$	$\tilde{D}_i$	θ_0	ω_0
30	G_{10}	4	5	-0.0839	1
31	G_2	3	4	0.0000	1
32	G_3	2.5	4	0.0325	1
33	G_4	4	6	0.0451	1
34	G_5	2	3.5	0.0194	1
35	G_6	3.5	3	-0.0073	1
36	G_7	3	7.5	0.1304	1
37	G_8	2.5	4	0.0211	1
38	G_9	2	6.5	0.1270	1
39	G_1	6	5	-0.2074	1

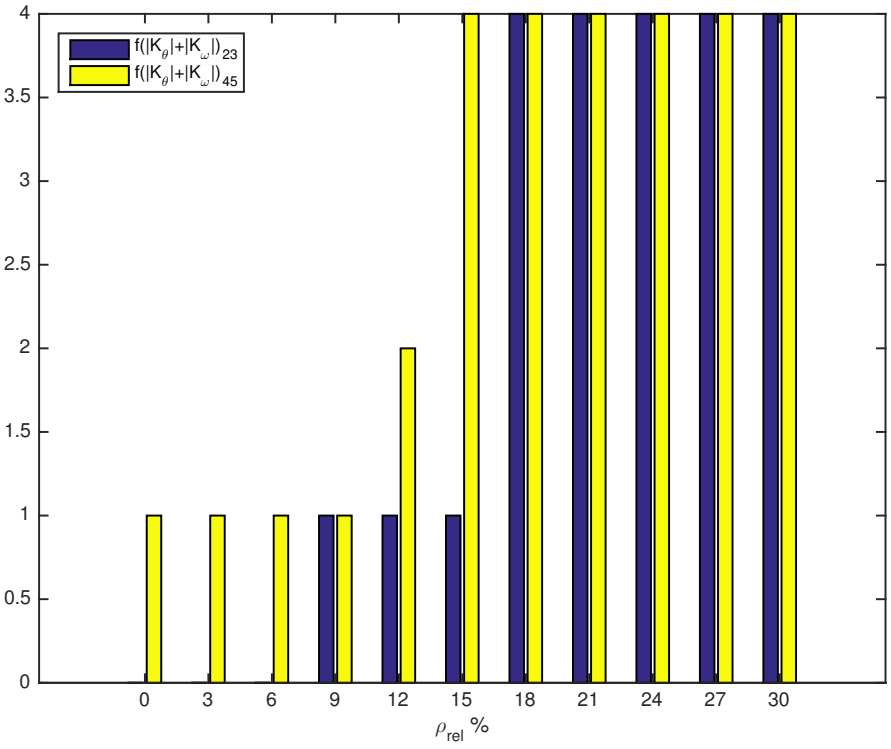
Table 4.1: Power parameters used in our numerical simulations.

ρ_{rel}	$\mathcal{R}_2$	$\mathcal{R}_\infty$	$\|K\|_0/\|\hat{K}\|_0$
0 %	21.35 %	49.42 %	4.5 %
30 %	36.31 %	88.71 %	7.5 %

Table 4.2: Performance and cardinality quantities for the case $\rho_{rel} \in \{0\ \%, 30\ \%\}$.

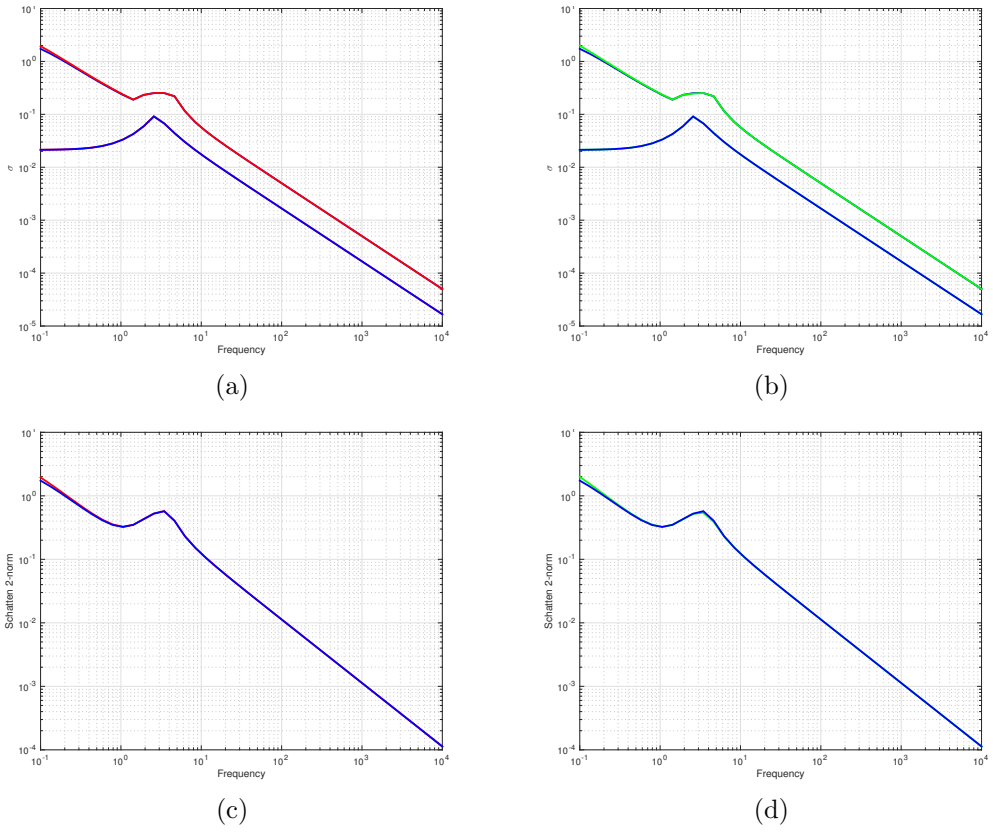


(a)



(b)

Figure 4.5: (a) $f(K_\theta) + f(K_\omega)$ versus ρ_{rel} % (b) $f(|K_\theta| + |K_\omega|)$ versus ρ_{rel} %.



(a) (b)

(c) (d)

Figure 4.6: Frequency characteristics of the closed loop systems controlled by the LQR (blue), the sparse controller (red) for the case $\rho_{rel} = 0$ %, and the sparse controller (green) for the case $\rho_{rel} = 30$ %. (a) and (b) depict maximum and minimum singular values for the cases of $\rho_{rel} = 0$ % and $\rho_{rel} = 30$ %, respectively. (c) and (d) exhibit the Schatten 2-norm of the closed loop system $\big($(c) case $\rho_{rel} = 0$ % (d) case $\rho_{rel} = 30$ %$\big)$.

Chapter 5

Sparse Memoryless LQR Design for Uncertain Linear Time-Delay Systems

5.1 Introduction

Time-Delay systems have been thoroughly investigated in control theory and its applications. Some fundamental works can be listed as works done by [98–107]. The existence of time-delay in characterization of dynamical systems is a realistic fact. Since neglecting the existence of time-delay simplifies solving the control theory problems, most of the research works in such an area fall into systems with no time-delay considerations. Thus, solving the control theory problems along with time-delay existence enables us to have much more accurate vision than when time-delay effects are ignored.

Dealing with uncertainty is another significant issue which arises in uncertain time-delay systems control and has been well-studied by [100, 101, 108–110]. Uncertainty is an undeniable concern in robust control applications. One of the fundamental works

in area of control against uncertainty is the work done by [94].

One of the control theory problems which has not been touched too much in domain of uncertain linear time-delay systems is sparsity-promoting optimal control of such systems. The necessity of sparse control designs has been highlighted in recent decade in research papers [1, 3, 7, 15, 19–21, 26–29, 38, 42]. It is clear that traditional centralized control methodologies are no longer of interest in power network control, control of platoons of vehicles, and multi-agent control systems. Thus, the tendency to utilize the structured, distributed, decentralized, or localized control methods has been tremendously increased in controlling such systems.

Inspired by achievements in uncertain linear time-delay systems and sparsity-promoting optimal control, we propose sparsity-promoting optimal control design for uncertain linear time-delay systems and investigate the effect of time-delay on sparsification process and performance-sparsity trade-off curves. In this work, the stability of uncertain linear time-delay system is characterized via linear matrix inequality (LMI) approach utilized by [101]. Such a sufficient condition is derived based on Lyapunov functionals introduced by [111].

In the following, the chapter is organized as follows: Section 5.2 presents our utilized mathematical notations. In Section 5.3, the sparse memoryless LQR design problem is formulated for uncertain linear time-delay systems. In Section 5.4, it is stated how our problem can equivalently be reformulated as an optimization problem with several LMIs and a rank constraint. Section 5.5 includes steps to be taken to tackle the rank-constrained optimization problem via bi-linear rank penalty technique. In Section 5.6, throughout the several numerical simulations, sparsity visualization of sparse memoryless LQR design, (time-delay)-(performance/sparsity) trade-offs, and (time-delay)-(performance-sparsity trade-off) behavior are visualized. Finally, Section 5.7 concludes the chapter along with sketching possible future insights.

5.2 Mathematical Notations

Throughout the chapter, the set of real numbers and the set of n by m real matrices are denoted by $\mathbb{R}$ and $\mathbb{R}^{n\times m}$, respectively. The n by n identity matrix and n by m zero matrix are shown by I_n and $0_{n\times m}$, respectively. Trace and rank of a matrix are specified by $\mathbf{Tr}(.)$ and $\mathbf{rank}(.)$, respectively. The transpose operator is represented by $(.)^T$. The Hadamard matrix product is denoted by $\circ$. The symbol $\|.\|_0$ symbolizes the number of non-zero elements of a matrix and symbols $\|.\|_1$ and $\|.\|_F$ symbolize the ℓ_1 and Frobenius norms, respectively. Also, the maximum singular value of a matrix is represented by $\|.\|$. A symmetric matrix is called positive definite (positive semi-definite) if all the eigenvalues are positive (non-negative). The space of positive definite (positive semi-definite) matrices are represented by $\mathbb{S}^n_{++}$ ($\mathbb{S}^n_{+}$) and the notation $X \succ Y$ ($X \succeq Y$) means $X - Y \in \mathbb{S}^n_{++}$ ($X - Y \in \mathbb{S}^n_{+}$).

Definition 4. *For a given $\epsilon \geq 0$ and matrix X, rank of X with tolerance ϵ is k and denoted by $\mathbf{rank}(X;\epsilon) = k$, if and only if k is the maximum number of singular values of X which are larger than ϵ.*

5.3 Problem Formulation

The uncertain linear time-delay system is characterized as follows:

$$\begin{cases} \dot{x}(t) = & (A+\Delta A)x(t) + (A_1+\Delta A_1)x(t-\tau) + (B+\Delta B)u(t) \\ x(t) = & \phi(t),\ t \in [-\tau, 0] \end{cases}, \tag{5.1}$$

where $A \in \mathbb{R}^{n\times n}$, $A_1 \in \mathbb{R}^{n\times n}$, $B \in \mathbb{R}^{n\times m}$ are known, positive τ denotes the constant time-delay, and $\phi(t)$ represents a vector-valued initial condition. Matrices ΔA, ΔA_1, and ΔB are matrix-valued functions which symbolize time-varying parameter uncertainties.

The form of parameter uncertainties is considered as follows:

$$\begin{bmatrix} \Delta A & \Delta B & \Delta A_1 \end{bmatrix} = DF(t) \begin{bmatrix} E_A & E_B & E_{A_1} \end{bmatrix},$$

where D, E_A, E_B, and E_{A_1} are known matrices which determine the structure of uncertainties and $F(t) \in \mathbb{R}^{i \times j}$ is an unknown matrix-valued function whose elements are Lebesgue measurable and it satisfies the following matrix inequality:

$$F(t)^T F(t) \preceq I_j.$$

Our goal is to design a sparse memoryless LQR

$$u(t) = Kx(t), \tag{5.2}$$

which minimizes the following quadratic cost functional:

$$J = \int_0^\infty \big(x(t)^T Q x(t) + u(t)^T R u(t)\big) dt, \tag{5.3}$$

subject to stability of closed-loop system under uncertainties. Matrices $Q \succeq 0$ and $R \succ 0$ represent state weight and input weight matrices, respectively.

To reach such a goal, we define the following optimization problem:

$$\begin{aligned} \underset{K}{\text{minimize}} \ & \int_0^\infty \big(x(t)^T Q x(t) + u(t)^T R u(t)\big) dt + \gamma \|K\|_0 \qquad (\mathbf{P1}) \\ \text{subject to: } & (5.1) \text{ and } (5.2), \\ & K : \text{stabilizing}, \end{aligned}$$

where γ is the sparsity-promoting parameter.

Remark 12. *Any convex structural constraint on K can be embedded into our sparse*

memoryless LQR design process. Distributed, decentralized, localized, and other topological assumptions are few examples of such a convex structural constraints on K.

For the case $\gamma = 0$, using the results of Theorem 3 stated by [101], implies that (**P1**) can be relaxed to the following convex optimization problem (assuming that $\gamma = 0$):

$$\begin{aligned}
&\underset{\delta,\alpha,M,X,Y,Z}{\text{minimize}} \ \alpha + \mathbf{Tr}(M) \qquad (\mathbf{P2})\\
&\text{subject to:} \begin{bmatrix} \tilde{A} & A_1 Z & \tilde{E}^T & X & Y^T & X \\ Z A_1^T & -Z & Z E_{A_1}^T & 0 & 0 & 0 \\ \tilde{E} & E_{A_1} Z & -\delta I_j & 0 & 0 & 0 \\ X & 0 & 0 & -Q^{-1} & 0 & 0 \\ Y & 0 & 0 & 0 & -R^{-1} & 0 \\ X & 0 & 0 & 0 & 0 & -Z \end{bmatrix} \prec 0,\\
&\begin{bmatrix} -\alpha & \phi(0)^T \\ \phi(0) & -X \end{bmatrix} \prec 0, \quad \begin{bmatrix} -M & N(\tau)^T \\ N(\tau) & -Z \end{bmatrix} \prec 0,
\end{aligned}$$

where $N(\tau)$ depends on time-delay τ and equals to the principal square root of matrix

$$\int_0^\tau \phi(t)\phi(t)^T dt. \tag{5.4}$$

Also, matrices $\tilde{A}$ and $\tilde{E}$ are equal to $AX+BY+(AX+BY)^T+\delta DD^T$ and E_AX+E_BY, respectively.

In fact, assuming the $\gamma = 0$, solving (**P2**) provides us with a sub-optimal solution $K = YX^{-1}$ to (**P1**) with upper bound

$$J^* = \phi(0)^T X^{-1} \phi(0) + \mathbf{Tr}\big(N(\tau)N(\tau)^T Z^{-1}\big), \tag{5.5}$$

on J defined by (5.3).

5.4 Equivalent Rank-Constrained Reformulation

In this part, we see how (**P1**) can be cast as an equivalent rank-constrained reformulation.

Motivated by formulation of (**P2**) and noting the NP-hardness of dealing with ℓ_0 sparsity measure, we define the weighted ℓ_1 relaxation of (**P1**) as follows:

$$\begin{aligned}
&\underset{\delta,\alpha,M,X,Y,Z}{\text{minimize}} \ \alpha + \mathbf{Tr}(M) + \gamma\|W \circ K\|_1 \qquad (\mathbf{P3})\\
&\text{subject to:} \begin{bmatrix} \tilde{A} & A_1 Z & \tilde{E}^T & X & Y^T & X \\ Z A_1^T & -Z & Z E_{A_1}^T & 0 & 0 & 0 \\ \tilde{E} & E_{A_1} Z & -\delta I_j & 0 & 0 & 0 \\ X & 0 & 0 & -Q^{-1} & 0 & 0 \\ Y & 0 & 0 & 0 & -R^{-1} & 0 \\ X & 0 & 0 & 0 & 0 & -Z \end{bmatrix} \prec 0,\\
&\begin{bmatrix} -\alpha & \phi(0)^T \\ \phi(0) & -X \end{bmatrix} \prec 0, \quad \begin{bmatrix} -M & N(\tau)^T \\ N(\tau) & -Z \end{bmatrix} \prec 0,\\
&Y = KX. \qquad (5.6)
\end{aligned}$$

As it is observed, the constraint (5.6) is non-convex and consequently, solving (**P3**) gets difficult in its current format. To overcome such an issue, we take an advantage of the following lemma which has been presented in [15].

Lemma 24. *([15]) Let $\mathcal{U} \in \mathbb{R}^{n\times n}$, $\mathcal{V} \in \mathbb{R}^{n\times m}$, $\mathcal{W} \in \mathbb{R}^{m\times m}$, and $\mathcal{Y} \in \mathbb{R}^{m\times n}$, with $\mathcal{U} \succ 0$. Then, $\mathbf{rank}(\mathcal{M}) = n$ if and only if $\mathcal{W} = \mathcal{Y}\mathcal{U}\mathcal{Y}^T$, $\mathcal{V}^T = \mathcal{Y}\mathcal{U}$, and $\mathcal{Z} = \mathcal{U}^{-1}$*

where

$$\mathcal{M} = \begin{bmatrix} \mathcal{U} & \mathcal{V} & I_n \\ \mathcal{V}^T & \mathcal{W} & \mathcal{Y} \\ I_n & \mathcal{Y}^T & \mathcal{Z} \end{bmatrix}.$$

Choosing the $\mathcal{V} = Y^T$, $\mathcal{Y} = K$, and $\mathcal{U} = X$ in Lemma 24, the constraint (5.6) gets equivalent to the following rank constraint:

$$\mathbf{rank}(M_1) = n, \tag{5.7}$$

where M_1 is described as follows:

$$M_1 = \begin{bmatrix} X & Y^T & I_n \\ Y & S & K \\ I_n & K^T & T \end{bmatrix}.$$

Although (5.7) implies that $S = KXK^T$ and $T = X^{-1}$ hold in addition to (5.6), they do not have any significant role in our next derivations.

Thus, utilizing the (5.7), the equivalent rank-constrained reformulation of (**P3**) is

obtained as follows:

$$\underset{\delta,\alpha,M,X,Y,Z,S,T,K}{\text{minimize}} \ \alpha + \mathbf{Tr}(M) + \gamma\|W \circ K\|_1 \qquad (\mathbf{P4})$$

$$\text{subject to:} \begin{bmatrix} \tilde{A} & A_1 Z & \tilde{E}^T & X & Y^T & X \\ Z A_1^T & -Z & Z E_{A_1}^T & 0 & 0 & 0 \\ \tilde{E} & E_{A_1} Z & -\delta I_j & 0 & 0 & 0 \\ X & 0 & 0 & -Q^{-1} & 0 & 0 \\ Y & 0 & 0 & 0 & -R^{-1} & 0 \\ X & 0 & 0 & 0 & 0 & -Z \end{bmatrix} \prec 0,$$

$$\begin{bmatrix} -\alpha & \phi(0)^T \\ \phi(0) & -X \end{bmatrix} \prec 0, \quad \begin{bmatrix} -M & N(\tau)^T \\ N(\tau) & -Z \end{bmatrix} \prec 0,$$

$$(5.7),$$

where W is element-wise non-negative weight matrix, i.e., all the elements take non-negative values.

5.5 Sparsification Algorithm via Bi-linear Rank Penalty Technique

Since (5.7) is non-convex, to deal with such a non-convexity, the bi-linear rank penalty technique is employed. Such a technique has been extensively utilized in recent works [26–28, 72, 77].

The core of such a technique is basically relaxing the constraint (5.7) with $M_1 \succeq 0$, adding a bi-linear penalty term to the objective function of the corresponding optimization problem, and then, iteratively solving two main convex sub-problems which will be explained in detail later on. It is worth emphasizing that the convergence

of such an iterative method is discussed through the facts provided by [72, 77, 78].

The following theorem is the main basis of bi-linear rank penalty technique and it is derived with few changes based on Theorem 5 proposed by [27].

Theorem 25. *Let us consider* (**P4**) *and define the following auxiliary optimization problem:*

$$
\begin{aligned}
&\underset{\mathcal{S}}{\text{minimize}}\ \alpha + \mathbf{Tr}(M) + \gamma\|W \circ K\|_1 + \nu\mathbf{Tr}(GM_1) \qquad (\mathbf{P5})\\
&\text{subject to:} \begin{bmatrix}
\tilde{A} & A_1 Z & \tilde{E}^T & X & Y^T & X \\
ZA_1^T & -Z & ZE_{A_1}^T & 0 & 0 & 0 \\
\tilde{E} & E_{A_1}Z & -\delta I_j & 0 & 0 & 0 \\
X & 0 & 0 & -Q^{-1} & 0 & 0 \\
Y & 0 & 0 & 0 & -R^{-1} & 0 \\
X & 0 & 0 & 0 & 0 & -Z
\end{bmatrix} \prec 0,\\
&\begin{bmatrix} -\alpha & \phi(0)^T \\ \phi(0) & -X \end{bmatrix} \prec 0, \quad \begin{bmatrix} -M & N(\tau)^T \\ N(\tau) & -Z \end{bmatrix} \prec 0,\\
&M_1 \succeq 0, 0 \preceq G \preceq I_{2n+m}, \mathbf{Tr}(G) = n+m,
\end{aligned}
$$

in which $\nu > 0$ *is bi-linear penalty parameter and* $\mathcal{S}$ *denotes the stack of variables* $\delta, \alpha, M, X, Y, Z, S, T, K, G$. *If* (**P4**) *is feasible, then there exists a positive constant* η $\big($*the optimal value of* (**P4**)$\big)$ *for which the optimal solution* M_1 *resulted from solving* (**P5**) *satisfies*

$$\mathbf{rank}(M_1; \eta\nu^{-1}) \ \leq \ n,$$

i.e., rank of M_1 *with tolerance threshold* $\eta\nu^{-1}$ *is less than or equal to* n *due to Definition 4.*

In addition to rank inequality derived by Theorem 25, the specific structure of

matrix M_1 yields that the inequality $\mathbf{rank}(M_1) \geq n$ holds as well. Thus, if n^{th} largest singular value of M_1 is at least $\eta\nu^{-1}$, then $\mathbf{rank}(M_1; \eta\nu^{-1}) = n$ holds.

As a consequence of Theorem 25, (**P5**) can be solved with an appropriately-chosen parameter ν to achieve a sub-optimal solution to (**P4**). For the sake of simplicity in our notations, the stack of all variables excluding variable G is denoted by H. The optimization problem (**P5**) can be reformulated as

$$\begin{aligned} &\underset{H,G}{\text{minimize}}\ \mathcal{F}(H, G) \\ &\text{subject to: } H \in \mathcal{C}_H, \quad G \in \mathcal{C}_G, \end{aligned}$$

where $\mathcal{C}_H$ is the convex set characterized by H, and the convex set $\mathcal{C}_G$ is generated by $\mathbf{Tr}(G) = n + m$ and $0 \preceq G \preceq I_{2n+m}$.

No need to say that $\mathcal{F}(H, G)$ denotes the bi-linear objective function in (**P5**). The above-mentioned rewritten optimization problem enables us to solve such a problem by iteratively minimizing the objective function for H and G. Consequently, the important steps of this iteratively implemented method can be summarized in two sub-problems: H-minimization and G-minimization sub-problems. Since both H-minimization and G-minimization steps are convex optimizations, they can be executed in an efficient way with existing convex solvers such as CVX developed by [112]. Fortunately, for the G-minimization step, there exists an analytic solution which is expressed in the following theorem similar to Theorem 6 stated by [27].

Theorem 26 ([27])**.** *The optimal solution to the G-minimization step is given by*

$$G^* = I_{2n+m} - \sum_{i=1}^{n} u_i {u_i}^T, \tag{5.8}$$

where vectors u_i for $i = 1, \ldots, n$ are the singular vectors corresponding to the n largest singular values of M_1.

The following sequence of iterations is employed to achieve the minimizer of (**P5**). The H-minimization and G-minimization sub-problems are expressed as follows:

$$H^{(k+1)} = \arg\underset{H\in\mathcal{C}_H}{\text{minimize}} \;\; \mathcal{F}(H, G^{(k)}), \tag{5.9}$$

$$G^{(k+1)} = I_{2n+m} - \sum_{i=1}^{n} u_i^{(k+1)} u_i^{(k+1)^T}, \tag{5.10}$$

where

$$M_1^{(k+1)} = \sum_{i=1}^{2n+m} \sigma_i^{(k+1)} u_i^{(k+1)} u_i^{(k+1)^T},$$

is the singular value decomposition (SVD) of $M_1^{(k+1)}$ which is simply M_1 at $(k+1)^{th}$ iteration. The stopping criterion is applied via $\varepsilon^{(k+1)} \leq \varepsilon^*$, where ε^* is the pre-specified precision, with the following update rule:

$$\varepsilon^{(k+1)} = \frac{\|K^{(k+1)} - K^{(k)}\|_2}{\|K^{(k+1)}\|_2}. \tag{5.11}$$

At last, negligible elements of obtained K, (e.g., those smaller than 5×10^{-5}) is truncated. In fact, the relatively small elements of K correspond to weakly-coupled links between the nodes in the information structure of the memoryless LQR. A summary of our proposed algorithm is described in bi-linear rank penalty technique sparsification (BRPTS) algorithm.

5.6 Numerical Simulations

In order to validate our utilized algorithm, we employ the class of spatially distributed systems which has been deeply studied by [42]. Firstly, we introduce how spatially distributed systems are characterized via their specific state-space realizations. Secondly, we design sparse memoryless LQR and visualize its sparsity visualization. Finally, the effect of time-delay on sparsification process and performance-sparsity

BRPTS Algorithm Solution to (**P5**)

Inputs: A, B, A_1, D, E_A, E_B, E_{A_1}, Q, R, γ, ν, W, $\phi(t)$, τ, and ε^*.

1: Initialization:
   Set $G^{(0)} = I_{2n+m}$, $\varepsilon^{(0)} > \varepsilon^*$, $K^{(0)} = 0_{m\times n}$ and $k = 0$.
2: **While** $\varepsilon^{(k)} > \varepsilon^*$ **Do**
3: Update $H^{(k+1)}$ by solving (5.9),
4: Update $G^{(k+1)}$ using (5.10),
5: Update $\varepsilon^{(k+1)}$ using (5.11),
6: $k \leftarrow k + 1$,
7: **End While**
8: Truncate K.

Output: K

trade-off are investigated. Although all of our derivations hold for uncertain linear time-delay systems, to purely assess the effect of time-delay on sparsification process and performance-sparsity trade-off, in current work, we assume that there is no uncertainty, i.e., $D = 0$, $E_A = 0$, $E_B = 0$, and $E_{A_1} = 0$. The effect of uncertainty on sparsification process and performance-sparsity trade-off can separately be investigated in future works.

5.6.1 Spatially Distributed Systems

Similar to the idea developed by [42], let us consider $N = 10$ randomly distributed (with a uniform distribution) nodes in a 10×10 box-shaped region (See Figure 5.1). Each node represents a linear sub-system which is coupled via its dynamics and the quadratic cost to other sub-systems. The dynamics of the i^{th} linear sub-systems is characterized as follows:

$$\dot{x}^{(i)}(t) = [A]_{ii}x^{(i)}(t) + \sum_{j=1,\ j\neq i}^{N} [A]_{ij}x^{(j)}(t) + [B]_{ii}u^{(i)}(t),$$

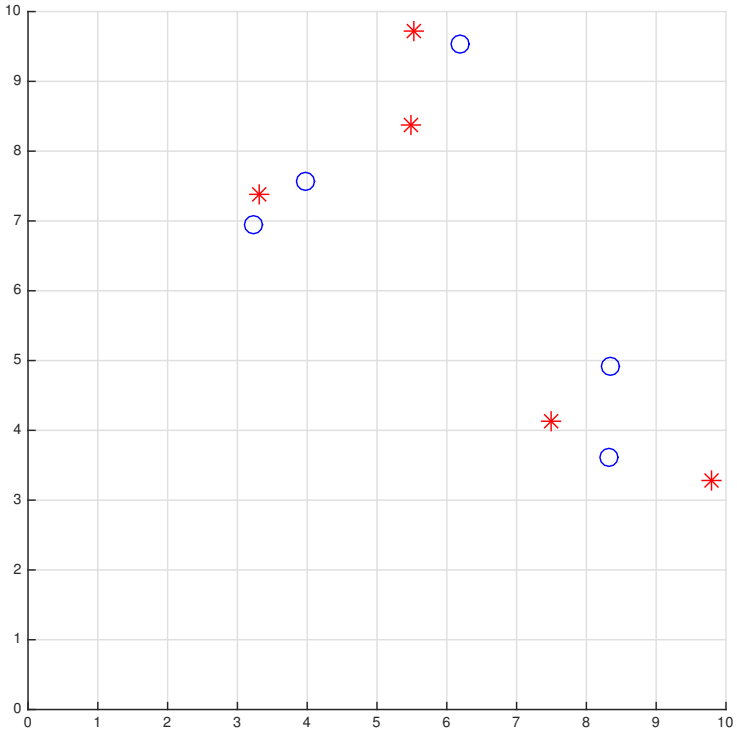


Figure 5.1: Positions of $N = 10$ randomly generated nodes in a 10×10 box-shape region.

where

$$[A]_{ii} = \begin{bmatrix} 1 & 1 \\ 1 & 2 \end{bmatrix}, \qquad B_{ii} = \begin{bmatrix} 0 \\ 1 \end{bmatrix} \text{ for nodes marked by red } *,$$

$$[A]_{ii} = \begin{bmatrix} -2 & 1 \\ 1 & -3 \end{bmatrix}, \qquad B_{ii} = \begin{bmatrix} 0 \\ 1 \end{bmatrix} \text{ for nodes marked by blue } \circ,$$

and

$$[A]_{ij} = \frac{1}{\mathcal{X}_\beta\big(\mathrm{dis}(i,j)\big)} \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}, B_{ij} = \begin{bmatrix} 0 \\ 0 \end{bmatrix}, \ \forall j \neq i,$$

where $\mathcal{X}_\beta$ represents the coupling characteristic function and $\text{dis}(i,j)$ symbolizes the Euclidean distance between nodes i and j in Figure 5.1. Several choices have been introduced for $\mathcal{X}_\beta$ by [42]. The one which we will utilize is the exponentially decaying operator which is defined as follows:

$$\mathcal{X}_\beta(x) = e^{\beta x}.$$

In general, the positive parameter β determines the spatially decaying rate in spatially-decaying operators.

In our numerical simulations, the matrix A is constructed in the above-mentioned manner. The matrix A_1 is chosen as a real multiple of A, i.e., $A_1 = \rho A$ where $\rho \in \mathbb{R}$.

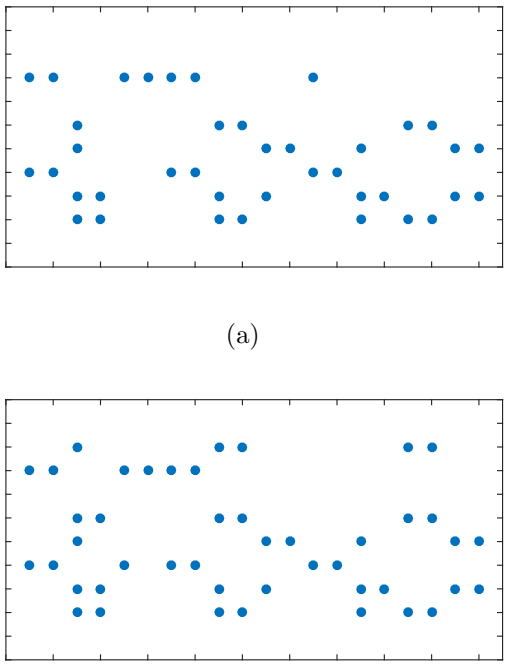
5.6.2 Sparse Memoryless LQR Design and Visualizations

In this chapter, we assume that $\phi(t) = x_0$ in whole time interval $[-\tau, 0]$ where x_0 is drawn from a standard normal distribution. Subsequently, $N(\tau)$ gets equal to the principal square root of $\tau x_0 x_0^T$. Also, to select x_0, we use command *randn* in MATLAB.

Considering the 20×20 randomly distributed system (β=1) drawn from $N = 10$ nodes in Figure 5.1 and setting $\rho = 0.1$, $\gamma = 0.1$, $\nu = 100$, $Q = I_{20}$, $R = 4I_{10}$, the sparsity visualization of sparse memoryless LQR designs for cases $\tau = 0$ and $\tau = 9$ are depicted in Figures 5.2(a) and 5.2(b), respectively.

5.6.3 Investigation of Effect of Time-Delay on Sparsification Process and Performance-Sparsity Trade-Off

For previously-mentioned 20×20 randomly distributed system and the same setup for all parameters except τ, by considering the 26 equidistant values for τ (with step



(a)

(b)

Figure 5.2: (a) Sparsity visualization of K for $\tau = 0$ (b) Sparsity visualization of K for $\tau = 9$. Blue dots represent the non-zero elements.

size 0.5), cardinality percentage

$$\frac{\|K\|_0}{\|K^{LQR}\|_0} \times 100,$$

and performance loss percentage

$$\frac{J^* - J^{LQR}}{J^{LQR}} \times 100,$$

are visualized versus time-delay τ in Figures 5.3(a) and 5.3(b), respectively. The superscript LQR is utilized to denote the quantities related to standard traditional LQR design corresponding to A when there is no time-delay (τ=0).

As Figures 5.3(a) and 5.3(b) demonstrate, as time-delay τ increases, cardinality percentage and performance loss percentage get increased. Thus, it is observed that

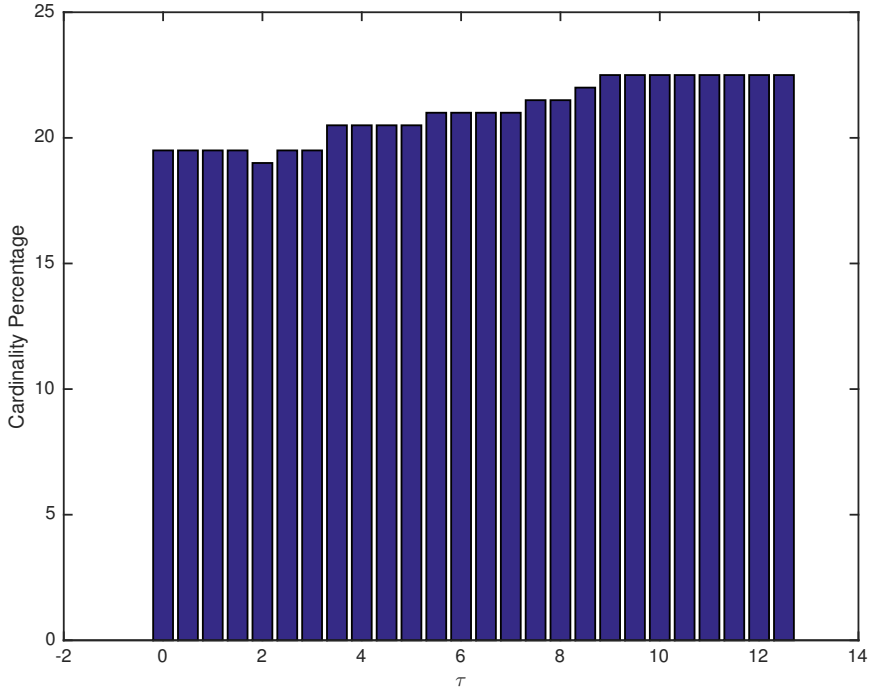
the larger time-delay we have, the poorer quality of sparsification process we get.

To assess the effect of time-delay on performance-sparsity trade-off, considering the same 20×20 randomly distributed system and choosing the fixed time-delay from set $\{0, 2.5, 5, 7.5, 10, 12.5\}$, we run BRPTS Algorithm for 20 logarithmically spaced sparsity-promoting parameter $\gamma \in [10^{-4}, 10^{-1}]$ which leads to plots depicted by Figure 5.4.

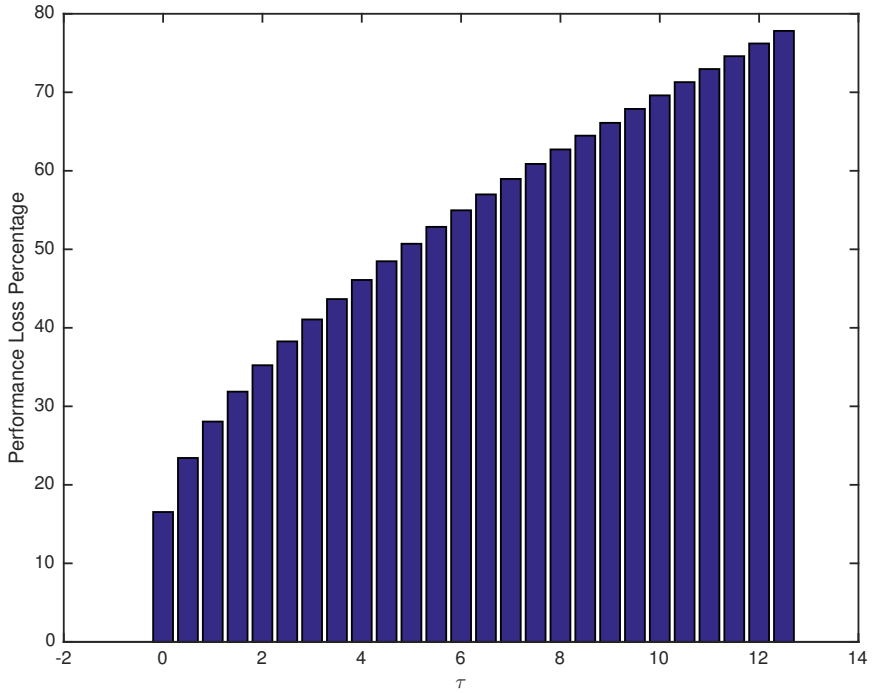
As Figure 5.4 showcases, when time-delay τ gets larger, the performance-sparsity trade-off gets worse. In other words, prescribing a fixed value of cardinality percentage (proportional to number of controller communication links), having a larger time-delay leads to higher performance loss which is not desired.

5.7 Conclusion

Considering the class of uncertain linear time-delay systems, the sparse memoryless LQR design is presented. Utilizing the LMI techniques, deriving the equivalent rank-constrained reformulation, and applying the bi-linear rank penalty technique, sub-optimal sparse memoryless LQR design is achieved. Employing the various numerical experiments, the negative effect of constant time-delay on sparsification process and performance-sparsity trade-off is observed. The improvement of sub-optimality level of utilized technique can be seen as a possible future work.



(a)



(b)

Figure 5.3: (a) Cardinality percentage versus τ (b) Performance loss percentage versus τ.

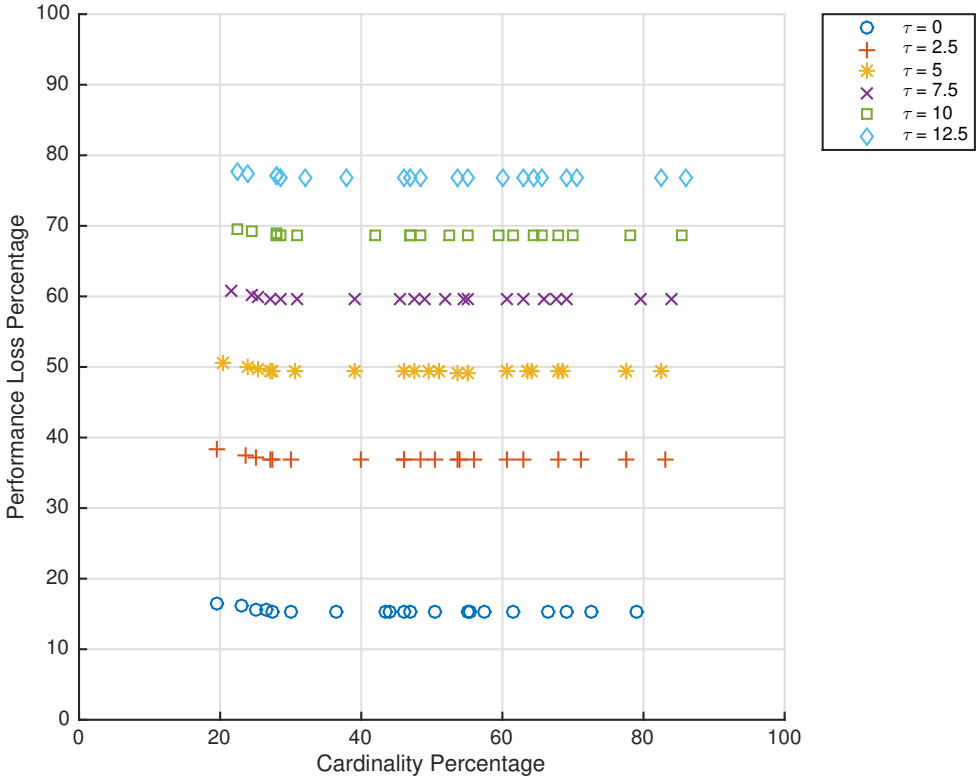


Figure 5.4: Performance-Sparsity trade-off for different values of 6 uniformly selected time-delay τ and 20 logarithmically spaced sparsity-promoting parameter $\gamma \in [10^{-4}, 10^{-1}]$.

Chapter 6

Row-Column Sparse Linear Quadratic Controller Design via Bi-Linear Rank Penalty Technique and Non-Fragility Notion

6.1 Introduction

Sparsity-Promoting control problems can be categorized as two important classes:
(i) sparse controller design,
(ii) row/column sparse controller design.
Some examples of the first category can be found in [3, 18, 19, 21, 26, 27, 30, 32, 42, 68, 89]. On the other side, row/column sparse controller design is investigated by [10, 28].

In this chapter, focusing on the second category of sparsity-promoting control problems, i.e., row/column sparse controller design, we consider row-column (r, c)-sparse controller design. In such a design problem, each node will communicate to

at most r other nodes and information of each node will be used by at most c other nodes. While in [10] the definition of row/column sparsity is different from ours and [28]'s, the new contribution of our work compared to [10] is capability of having row and column sparsity at the same time with possibly distinct values for r and c. Also, in comparison with [28], instead of using majorization theory and computationally expensive algorithm, we utilize the non-fragility notion provided by [33] to have a purely utilized bi-linear rank penalty technique and relatively fast algorithm. It is also remarkable that numerical simulations provided by [28], do not satisfy the sparsity constraints for all rows or columns. As we will see in our simulation section, all the sparsity constraints hold for all rows and all columns.

The chapter is arranged as follows: The section 6.2 is dedicated to explain our mathematical notations which are used along the chapter. In Section 6.3, we express the problem which we aim at solving. In Section 6.4, we show how our problem can equivalently be translated to an optimization problem constrained to several linear matrix inequalities and $m + n + 1$ rank constraints. Section 6.5 provides the vision to our bi-linear rank penalty technique and its details. Our numerical simulations are visualized for the class of randomly-generated systems in Section 6.6. At last, we finish the chapter with drawing some future directions in Section 6.7.

6.2 Mathematical Notations

Throughout the chapter, matrices are denoted by capital letters, and the elements are shown by capital letters with subscripts. The vectors, on the other hand, are represented by lower-case letters, with elements denoted by the same letter with subscripts. The identity matrix of size $n \times n$ is denoted by I_n. The Hadamard matrix product is denoted by $\circ$. The number of non-zero elements of a matrix is denoted by $\|.\|_0$. The ℓ_2 norm of a matrix is represented by $\|.\|_2$. The notation$\|X\|_{\max}$ represents

the maximum absolute value of all elements of matrix X. The trace operator is denoted by **Tr**(.) and the rank operator is demonstrated by **rank**(.). The block diagonal matrix construction operator is shown by **diag**(.). The operator **sign**(.) is used to take element-wise sign of a matrix. The element-wise comparison between two matrices is denoted by usual $\geq$. A matrix is called Hurwitz if its all eigenvalues lie in the open left half of the complex plane. The set of $n \times 1$ real vectors and $m \times n$ real matrices are represented by $\mathbb{R}^n$ and $\mathbb{R}^{m \times n}$, respectively. A symmetric matrix is called positive definite (positive semi-definite) if all of its eigenvalues are positive (non-negative). The space of positive definite (positive semi-definite) matrices is denoted by $\mathbb{S}^n_{++}$ ($\mathbb{S}^n_{+}$) and $X - Y \in \mathbb{S}^n_{+}$ ($X - Y \in \mathbb{S}^n_{++}$) is symbolized with $X \succeq Y$ ($X \succ Y$). The normal distribution with zero mean and unit variance is represented by $\mathcal{N}(0, 1)$ and the expected value is represented by $\mathbb{E}$. The i^{th} row and j^{th} column of matrix X are shown by $X(i, :)$ and $X(:, j)$, respectively.

Definition 5. *For a given $\epsilon \geq 0$ and matrix X, rank of X is k with tolerance ϵ and denoted by* **rank**$(X; \epsilon)$*, if exactly k singular values of X are greater than ϵ.*

6.3 Problem Formulation

6.3.1 Linear Time-Invariant System Controlled by Linear Controller

The linear time-invariant (LTI) system controlled by linear controller $K \in \mathbb{R}^{m \times n}$ is considered as follows:

$$\begin{aligned} \dot{x}(t) &= Ax(t) + Bu(t),\ x(0) = x_0, \\ u(t) &= Kx(t), \end{aligned} \tag{6.1}$$

where $x(t) \in \mathbb{R}^n$ denotes the state vector, $A \in \mathbb{R}^{n\times n}$ denotes the state matrix, $B \in \mathbb{R}^{n\times m}$ denotes the input matrix, $u(t) \in \mathbb{R}^m$ denotes the control input, and $x_0 \in \mathcal{N}(0,1)$ denotes the initial condition of the system.

6.3.2 Row-Column Sparse Linear Quadratic Controller (LQC) Design

Suppose that a linear time-invariant (LTI) system along with linear controller is given as shown in (6.1).

Definition 6. *A stabilizing controller K is called row-column (r,c)-sparse if and only if each of its rows/columns has at most r/c non-zero elements. Mathematically describing, it means that for $1 \leq i \leq m$ and for $1 \leq j \leq n$, $m+n$ constraints $\|K(i,:)\|_0 \leq r$ and $\|K(:,j)\|_0 \leq c$ hold, respectively.*

Given a non-negative integer ordered pair (r,c), the goal is to design a row-column (r,c)-sparse LQC design which minimizes the following quadratic functional:

$$J(K) := \mathbb{E}\bigg(\int_0^\infty \big(x(t)^T Q x(t) + u(t)^T R u(t)\big)dt\bigg), \tag{6.2}$$

where $Q \succeq 0$ and $R \succ 0$ are corresponding state and input weight matrices, respectively.

Our goal can mathematically be translated to solve the following optimal control problem:

$$\underset{K}{\text{minimize}}\ J(K) \tag{6.3a}$$

$$\text{subject to: } K:\ \text{Row-Column } (r,c)\text{-sparse.} \tag{6.3b}$$

6.4 Rank-Constrained Optimization Reformulation

In order to reformulate the constraints of an optimization problem as rank-constrained linear matrix inequalities, we state the following lemma proposed by [15].

Lemma 27 ([15]). *Let $\mathcal{U} \in \mathbb{R}^{n\times n}$, $\mathcal{V} \in \mathbb{R}^{n\times m}$, $\mathcal{W} \in \mathbb{R}^{m\times m}$, and $\mathcal{Y} \in \mathbb{R}^{m\times n}$, with $\mathcal{U} \succ 0$. Then,* $\mathbf{rank}(\mathcal{M}) = n$ *if and only if $\mathcal{W} = \mathcal{Y}\mathcal{U}\mathcal{Y}^T$, $\mathcal{V}^T = \mathcal{Y}\mathcal{U}$, and $\mathcal{Z} = \mathcal{U}^{-1}$ where*

$$\mathcal{M} = \begin{bmatrix} \mathcal{U} & \mathcal{V} & I_n \\ \mathcal{V}^T & \mathcal{W} & \mathcal{Y} \\ I_n & \mathcal{Y}^T & \mathcal{Z} \end{bmatrix}.$$

Assuming that $x_0 \in \mathcal{N}(0,1)$, considering the fact $\mathbb{E}(x_0 x_0^T) = I$, and doing simple expected value calculations it is resulted that

$$J(K) = \mathbf{Tr}(QX_{11} + K^T RKX_{11}),$$

where X_{11} denotes the unique positive definite solution of the following Lyapunov equation:

$$(A + BK)X_{11} + X_{11}(A + BK)^T + I_n = 0.$$

Utilizing the Lemma 27 with $\mathcal{U} = X_{11}$, $\mathcal{V} = X_{12}$, and $\mathcal{W} = X_{22}$, we construct the following equivalent rank-constrained optimization problem of problem (6.3):

$$\underset{K,X_{11},X_{22},X_{12},Z_{11}}{\text{minimize}} \quad \mathbf{Tr}(QX_{11}) + \mathbf{Tr}(RX_{22}) \tag{6.4}$$

$$\begin{aligned}
\text{subject to: } & K: \text{ Row-Column } (r,c)\text{-sparse,} \\
& X_{11} \succ 0, \\
& AX_{11} + X_{11}A^T + BX_{12}^T + X_{12}B^T + I_n = 0, \\
& \mathbf{rank}\left(\begin{bmatrix} X_{11} & X_{12} & I_n \\ X_{12}^T & X_{22} & K \\ I_n & K^T & Z_{11} \end{bmatrix}\right) = n,
\end{aligned}$$

Now, we translate the row-column (r,c)-sparsity of K into rank constraints via the following proposition.

Proposition 28. *The optimization problem (6.4) can equivalently be reformulated into the following rank-constrained optimization problem:*

$$\underset{K,X_{11},X_{22},X_{12},Z_{11}}{\textit{minimize}} \quad \mathbf{Tr}(QX_{11}) + \mathbf{Tr}(RX_{22}) \tag{6.5a}$$

$$\textit{subject to: } X_{11} \succ 0, \tag{6.5b}$$

$$AX_{11} + X_{11}A^T + BX_{12}^T + X_{12}B^T + I_n = 0, \tag{6.5c}$$

$$\mathbf{rank}\left(\begin{bmatrix} X_{11} & X_{12} & I_n \\ X_{12}^T & X_{22} & K \\ I_n & K^T & Z_{11} \end{bmatrix}\right) = n, \tag{6.5d}$$

$$\mathbf{rank}\Big(\mathbf{diag}\big(K(i,:)\big)\Big) \leq r,\ 1 \leq i \leq m, \tag{6.5e}$$

$$\mathbf{rank}\Big(\mathbf{diag}\big(K(:,j)\big)\Big) \leq c,\ 1 \leq j \leq n. \tag{6.5f}$$

For the sake of simplicity in our notations let us assume that

$$H = \begin{bmatrix} X_{11} & X_{12} & I_n \\ X_{12}^T & X_{22} & K \\ I_n & K^T & Z_{11} \end{bmatrix}.$$

6.5 Bi-Linear Rank Penalty Technique for Computing Row-Column (r, c)-Sparse LQC Design

All the expressions in optimization problem (6.5) are convex excluding the $m+n+1$ rank constraints (6.5d), (6.5e), and (6.5f). In this section, we describe our strategy in facing such non-convex constraints.

It is a well-known fact that existence of the rank constraint leads to computational difficulties in the corresponding optimization problem. Hence, to handle such a difficulty, we employ the bi-linear rank penalty technique which has been shown to be effective in recent research works by [26–28, 72, 77]. Basically, such a methodology is proposed on the basis of substituting the rank constraints with convex relaxations on positive semi-definite cone and subsequently, establishing additional convex constraints along with a bi-linear term to the objective function. Thus, (6.5d) is replaced by the positive semi-definite constraint $H \succeq 0$ while establishing such additional convex constraints along with a bi-linear term to the corresponding objective function.

To make a similar convex relaxation regarding the rank constraints (6.5e), and (6.5f), motivated by sparsification via non-fragility notion presented by [33], we make the assumption that for all elements, our resulted row-column (r, c)-sparse K has the same sign of corresponding standard LQR design namely K^{LQR}. In fact, in [33], the ij^{th} element of a class of (non-fragility)-based sparsified controller K^{nfs} is obtained

via the following rule:

$$K_{ij}^{nfs} = 0, \qquad \text{if } |K_{ij}^{LQR}| \leq \rho,$$
$$K_{ij}^{nfs} = K_{ij}^{LQR}, \qquad \text{if } |K_{ij}^{LQR}| > \rho,$$

where ρ denotes the non-fragility of K^{LQR}.

Thus, such an assumption on sign of elements of our proposed row-column (r, c)-sparse K can be embedded in:

$$K \circ \mathbf{sign}(K^{LQR}) \geq 0,$$

which enables us to relax the $m + n$ rank constraints (6.5e) and (6.5f) which leads to obtain a problem with all convex constraints and auxiliary bi-linear terms in the objective function which can be handled utilizing the iterative algorithm implemented by bi-linear rank penalty technique.

Remark 13. *To provide more details on ρ, we express the definition of non-fragility introduced by [33] as follows:*

$$\rho := \sup\{\alpha > 0 : A + B(K + \Delta K) \text{ is Hurwitz}, \ \forall \Delta K \in S_\alpha\},$$

where

$$S_\alpha := \{X \in \mathbb{R}^{m \times n} : \|X\|_{\max} < \alpha\}.$$

Note that S_ρ denotes the maximal stabilizing hypercube with side length 2ρ, i.e., if we add any point ΔK of S_ρ to K, it will give us a stabilizing static feedback controller $K + \Delta K$.

Inspired by Theorem 5 in [27], we state the following similar theorem.

Theorem 29. *Let us consider the rank-constrained optimization problem (6.5) and employ the following auxiliary optimization problem:*

$$\begin{aligned}
\underset{K,X_{11},X_{22},X_{12},Z_{11},G's}{\text{minimize}} \ & \mathbf{Tr}(QX_{11}) + \mathbf{Tr}(RX_{22}) + \nu\mathbf{Tr}(GH) + \nu_r \sum_{i=1}^{m} \mathbf{Tr}(G_r^i H_r^i) \\
& + \nu_c \sum_{j=1}^{n} \mathbf{Tr}(G_c^j H_c^j) \qquad (6.6) \\
\text{subject to: } & (6.5b) - (6.5c), \\
& 0 \preceq G \preceq I_{2n+m},\ \mathbf{Tr}(G) = n + m, \\
& H \succeq 0, \\
& 0 \preceq G_r^i \preceq I_n,\ \mathbf{Tr}(G_r^i) = n - r,\ 1 \leq i \leq m, \\
& H_r^i \succeq 0,\ 1 \leq i \leq m, \\
& 0 \preceq G_c^j \preceq I_m,\ \mathbf{Tr}(G_c^j) = m - c,\ 1 \leq j \leq n, \\
& H_c^j \succeq 0,\ 1 \leq j \leq n,
\end{aligned}$$

in which $\nu > 0$, $\nu_r > 0$, *and* $\nu_c > 0$ *are some given design parameters and* $H_r^i = \mathbf{diag}\Big(K(i,:) \circ \mathbf{sign}\big(K^{LQR}(i,:)\big)\Big)$ *and* $H_c^j = \mathbf{diag}\Big(K(:,j) \circ \mathbf{sign}\big(K^{LQR}(:,j)\big)\Big)$ *are considered for* $1 \leq i \leq m$ *and* $1 \leq j \leq n$, *respectively. By* G*'s in subscript of minimize, we mean* G, G_r^i*'s, and* G_c^j*'s. If problem (6.5) is feasible, then there exist constants* $\eta > 0$, $\eta_r > 0$, *and* $\eta_c > 0$ *for which the optimal solutions* H, H_r^i*'s, and* H_c^j*'s achieved by solving the* (6.6) *satisfy*

$$\begin{aligned}
\mathbf{rank}(H; \eta\nu^{-1}) &\leq n, \\
\mathbf{rank}(H_r^i; \eta_r {\nu_r}^{-1}) &\leq r,\ 1 \leq i \leq m, \\
\mathbf{rank}(H_c^j; \eta_c {\nu_c}^{-1}) &\leq c,\ 1 \leq j \leq n.
\end{aligned}$$

According to $H \succeq 0$ and the specific structure of matrix H the inequality $\mathbf{rank}(H) \geq n$ is satisfied. As a result of Theorem 29, we are able to solve the optimization problem

(6.6) for an appropriately-selected parameter ν to obtain a sub-optimal solution to the problem (6.5). To simplify our notations, we denote the stack of all optimization variables except the variables G, G_r^i's, and G_c^j's by Z. The optimization problem (6.6) can be restated as

$$\begin{aligned} \underset{Z,G,G_r^{i\prime}s,G_c^{j\prime}s}{\text{minimize}} \quad & \mathcal{F}(Z, G, G_r^{i\prime}s, G_c^{j\prime}s) \\ \text{subject to:} \quad & Z \in \mathcal{C}_Z, \\ & G \in \mathcal{C}_G, \\ & G_r^i \in \mathcal{C}_{G_r^i},\ 1 \leq i \leq m, \\ & G_c^j \in \mathcal{C}_{G_c^j},\ 1 \leq j \leq n, \end{aligned}$$

where $\mathcal{C}_Z$ is the convex set formed by the constraints (6.5b)-(6.5c) in company with $H \succeq 0$, $H_r^i \succeq 0$, and $H_c^j \succeq 0$ and the convex set $\mathcal{C}_G$ is characterized by $\mathbf{Tr}(G) = n+m$ and $0 \preceq G \preceq I_{2n+m}$. The similar notational agreement to G_r^is and G_c^js are applied as well. We mention that $\mathcal{F}(Z, G, G_r^{i\prime}s, G_c^{j\prime}s)$ represents the bi-linear objective function in the optimization problem (6.6). The previously mentioned reformulation enables us to tackle this problem by iteratively minimizing the objective function for Z and G's. The key steps of such an iterative method are twofold: Z-minimization sub-problems and G's-minimization sub-problems. As both Z-minimization and G's-minimization steps are convex optimization problems, they can be executed in an efficient way. Fortunately, for the G's-minimization, there exists an analytic solution, expressed by the next theorem which is built on Theorem 6 by [27].

Theorem 30. *The optimal solution of the* G's-minimization *step is presented with*

$$G^* = I_{2n+m} - \sum_{p=1}^{n} u_p {u_p}^T, \tag{6.7}$$

$$G_r^{i\,*} = I_n - \sum_{p_r=1}^{r} u_{p_r}^i {u_{p_r}^i}^T, \ 1 \leq i \leq m, \tag{6.8}$$

$$G_c^{j\,*} = I_m - \sum_{p_c=1}^{c} u_{p_c} {u_{p_c}^j}^T, \ 1 \leq j \leq n, \tag{6.9}$$

where vectors u_p *for* $p = 1, \ldots, n$ *represent the singular vectors corresponding to the* n *greatest singular values of* H*. Vectors* $u_{p_r}^i$ *and* $u_{p_c}^j$ *are defined similarly for* $p_r = 1, \ldots, r$ *and* $p_c = 1, \ldots, c$*, respectively.*

We employ the following sequence of iterations to achieve the minimizer of the constrained problem (6.6). First, we solve the Z-minimization and G's-minimization sub-problems

$$Z^{(k+1)} = \arg\underset{Z \in \mathcal{C}_Z}{\text{minimize}}\ \mathcal{F}(Z, G^{(k)}, G_r^{i,(k)}, G_c^{j,(k)}), \tag{6.10}$$

$$G^{(k+1)} = I_{2n+m} - \sum_{p=1}^{n} u_p^{(k+1)} u_p^{(k+1)^T}, \tag{6.11}$$

$$G_r^{i,(k+1)} = I_n - \sum_{p_r=1}^{r} u_{p_r}^{i,(k+1)} u_{p_r}^{i,(k+1)^T}, \ 1 \leq i \leq m, \tag{6.12}$$

$$G_c^{j,(k+1)} = I_m - \sum_{p_c=1}^{c} u_{p_c}^{j,(k+1)} u_{p_c}^{j,(k+1)^T}, \ 1 \leq j \leq n, \tag{6.13}$$

where $H^{(k+1)} = \sum_{p=1}^{2n+m} \sigma_p^{(k+1)} u_p^{(k+1)} u_p^{(k+1)^T}$ is the singular value decomposition of $H^{(k+1)}$. We have similar notations for H_r^i's and H_c^j's. The stopping criterion is established by $\epsilon^{(k+1)} \leq \epsilon^*$, where ϵ^* denotes the pre-specified desired precision, with the following update rule:

$$\epsilon^{(k+1)} = \frac{\|K^{(k+1)} - K^{(k)}\|_2}{\|K^{(k+1)}\|_2}. \tag{6.14}$$

Algorithm 1: Solution to problem (6.6)

Inputs: A, B, Q, R, ν, ν_r, ν_c *and* ϵ^*.

1: Initialization:

Set $G^{(0)} = I_{2n+m}$, $G_r^{i,(0)} = I_m \;\; 1 \leq i \leq m$,

$G_c^{j,(0)} = I_n, \;\; 1 \leq j \leq n$, $\epsilon^{(0)} > \epsilon^*$, $K^{(0)} = 0$, *and* $k = 0$.

2: While $\epsilon^{(k)} > \epsilon^*$ *Do*

3: *Update* $Z^{(k+1)}$ *solving the (6.10),*

4: *Update* $G^{(k+1)}$ *via (6.11),*

5: *Update* $G_r^{i,(k+1)}$*'s via (6.12),*

6: *Update* $G_c^{j,(k+1)}$*'s via (6.13),*

5: *Update* $\epsilon^{(k+1)}$ *via (6.14),*

6: $k \leftarrow k + 1$,

7: End While

8: Truncate K.

Output: K

In the final step of the algorithm, we truncate unimportant elements, e.g., the ones less than 5×10^{-5}, of the computed K. The small enough elements showcase relatively weak couplings among the nodes in the controller design. Our proposed algorithm is summarized in *Algorithm 1.*

Remark 14. *It is remarkable that in implementation phase of our algorithm, utilization of a constraint to bound* $\mathbf{Tr}(QX_{11}) + \mathbf{Tr}(RX_{22})$ *will help us to find a better locally optimal solutions. Because, otherwise, by removing them, convex optimization solver would not be able to give row-column* (r, c)*-sparse designs with higher quality in terms of sparsity-performance specifications. Also, due to some practical purposes, sometimes, it is not permitted to have a quadratic performance loss larger than a certain value which highlights the necessity of using such an optimization constraint.*

Remark 15. *By putting* $r = m$ *our row-column* (r, c)*-sparse design reduces to column* c*-sparse design. Likewise, substituting* $c = n$ *changes our row-column* (r, c)*-sparse design to row* r*-sparse design. Additionally, our row-column* (r, c)*-sparse design can be generalized to any sparsity-promoting control problem with sparsity constraints in the form of* $\|\mathcal{S}(K)\|_0 \leq s$ *in which* $\mathcal{S}(K)$ *is an arbitrarily-chosen set of controller*

elements K_{ij}'s and s is a pre-fixed non-negative integer number.

6.6 Numerical Simulations

To assess the efficacy of our proposed methodology, we consider the class of randomly-generated linear systems. Such systems can be generated via MATLAB command *randn*.

Also, we define the following performance measure to make a comparison between the given controller and row-column (r, c)-sparse one:

$$\mathcal{R} = 100 \times \frac{J(K) - J(K^{LQR})}{J(K^{LQR})},$$

where $J(K)$ and $J(K^{LQR})$ denotes the corresponding quadratic performance losses for K and K^{LQR}, respectively.

6.6.1 Row-Column (r, c)-Sparse LQC Design

Let us consider 25×25 randomly-generated system A and 25×20 randomly-generated input matrix B. Suppose that $Q = 2I_{25}$, $R = I_{20}$, $r = 10$, $c = 13$, $\epsilon^* = 0.0005$, $\nu = \nu_r = \nu_c = 5000$.

Running the *Algorithm 1*, we obtain row-column $(10, 13)$-sparse LQC design for which the sparsity pattern is depicted by Figure 6.1. The measure $\mathcal{R}$ for such a design is equal to 39.08 % while 60 % of controller elements has been sparsified.

6.6.2 Sparsity-Performance Trade-Offs in terms of (r,$\mathcal{R}$) and (c,$\mathcal{R}$) Pairs

Considering the same randomly-generated matrices A and B, we investigate the relationship between $\mathcal{R}$ % and r by fixing the c and vice versa. Firstly, we assume that

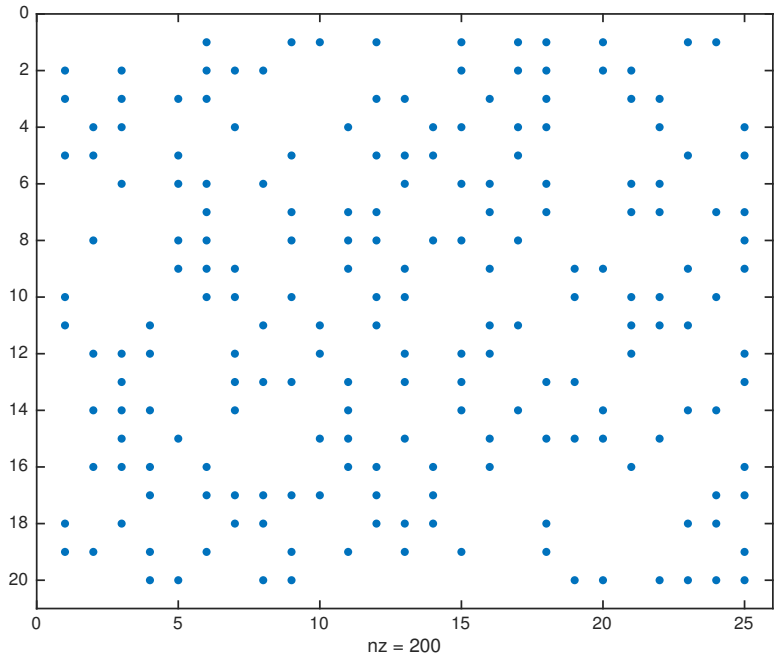


Figure 6.1: Sparsity pattern of row-column $(10, 13)$-sparse LQC design for 25×25 randomly-generated system with 25×20 randomly-generated input matrix B. Blue dots represent the non-zero elements and it nz denotes the number of non-zero elements.

$c = 13$ is fixed, then for $10 \leq r \leq 16$, then setting the $Q = 2I_{25}$, $R = I_{20}$, $\epsilon^* = 0.0005$, $\nu = \nu_r = \nu_c = 5000$, Figure 6.2 is plotted. As it is observed, there exists a fundamental trade-off between r (maximum allowed density level for each row of controller) and performance loss percentage $\mathcal{R}$ %. Secondly, assuming the $r = 20$ and $10 \leq c \leq 16$, and substituting the $Q = 2I_{25}$, $R = I_{20}$, $\epsilon^* = 0.0005$, $\nu = \nu_r = \nu_c = 5000$, Figure 6.3 is drawn which depicts the fundamental trade-off between c (maximum allowed density level for each column of controller) and performance loss percentage $\mathcal{R}$ %.

6.7 Conclusion

We utilize the bi-linear rank penalty technique and non-fragility notion for the design of row-column (r, c)-sparse LQC for LTI systems. We do not use the ℓ_1-norm

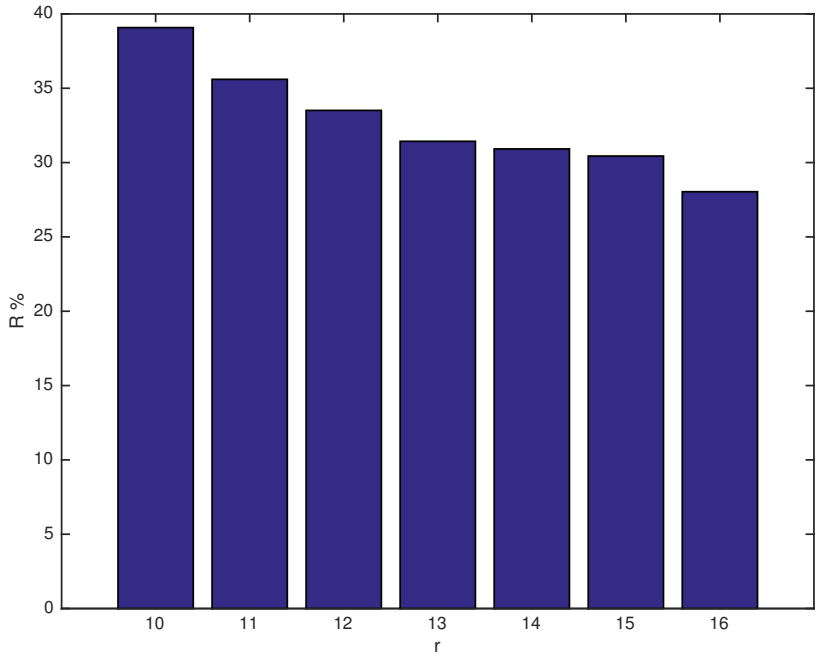


Figure 6.2: Relationship between r and $\mathcal{R}$ % for 25×25 randomly-generated system with 25×20 randomly-generated input matrix B.

relaxation. Instead, utilizing the non-fragility notion motivates us to translate the row-column (r, c)-sparse LQC design problem into a rank-constrained optimization problem which can sub-optimally be handled via bi-linear rank penalty technique. As future directions, the extension of row-column (r, c)-sparse LQC design to large-scale systems and improving the sub-optimality caused by non-fragility notion can be interesting topics.

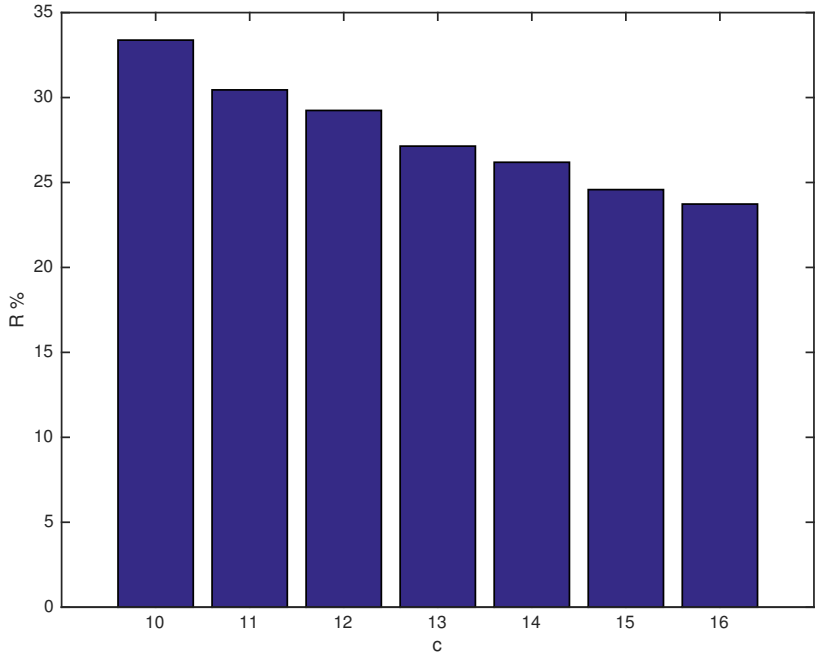


Figure 6.3: Relationship between c and $\mathcal{R}$ % for 25×25 randomly-generated system with 25×20 randomly-generated input matrix B.

Chapter 7

State Feedback Controller Sparsification via Non-Fragility Notion

7.1 Introduction

The notion of fragility has attained growing attention in control theory literature. In past two decades, to name a few, we can list the following research works [113–124]. Such a notion basically refers to the sensitivity of the controller design parameters with respect to the stability guarantee. Two fundamental reasons are expressed to highlight the importance of the fragility in controller implementation and control theoretical design; (i) Imprecision in analog-digital and digital-analog conversions, finite word length, and finite resolution measuring instruments and round-off errors in numerical computations [113]. (ii) Every theoretical design needs readjustment because no scalar index can describe all the performance requirements in a control system [113]. Consequently, a certain level of tolerance against the changes in controller parameters is necessary.

Considering the trade-off between robustness and non-fragility [113], researchers have made an effort to design robust and non-fragile controllers simultaneously. In [118–121] and [123], non-fragile $\mathcal{H}_\infty$ controllers are proposed for diverse structural uncertainties including multiplicative and additive via LMI approach. Also, a set of stabilizing robust non-fragile controllers are formulated in [122]. In [124], it is observed that the traditional Ziegler-Nichols PID tuning method is surprisingly non-fragile, in the sense that it remains in the admissible stabilizing region. Thus, it has been shown that the notion of non-fragility has been significantly effective in control theoretic designs and applications.

After highlighting the effectiveness of the non-fragility in controller design, from another different point of view, we note that during the past decade, the area of sparsity-promoting optimal control has attained considerable achievements [1, 13, 14, 18, 21, 26–30, 32, 35, 38, 40, 42]. The main aim in sparsity-promoting optimal control is to decrease the number of communication links among nodes preserving the guaranteed level of performance.

All the above-mentioned sparsity-promoting control methods are unfortunately unable to deal with large-scale systems and fail to propose a sparse feedback controller for such systems. In this chapter, we will show how a notion of non-fragility can be utilized as an effective tool to find sparse stabilizing feedback controllers in the vicinity of a given stabilizing feedback controller even for a large-scale system. However, a drawback of our proposed method is that an upper bound exists for the rate of sparsification while other sparsity-promoting optimal controller design methods do not have such an issue when they are applied to medium-size systems. Another drawback is that we do not consider any performance measure in our sparsification approach and it is intentional, since we want to propose a sparsification method which is capable of being applied to large-scale systems. However, we will take advantage of $\mathcal{H}_2$ performance measure to improve the closed-loop performance in the case of

small-size systems. In addition to deriving the lower and upper bounds on our introduced non-fragility, via extensive numerical simulations, we visualize the trade-off between upper bound on non-fragility and sparsity level of the feedback controller. Also, it is illustrated through case studies that the (non-fragilty)-based sparsification procedure can outperform a well-respected existing method in the literature, in terms of sparsity-performance trade-off behavior. In fact, our numerical simulations show that although our proposed sparsification method does not employ any performance measure, surprisingly, its closed-loop performance loss is reasonable and not quite much. Moreover, for the case of small-size systems, utilizing a $\mathcal{H}_2$ performance measure, two greedy algorithms are proposed to obtain a set of sparse feedback controllers out of a given stabilizing feedback controller.

This chapter is structured as follows: After stating the mathematical notations in Section 7.2, Section 7.3 defines the concept of non-fragility for a feedback controller and then provides analytic lower and upper bounds on such a defined non-fragility notion. Section 7.4 explains how the introduced non-fragility notion can be helpful to sparsify a given feedback controller. Section 7.5 by providing the extensive numerical simulations evaluates the effectiveness of our non-fragility based sparsification method for large-scale systems, points out some notes about performance loss, sparsity level, and their relationship including that the sparser feedback controller we consider, the more fragility is observed. As further, two greedy algorithms (brute force greedy and gradient-based greedy) are proposed to obtain a set of sparse feedback controllers. Section 7.6 mentioning some discussions and concluding remarks ends the chapter.

7.2 Mathematical Notations

In this chapter, vectors and matrices are shown with lower-case and upper-case letters, respectively. Set of real numbers, $n \times 1$ real vectors, and $m \times n$ real matrices are

represented by $\mathbb{R}$, $\mathbb{R}^n$, and $\mathbb{R}^{m\times n}$, respectively. The transpose of a matrix is denoted by superscript T. The i^{th} column of matrix M is denoted by $M(:,i)$. Vector e_i is defined as $I(:,i)$ where I denotes the identity matrix. The Kronecker matrix product and Hadamard matrix product are represented by $\otimes$ and $\circ$, respectively. The Euclidean norm of vector v is denoted by $\|v\|_2$. The notation $\|M\|_{\max}$ represents the maximum absolute value of all elements of matrix M. By $\lambda_i(M)$, we mean the eigenvalue of M which has the i^{th} largest real part and $\lambda_{\max}$ is defined as $\lambda_{\max} := \lambda_1$. A matrix M is called Hurwitz if and only if $\lambda_{\max}(M) < 0$. The positive definiteness and negative definiteness are shown by $\succ 0$ and $\prec 0$, respectively. The ℓ_1 norm of matrix M is sum of absolute value of its elements and denoted by $\|M\|_1$. The number of non-zero elements of matrix M is denoted by $\|M\|_0$ and called ℓ_0 sparsity measure. Trace of square matrix M is sum of its eigenvalues and shown by $\mathbf{Tr}(M)$. The element-wise sign function of matrix M is represented by $\mathbf{sign}(M)$. The supremum of a set is denoted by sup. The set subtraction and union are denoted by $\setminus$ and $\cup$, respectively. The empty set is represented by $\emptyset$. The big O time complexity is denoted by $\mathcal{O}$.

7.3 Non-Fragility Notion: Definition, Lower and Upper Bounds

Suppose that the linear time-invariant (LTI) system

$$\dot{x}(t) = Ax(t) + Bu(t) + Dd(t), \tag{7.1}$$

is controlled by

$$u(t) = Fx(t), \tag{7.2}$$

where $x(t) \in \mathbb{R}^n$, $u(t) \in \mathbb{R}^m$, $d(t) \in \mathbb{R}^p$, and F denote the state vector, control input, disturbance input, and state feedback controller, respectively.

Given a triplet $\Sigma = (A, B, F)$ wherein F is an arbitrary stabilizing state feedback controller, the *non-fragility* is defined as follows:

$$\rho(\Sigma) := \sup\{r | A + B(F + \Delta) \text{ stays Hurwitz } \forall \Delta \in \mathcal{S}_r\}, \tag{7.3}$$

wherein

$$\mathcal{S}_r := \{X \in \mathbb{R}^{m\times n} | r > 0,\ \|X\|_{\max} < r\}. \tag{7.4}$$

It is noteworthy that $\mathcal{S}_{\rho(\Sigma)}$ denotes the largest stabilizing hypercube with side length $2\rho(\Sigma)$, i.e., if we add any point Δ of $\mathcal{S}_{\rho(\Sigma)}$ to F, it will give us a stabilizing state feedback controller $F + \Delta$. In the sequel, we present Theorem 31 to enlighten the point that our introduced non-fragility is a strictly positive number, in other words, it is well-posed.

Theorem 31. *Given a triplet $\Sigma = (A, B, F)$, the non-fragility $\rho(\Sigma)$ defined by* (7.3) *is a strictly positive number.*

Proof. Let us define the following scalar-valued function:

$$g(V) := \lambda_{\max}(A + BV) = \lambda_{\max}\big(A + B\sum_{i=1}^{n} e_i^T \otimes V(:, i)\big).$$

We know that the function $g(v)$ is a continuous function of v where

$$v := [V(:, 1)^T \dots V(:, n)^T]^T.$$

Because, it is a composition of a series of continuous operations including taking maximum, taking real part, and calculating eigenvalues of affine expression of state feedback controller. Thus, defining the

$$f := [F(:, 1)^T \dots F(:, n)^T]^T,\ \delta := [\Delta(:, 1)^T \dots \Delta(:, n)^T]^T,$$

and considering the continuity property of g, we can say that

$$g(f+\delta) - g(f) \leq \epsilon, \tag{7.5}$$

holds for all $\|\delta\|_2 \leq c(\epsilon)$ where $c(\epsilon)$ is a strictly positive number which depends on ϵ. To guarantee $g(f+\delta) < 0$, we enforce $g(f) + \epsilon < 0$ to hold, i.e., we choose ϵ as follows:

$$\epsilon < -g(f) = -\lambda_{\max}(A + BF).$$

The side length of the largest hypercube inscribed by $\|\delta\|_2 \leq c(\epsilon)$ (equivalently $\|\Delta\|_{\max}$ in space of $\mathbb{R}^{m\times n}$) is equal to $\frac{2c(\epsilon)}{\sqrt{mn}}$. Because, all the 2^{mn} corner points of the hypercube, i.e., $(\pm\frac{c(\epsilon)}{\sqrt{mn}}, \ldots, \pm\frac{c(\epsilon)}{\sqrt{mn}})$ satisfies

$$\sum_{i=1}^{mn} \delta_i^2 \leq c(\epsilon)^2,$$

and particularly, the equality holds which implies the maximality of such a hypercube. Thus, regarding the non-fragility $\rho(\Sigma)$ we deduce

$$\rho(\Sigma) \geq \frac{c(\epsilon)}{\sqrt{mn}}, \ \forall \epsilon \in \big(0, -\lambda_{\max}(A + BF)\big). \tag{7.6}$$

Since $\frac{c(\epsilon)}{\sqrt{mn}} > 0$ holds, then non-fragility $\rho(\Sigma)$ is strictly positive and proof is done. □

Remark 16. *To obtain the best lower bound on non-fragility $\rho(\Sigma)$, we can take supremum from lower bound $\frac{c(\epsilon)}{\sqrt{mn}}$ over all choices $\epsilon \in \big(0, -\lambda_{\max}(A + BF)\big)$. It can easily be checked that $c(\epsilon)$ is an increasing function of ϵ. Thus, the best lower bound would be $\lim_{\epsilon\to-\lambda_{\max}(A+BF)} c(\epsilon)$.*

Remark 17. *It is noteworthy that the result of Theorem 31 may seem trivial because of $r > 0$ in definition of S_r. However, it is not the case and the result of Theorem 31*

is nontrivial. Because, Theorem 31 basically shows that in (7.3)*, the set on which the supremum is taken, i.e., the following set:*

$$\{r | A + B(F + \Delta) \text{ stays Hurwitz } \forall \Delta \in S_r\},$$

is not empty. If such a fact is not shown, then taking the supremum will not be possible and consequently, the strict positivity of the non-fragility $\rho(\Sigma)$ *defined by* (7.3) *cannot be implied trivially.*

Remark 18. *In the rest of the chapter, for the sake of the simplicity in our notations, we drop the argument* Σ *from* $\rho(\Sigma)$ *and simply use* ρ*.*

To compute ρ in an exact way, the infinite set of feasibility problems should be considered. However, such an approach is not computationally cheap nor practical. In [125], it is mentioned that calculating the exact minimum destabilizing real perturbation is unfortunately impossible. Thus, we present the following theorems which suggest an analytic upper bound on ρ.

Theorem 32. *Given a triplet* $\Sigma = (A, B, F)$*, the non-fragility* $\rho(\Sigma)$ *defined by* (7.3) *is upper bounded by*

$$\hat{\rho} = -\frac{\mathbf{Tr}(A + BF)}{\|B\|_1}. \tag{7.7}$$

Proof. The expression $\mathbf{Tr}(A + BF + B\Delta)$ is sum of eigenvalues of $A + BF + B\Delta$ and must be negative. Thus, we have

$$\mathbf{Tr}(B\Delta) < -\mathbf{Tr}(A + BF), \tag{7.8}$$

for all Δ in $\mathcal{S}_\rho$. Since

$$(\rho - \zeta)\mathbf{sign}(B^T) \in \mathcal{S}_\rho,$$

then (7.8) holds for

$$\Delta = (\rho - \zeta)\mathbf{sign}(B^T),$$

$\big($In fact, the left hand side of (7.8) takes its maximum value for $\Delta = (\rho-\zeta)\mathbf{sign}(B^T)\big)$. Equivalently, we have

$$\mathbf{Tr}(B\Delta) = \mathbf{Tr}\big(B(\rho - \zeta)\mathbf{sign}(B^T)\big) = (\rho - \zeta)\|B\|_1 < -\mathbf{Tr}(A + BF).$$

Taking the supremum from both sides, (7.7) is resulted. □

Stating the following theorem, we improve the upper bound $\hat{\rho}$.

Theorem 33. *Given a triplet $\Sigma = (A, B, F)$, the non-fragility $\rho(\Sigma)$ defined by* (7.3) *is upper bounded by $\gamma\hat{\rho}$ where*

$$\gamma = \sup\{\alpha | \lambda_{\max}\big(A + BF + \beta\hat{\rho}B\mathbf{sign}(B^T)\big) < 0, \forall\beta \in [0, \alpha]\}. \tag{7.9}$$

and the parameter γ is less than or equal to 1*.*

Proof. Substituting the $\alpha = 0$, we observe that

$$\lambda_{\max}\big(A + BF + \beta\hat{\rho}B\mathbf{sign}(B^T)\big) = \lambda_{\max}(A + BF) < 0.$$

Because, $A + BF$ is Hurwitz. Since $\lambda_{\max}\big(A + BF + \beta\hat{\rho}B\mathbf{sign}(B^T)\big)$ is a continuous function of β and it takes the negative value of $\lambda_{\max}(A + BF)$ at $\beta = 0$, then $\alpha > 0$. Since for $\gamma + \theta$, there exists a $\beta \in [\gamma, \gamma + \theta]$ for which we have

$$\lambda_{\max}\big(A + BF + \beta\hat{\rho}B\mathbf{sign}(B^T)\big) \geq 0,$$

then $\rho < (\gamma + \theta)\hat{\rho}$ is resulted. By taking the infimum from both sides, the proof of

first part of the theorem is done.

To prove the second part, for $\beta = 1$, we claim that

$$\lambda_{\max}\big(A + BF + \beta\hat{\rho}B\mathbf{sign}(B^T)\big) \geq 0.$$

Because,

$$\sum_{i=1}^{n} \lambda_i\big(A + BF + \hat{\rho}B\mathbf{sign}(B^T)\big) = \mathbf{Tr}\big(A + BF + \hat{\rho}B\mathbf{sign}(B^T)\big) =$$

$$\mathbf{Tr}(A + BF) + \mathbf{Tr}\big(\hat{\rho}B\mathbf{sign}(B^T)\big) = \mathbf{Tr}(A + BF) + \hat{\rho}\mathbf{Tr}\big(B\mathbf{sign}(B^T)\big) = 0,$$

and since the sum of all real parts of eigenvalues is 0, then $\lambda_{\max}$ cannot be less than 0. Meanwhile, the last line of the above-mentioned lines is resulted from (7.7). Thus, α cannot be greater than or equal to 1. Since $\alpha < 1$, then according to the definition of supremum, it is resulted that $\gamma \leq 1$. □

7.4 State Feedback Controller Sparsification Procedure

After obtaining the upper bounds on non-fragility, by taking advantage of introduced non-fragility notion, a procedure to sparsify a given stabilizing state feedback controller is presented. The following theorem enlightens such a procedure and shows how the notion of non-fragility can be utilized as an effective sparsification tool, specifically, for the case of large-scale systems.

Theorem 34. *Given a triplet $\Sigma = (A, B, F)$, a class of stabilizing state feedback controllers F^{nf} consisting of sparse stabilizing state feedback controllers F^{nfs} is characterized as follows:*

$$F^{nf} = G^{\rho} \circ F, \tag{7.10}$$

where

$$\begin{cases} G_{ij}^{\rho} = 0 & \text{if } (i,j) \in \mathcal{S}, \\ G_{ij}^{\rho} \in (1 - \frac{\rho}{|F_{ij}|}, 1 + \frac{\rho}{|F_{ij}|}) \backslash \{0\} & \text{otherwise}, \end{cases} \tag{7.11}$$

and sparsity structure $\mathcal{S} \subseteq \mathcal{I}_{\rho} = \{(i,j) |\ |F_{ij}| < \rho\}$. *When* $\mathcal{S} \neq \emptyset$, F^{nf} *is taken to account as a sparse stabilizing state feedback controller which is called* F^{nfs}.

Proof. To prove that $F^{\mathrm{n}f}$ is stabilizing, we show that $(F^{\mathrm{n}f} - F) \in \mathcal{S}_{\rho}$. First, let us consider $(i,j) \in \mathcal{S}$. Then,

$$|F_{ij}^{\mathrm{n}f} - F_{ij}| = |0 - F_{ij}| < \rho,$$

holds, since $(i,j) \in \mathcal{S}$ and $\mathcal{S} \subseteq \mathcal{I}_{\rho}$. Second, assume that $(i,j) \notin \mathcal{S}$. Thus, we get

$$|F_{ij}^{\mathrm{n}f} - F_{ij}| = |(G_{ij}^{\rho} - 1)F_{ij}| = |G_{ij}^{\rho} - 1||F_{ij}|.$$

Since $(i,j) \notin \mathcal{S}$, then

$$-\frac{\rho}{|F_{ij}|} < G_{ij}^{\rho} - 1 < \frac{\rho}{|F_{ij}|},$$

or equivalently

$$|G_{ij}^{\rho} - 1| < \frac{\rho}{|F_{ij}|}.$$

Thus,

$$|F_{ij}^{\mathrm{n}f} - F_{ij}| < \rho,$$

is resulted.

Since for all elements of $F^{\mathrm{n}f} - F$, $|F_{ij}^{\mathrm{n}f} - F_{ij}| < \rho$ is satisfied, then

$$\|F^{\mathrm{n}f} - F\|_{\max} < \rho,$$

holds, i.e., $(F^{\mathrm{n}f} - F) \in \mathcal{S}_{\rho}$ and proof is done. □

Remark 19. *Note that in the case that $(i,j) \notin \mathcal{S}$ is satisfied, G_{ij}^{ρ} can be set to* 1*, when we desire to remove just structured weak links and let the other links to remain unchanged.*

Remark 20. *In our proposed sparsification procedure, to mention the dependency of F^{nfs} on $\tilde{\rho}$, we will simply use $F^{nfs}(\tilde{\rho})$ whenever it is needed.*

Due to the fact that we cannot compute the exact value of non-fragility ρ, we develop the following sparsification procedure based on our obtained upper bounds on non-fragility ρ:

Non-Fragility Sparsification (NFS) Procedure

---

1. *Begin*
2. *Compute $\hat{\rho}$ via* (7.7)*,*
3. *Compute γ via* (7.9)*,*
4. *Compute $\tilde{\rho} = \gamma\hat{\rho}$,*
5. *Compute $F^{nfs}(\tilde{\rho})$,*
6. *If $\lambda_{\max}\big(A + BF^{nfs}(\tilde{\rho})\big) < 0$, go to step 7,*
   else, update $\tilde{\rho}$ with $(\eta - \varepsilon)\gamma\hat{\rho}$ and go to step 5,
   where $\eta < 1$ is computed via
   $\lambda_{\max}\big(A + BF^{nfs}(\eta\gamma\hat{\rho})\big) = 0$,
   and ε is an infinitesimal strictly positive number,
7. *End.*

Remark 21. *In addition to the advantages of (NFS) procedure, including its high speed, simplicity, and applicability to large-scale systems, the sparse state feedback controller proposed by this method can be utilized as an initialization for methods presented by [1, 21].*

An immediate implication of Theorem 34 is stated in the following corollary.

Corollary 35. *Given a triplet $\Sigma = (A, B, F)$, the non-fragility $\rho(\Sigma)$ defined by* (7.3) *is upper bounded by $\eta\gamma\hat{\rho}$.*

Remark 22 (Time Complexity)**.** *In the worst-case scenario, the time complexity of (NFS) procedure is equal to $\mathcal{O}\big(n^3 s + mn + (n^3 + mn)s\big)$ or equivalently $\mathcal{O}\big((n^3 + mn)s\big)$ wherein s refers to the time complexity of solving the equation $\lambda_{\max}(A + BF^{nfs}) = 0$ with a prespecified precision. The expression n^3 is the corresponding term for the eigenvalue decomposition (via Cholesky factorization) [126]. Since we need to find the $\lambda_{\max}$ in (NFS) procedure, such a term appears in the time complexity. The term mn shows the time complexity of computation of F^{nfs} via Hadamard matrix product and element-wise comparison. Assuming the $m \leq n$, the dominant term in $\mathcal{O}\big((n^3+mn)s\big)$ is $n^3 s$. As another sparsification method, the method proposed by [1], attains the time complexity of $\mathcal{O}\big((n^3 + mn)q\big)$ in which n^3, mn, and q are corresponding terms to (Lyapunov and Sylvester equations), (matrix addition, Hadamard matrix product, and element-wise comparison), and Frobenius norm stopping criteria of the algorithm, respectively. Assuming the $m \leq n$, the dominant term in $\mathcal{O}\big((n^3 + mn)q\big)$ is $n^3 q$. It is noteworthy that (NFS) procedure is finished after at most 2 iterations and s is determined based on the chosen nonlinear optimization method while the method proposed by [1] is finished whenever the Frobenius norm stopping criteria are fulfilled and there is no analytical determination about q.*

7.5 Numerical Simulations

This section evaluates the effectiveness of our proposed method in terms of capability of being applied to large-scale systems, relative performance/sparsity specifications, and trade-off between upper bound on non-fragility and sparsity level of state feedback controller. Moreover, for the certain case of the small-size systems, two greedy algorithms are proposed to obtain a set of sparse state feedback controllers given a stabilizing state feedback controller. Unlike the case of large-scale systems, the small size of such systems allows us to consider and improve the closed-loop performance.

It is remarkable that throughout the section, to choose the state feedback controller to be sparsified, i.e., F, we will utilize the standard linear-quadratic regulator (LQR) design unless otherwise it is stated.

7.5.1 Sparsified State Feedback Controller via Non-Fragility for Large-Scale Systems

Considering the large-scale randomly generated systems and spatially-decaying systems, the effectiveness of our non-fragility sparsification (NFS) procedure is assessed.

Randomly Generated System

Utilizing the MATLAB command $randn(n)$, we produce $10,000$ by $10,000$ randomly generated matrices A and B. Assuming the LQR state-weight matrix $Q = 2I$, LQR input-weight matrix $R = 5I$, $D = B$, $\mathcal{S} = \mathcal{I}_\rho$, $G_{ij}^\rho = 1$ for all links satisfying $(i,j) \notin \mathcal{S}$, and running the (NFS) procedure, for such a large-scale system, F^{nfs} is obtained. Figures 7.1 and 7.2 visualize the eigenvalues of open loop and closed loop for both cases, (LQR) design F and (NFS) procedure F^{nfs}, respectively. As Figures 7.1 and 7.2 demonstrate, the open loop is unstable while both closed loops are stable. Costs and density levels for both cases, F (LQR) and $F^{\mathrm{n}fs}$ (NFS), are demonstrated

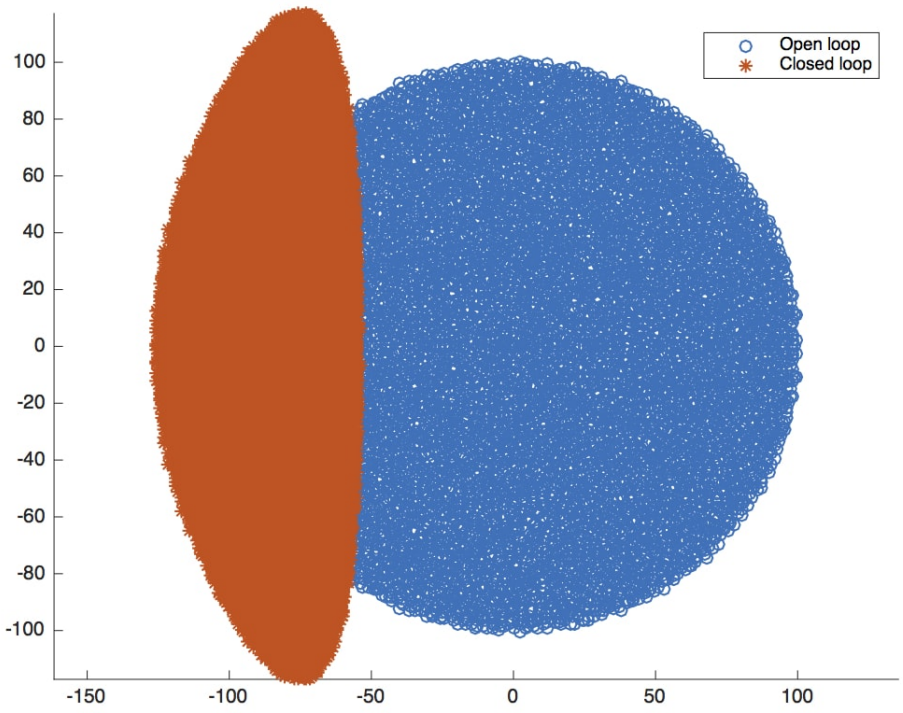


Figure 7.1: Eigenvalues of open loop and closed loop for F (LQR) $(10,000 \times 10,000$ randomly generated system).

$J(F)$	$J(F^{\mathrm{n}fs})$
$4,191,252.0732$	$4,318,583.8859$
$\|F\|_0$	$\|F^{\mathrm{n}fs}\|_0$
$100,000,000$	$65,126,579$

Table 7.1: Cost and cardinality quantities for F (LQR) and $F^{\mathrm{n}fs}$ (NFS) $(10,000 \times 10,000$ randomly generated system).

by Table 7.1. Table 7.1 depicts that compared to the LQR design, 34.8734 % of links, (i.e., structured weak links) is removed and the payoff is just 3.0380 % which is reasonable. Although the (NFS) procedure does not take advantage of optimal control techniques nor optimization-based methods such as semi-definite program (SDP) and second-order cone programming (SOCP), its corresponding specifications consisting of density level and performance loss are considerably desirable.

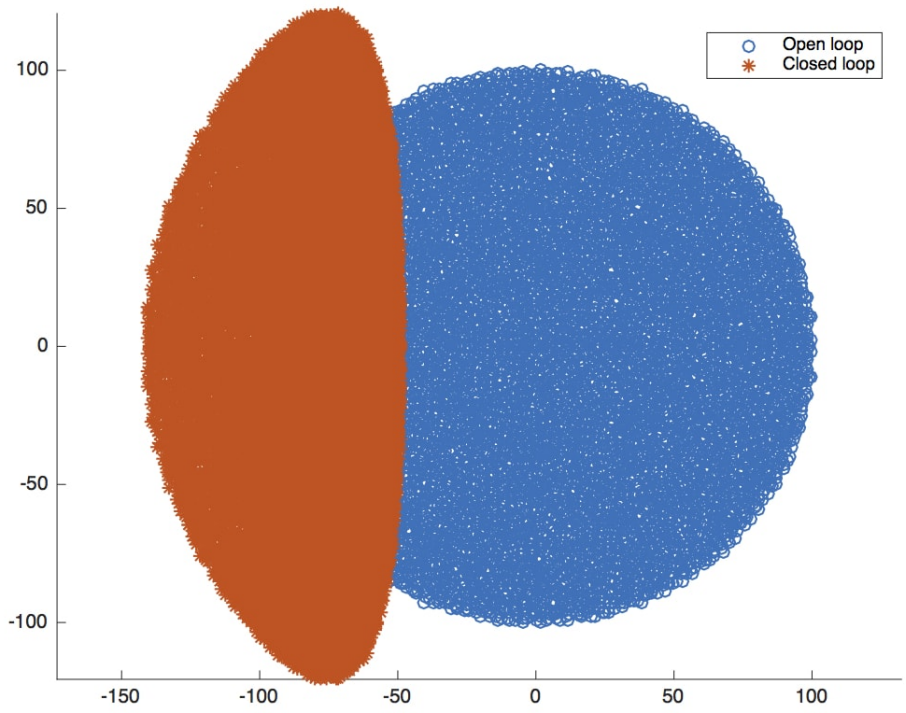


Figure 7.2: Eigenvalues of open loop and closed loop for $F^{\mathrm{n}fs}$ (NFS) ($10,000 \times 10,000$ randomly generated system).

Sub-Exponentially Spatially-Decaying System

The ij^{th} element of sub-exponentially spatially-decaying system X is defined as follows:

$$X_{ij} = \xi \mathcal{X}_{ij} e^{-\alpha |i-j|^{\beta}},$$

wherein $\mathcal{X}_{ij}$ is a normally-distributed random variable with zero mean and unit variance, i.e., it belongs to $\mathcal{N}(0,1)$, α determines the band-width of matrix X, β specifies the rate of spatially-decaying in such a system, and ξ is a positive constant.

Let us consider a $10,000 \times 10,000$ sub-exponentially spatially-decaying system with $\alpha_A = 0.5$, $\beta_A = 0.5$, and $\xi_A = 10$. Assuming the $\alpha_B = 0.25$, $\beta_B = 0.75$, $\xi_B = 10$, $Q = 6I$, $R = 4I$, $D = B$, $\mathcal{S} = \mathcal{I}_{\rho}$, $G_{ij}^{\rho} = 1$ for all links satisfying $(i,j) \notin \mathcal{S}$, and running the (NFS) procedure, for such a large-scale system, F^{nfs} is obtained. Figures

7.3 and 7.4 visualize the eigenvalues of open loop and closed loop for both cases, F (LQR) and F^{nfs} (NFS), respectively. Similar to the randomly generated system case, the open loop is unstable while both closed loops are stable. Costs and density

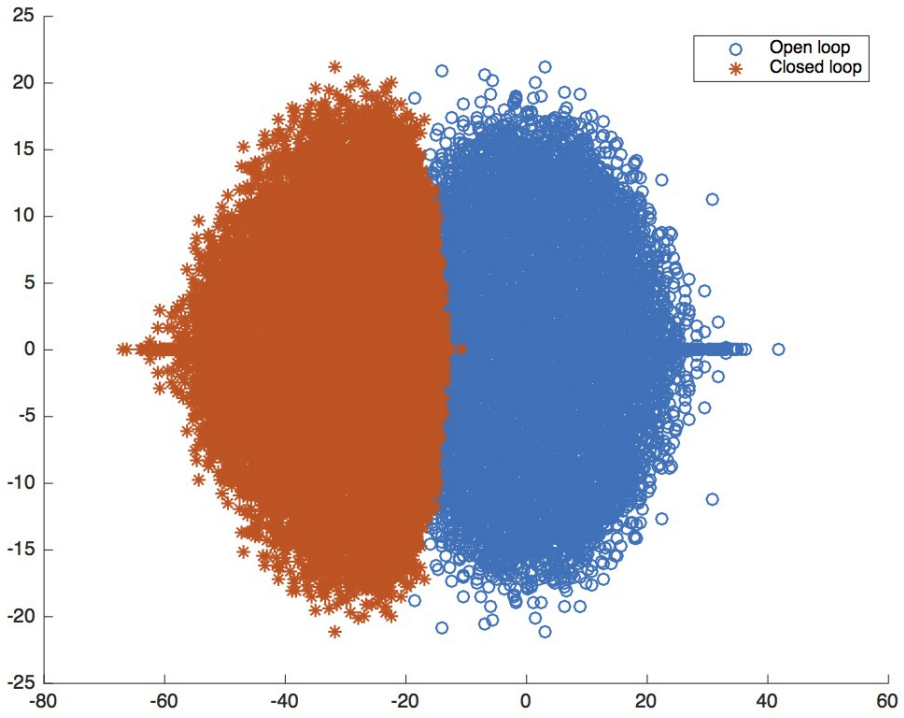


Figure 7.3: Eigenvalues of open loop and closed loop for F (LQR) ($10,000 \times 10,000$ sub-exponentially spatially-decaying system).

levels for both cases, F (LQR) and $F^{\mathrm{n}fs}$ (NFS), are shown in Table 7.2. Table 7.2 showcases that compared to the LQR design, 98.9600 % of links, (i.e., structured weak links) is removed and the payoff is just 0.1333 % which is negligible. Again, (NFS) procedure acts well and its corresponding specifications consisting of density level and performance loss are notably convincing.

The sparsity pattern of $F^{\mathrm{n}fs}$ (NFS) is depicted by Figure 7.5 and Figure 7.6 visualizes the sparsity pattern of first $1,000 \times 1,000$ diagonal sub-block of $F^{\mathrm{n}fs}$ (NFS).

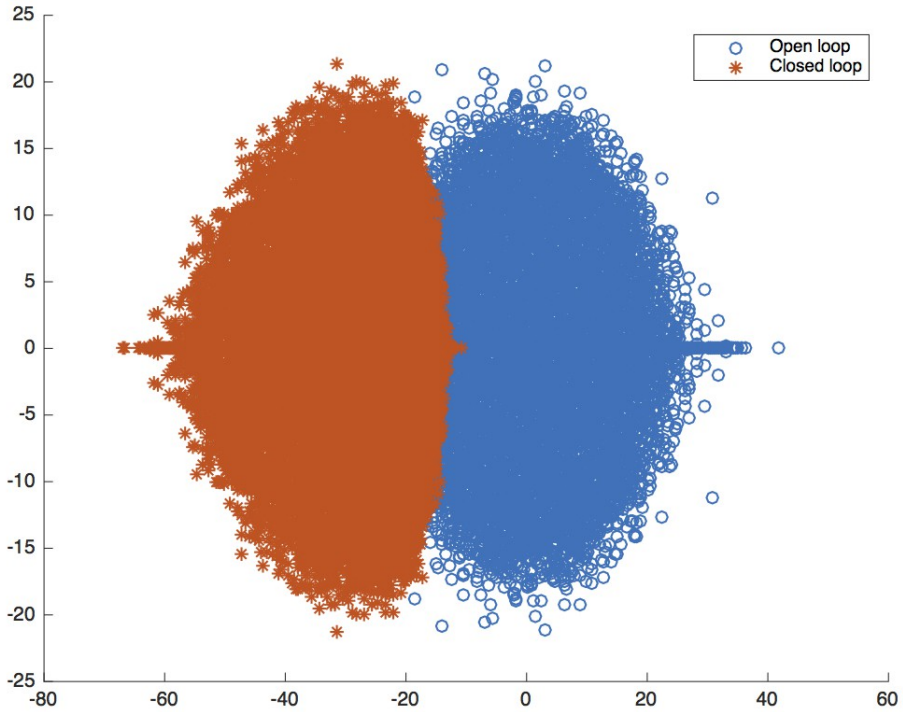


Figure 7.4: Eigenvalues of open loop and closed loop for $F^{\mathrm{n}fs}$ (NFS) ($10,000 \times 10,000$ sub-exponentially spatially-decaying system).

$J(F)$	$J(F^{\mathrm{n}fs})$
$1,241,286.5123$	$1,242,941.7377$
$\|F\|_0$	$\|F^{\mathrm{n}fs}\|_0$
$100,000,000$	$1,040,037$

Table 7.2: Cost and cardinality quantities for F (LQR) and $F^{\mathrm{n}fs}$ (NFS) ($10,000 \times 10,000$ sub-exponentially spatially-decaying system).

7.5.2 Investigation of Relative Performance/Sparsity Specifications for Medium-Size Systems

For a given F, density level of $F^{\mathrm{n}fs}$ is defined by

$$\sigma_D := 100 \times \frac{\|F^{\mathrm{n}fs}\|_0}{\|F\|_0}, \tag{7.12}$$

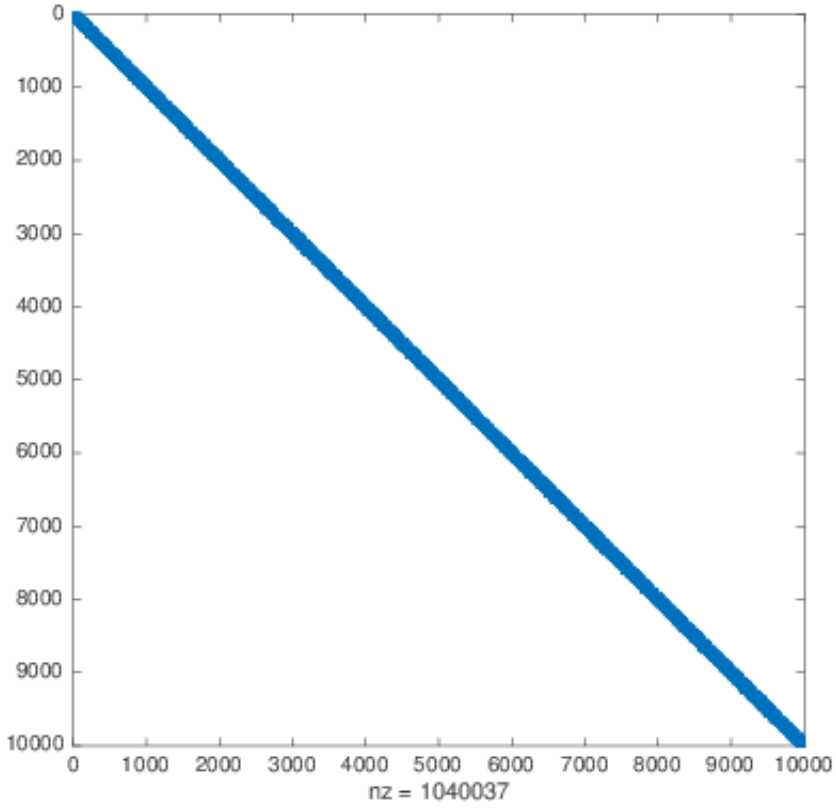


Figure 7.5: Sparsity pattern of $F^{\mathrm{n}fs}$ (NFS); Each blue spot represents a nonzero element of the state feedback controller and number of nonzero elements of the state feedback controller is denoted by nz ($10,000 \times 10,000$ sub-exponentially spatially-decaying system).

sparsity level of F is defined by

$$\sigma_S := 100 - \sigma_D, \tag{7.13}$$

and performance loss is defined by

$$\sigma_P := 100 \times \frac{J(F^{\mathrm{n}fs}) - J(F)}{J(F)}, \tag{7.14}$$

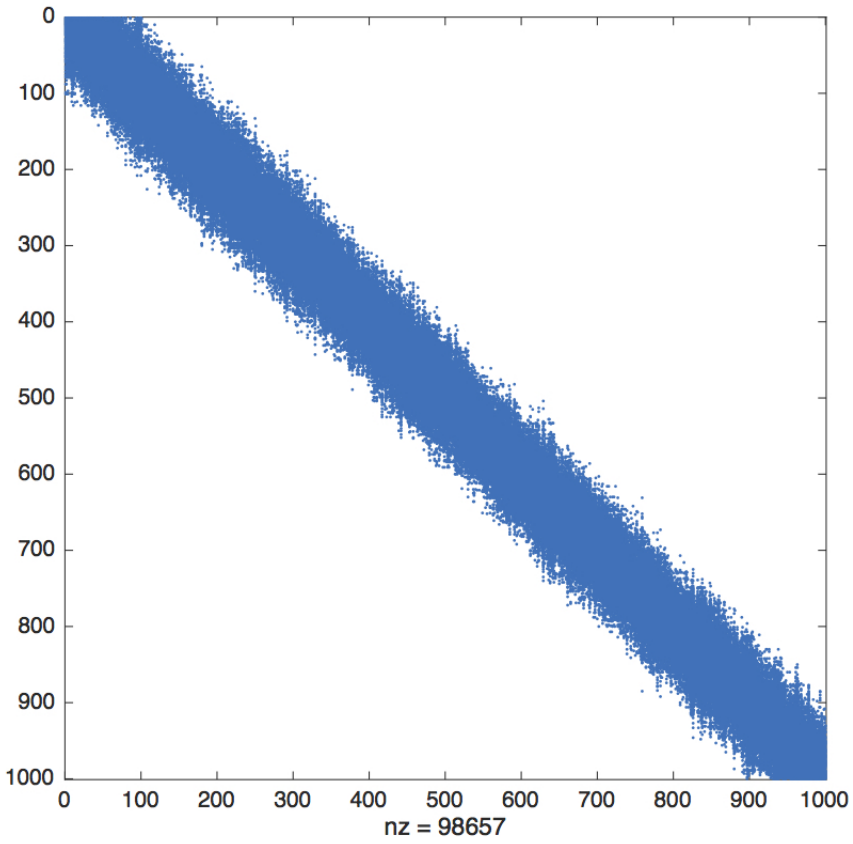


Figure 7.6: Sparsity pattern of first $1,000 \times 1,000$ diagonal sub-block of F^{nfs} (NFS); Each blue spot represents a nonzero element of the state feedback controller and number of nonzero elements of the state feedback controller is denoted by nz ($10,000 \times 10,000$ sub-exponentially spatially-decaying system).

wherein $J(.)$ denotes the squared $\mathcal{H}_2$-norm of the corresponding closed loop system [1].

IEEE 39-Bus New England Power System

Let us consider a 130×130 linear model of IEEE 39-Bus New England power system and a 130×10 input matrix B. Running the (NFS) procedure with $Q = 7I$, $R = 3I$, $D = B$, $\mathcal{S} = \mathcal{I}_\rho$, $G_{ij}^{\rho} = 1$ for all links satisfying $(i, j) \notin \mathcal{S}$, F^{nfs} is obtained. The sparsity pattern of F^{nfs} is depicted in Figure 7.7. Table 7.3 showcases the relative

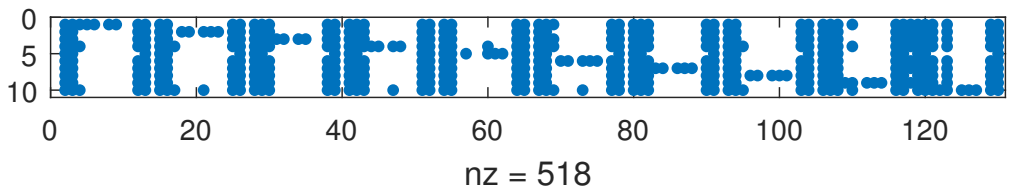


Figure 7.7: Sparsity pattern of $F^{\mathrm{n}fs}$ (NFS); Each blue spot represents a nonzero element of the state feedback controller and number of nonzero elements of the state feedback controller is denoted by nz (130×130 IEEE 39-bus New England power system).

σ_S (%)	σ_P (%)
60.1538	0.007560

Table 7.3: Relative performance/sparsity specifications for closed loops with F (LQR) and $F^{\mathrm{n}fs}$ (NFS) (130×130 IEEE 39-bus New England power system).

performance/sparsity specifications corresponding to the closed loops with F (LQR) and $F^{\mathrm{n}fs}$ (NFS). As data in Table 7.3 illustrates, the relative performance/sparsity characteristics match the desired levels.

Randomly Generated System

Considering the 100×100 randomly generated matrices A and B, $Q = R = I$, $\mathcal{S} = \mathcal{I}_\rho$, $G^\rho_{ij} = 1$ for all links satisfying $(i, j) \notin \mathcal{S}$, and running the (NFS) procedure and algorithm proposed by [1], we obtain $F^{\mathrm{n}fs}$ and sparse LQR F, respectively. For such designs, Figures 7.8 and 7.9 depict the corresponding sparsity patterns.

Table 7.4 showcases the corresponding values of cost and density levels for both cases. According to the data presented in Table 7.4, our proposed state feedback controller outperforms the state feedback controller proposed by [1], because it gives sparser state feedback controller with less quadratic cost. It is worth emphasizing that running the (NFS) procedure for such a medium-size system takes few seconds while optimal sparsification methods proposed by [1, 18, 21, 27, 28] are not able to do so, i.e., our sparsification method is the fastest one among all. Also, we should

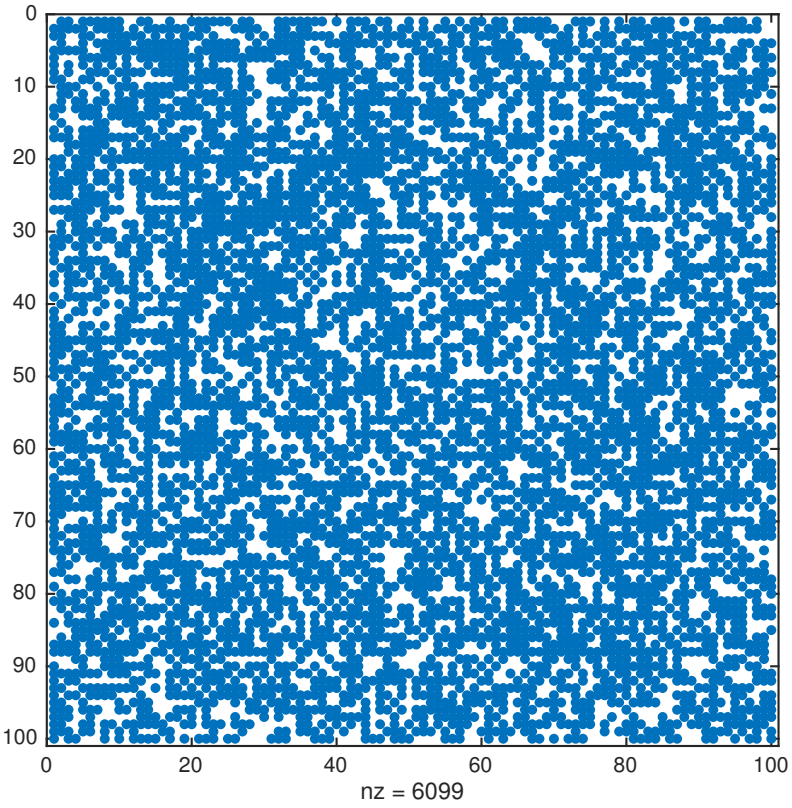


Figure 7.8: Sparsity pattern of $F^{\mathrm{n}fs}$ (NFS); Each blue spot represents a nonzero element of the state feedback controller and number of nonzero elements of the state feedback controller is denoted by nz (100×100 randomly generated system).

mention that all the tests were performed on an iMac with CPU 4 GHz Intel Core i7.

Remark 23. *It should be clarified that it is not claimed that in the case of medium-size systems, (NFS) procedure always outperforms the other sparsification methods in terms of performance-sparsity trade-off, via such a numerical simulation, it is just shown that it may outperform the other sparsification methods in some cases. In other words, such an outperforming highlights that although (NFS) procedure does not embed any performance minimization along with Lyapunov/Sylvester equations, its sparsity-performance trade-off behavior could be even better than other sparsification methods. Also, in general, it is true that (NFS) procedure is the fastest among all at*

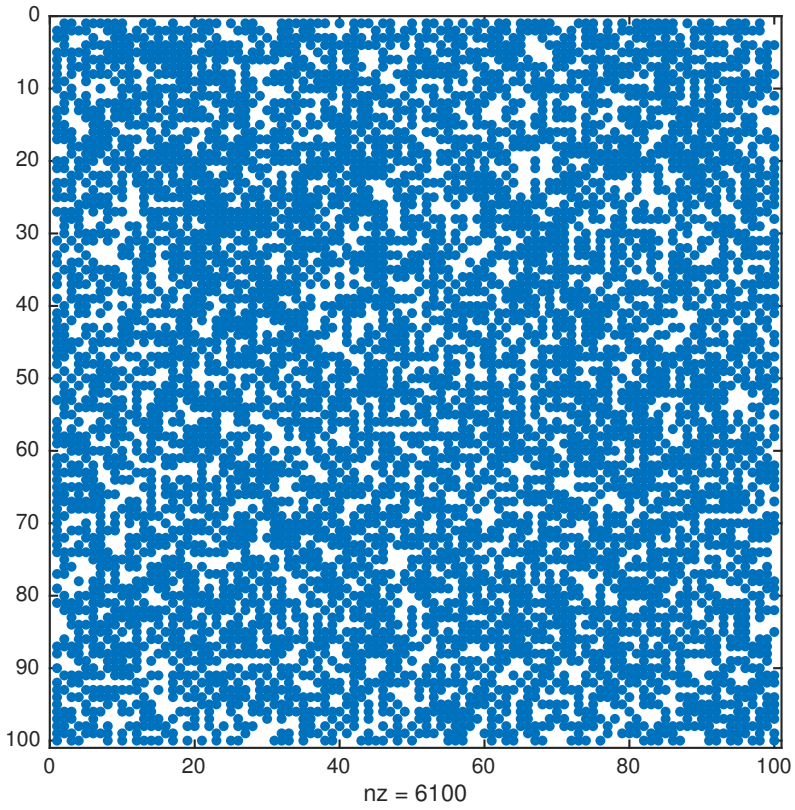


Figure 7.9: Sparsity pattern of F (Sparse LQR); Each blue spot represents a nonzero element of the state feedback controller and number of nonzero elements of the state feedback controller is denoted by nz (100×100 randomly generated system).

the cost of utilizing the (non-fragility)-based stability guarantee rather than utilizing the Lyapunov/Sylvester equations. In other words, the specific advantage of (NFS) procedure is its applicability to the large-scale systems (its high speed) while other sparsification methods are slow Lyapunov-based methods which generally attain better sparsity-performance trade-off in the case of medium-size systems.

Remark 24. *In [1] a structured $\mathcal{H}_2$ design method is proposed by which, the $\mathcal{H}_2$ quadratic cost of a proposed sparse design can be improved. Utilizing such a structured method can enable us to improve the $\mathcal{H}_2$ quadratic cost in the case that we utilize the*

$J(F)$	$J(F^{\mathrm{n}fs})$	$\|F\|_0$	$\|F^{\mathrm{n}fs}\|_0$
1162.5	1154.6	6100	6099

Table 7.4: Cost and cardinality quantities for F (Sparse LQR) and $F^{\mathrm{n}fs}$ (NFS) (100×100 randomly generated system).

$J(.)$ *as a performance measure.*

7.5.3 The Trade-Off Between Upper Bound on Non-Fragility and Sparsity Level

Let us consider a set of sparse state feedback controllers designed by [1] and then compute the sparsity level and the upper bound on non-fragility $\eta\gamma\hat{\rho}$. The procedure is simply as follows:

For a 100×100 randomly generated system, after collecting the set of 50 sparse LQRs F designed by [1] for $B = randn(100)$, $Q = I$, $R = I$, $D = B$, we run the (NFS) procedure with $Q = I$, $R = I$, $D = B$, $\mathcal{S} = \mathcal{I}_\rho$, $G_{ij}^{\rho} = 1$ for all links satisfying $(i, j) \notin \mathcal{S}$. Figure 7.10 depicts the trade-off between upper bound on non-fragility $\eta\gamma\hat{\rho}$ and sparsity level of those 50 sparse LQRs and their corresponding (NFS) sparsified state feedback controllers.

As Figure 7.10 illustrates, the sparser design considered for both sparse LQR F and F^{nfs} (NFS), the smaller upper bound on non-fragility is obtained. In other words, sparser state feedback controllers will be more fragile in terms of notion of the non-fragility. In Figure 7.10, such a trade-off is visualized via blue points $\circ$ for set of 50 sparse LQRs F and via red points $\circ$ for set of 50 sparsified state feedback controllers F^{nfs}, respectively.

In Figure 7.10, another similar observation is that for any of 50 sparse LQRs F, the non-fragility of sparsified state feedback controller $F^{\mathrm{n}fs}$ is less than non-fragility of the corresponding sparse LQR F. Thus, similar to the previously mentioned trade-off,

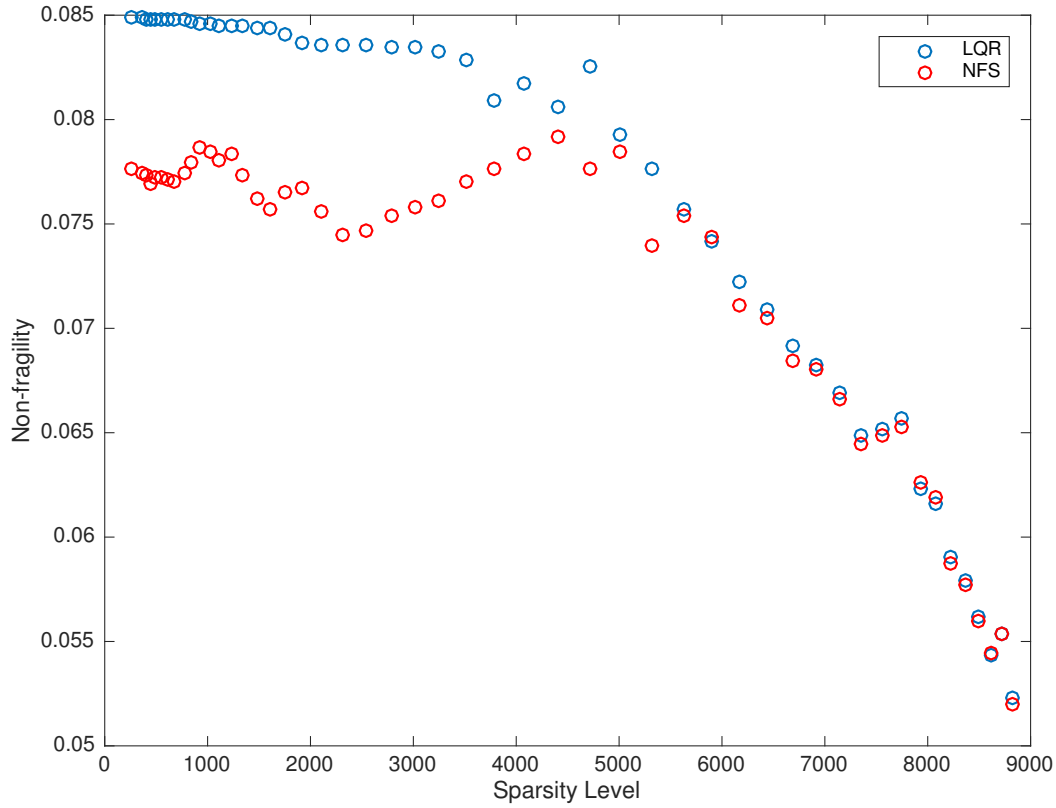


Figure 7.10: The trade-off between upper bound on non-fragility and sparsity level for Sparse LQRs F designed by [1] (Blue) and $F^{\mathrm{n}fs}$ (NFS) (Red) (100×100 randomly generated system).

a trade-off exists between upper bound on non-fragility and sparsity level.

7.5.4 Two Greedy Algorithms to Obtain a Set of Sparse State Feedback Controllers

This subsection utilizes two greedy algorithms to determine the greedy sparsity structures $\mathcal{S}_k$'s where $\mathcal{S}_0 := \mathcal{S}$ and $\mathcal{S}_k \subset \mathcal{S}$ for all $k \in \{1, \ldots, \|F\|_0 - \|F^{nfs}\|_0\}$ and obtain the set of sparse state feedback controllers F^k between $F^0 := F$ and $F^{\|F\|_0 - \|F^{nfs}\|_0} = F^{nfs}$ in terms of the ℓ_0 measure. As it was previously mentioned, because of scalability complexities, such algorithms are applicable to small-size systems.

Brute Force Greedy Algorithm

To reach the previously mentioned goal, starting from $F^0 = F$, at k^{th} step, we compute $J(.)$ for all

$$F^{k-1} - F_{ij}^{k-1} e_i e_j^T,$$

where $(i,j) \in \mathcal{S}_{k-1}$ and then by removing the link corresponding to the minimum achievable cost (i_k^*, j_k^*), we obtain F^k as

$$F^k = F^{k-1} - F_{i_k^* j_k^*}^{k-1} e_{i_k^*} e_{j_k^*}^T.$$

Such an iterative approach can be summarized as the following greedy algorithm:

Brute Force Greedy Algorithm

---

1. *Begin*
2. *Set* $F^0 = F$, $\mathcal{S}_0 := \mathcal{S} = \mathcal{I}_\rho$, *and* $k = 1$.
3. *Find* $(i_k^*, j_k^*) = \underset{(i,j)\in\mathcal{S}_{k-1}}{\arg\min} J\big(F^{k-1} - F_{ij}^{k-1} e_i e_j^T\big)$.
4. *Update* $F^k = F^{k-1} - F_{i_k^* j_k^*}^{k-1} e_{i_k^*} e_{j_k^*}^T$.
5. *Update* $\mathcal{S}_k = \mathcal{S}_{k-1} \backslash \{(i_k^*, j_k^*)\}$.
6. *If* $k = \|F\|_0 - \|F^{nfs}\|_0$,
   then go to step 8.
7. *Set* $k \to k+1$ *and go to step 3.*
8. *End.*

Remark 25. *Considering the* (7.10) *and* (7.11) *and defining the* $\mathcal{U}_k$ *as*

$$\mathcal{U}_k := \cup_{l=1}^{k} \{(i_l^*, j_l^*)\}, \tag{7.15}$$

it is followed that by setting $\mathcal{S} = \mathcal{U}_k$ *in* (7.10) *and* (7.11), $G_{ij}^{\rho} = 1$ *for all links satisfying* $(i, j) \notin \mathcal{S}$, *and applying such formulas to* $F^0 = F$, F^k *is resulted.*

Let us consider the 15×15 randomly generated matrices A and B. Suppose that $Q = I$ and $R = I$. Applying the greedy algorithm, we visualize σ_P (%) versus σ_D (%) in Figure 7.11.

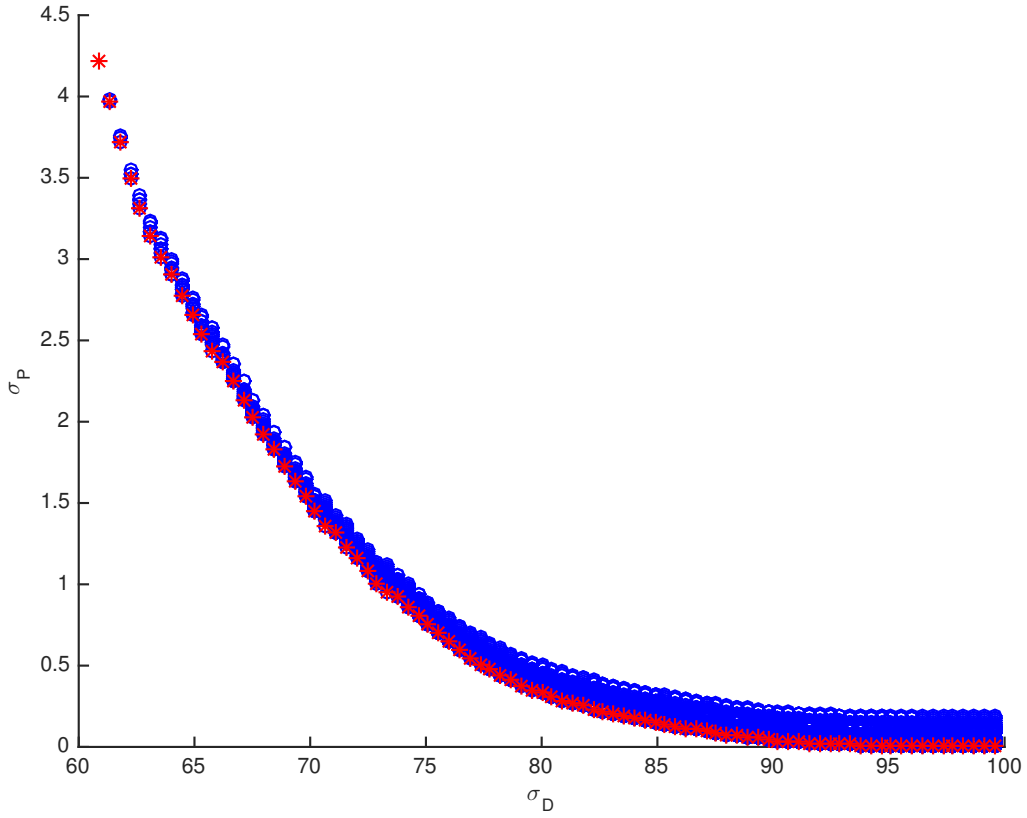


Figure 7.11: The visualization of σ_P (%) versus σ_D (%) obtained from brute force greedy algorithm (15×15 randomly generated system).

In such an illustration, blue $\circ$'s and red $*$'s represent $(i, j) \in \mathcal{S}_{k-1}$'s and (i_k^*, j_k^*)'s,

respectively. As it is observed, points shown by red $*$ specify a lower bound for performance-sparsity trade-off curves. In other words, such a greedy curve provides an improvement regrading the performance loss for our sparse state feedback controllers. However, it is clear that it does not suggest the best optimal performance-sparsity trade-off curve. In terms of scalability, the greedy algorithm is appropriate for medium-size systems; unfortunately, it is not applicable to large-scale systems.

Gradient-Based Greedy Algorithm

Here, instead of computing the measure $J(.)$, over all possible choices at each step, we compute the gradient of J for F^{k-1}. Then, we remove the link which attains the minimum absolute value of gradient term of J for F^{k-1}. Specifically, for the case of the measure $J(.)$, this method reduces the time complexity of brute force method by $\|F\|_0$ (mn in the case of fully dense F). Because, we no longer need to solve Lyapunov equations for all possible choices at each step and in return, we solve just 2 Lyapunov equations at each step. The only pay-off is the increase in performance loss. This method enables us to obtain a set of sparse state feedback controllers for larger systems.

Now, the gradient-based greedy algorithm is explained by reconsidering the measure J. According to [1, 127], the gradient of the measure J is computed via:

$$\nabla J\big(F^{k-1} - F^{k-1}_{ij} e_i e_j^T\big) = 2(-RF - B^T P)L,$$

where observability Gramian P and controllability Gramian L represent the unique positive definite solutions of the following Lyapunov equations:

$$(A + BF)^T P + P(A + BF) = -(Q + F^T RF),$$
$$(A + BF)L + L(A + BF)^T = -DD^T.$$

Starting from $F^0 = F$, at k^{th} step, we compute $J(.)$ for all

$$F^{k-1} - F_{ij}^{k-1} e_i e_j^T,$$

where $(i, j) \in \mathcal{S}_{k-1}$ and then by removing the link corresponding to the minimum absolute value of the gradient term (i_k^*, j_k^*), we obtain F^k as

$$F^k = F^{k-1} - F_{i_k^* j_k^*}^{k-1} e_{i_k^*} e_{j_k^*}^T.$$

Such an iterative approach can be summarized as the following greedy algorithm:

Gradient-Based Greedy Algorithm

---

1. *Begin*
2. *Set* $F^0 = F$, $\mathcal{S}_0 := \mathcal{S} = \mathcal{I}_\rho$, *and* $k = 1$.
3. *Find* $(i_k^*, j_k^*) = \underset{(i,j)\in\mathcal{S}_{k-1}}{\arg\min} |\nabla J\big(F^{k-1} - F_{ij}^{k-1} e_i e_j^T\big) F_{ij}^{k-1}|$.
4. *Update* $F^k = F^{k-1} - F_{i_k^* j_k^*}^{k-1} e_{i_k^*} e_{j_k^*}^T$.
5. *Update* $\mathcal{S}_k = \mathcal{S}_{k-1} \backslash \{(i_k^*, j_k^*)\}$.
6. *If* $k = \|F\|_0 - \|F^{nfs}\|_0$,
   then go to step 8.
7. *Set* $k \to k + 1$ *and go to step 3.*
8. *End.*

Considering the 20×20 randomly generated matrices A and B, assuming that $Q = I$

and $R = I$, and applying the greedy algorithm, we visualize σ_P (%) versus σ_D (%) in Figure 7.12.

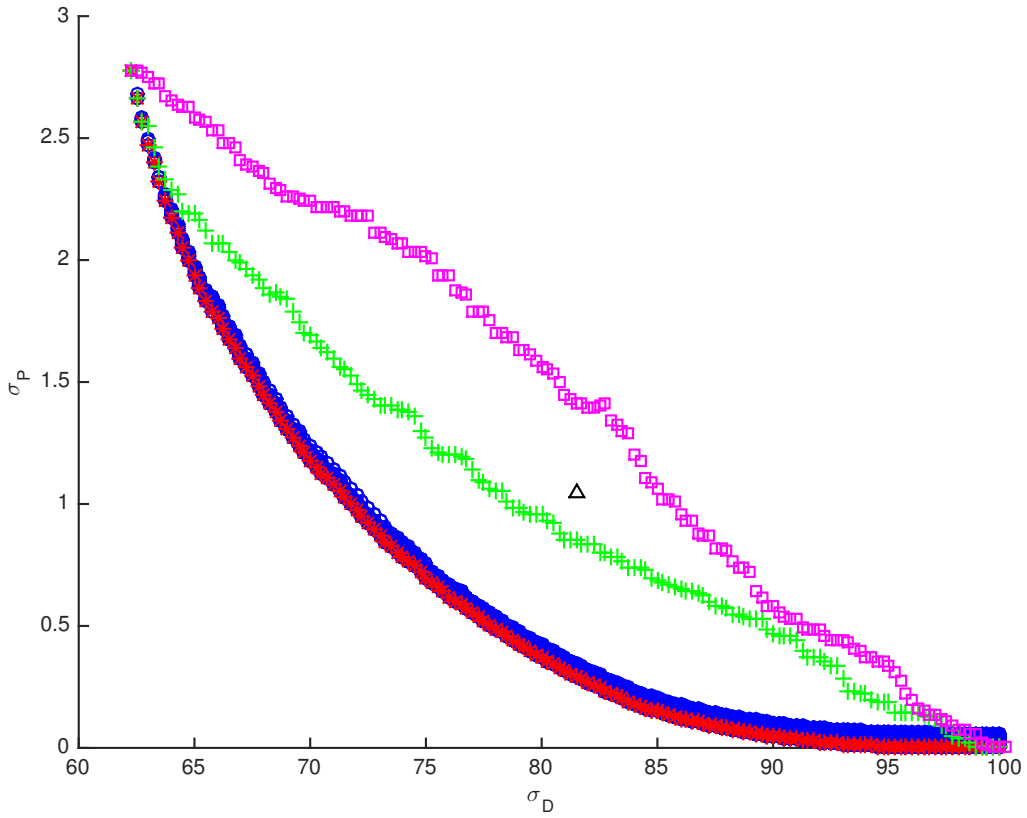


Figure 7.12: The visualization of σ_P (%) versus σ_D (%) obtained from brute force and gradient-based greedy algorithms (20×20 randomly generated system).

In such an illustration, blue $\circ$'s and red $*$'s represent $(i, j) \in \mathcal{S}_{k-1}$'s and (i_k^*, j_k^*)'s achieved by brute force method, respectively. The green $+$'s and magenta $\square$'s demonstrate the (i_k^*, j_k^*)'s achieved by gradient-based greedy method and randomly sparsified (i_k, j_k)'s, respectively. The black $\triangle$ depicts the state feedback controller achieved by randomly chosen sparsity structure. As it is observed, points shown by green $+$ specify sub-optimal performance-sparsity trade-off curve. In other words, such a greedy curve provides a cheap computational method while attaining a relatively poor performance loss for our sparse state feedback controllers. As depicted, the state feedback

controller obtained from randomly chosen sparsity structure and randomly sparsified state feedback controllers are outperformed by gradient-based greedy method.

7.6 Conclusion

A notion of non-fragility is introduced and some lower and upper bounds are derived for such a notion of non-fragility. On the basis of such a notion of non-fragility, a sparsification procedure is presented. Sparsity/performance features of such sparsified state feedback controllers are evaluated via large-scale and medium-size systems. Considering the set of sparse state feedback controllers for medium-size systems, a trade-off between our achieved upper bound on non-fragility and sparsity level of those state feedback controllers is observed for both sparse state feedback controllers and their sparsified ones. Also, there exists a trade-off between upper bound on non-fragility and sparsity level when we compare any of sparse LQRs with its corresponding sparsified state feedback controller. In addition, two greedy algorithms are proposed to obtain a set of sparse state feedback controllers, the brute force one and the gradient-based one which is computationally cheaper while attaining a reasonably higher sub-optimality level. A remaining problem could be the improvement of limited sparsification rate of the (non-fragility)-based sparsification method.

Chapter 8

Improving Sparsity in Time and Space via Self-Triggered Sparse Optimal Controllers

8.1 Introduction

The area of distributed control systems has rapidly been growing in the past decade, and it has been applied to various real-world problems including formation control of autonomous vehicles, power systems wide area control, wireless networks stochastic control, and so on. In several important applications, the centralized control methodologies fail, since it requires a dense communication graph which is not practically achievable in most cases. Also, the less communication we have, the less we are faced with privacy issues; therefore network information is kept more secure. Thus, minimizing the number of communication links with pre-specified guarantee on performance loss becomes crucial which is the main goal of the *Spatial Sparse Optimal Control* [1].

Sampled-data control systems have thoroughly been investigated in previous decades

[81, 83]. In such control systems, the system to be controlled is considered in continuous time; however, the controller is synthesized in a discrete manner [83]. One of the significant issues in such an area is to decrease the number of samplings. In other words, the longer maximum allowable time interval is desired in terms of sampling cost. Achieving such an objective, leads to smarter CPU task scheduling in embedded systems and more battery life in networked control systems. Both previously mentioned applications are categorized as *Temporal Sparse Optimal Control* [56].

Recently, the spatial sparse optimal control has attained much attention in control theory and applications [1, 13, 21, 26, 27, 38]. The main goal in such an area is to decrease the number of communication links among nodes while preserving the guaranteed performance level. Such a balance between performance loss and density level of feedback controller is customarily made via some ℓ_1-regularized term which is augmented to the performance loss term appeared in objective function of the optimization problem [1, 21, 26, 27]. Also, in the case of sparsity promotion in space of control actions, in [128], it is proved that ℓ_1 relaxation provides an effective tool to obtain sparsity in the space of control actions, i.e., having the minimum support length over time horizon. Moreover, they propose a self-triggered maximum hands-off control design which is numerically evaluated for specific single-input systems.

One of the main switching control methods to achieve temporal sparse optimal control is called *Self-triggered Control* [43]. The fundamental advantage of self-triggered control is having no control update when there is no need to take new updates (samplings) [52, 56, 58, 59, 85]. In fact, a performance-preserving condition or Lyapunov-based stability condition (known as *self-triggering condition*) is checked to determine whether new sampling is necessary or not. Also, the next update time is solely computed based on current state information [56]. However, in event-triggered control (another well-established aperiodic control method), triggering condition is monitored continuously and whenever it is satisfied, the event is triggered. Thus, it

requires a dedicated hardware which is not available in most cases [52]. In [64], a roll-out event-triggered control approach is proposed which outperforms the traditional periodic LQR design while having the same sampling rate. In such an approach, inter-execution times are computed via event-triggering condition and control action u_k is set to either 0 or $F_k x_k$ where F_k's are dense time-varying gains which is obtained from corresponding Riccati equations and x_k's denote the triggering states. Moreover, all such calculations are done after transforming the continuous time setup to discrete one. In [62], assuming the finite sequence of interval lengths and their corresponding spatially-varying stabilizing controllers, a self-triggered method is proposed to find the switching rule on the basis of current state information. In [85], the proposed self-triggering method, ensures $\mathcal{L}_2$ stability of the closed loop system and the average time period has an increasing behavior versus $\mathcal{L}_2$ gain. In [62], some similar but not exact relationship between $\mathcal{H}_{2,\infty}$ performance indices and the average time period is expressed.

Merging advantages of spatial sparse optimal control and temporal sparse optimal control, the *self-triggered sparse optimal control (SSOC)* problem is proposed. This chapter is improving the spatial sparsity of the feedback gains/actuators utilized by self-triggered control design while achieving a pre-specified bound on performance loss and stability guarantee for any arbitrarily-chosen initial condition. At each time interval, our design procedure breaks into two main parts: (i) Via solving a nonlinear optimization, the maximum allowable inter-execution time is computed by taking advantage of commonly-used tools such as discretization rule. (ii) For a fixed value of inter-execution time, we design optimal sparse feedback gains via minimizing the ℓ_1 sparsity-promoting terms corresponding to both feedback gain and actuation over such a time interval subject to performance constraint. Investigating the feasibility of the corresponding optimization problems, the closed-loop stability is immediate. The effectiveness of the SSOC design is assessed by utilizing the spatially distributed

systems. The extensive numerical simulations show that, compared to the periodic time-triggered LQR design, the average density level of feedback gains is appropriately improved, the number of utilized actuators is meaningfully reduced, and comparatively less sensing is required. Spatial sparsity and temporal sparsity are effectively improved by our proposed SSOC design while preserving the guaranteed performance loss and stability for any arbitrarily-chosen initial condition. Also, a tradeoff between pre-specified performance bound and sparsity in time/space is observed. In other words, the tradeoff between performance bound and average inter-execution time means that the higher allowed performance loss results in less frequently sampling. Furthermore, the effect of spatially decaying rate on sparsification process is visualized. Additionally, the effect of penalizing parameters on sparsification process is depicted.

The structure of the chapter is described as follows: Section 8.2 introduces the mathematical notations that are used. Section 8.3 is devoted to the statement of the problem to be solved. In Section 8.4, an equivalent reformulation of the self-triggered sparse optimal control problem is presented. Section 8.5 guarantees the feasibility of our formulated problem. Section 8.6 includes the stability proof, assuming the feasibility of sequence of corresponding optimization problems. Section 8.7 details the self-triggered performance-based method. Section 8.8 presents the corresponding algorithm which consists of constrained convex programming and nonlinear equation solving. Considering a class of spatially-distributed systems and utilizing the algorithm proposed by Section 8.8, Section 8.9 assesses the effectiveness of our proposed method. Finally, Section 8.10 concludes the chapter with drawing some future insights.

8.2 Mathematical Notations

The set of real numbers, positive real numbers, positive integer numbers, and non-negative integer numbers are denoted by $\mathbb{R}$, $\mathbb{R}_{++}$, $\mathbb{N}$, and $\mathbb{Z}_+$, respectively. The set of real-valued $n \times 1$ vectors and set of real-valued $m \times n$ matrices are represented by $\mathbb{R}^n$ and $\mathbb{R}^{m\times n}$, respectively. The supremum of a subset of real numbers is denoted by sup. The derivative of time-dependent function f with respect to time t is shown by f' and partial derivative of multivariate function g with respect to x is represented by $\partial g/\partial x$. The positive semi-definiteness and positive definiteness are shown by $\succeq 0$ and $\succ 0$, respectively. The identity matrix is represented by I as usual. The Euclidean norm of vector v is denoted by $\|v\|_2$. Cardinality (ℓ_0 sparsity measure), ℓ_1 norm, and largest singular value of matrix M are represented by $\|M\|_0$, $\|M\|_1$, and $\|M\|$, respectively where $\|M\|_0$ is identical to number of nonzero elements of M and $\|M\|_1$ refers to sum of absolute values of elements of M. Matrix M is called to be Hurwitz if any of its eigenvalues has a negative real part. Vectorization and determinant of matrix M are denoted by $\mathbf{vec}(M)$ and $\det(M)$, respectively. The Kronecker matrix product is denoted by $\otimes$. The mathematical limit operator is represented by lim.

8.3 Problem Formulation

Let us consider the class of linear time invariant (LTI) systems described by

$$\dot{x}(t) = Ax(t) + Bu(t),\ x(t_0) = x_0, \tag{8.1}$$

where $x \in \mathbb{R}^n$ denotes the state vector, $u \in \mathbb{R}^m$ represents the control input (control action), $t_0 = 0$, and x_0 is an arbitrarily-chosen initial condition. The control objective for system (8.1) is to stabilize the system using the following class of sample-and-hold

control laws:

$$u(t) = \psi\big(x(t)\big) = u_k = F_k x(t_k) = F_k x_k, \tag{8.2}$$

for all $t \in [t_k, t_{k+1})$ and $k \in \mathbb{Z}_+$. The time instants t_k's are called *triggering times* and their sequence is denoted by $\{t_k\}_{k=0}^{\infty}$. The control law (8.2) is defined based on a time-varying feedback gain F_k where its sequence is shown by $\{F_k\}_{k=0}^{\infty}$. Also, let us define the k^{th} *inter-execution time* as difference of k^{th} and $(k+1)^{th}$ triggering times, i.e., as follows:

$$\delta_k := t_{k+1} - t_k.$$

Considering system (8.1) with control law (8.2), the cost functional corresponding to $[t_k, t_k + \xi)$ is denoted by $J_k(F_k, \xi; x_k)$ and defined by

$$J_k(F_k, \xi; x_k) := \int_{t_k}^{t_k+\xi} \big(x(t)^T Q x(t) + u(t)^T R u(t)\big) dt, \tag{8.3}$$

for $\xi \in [0, \delta_k)$ where $Q \succ 0$ and $R \succ 0$ are corresponding state-weight and input-weight matrices, respectively.

Assumption 4. *The pair (A, B) is controllable.*

Assumption 4 is standard in the literature and along with observability of pair (Q, A) which is implied by positive definiteness of Q, it ensures the uniqueness of the LQR solution and boundedness of the total cost value.

The method proposed by [56] solely takes advantage of the Lyapunov function evolutions. However, our proposed problem setup is mostly similar to the method proposed by [59] in which $\forall \xi \in [0, t_{k+1} - t_k)$, a performance-based condition is considered in addition to utilization of the Lyapunov function. In our problem setup, the Lyapunov function V is considered in the form of $V(x) = x^T \tilde{P} x$ where $\tilde{P}$ denotes

the unique positive definite solution of the following Lyapunov equation:

$$(A+B\tilde{F})^T\tilde{P}+\tilde{P}(A+B\tilde{F})+Q+\tilde{F}^TR\tilde{F}=0, \tag{8.4}$$

where the feedback gain $\tilde{F}$ is a predesigned well-performing feedback gain which stabilizes system (8.1) via control law $u(t)=\tilde{F}x(t)$, i.e., $A+B\tilde{F}$ is Hurwitz. It is noteworthy that positive definiteness of $\tilde{P}$ is resulted from positive definiteness of $Q+\tilde{F}^TR\tilde{F}$, the following equation:

$$\tilde{P}=\int_0^\infty e^{(A+B\tilde{F})^Tt}(Q+\tilde{F}^TR\tilde{F})e^{(A+B\tilde{F})}dt,$$

and the fact that $A+B\tilde{F}$ is Hurwitz. Also, the stability of $A+B\tilde{F}$ implies the uniqueness of $\tilde{P}$. We utilize a parameter $\alpha>1$ which specifies the pre-given upper bound on performance loss. A possible choice for selection of $\tilde{F}$ can be standard LQR design F_{LQR}. The cost value $\tilde{J}=J_{LQR}$ for such a controller is computed via solving an optimal control problem (via solving a Riccati Equation). Thus, it is rather reasonable to make such a choice.

The control objectives of our chapter can be listed as follows:

(i) reduced sensing requirements,

(ii) minimum inter-subsystems communication requirements,

(iii) minimum number of utilized actuators,

(iv) guaranteed closed-loop performance losses.

In order to achieve the first objective, we minimize $-\delta_k$ because maximizing δ_k leads to having a decreased number of required sensing instants. The second and third objectives can be achieved via minimization of sparsity-promoting terms $\gamma\|F_k\|_0$ (feedback gain) and $\eta\|u_k\|_0$ (control action), respectively and the fourth objective can be realized via enforcing a pre-specified upper bound on performance loss at each time interval. Such a four control objectives can be realized by computing time-varying

feedback gains $\{F_k\}_{k=0}^{\infty}$ utilizing the following self-triggered sparse optimal control (SSOC) problem:

$$\underset{F_k,\delta_k}{\text{minimize}} \; -\delta_k + \gamma\|F_k\|_0 + \eta\|u_k\|_0 \tag{\textbf{P1}}$$

subject to: (8.1) and (8.2),

$$\forall \xi \in [0, \delta_k) : J_k(F_k, \xi; x_k) \leq \alpha\Big(V\big(x(t_k)\big) - V\big(x(t_k+\xi)\big)\Big), \tag{8.5}$$

It is noteworthy that $J_k(F_k, \delta_k; x_k)$ denotes the k^{th} time interval cost, i.e., the cost corresponding to $[t_k, t_k + \delta_k)$. Parameters γ and η adjust the balance between communication/actuation spatial density levels and temporal density level.

It is emphasized that in general, the simultaneous satisfaction of all four objectives by imposing (**P1**) is a hard task because each objective has its own computational complexities. Thus, they need to be relaxed and subsequently, the control objectives are sub-optimally achieved. However, throughout the chapter, we see how (**P1**) enables us to reflect the fundamental facts regarding the self-triggered sparse optimal control (SSOC) design.

In the rest of the chapter, we aim at solving (**P1**) to sub-optimally achieve the control objectives.

8.4 Equivalent Reformulation

Throughout this section, we show how (**P1**) can be reformulated as a regularized quadratically-constrained quadratic program (QCQP) when δ_k is kept fixed. In the following, we explain all necessary steps to be taken to achieve such a reformulation.

The feedback control law (8.2) can be decomposed as

$$u(t) = F_k N_k x_0,$$

for all $t \in [t_k, t_{k+1})$ where $N_0 = I$,

$$N_k = M_{k-1}(\delta_{k-1})N_{k-1},\ M_j(\xi) = e^{A\xi}\big(I + Z(\xi)BF_j\big),$$
$$Z(\xi) = \int_0^{\xi} e^{-A\tau}d\tau,$$

for all $k \in \mathbb{N}$ and $j \in \mathbb{Z}_+$. Because, solving system (8.1) and (8.2) for time interval $[t_k, t_{k+1})$, we get

$$\begin{aligned} x(t) &= e^{A(t-t_k)}x(t_k) + \int_{t_k}^{t} e^{A(t-\phi)}BF_k x(t_k)d\phi, \\ &= e^{A(t-t_k)}(I + \int_0^{t-t_k} e^{A(-\tau)}BF_k d\tau)x(t_k), \\ &= e^{A(t-t_k)}\big(I + Z(t-t_k)BF_k\big)x(t_k), \\ &= M_k(t-t_k)x(t_k). \end{aligned}$$

Thus, assuming the continuity of $x(t)$ at $t = t_{k+1}$, we have $x(t_{k+1}) = e^{A\delta_k}\big(I + Z(\delta_k)BF_k\big)x(t_k) = M_k(\delta_k)x(t_k)$. Utilizing the principle of mathematical induction, it turns out that $x(t_k) = N_k x_0$.

Next lemma expresses how we can explicitly calculate the k^{th} time interval cost, i.e., $J_k(F_k, \delta_k; x_k)$.

Lemma 36. *The k^{th} time interval cost $J_k(F_k, \delta_k; x_k)$ can be expressed as*

$$J_k(F_k, \delta_k; x_k) = x_k^T Y_k(F_k, \delta_k)x_k,$$

where

$$Y_k(F_k, \delta_k) = H_0(\delta_k) + F_k^T H_1(\delta_k)^T + H_1(\delta_k)F_k + F_k^T H_2(\delta_k)F_k,$$

for all $k \in \mathbb{Z}_+$ *and*

$$H_0(\xi) = \int_0^{\xi} e^{A^T\tau} Q e^{A\tau} d\tau,$$
$$H_1(\xi) = \int_0^{\xi} e^{A^T\tau} Q e^{A\tau} Z(\tau) B d\tau,$$
$$H_2(\xi) = \int_0^{\xi} \big(e^{A\tau} Z(\tau) B\big)^T Q \big(e^{A\tau} Z(\tau) B\big) d\tau + \xi R.$$

Proof. The k^{th} time interval cost $J_k(F_k, \delta_k; x_k)$ can be written as a sum of two terms as follows:

$$J_k(F_k, \delta_k; x_k) = J_k^x(F_k, \delta_k; x_k) + J_k^u(F_k, \delta_k; x_k),$$

wherein

$$\begin{aligned} J_k^x(F_k, \delta_k; x_k) &= \int_{t_k}^{t_{k+1}} x_k^T M_k(t - t_k)^T Q M_k(t - t_k) x_k dt \\ &= x_k^T \int_0^{\delta_k} M_k(\tau)^T Q M_k(\tau) d\tau x_k, \end{aligned}$$

and

$$J_k^u(F_k, \delta_k; x_k) = \int_{t_k}^{t_{k+1}} x_k^T F_k^T R F_k x_k dt = x_k^T (\delta_k F_k^T R F_k) x_k.$$

Then, we get $J_k(F_k, \delta_k; x_k) = x_k^T Y_k x_k$ wherein

$$Y_k(F_k, \delta_k) = \int_0^{\delta_k} M_k(\tau)^T Q M_k(\tau) d\tau + \delta_k F_k^T R F_k. \tag{8.6}$$

Substituting the $M_k(\tau) = e^{A\tau}\big(I + Z(\tau) B F_k\big)$ in the right side of (8.6), $Y_k(F_k, \delta_k)$ is expressed in terms of $H_0(\delta_k)$, $H_1(\delta_k)$, and $H_2(\delta_k)$. □

The following proposition states an important property about matrix $H_2(\zeta)$ which will lead to the convex reformulation of (**P1**) (in terms of variable F_k) when δ_k is

kept fixed.

Proposition 37. *For all $\xi > 0$, the matrix $H_2(\xi)$ is positive definite, i.e., $H_2(\xi) \succ 0$.*

Proof. To prove the positive definiteness of matrix $H_2(\xi)$ for all positive values of ξ, we consider an arbitrary nonzero vector $v \in \mathbb{R}^m$ $(v \neq 0)$. Then, we showcase that $v^T H_2(\xi) v$ is positive. Let us calculate $v^T H_2(\xi) v$ as follows:

$$\begin{aligned} v^T H_2(\xi) v &= v^T \Big(\int_0^\xi \big(e^{A\tau} Z(\tau) B\big)^T Q \big(e^{A\tau} Z(\tau) B\big) d\tau + \xi R \Big) v \\ &= \int_0^\xi \big(e^{A\tau} Z(\tau) B v\big)^T Q \big(e^{A\tau} Z(\tau) B v\big) d\tau + \xi v^T R v. \end{aligned}$$

Since $Q \succ 0$ and $R \succ 0$ hold, the term appeared inside the last integral is non-negative and $\xi v^T R v$ is positive, respectively. It is known the definite integral of a non-negative function over a finite interval results in a non-negative value. Thus, $v^T H_2(\xi) v > 0$ and proof is done. □

The following proposition is a consequence of Lemma 36.

Proposition 38. *The self-triggered sparse optimal control (SSOC) problem* (**P1**) *can equivalently be cast as follows:*

$$\underset{f_k, \delta_k}{\textit{minimize}} \; -\delta_k + \gamma \|f_k\|_0 + \eta \|(x_k^T \otimes I) f_k\|_0 \tag{**P2**}$$

$$\textit{subject to: } \forall \xi \in [0, \delta_k) : \frac{1}{2} f_k^T P_1(\xi) f_k + q_1(\xi)^T f_k + r_1(\xi) \leq 0, \tag{8.7}$$

where $f_k = \mathbf{vec}(F_k)$,

$$\begin{aligned} P_1(\xi) &= (x_k x_k^T) \otimes \big(2H_2(\xi) + 2\alpha B^T Z(\xi)^T e^{A^T \xi} \tilde{P} e^{A\xi} Z(\xi) B\big), \\ q_1(\xi) &= 2\mathbf{vec}\Big(\big(H_1(\xi)^T + \alpha B^T Z(\xi)^T e^{A^T \xi} \tilde{P} e^{A\xi}\big) x_k x_k^T \Big), \\ r_1(\xi) &= x_k^T \big(H_0(\xi) + \alpha(e^{A^T \xi} \tilde{P} e^{A\xi} - \tilde{P})\big) x_k. \end{aligned}$$

Proof. The building block of the proof is the following linear algebraic identity:

$$\mathbf{vec}(UVW) = (W^T \otimes U)\mathbf{vec}(V),$$

for any triplet (U, V, W). Utilizing the identity above for $U = I$, $V = F_k$, and $W = x_k$, the objective function of (**P1**) takes the form appeared in objective function of (**P2**). Likewise, (8.7) is derived from (8.5), by repeated utilizations of such an identity to corresponding appropriate matrices. □

From (8.6) and considering Proposition 12, it is evident that $J_k(F_k, \delta_k; x_k)$ is a quadratic convex function in terms of $f_k = \mathbf{vec}(F_k)$. Thus, when δ_k is kept fixed, (**P2**) takes the regularized quadratically-constrained quadratic program (QCQP) form. Substituting the ℓ_0 sparsity measure with its convex surrogate, i.e., ℓ_1 norm, (**P2**) takes the following form:

$$\begin{aligned} \underset{f_k, \delta_k}{\text{minimize}} \ & -\delta_k + \gamma \|f_k\|_1 + \eta \|(x_k^T \otimes I) f_k\|_1 \\ \text{subject to: } \forall \xi \in [0, \delta_k) : & \ \frac{1}{2} f_k^T P_1(\xi) f_k + q_1(\xi)^T f_k + r_1(\xi) \leq 0. \end{aligned} \tag{**P3**}$$

Although the ℓ_1 relaxation makes (**P2**) more tractable, still there may exist a non-convexity in the nature of constraint of (**P3**) in terms of form of dependency on argument ξ. Figure 8.1 showcases a possible case in which the expression on left hand side of (8.7) is a non-convex function of argument ξ. Thus, solving (**P3**) for optimal solutions generally remains as a difficult non-convex task with $mn + 1$ variables. In order to sub-optimally solve (**P3**), the form of (**P3**) motivates us to define the following two main sub-problems:

1. The optimal control problem (**P3**) while F_k is kept fixed which boils down to solve a nonlinear equation.

2. The optimal control problem (**P3**) while δ_k is kept fixed which fortunately

reduces to solve a tractable convex problem.

One may say that since (**P3**) is a tractable convex problem when δ_k is kept fixed, we can solve (**P3**) via solving it when fixed δ_k's are chosen among the set of discretized values and finally the optimal solution will be the one which produces the minimum value for objective function of (**P3**). However, such an approach is highly costly in terms of computational concerns which makes it practically inefficient. In addition, for each calculated F_k while a fixed discretized δ_k is pre-considered, it should be monitored if such a F_k is valid for $\xi \in [0, \delta_k)$ or not and this adds extra computational complexity. Further detailed points regarding the procedure of solving (**P3**) via solving such two sub-problems are included later on.

8.5 Feasibility

Although the feasibility analysis of our formulated problem is straightforward, for the sake of clarification, we state the following remark on feasibility analysis.

Remark 26. *The self-triggered sparse optimal control (SSOC) problem* (**P1**) *is feasible.*

Inspired by form of (8.5)*, we define the following scalar-valued function:*

$$g_k(\xi) = x_k^T \Big(Y_k(F_k, \xi) + \alpha\big(-\tilde{P} + M_k(\xi)^T \tilde{P} M_k(\xi)\big)\Big) x_k. \tag{8.8}$$

The proof is done, if we show that for some F_k, there exists a positive θ_k such that $\forall \xi \in [0, \theta_k)$ inequality $g_k(\xi) \leq 0$ holds. Knowing that the function g_k is differentiable, we show that $g_k'(0) < 0$ holds and implies that for $F_k = \tilde{F}$, there exists a positive θ_k such that $\forall \xi \in [0, \theta_k)$ inequality $g_k(\xi) \leq 0$ holds. According to the definition of right

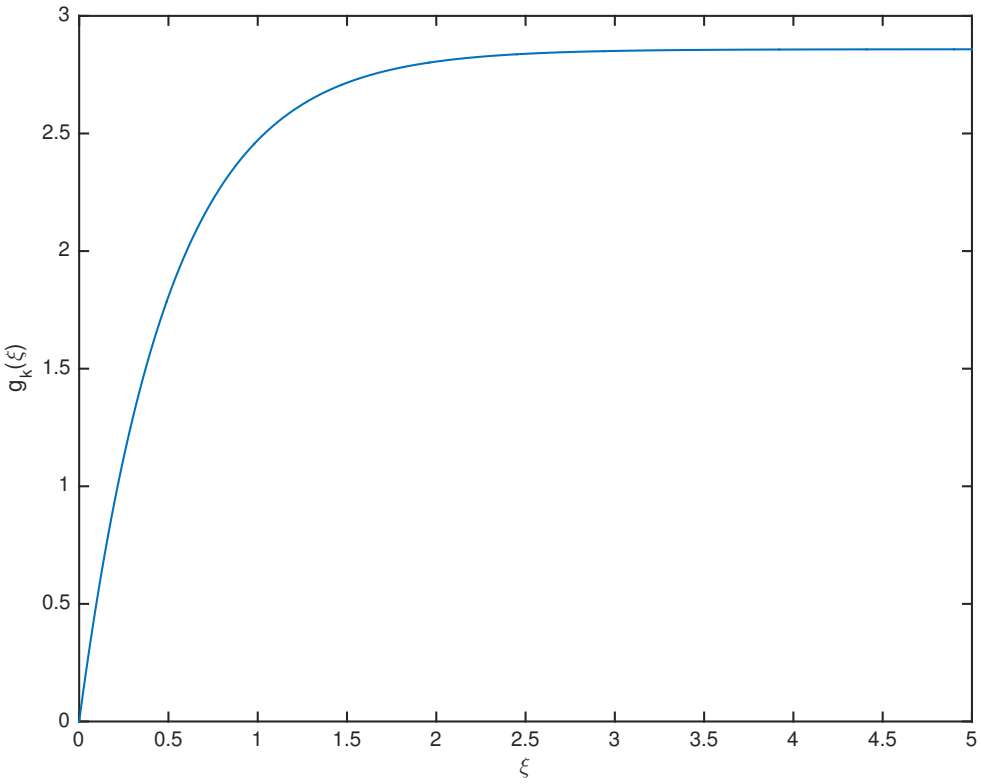


Figure 8.1: A non-convexity in the nature of constraint of (**P3**) in terms of form of dependency on argument ξ.

derivative and noting that $g_k(0) = 0$ *holds, we have*

$$g_k'(0) = \lim_{\xi \to 0^+} \frac{g_k(\xi)}{\xi}. \tag{8.9}$$

Due to (8.9) and definition of mathematical right limit, we have

$$\forall \epsilon > 0, \exists \theta_k > 0 \ s.t. \ \forall \xi \in (0, \theta_k), \ |\frac{g_k(\xi)}{\xi} - g_k'(0)| < \epsilon.$$

Consequently

$$g_k(\xi) < (g_k'(0) + \epsilon)\xi, \tag{8.10}$$

is resulted. Doing simple calculations implies that

$$g_k'(0) = x_k^T \Big(Q + F_k^T R F_k + \alpha\big((A + BF_k)^T \tilde{P} + \tilde{P}(A + BF_k)\big) \Big) x_k.$$

According to (8.4), for $F_k = \tilde{F}$*, we get*

$$g_k'(0) = (1 - \alpha) x_k^T (Q + \tilde{F}^T R \tilde{F}) x_k.$$

Since $\alpha > 1$ *and* $Q + \tilde{F}^T R \tilde{F}$ *is positive definite,* $g_k'(0) < 0$ *gets satisfied for* $F_k = \tilde{F}$*. Thus, due to* (8.10)*, the choice of* $\epsilon < -g_k'(0)$ *implies that for some* F_k *(namely* $\tilde{F}$ *in this case), there exists a positive* θ_k *such that* $g_k(\xi) < 0$ *holds* $\forall \xi \in (0, \theta_k)$ *and since* $g_k(0) = 0$ *holds, subsequently* $g_k(\xi) \leq 0$ *holds* $\forall \xi \in [0, \theta_k)$ *which completes the proof.*

Next, we present a lemma which expresses a necessary condition. Later, we will see how it will help us in our design procedure.

Lemma 39. *Given the function* $g_k(\xi)$ *defined as* (8.8)*, the feasibility of* (**P1**)*, implies that the following inequality holds:*

$$g_k'(0) \leq 0, \tag{8.11}$$

where

$$g_k'(0) = x_k^T \Big(Q + F_k^T R F_k + \alpha\big((A + BF_k)^T \tilde{P} + \tilde{P}(A + BF_k)\big) \Big) x_k.$$

Proof. According to (8.5), $g_k(\xi) \leq 0$ holds for all $\xi \in [0, \delta_k)$. Thus, due to (8.9), it implies that $g_k'(0) \leq 0$ holds. □

In the following, we see a proposition which firstly suggests an equivalent Linear Matrix Inequality (LMI) to (8.11) and secondly provides a sufficient condition to

guarantee (8.11).

Proposition 40. *(i) The condition (8.11) is equivalent to*

$$\begin{bmatrix} R^{-1} & F_k x_k \\ x_k^T F_k^T & -x_k^T \Big(Q + \alpha\big((A+BF_k)^T \tilde{P} + \tilde{P}(A+BF_k)\big) \Big) x_k \end{bmatrix} \succeq 0.$$

(ii) The condition (8.11) is satisfied, if the following LMI holds:

$$\begin{bmatrix} R^{-1} & F_k \\ F_k^T & -\alpha\big((A+BF_k)^T \tilde{P} + \tilde{P}(A+BF_k)\big) - Q \end{bmatrix} \succeq 0. \tag{8.12}$$

Proof. (i) Applying the Schur complement [129] to (8.11), proof is achieved.
(ii) Considering the form of (8.11) implies that satisfaction of the following matrix inequality:

$$Q + F_k^T R F_k + \alpha\big((A+BF_k)^T \tilde{P} + \tilde{P}(A+BF_k)\big) \preceq 0. \tag{8.13}$$

leads to guarantee such a condition. Applying the Schur complement to (8.13), (8.12) is obtained and proof is completed. □

Remark 27. *Since we aim to propose a design procedure which works for any arbitrarily-chosen initial condition x_0, the sufficient condition* (8.12) *becomes necessary, i.e.,* (8.12) *would be necessary and sufficient condition in our design procedure when we solve for F_k along with a fixed δ_k. In other words, necessary condition* (8.11) *holds for any arbitrarily-chosen initial condition which leads to necessity of* (8.12)*. It is worth emphasizing that x_k can be interpreted as an initial condition of k^{th} time interval $[t_k, t_{k+1})$ and since the design procedure should work for all arbitrarily-chosen initial condition x_0's, such a conservative interpretation makes sense. The conservatism comes from the complicated nature of the dependency of x_k on x_0.*

Remark 28. *It is worth considering the case that $F_0 = 0$ and $\delta_0 = \infty$ are solutions*

for problem (**P1**). *Evaluating function* $g_k(\xi)$ *in* (8.8) *with* $F_0 = 0$, *we get the following expression:*

$$g_0(\xi) = \int_0^{\xi} x_0^T e^{A^T\tau}\big(Q + \alpha(A^T\tilde{P} + \tilde{P}A)\big)e^{A\tau}x_0 d\tau. \tag{8.14}$$

Thus, (8.14) *is non-positive for any arbitrarily chosen* x_0 *and for any non-negative* ξ *if and only if*

$$A^T\tilde{P} + \tilde{P}A + \frac{1}{\alpha}Q \preceq 0, \tag{8.15}$$

holds. Thus, $F_0 = 0$ *and* $\delta_0 = \infty$ *are solutions for problem* (**P1**), *for any arbitrarily chosen* x_0, *if and only if* (8.15) *holds. Moreover, according to the linear quadratic Lyapunov theory, satisfaction of* (8.15) *implies that* A *is Hurwitz (stable).*

Also, if the following condition holds:

$$A^T\tilde{P} + \tilde{P}A + \frac{1}{\alpha}Q \not\succ 0, \tag{8.16}$$

then $\exists x_0$ *for which,* $F_0 = 0$ *and* $\delta_0 = \infty$ *are solutions for problem* (**P1**).

Motivated by Remark 28, for the rest of the chapter we assume the following assumption to avoid the trivial solutions for any arbitrarily chosen x_0.

Assumption 5. *For a given* 5*-tuple* (A, B, Q, R, α), (8.15) *does not hold.*

8.6 Stability

Section 8.6 investigates the stability of the closed-loop system which is controlled by sequence of performance-based self-triggered sparse optimal controllers (8.2). The remark which we will state on stability guarantee is a well-investigated concept in model predictive control (MPC) literature [59].

Before commenting on the stability of our proposed controller design, we define the total cost for our proposed SSOC problem by summing up all elements of sequence

$\{J_k(F_k, \delta_k; x_k)\}_{k=0}^{\infty}$ and obtain

$$J := \sum_{k=0}^{\infty} J_k(F_k, \delta_k; x_k).$$

The above mentioned definition is constructed based on the fact that (**P1**) is feasible according to Remark 26. Otherwise, it is not possible to utilize such a definition in our controller design problem. Knowing that, the remark on stability is presented in the following.

Remark 29. *The optimal control problem* (**P1**) *results in stabilizing* $\{F_k\}_{k=0}^{\infty}$ *and the relative performance loss percentage* $\nu = 100 \times \frac{J-\tilde{J}}{\tilde{J}}$ *is upper bounded by* $100 \times (\alpha - 1)$ *where* $\tilde{J} = x_0^T \tilde{P} x_0$.

Let us define S_l *as follows:*

$$S_l := \sum_{k=0}^{l} J_k(F_k, \delta_k; x_k).$$

Taking the sum of both sides of (8.5) for first $l+1$ *terms, we get*

$$0 \leq S_l \leq \alpha(x_0^T \tilde{P} x_0 - x_{l+1}^T \tilde{P} x_{l+1}) < \alpha x_0^T \tilde{P} x_0 = \alpha \tilde{J}. \tag{8.17}$$

The sequence $\{S_l\}_{l=0}^{l=\infty}$ *is an increasing sequence. According to (8.17), it is bounded. Thus, it implies that the sequence* $\{S_l\}_{l=0}^{l=\infty}$ *is convergent. Taking the limit from both sides of (8.17), we get*

$$J = \sum_{k=0}^{\infty} J_k(F_k, \delta_k; x_k) = \lim_{l \to \infty} S_l < \lim_{l \to \infty} \alpha \tilde{J} = \alpha \tilde{J}. \tag{8.18}$$

Doing simple calculation on (8.18) shows that the following inequality holds:

$$\nu = 100 \times \frac{J - \tilde{J}}{\tilde{J}} < 100 \times (\alpha - 1).$$

8.7 Self-Triggered Sparse Optimal Control (SSOC): Performance-Based Method

This section consists of three parts where first two parts enable us to solve (**P3**) via solving two sub-problems (**P4**) and (**P5**). The first part describes the case for which (**P3**) is solved for δ_k when F_k is kept fixed. The second part is dedicated to solve (**P3**) for F_k while δ_k is kept fixed. The third part provides us with lower bounds on inter-execution times. Such lower bounds play an important role in computation of inter-execution times, because they provide an appropriate estimate for initial guess of solution of the corresponding nonlinear equation.

8.7.1 Solving (P3) for δ_k when F_k is Kept Fixed

It can simply be verified that inequality (8.7) can equivalently be cast as the following quadratic constraint:

$$\frac{1}{2}u_k^T P_2(\xi)u_k + q_2(\xi)^T u_k + r_1(\xi) \leq 0, \tag{8.19}$$

where $u_k = F_k x_k$,

$$P_2(\xi) = 2H_2(\xi) + 2\alpha B^T Z(\xi)^T e^{A^T\xi}\tilde{P}e^{A\xi}Z(\xi)B,$$
$$q_2(\xi) = \big(2H_1(\xi)^T + 2\alpha B^T Z(\xi)^T e^{A^T\xi}\tilde{P}e^{A\xi}\big)x_k.$$

Assuming the fixed value for F_k and according to (8.5), (**P3**) boils down to the following auxiliary nonlinear optimization problem:

$$\begin{aligned} &\underset{\delta_k}{\text{minimize}} \;\; -\delta_k && \textbf{(P4)}\\ &\text{subject to: } \forall\xi \in [0,\delta_k) : \frac{1}{2}u_k^T P_2(\xi)u_k + q_2(\xi)^T u_k + r_1(\xi) \leq 0. \end{aligned}$$

Considering (**P4**), it implies that the k^{th} inter-execution time δ_k can be obtained as follows:

$$\delta_k = \sup\big\{\theta_k \in \mathbb{R}_{++} | (8.5) \text{ is satisfied } \forall \xi \in [0, \theta_k)\big\}. \tag{8.20}$$

In order to solve (8.20), we utilize the simple, commonly-used discretization rule which has been employed by [49, 56] as well. To do so, we will utilize the derived lower bounds on inter-execution times which will be discussed in detail in next subsection. It is worth mentioning that other computational tools such as nonlinear optimization-based methods can be effective to solve (8.20) too. In particular, for the case that we have just 1 input, i.e., single-input systems (such as one of numerical simulations provided by [128]), the method proposed by [130] can be utilized.

8.7.2 Solving (**P3**) for F_k when δ_k is Kept Fixed

Since solving (**P3**) for F_k when δ_k is kept fixed, requires satisfaction of infinitely many constraints, it is undoubtedly a computationally expensive task. To overcome such an issue, instead of checking the all set of infinitely many constraints, we only consider the endpoint of the corresponding time interval, i.e., just δ_k.

Applying the Schur complement to (8.19) and changing the arguments to δ_k, the following LMI constraint is obtained:

$$\begin{bmatrix} 2P_2(\delta_k)^{-1} & F_k x_k \\ x_k^T F_k^T & -q_2(\delta_k)^T F_k x_k - r_1(\delta_k) \end{bmatrix} \succeq 0,$$

Then, assuming a fixed value for δ_k, we get the following regularized SDP:

$$\begin{aligned} &\underset{F_k}{\text{minimize}}\ \gamma\|F_k\|_1 + \eta\|F_k x_k\|_1 \qquad \textbf{(P5)} \\ &\text{subject to: } \begin{bmatrix} 2P_2(\delta_k)^{-1} & F_k x_k \\ x_k^T F_k^T & -q_2(\delta_k)^T F_k x_k - r_1(\delta_k) \end{bmatrix} \succeq 0. \end{aligned}$$

According to Remark 27, in process of solving (**P5**) for F_k, we strictly include LMI (8.12) as an additional constraint.

It should be emphasized that F_k obtained from solving (**P5**) may not satisfy inequality (8.7) for all $\xi \in [0, \delta_k)$. This is an expected issue which obviously arises from ignoring the continuously satisfaction of the optimization constraint. However, solving (**P3**) for δ_k when F_k is set to the value obtained from solving (**P5**), enables us to check that if obtained F_k from solving (**P5**) is valid or not. In other words, to see if such an F_k satisfies inequality (8.7) for all $\xi \in [0, \delta_k)$ or not. If it does not satisfy such a condition, then, we will repeat solving (**P5**) for a decreased value of δ_k and again check the validity of the newly calculated F_k. This process will definitely be stopped because due to satisfaction of (8.12) which is strictly implemented in the design procedure, $g_k'(\xi) < 0$ holds and subsequently $g_k(\xi) < 0$ continuously holds for all $\xi \in [0, \delta_k)$.

8.7.3 Lower Bounds and Constraints on Inter-Execution Times

The following two propositions suggest lower bounds on inter-execution times which will later be utilized as an effective tool in computation of inter-execution times via solving (**P4**).

Proposition 41. *The k^{th} inter-execution time δ_k given by* (8.20) *is lower bounded*

by δ_k^* which is defined as follows:

$$\delta_k^* := \sup\{\theta_k \in \mathbb{R}_{++} | \mathcal{M}_k \succeq 0,\ \forall \xi \in [0, \theta_k)\}, \tag{8.21}$$

where

$$\mathcal{M}_k = \begin{bmatrix} \frac{1}{\alpha}\tilde{P}^{-1} & 0 & M_k \\ 0 & H_2^{-1} & F_k \\ M_k^T & F_k^T & \alpha\tilde{P} - H_0 - H_1 F_k - F_k^T H_1^T \end{bmatrix}.$$

Proof. A sufficient condition which implies (8.5) is as follows:

$$Y_k(F_k, \xi) + \alpha\big(- \tilde{P} + M_k(\xi)^T \tilde{P} M_k(\xi)\big) \preceq 0. \tag{8.22}$$

Knowing that

$$Y_k(F_k, \xi) = H_0(\xi) + F_k^T H_1(\xi)^T + H_1(\xi)F_k + F_k^T H_2(\xi) F_k,$$

holds, substituting it in (8.22), and applying the Schur complement we get $\mathcal{M}_k \succeq 0$. Thus, δ_k^* gives us a lower bound on δ_k. □

Remark 30. *Similar to the point mentioned by Remark 27, since robust stability is desired in terms of dealing with arbitrarily-chosen initial condition, condition $\mathcal{M}_k \succeq 0$ becomes necessary, i.e., $\mathcal{M}_k \succeq 0$ would be a necessary and sufficient condition when we solve for F_k assuming a fixed value for δ_k.*

Proposition 42. *The k^{th} inter-execution time δ_k given by* (8.20) *is lower bounded by $\delta_k^\dagger$ which is defined as follows:*

$$\delta_k^\dagger := \min\{\xi \in \mathbb{R}_{++} | \det\big(\mathcal{N}_k(\xi)\big) = 0\}, \tag{8.23}$$

where

$$\mathcal{N}_k(\xi) = Y_k(F_k, \xi) + \alpha\big(- \tilde{P} + M_k(\xi)^T \tilde{P} M_k(\xi)\big).$$

Proof. Let us define $\hat{g}_k$ as follows:

$$\hat{g}_k(\xi, x_k) = x_k^T \mathcal{N}_k(\xi) x_k.$$

In order to find a lower bound on δ_k which is the smallest positive solution of $\hat{g}_k(\xi, x_k) = 0$, it can be assumed that $\xi = h_k(x_k)$ where h_k is an implicit mapping. In other words, we have $\hat{g}_k\big(h_k(x_k), x_k\big) = 0$. Thus, similar to the idea used in [49], setting the derivative of h_k with respect to components of x_k equal to 0, it implies that

$$\partial \hat{g}_k / \partial x_k = 0.$$

Since $\mathcal{N}_k(\xi)$ is a symmetric matrix, $\partial \hat{g}_k / \partial x_k$ would be equal to $2\mathcal{N}_k(\xi)$. Thus, we get $\mathcal{N}_k(\xi) x_k = 0$ or equivalently $\det\big(\mathcal{N}_k(\xi)\big) = 0$. □

The following corollary is immediately resulted from merging Propositions 41 and 42.

Corollary 43. *The k^{th} inter-execution time δ_k given by* (8.20) *is lower bounded by $\underline{\delta}_k$ which is defined as follows:*

$$\underline{\delta}_k = \max\{\delta_k^*, \delta_k^\dagger\},$$

wherein δ_k^* and $\delta_k^\dagger$ are defined by (8.21) *and* (8.23)*, respectively.*

Using (8.19), the following proposition is derived which suggests a property of δ_k.

Proposition 44. *For $\xi = \delta_k$ $\big($the k^{th} inter-execution time given by (8.20)$\big)$ the following inequality holds:*

$$-\frac{1}{2}q_2(\xi)^T P_2(\xi)^{-1} q_2(\xi) + r_1(\xi) \leq 0. \tag{8.24}$$

Proof. If the quadratic term on the left side of (8.19) is non-positive, then the minimum value of such a quadratic term would also be less than or equal to zero. Since $P_2(\delta_k)$ is positive definite, then the unique minimizer for such a quadratic term would be $-P_2(\delta_k)^{-1}q_2(\delta_k)$. The corresponding minimum value for such a minimizer would be the left side of (8.24). Thus, proof is achieved. □

8.8 Algorithm

In section 8.8, we develop an algorithm to find the sequence of self-triggered sparse optimal controllers. The main scheme of the algorithm is described as follows:

For each non-negative k, at k^{th} time interval, firstly, setting $F_k = \tilde{F}$, solving (8.20), we find $\delta_k^{(0)}$. Then, setting $\delta_k = \delta_k^{(0)}$, we solve (**P5**) to get F_k. Secondly, for such an obtained F_k, solving (8.20), we find $\delta_k^{(1)}$ and compare it with $\delta_k^{(0)}$, if $\delta_k^{(0)} \leq \delta_k^{(1)}$ holds then we update δ_k by obtained $\delta_k^{(1)}$ and process is done. Otherwise, we update δ_k by $\delta_k - \omega\delta_k^{(0)}$ where $0 < \omega < 1$ holds and repeat the previous step. As it is explained in subsection 8.7.2, such a repetition will definitely be stopped after finite iterations. The optimal control problem (**P5**) can be solved by convex solvers such as CVX [112] with MOSEK solver [131]. A big picture of our algorithm is simply stated as Algorithm 1.

It is noteworthy that all matrices $H_0(\xi)$, $H_1(\xi)$ and $H_2(\xi)$ can separately be computed ahead of time as their values only depend on the state-space matrices A and B, the weight matrices Q and R, and the time argument ξ. Also, x_{k+1} is computed via $M_k(\delta_k)x_k$ wherein x_k is pre-known from the previous time interval.

Algorithm 1: Self-Triggered Sparse Optimal Control (SSOC) Design

Inputs: A, B, Q, R, $n_{\max}$, ω, γ, η, and α.
For $k = 0 : n_{\max}$
  If $k = 0$ *then*
  $N_k = I$,
  End
Compute $H_0(\xi)$, $H_1(\xi)$, *and* $H_2(\xi)$,
Solve (8.20) for $\delta_k^{(0)}$ *with setting* $F_k = \tilde{F}$,
Solve (**P5**) *for* $F_k^{(0)}$, *with setting* $\delta_k = \delta_k^{(0)}$,
Solve (8.20) for $\delta_k^{(1)}$ *with setting* $F_k = F_k^{(0)}$,
  While $\delta_k^{(0)} > \delta_k^{(1)}$
  Solve (**P5**) *for* $F_k^{(1)}$, *with setting* $\delta_k \leftarrow \delta_k - \omega\delta_k^{(0)}$,
  Solve (8.20) for $\delta_k^{(1)}$ *with setting* $F_k = F_k^{(1)}$,
  End
Put $\delta_k = \delta_k^{(1)}, F_k = F_k^{(1)}$,
Compute $M_k(\delta_k)$ *via* $M_k(\tau) = e^{A\tau}(I + Z(\tau)BF_k)$,
Update N_{k+1} *and* x_{k+1} *via* $N_{k+1} = M_k(\delta_k)N_k$
and $x_{k+1} = N_{k+1}x_0$, *respectively,*
$k \leftarrow k + 1$,
End
Output: $\{F_k\}_{k=0}^{n_{\max}}$ and $\{\delta_k\}_{k=0}^{n_{\max}}$.

8.9 Numerical Simulations

To assess the effectiveness of our self-triggered sparse optimal control (SSOC) design, we consider the class of spatially distributed systems. Such a class of systems has thoroughly been investigated in [42].

8.9.1 Spatially Distributed Systems

Similar to the methodology utilized by [42], let us consider $N = 10$ randomly distributed (with a uniform distribution) nodes in a 10×10 box-shaped region (See Figure 8.2).

Remark 31. *Since* (**P5**) *is in the form of SDP, for large-scale systems, our algorithm*

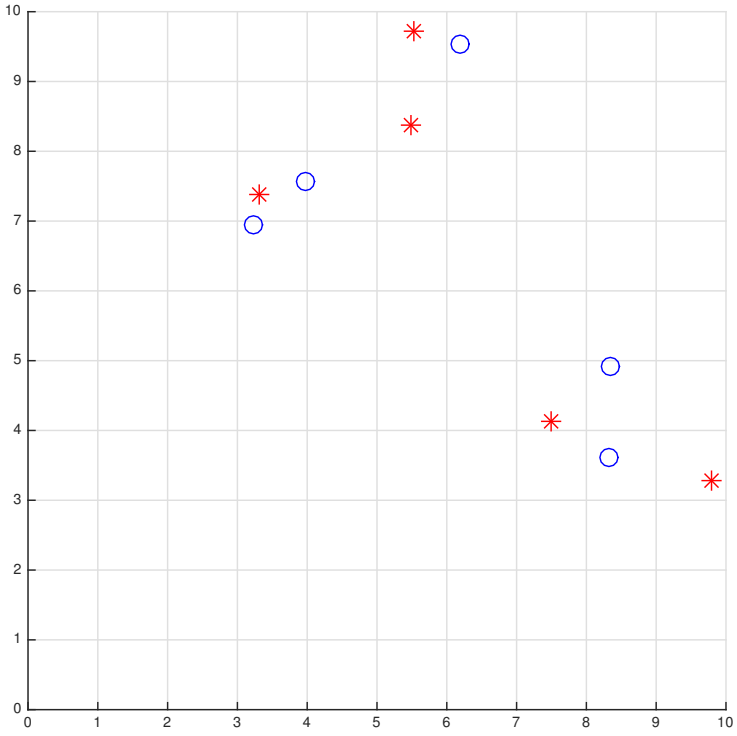


Figure 8.2: Positions of $N = 10$ randomly generated nodes in a 10×10 box-shape region.

will be highly costly in terms of time complexity. As a result, we choose a reasonable value for N such as 10*. However, we can increase the number of nodes, i.e., N, the payoff would be the higher computational cost.*

Each node represents a linear sub-system which is coupled via its dynamics and the linear-quadratic cost to the other sub-systems. The dynamics of the i^{th} linear sub-system is characterized as follows:

$$\dot{x}^{(i)}(t) = [A]_{ii}x^{(i)}(t) + \sum_{j=1,\ j\neq i}^{N} [A]_{ij}x^{(j)}(t) + [B]_{ii}u^{(i)}(t),$$

where

$$[A]_{ii} = \begin{bmatrix} 1 & 1 \\ 1 & 2 \end{bmatrix}, \qquad [B]_{ii} = \begin{bmatrix} 0 \\ 1 \end{bmatrix} \text{ for nodes marked by red } *,$$

$$[A]_{ii} = \begin{bmatrix} -2 & 1 \\ 1 & -3 \end{bmatrix}, \qquad [B]_{ii} = \begin{bmatrix} 0 \\ 1 \end{bmatrix} \text{ for nodes marked by blue } \circ,$$

and

$$[A]_{ij} = \frac{1}{e^{\beta \text{dis}(i,j)}} \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}, [B]_{ij} = \begin{bmatrix} 0 \\ 0 \end{bmatrix}, \ \forall j \neq i,$$

where β determines the spatially decaying rate in spatially-decaying operators and $\text{dis}(i,j)$ denotes the Euclidean distance between nodes i and j in Figure 8.2. Later on, we will visualize the effect of β on sparsification process.

8.9.2 Spatial/Temporal Sparsity Visualizations for SSOC Design

Considering a 20×20 randomly distributed system (β=1) drawn from $N = 10$ nodes in Figure 8.2 and setting the parameters γ and η to 0.001 and 0.001, respectively, $n_{\max} = 49$, $\omega = 0.05$, $\alpha = 1.15$, $Q = I$, and $R = 2I$, Figures 8.3(a) and 8.3(b) are obtained which depict the relative cardinality of controllers

$$100 \times (\|F_k\|_0 / \|F_{LQR}\|_0),$$

and relative cardinality of control inputs

$$100 \times (\|u_k\|_0 / \|u_k^{LQR}\|_0),$$

respectively. In Figure 8.3(a), at each triggering time t_k, the corresponding bar shows the relative cardinality of controllers. In Figure 8.3(b), at each triggering time t_k, the corresponding bar depicts the relative cardinality of control inputs. As Figures 8.3(a) and 8.3(b) express, both relative cardinality of controllers and relative cardinality of control inputs attain values less than 100 % which means that SSOC improves the spatial sparsity compared to the periodic time triggered LQR design. Specifically, on average, cardinalities of controllers and control inputs are improved by 38.3857 % and 50.3325 %, respectively while the corresponding payoff is 15 % performance loss.

Figure 8.4(a) compares the Euclidean norm of state trajectories of SSOC with the Euclidean norm of state trajectories of periodic time-triggered LQR design. Dividing the each time interval to 20 equidistant sub-intervals and evaluating the $x(t)$ in such points via $x(t) = M_k(t-t_k)x_k$, Figure 8.4(b) visualizes the state trajectories of SSOC starting from an arbitrarily-chosen x_0.

Figure 8.5 showcases the inter-execution times δ_k versus time t. To measure the average value of relative cardinality of controllers (spatial sparsity) over time, the following quantity is defined:

$$R_F := 100 \times \frac{\sum_{k=0}^{n_{\max}} \delta_k (\|F_k\|_0 / \|F_{LQR}\|_0)}{\sum_{k=0}^{n_{\max}} \delta_k}.$$

Likewise, the average relative cardinality of control inputs (spatial sparsity) over time can be defined as follows:

$$R_u := 100 \times \frac{\sum_{k=0}^{n_{\max}} \delta_k (\|u_k\|_0 / \|u_k^{LQR}\|_0)}{\sum_{k=0}^{n_{\max}} \delta_k}.$$

Also, the average inter-execution time (temporal sparsity) is defined as follows:

$$D = \frac{\sum_{k=0}^{n_{\max}} \delta_k}{n_{\max}}.$$

The dependency of quantities R_F, R_u, and D on parameter α is captured in Table 8.1. As it is observed, the trend demonstrates that the higher performance loss, the sparser control design we get in terms of both spatial sparsity, i.e., R_F/R_u and temporal sparsity, i.e., D.

α	R_F	R_u	D
1.05	85.5552 %	62.5386 %	0.2823
1.10	72.5959 %	66.8855 %	0.2715
1.15	61.6143 %	49.6675 %	0.3322
1.20	53.8641 %	41.0432 %	0.3247
1.25	50.6312 %	36.8142 %	0.3629
1.30	49.0471 %	31.9999 %	0.3529

Table 8.1: Dependency of quantities R_F, R_u, and D on parameter α.

8.9.3 Effect of Spatially Decaying Rate β on Sparsification Process

Having the same numerical specifications from previous subsection except for the β and fixing the values of γ and η, the effect of β on R_F, R_u, and D is studied.
Here, we assume that γ and η are set equal to 0.001 and 0.001, respectively. Figures 8.6(a), 8.6(b), and 8.7 demonstrate the dependency of R_F and R_u on β, respectively.

As Figures 8.6(a) and 8.6(b) depict, there is a tradeoff between spatially decaying rate β and R_F/R_u, respectively. Such a tradeoff is not unexpected. Because, as β increases, the spatially distributed system automatically tends to be sparser and consequently, the controller gains F_k's and control inputs u_k's tend to be sparser. Also, according to Figure 8.7, the similar observation is true for temporal sparsity, i.e., as the spatially distributed system gets sparser, the average inter-execution time increases which means that less number of samplings will be required.

8.9.4 Effect of Penalizing Parameters γ/η on Sparsification Process

In this subsection, considering the previously considered setup, the effect of parameters γ/η on quantities R_F/R_u is investigated. To investigate the effect of penalizing parameter γ on R_F, assuming the $\alpha = 1.15$ (at most 15 % performance loss), fixing the $\eta = 0.001$, and choosing 20 log-scaled values for γ varying between 10^{-5} and 10^{-3}, and running Algorithm 1, the decreasing behavior between γ and R_F is visualized via Figure 8.8(a). Such a decreasing behavior is not unexpected. Because, the penalizing parameter γ appeared in objective function of (**P5**), is the coefficient multiplied by the ℓ_1 norm of controller F_k. Thus, when it increases, the cardinality is supposed to be decreased. In such a case, the number of communications among nodes is decreased.

Figure 8.8(b) illustrates the relationship between η and R_u, for setting $\alpha = 1.15$, $\gamma = 10^{-3}$, and 20 log-scaled values for η varying between 10^{-5} and 10^{-3}. Similar to R_F-γ relationship, there exists a trade-off between R_u and η. In other words, as penalizing parameter η increases, it enforces the components of control inputs to be equal to zero and as a consequence less number of utilized actuators.

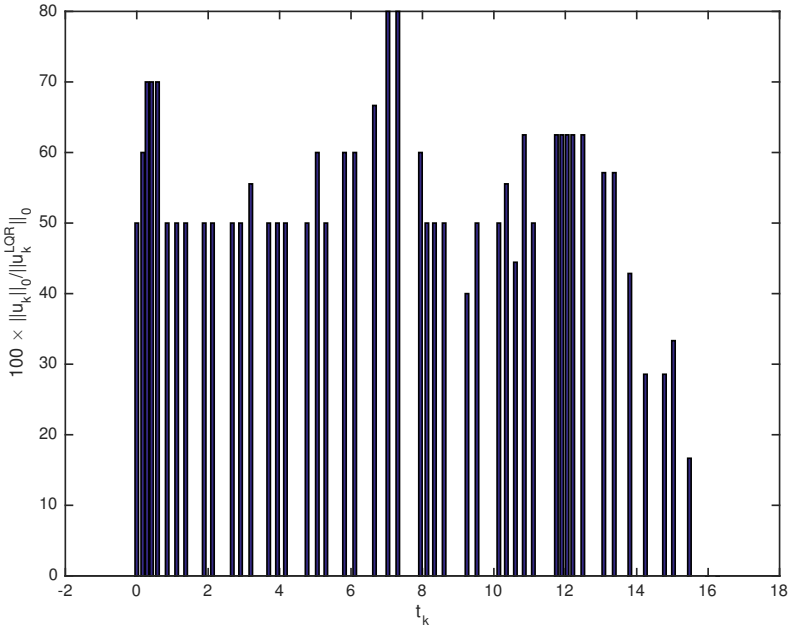
Remark 32. *The reason which we choose an upper bound for γ/η in our demonstrations is implicitly implied by the fact that we assume the upper bound α for the performance loss. In other words, since as γ/η enlarges, the corresponding performance loss increases accordingly, we cannot increase the upper bound for γ/η arbitrarily for a pre-specified α.*

8.10 Conclusion

We present a mixture of self-triggered control and sparse optimal control. Each time interval is divided into two main levels: (i) computation of the inter-execution time while feedback gain is kept fixed (via nonlinear optimization). (ii) design of the

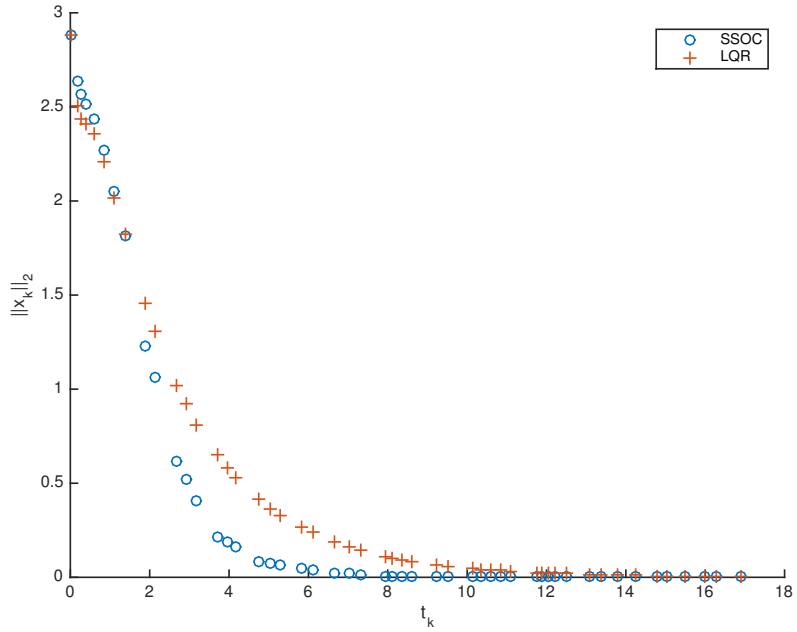
sparse optimal controller while inter-execution time is kept fixed (via convex optimization). At both previously mentioned levels, stability is guaranteed via an enforced performance-based constraint. The numerical simulations show that SSOC improves sparsity both in time and space. In other words, the average sampling rate and average cardinalities are less compared to the periodic time-triggered LQR design and the performance loss payoff is not much comparatively. Meanwhile, its performance loss can be upper bounded by means of a pre-specified parameter. Also, it is observed the average relative cardinality of controller gains, average relative cardinality of control inputs, and average sampling rate decrease as performance loss increases. Additionally, in the case of spatially distributed systems, it is verified that there exists a trade-off between spatially decaying rate and spatial sparsity quantities. Another visualized trade-off is the one between penalizing parameters and spatial sparsity quantities. A future work can be the distributed version of our proposed method in which each node should compute its control inputs accordingly. Furthermore, the improvement on sub-optimality of our proposed method can be thought as another future direction.

(a)

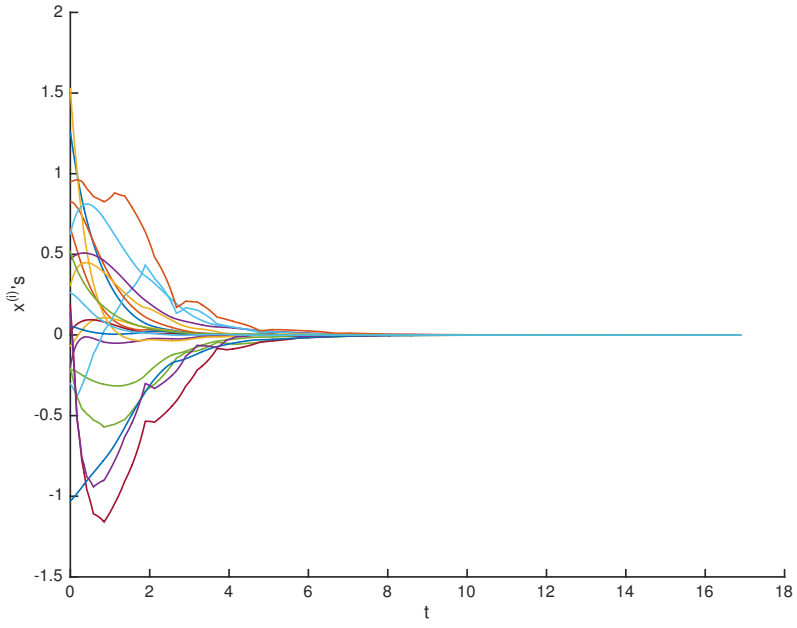


(b)

Figure 8.3: (a) Relative cardinality of controllers $100 \times (\|F_k\|_0 / \|F_{LQR}\|_0)$ versus triggering times t_k (b) Relative cardinality of control inputs $100 \times (\|u_k\|_0 / \|u_k^{LQR}\|_0)$ versus triggering times t_k.



(a)



(b)

Figure 8.4: (a) The Euclidean norm of state trajectories of SSOC and periodic time-triggered LQR design $\|x_k\|_2$ versus triggering times t_k (b) State trajectories $x^{(i)}$'s of SSOC starting from an arbitrarily-chosen x_0 for $i \in \{1, 2, \cdots, 19, 20\}$.

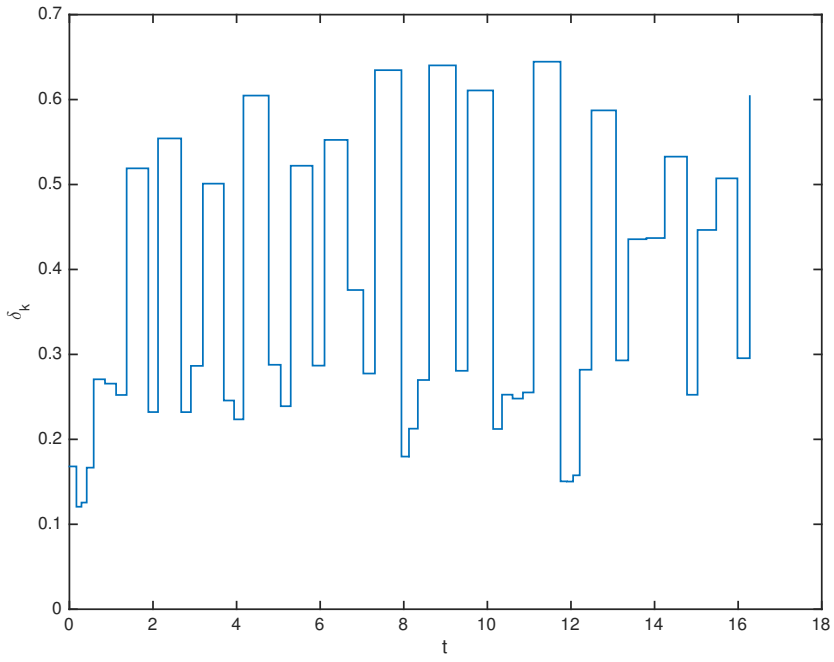
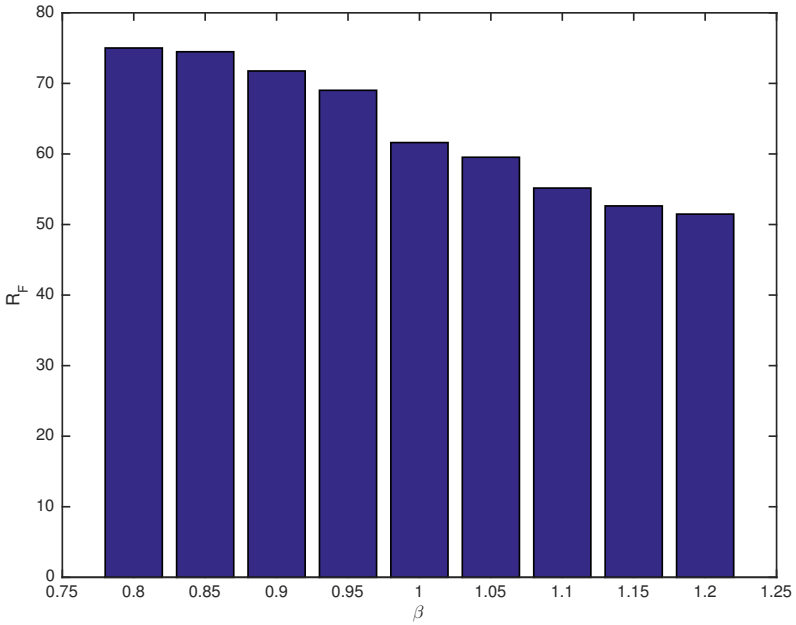
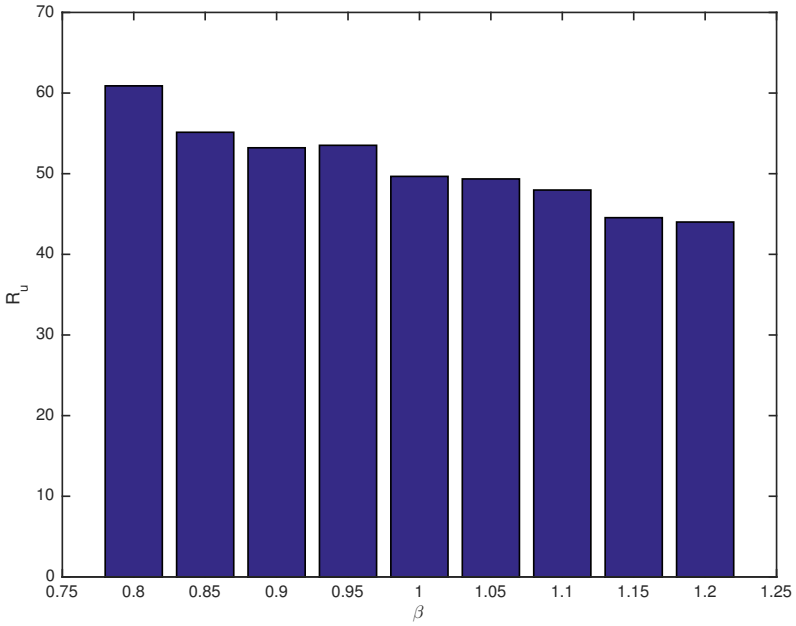


Figure 8.5: Inter-Execution times δ_k versus time t.



(a)



(b)

Figure 8.6: (a) Average relative cardinality of controllers R_F versus spatially decaying rate β. (b) Average relative cardinality of control inputs R_u versus spatially decaying rate β.

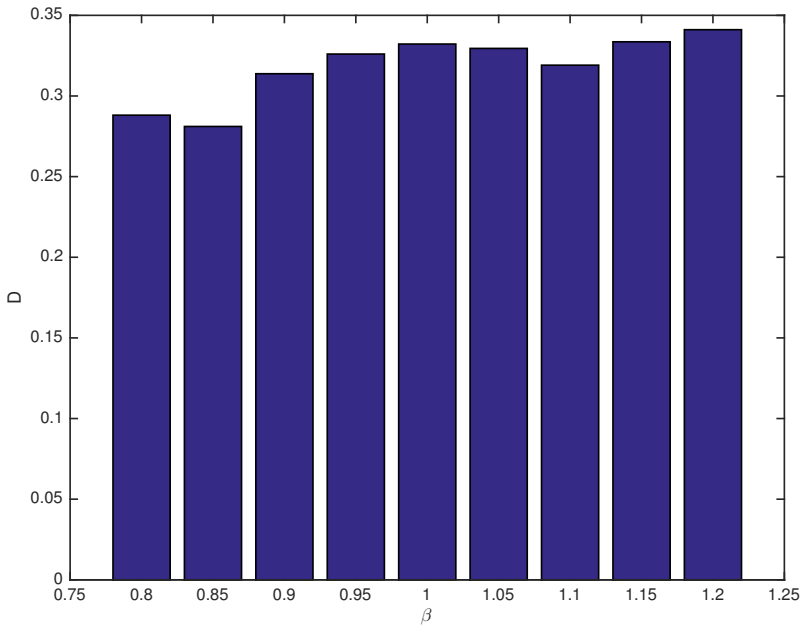
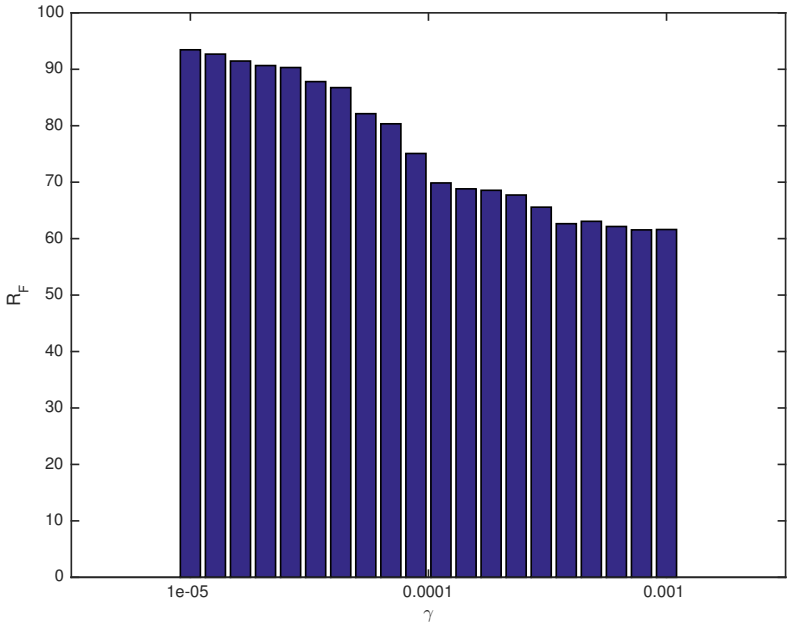
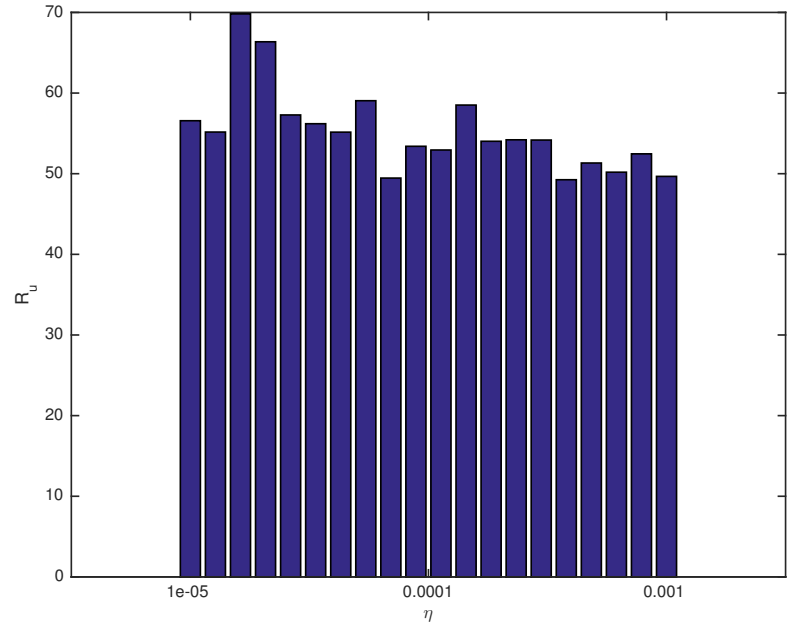


Figure 8.7: Average inter-execution time D versus spatially decaying rate β.



(a)



(b)

Figure 8.8: (a) Average relative cardinality of controllers R_F versus penalizing parameter γ. (b) Average relative cardinality of control inputs R_u versus penalizing parameter η.

Chapter 9

Feedback Controller Sparsification via Quasi-Norms

9.1 Introduction

During the past two decades, several research works have been done in the area of sparsity-promoting optimal control. To address some of such works, we encourage the interested reader to see [1, 7, 13, 14, 18, 20, 21, 26–28, 30, 33, 34, 38, 40, 42, 80, 132–135]. The main goal in sparsity-promoting optimal control is to make a reasonable balance between the number of communication links among nodes and the network performance loss.

All of the above-mentioned methods except the [33], are unfortunately unable to deal with large-scale systems and fail to propose a sparse feedback controller for such systems. Throughout the chapter, we show how $q \in (0, 1)$ quasi-norms enable us to find sparse stabilizing feedback controllers in the geometrical norm based vicinity of a given dense centralized feedback controller for a large-scale system. However, the drawback of our proposed method is that an upper bound exists for the sparsification rate while most of the other sparsity-promoting optimal controller design methods

do not face with such an issue when they are utilized to design a sparse feedback controller for medium size systems.

This chapter is structured as follows: After stating the mathematical notations in Section 9.2 and subsequently formulating the problem to be solved in Section 9.3, Section 9.4 explains how $q \in (0, 1)$ quasi-norms can be considered as a handy tool to sparsify a given feedback controller. Providing the extensive numerical solutions, Section 9.5 assesses the effectiveness of our $q \in (0, 1)$ quasi-norm based sparsification method for large-scale systems, showcases that $q \in (0, 1)$ quasi-norm based method can outperform the truncation based cardinality minimization in some cases, and investigates the relationship between q and the sparsity-performance trade-off (in particular, it is observed that the less q we consider, the better sparsity-performance balance we get). Section 9.6 mentioning the concluding remarks and possible future directions, ends the chapter.

9.2 Mathematical Notations

Throughout the chapter, vectors and matrices are shown with lower-case and upper-case letters, respectively. The set of real numbers, $n \times 1$ real vectors, and $m \times n$ real matrices are represented by $\mathbb{R}$, $\mathbb{R}^n$, and $\mathbb{R}^{m\times n}$, respectively. The transpose of a matrix is denoted by superscript T. The vector of all ones is denoted by $\mathbf{1}$. The symbol I denotes the identity matrix. A matrix M is called Hurwitz if and only if all of its eigenvalues lie on the complex open left half-plane. The ℓ_0 sparsity measure of matrix M is denoted by $\|M\|_0$ which simply equals to its number of non-zero elements. The ℓ_1 norm of matrix M is sum of absolute value of its elements and denoted by $\|M\|_1$. The trace of square matrix M is sum of its eigenvalues and shown by $\mathbf{Tr}(M)$. The

$q \in (0,1)$ quasi-norm of matrix M is defined as follows:

$$\|M\|_q = (\sum_{i,j} |M_{ij}|^q)^{\frac{1}{q}}.$$

The Frobenius norm of matrix M is denoted by $\|M\|_F$ and defined as $\sqrt{\mathbf{Tr}(M^T M)}$. The sign function of a scalar s is represented by $\mathbf{sign}(s)$. The big O complexity notation is denoted by $\mathcal{O}$. A normally-distributed random variable u with zero mean and unit variance is denoted by $u \in \mathcal{N}(0,1)$.

9.3 Problem Formulation

In the area of sparsity-promoting control, to sparsify the feedback controller, diverse operators have been utilized such as truncation operator (cardinality minimization), soft thresholding operator (ℓ_1 norm minimization), and sum of logs [1]. However, the $q \in (0,1)$ quasi-norms have not been utilized in this specific area in general. Such quasi-norms have thoroughly been studied and utilized by [7, 14, 30] to measure the sparsity and sparsify the feedback controller in control of a special class of systems, i.e., the class of spatially-decaying systems. In addition to the control theory field, such quasi-norms have been helpful in other fields such as data compression, image signal processing, and linear least squares. In [136], utilizing the $q \in (0,1)$ quasi-norms, sparse solutions are obtained for a linear least squares problem.

According to the works authored by [33, 40, 133], it is reasonable to seek for sparse feedback controllers in the geometrical norm based vicinity of the given well-performing feedback controllers.

Thus, merging the notion of $q \in (0,1)$ quasi-norms and concept of geometrical norm based vicinity, we formulate the following unconstrained optimization problem:

$$\underset{K}{\text{minimize}} \ \frac{1}{2}\|K - F\|_F^2 + \gamma \|K\|_q^q, \tag{9.1}$$

where F is a given well-performing feedback controller which stabilizes the following LTI system:

$$\dot{x}(t) = Ax(t) + Bu(t) + Dd(t), \tag{9.2}$$

via $u(t) = Fx(t)$ and K denotes the sparse feedback controller which stabilizes (9.2) via $u(t) = Kx(t)$. Meanwhile, keep in mind that $A \in \mathbb{R}^{n\times n}$, $B \in \mathbb{R}^{n\times m}$, and $D \in \mathbb{R}^{n\times p}$ denote the state matrix, control input matrix, and disturbance input matrix, respectively.

The goal is to design the sparse feedback controller K via solving problem (9.1).

Remark 33. *For the sake of simplicity in our notations, with a little bit abuse of notation, we show the optimal solution of problem* (9.1)*, with K, i.e., the same notation used for its corresponding optimization variable.*

Remark 34. *Since considering the stability constraint on K, generally makes problem* (9.1) *complicated (NP-hard in the case of ℓ_0 sparsity measure), we loosen such a constraint and after obtaining the K, we check that if $A + BK$ is Hurwitz or not. Since the closed-loop stability is heuristically imposed via the Frobenius term appeared in objective function of problem* (9.1)*, $A + BK$ is not necessarily Hurwitz. Thus, we hope to get a sparse stabilizing K out of a given dense F which can be seen as a drawback of our proposed method. However, in Section 7.5, our extensive numerical simulations showcase that our proposed method properly obtains a sparse stabilizing K out of a given dense F.*

9.4 Feedback Controller Sparsification via $q \in (0, 1)$ Quasi-Norms

The following theorem includes the main part of this chapter which is employed as a basis for constructing the sparsification algorithm.

Theorem 45. *Given a 4-tuple (A, B, D, F) for system (9.2), the analytic solution (sparse feedback controller) of unconstrained optimization problem (9.1) is characterized as follows:*

$$\begin{aligned}
&\textit{If}\ \ F_{ij} = 0, && \textit{then}\ \ K_{ij} = 0,\\
&\textit{If}\ \ F_{ij} \neq 0\ \&\ \gamma > \gamma_{ij}, && \textit{then}\ \ K_{ij} = 0,\\
&\textit{If}\ \ F_{ij} \neq 0\ \&\ \gamma = \gamma_{ij}, && \textit{then}\ \ K_{ij} = 0\ \ \textit{or}\ \ c(q)F_{ij},\\
&\textit{If}\ \ F_{ij} \neq 0\ \&\ \gamma < \gamma_{ij}, && \textit{then}\ \ K_{ij} = X_{ij},
\end{aligned}$$

where

$$\gamma_{ij} = \frac{\big(2(1-q)\big)^{1-q}}{(2-q)^{2-q}} |F_{ij}|^{2-q}, \tag{9.3}$$

$$c(q) = \frac{2(1-q)}{2-q}, \tag{9.4}$$

and X_{ij} denotes the solution of the following equation which has the larger absolute value:

$$X_{ij} + \gamma q \mathbf{sign}(X_{ij}) |X_{ij}|^{q-1} - F_{ij} = 0. \tag{9.5}$$

Proof. It is clear that to solve unconstrained optimization problem (9.1), we can solve it element-wise. Hence, let us consider the following scalar function:

$$f(K_{ij}) = \frac{1}{2}(K_{ij} - F_{ij})^2 + \gamma |K_{ij}|^q.$$

If $F_{ij} = 0$, then it is obvious that the $K_{ij} = 0$ would be the optimal solution. Now, assume that $F_{ij} \neq 0$. Let us define $g(K_{ij})$ as follows:

$$g(K_{ij}) := f(K_{ij}) - \frac{1}{2}F_{ij}^2 = \frac{1}{2}K_{ij}^2 - K_{ij}F_{ij} + \gamma |K_{ij}|^q.$$

We observe that $g(0) = 0$. If $\mathbf{sign}(K_{ij}) = -\mathbf{sign}(F_{ij})$, then $F_{ij} \neq 0$ implies that $K_{ij} \neq 0$ holds and subsequently

$$g(K_{ij}) = \frac{1}{2}K_{ij}^2 - K_{ij}F_{ij} + \gamma|K_{ij}|^q > 0 + 0 + 0 = g(0),$$

which contradicts the optimality of K_{ij}, $\big($i.e., $g(K_{ij}) \leq g(0)$ $\big)$. Thus, the optimal K_{ij} is either 0 or has the same sign as F_{ij}. We claim that depending on value of γ, the number of roots of function g can be 1, 2, or 3. We know that (w.l.o.g. assume that $F_{ij} > 0$)

$$\lim_{K_{ij}\to 0^+} g''(K_{ij}) = \lim_{K_{ij}\to 0^+} 1 + \gamma q(q-1){K_{ij}}^{q-2} = -\infty,$$
$$\lim_{K_{ij}\to +\infty} g''(K_{ij}) = \lim_{K_{ij}\to +\infty} 1 + \gamma q(q-1){K_{ij}}^{q-2} = 1,$$
$$g'''(K_{ij}) = \gamma q(q-1)(q-2){K_{ij}}^{q-3} > 0,$$

hold which along with intermediate value theorem, imply that $g''(K_{ij})$ is a strictly increasing function that exactly has 1 root. We prove the claim by contradiction. If function g has more than 3 roots, then sequential applying of Rolle's theorem to functions g and g' implies that function g'' has at least 2 roots which is a contradiction.

To classify the 3 possible cases, we firstly specify the crucial case in which g has 2 roots because it automatically classifies other two cases. When g has 2 roots, $g(K_{ij}) \geq 0$ holds for all values of K_{ij}. Also, in addition to 0, another optimal solution exists for which both $g(K_{ij}) = 0$ and $g'(K_{ij}) = 0$ hold. Solving such a pair of equations implies that g has 2 roots when $\gamma = \gamma_{ij}$ holds in which γ_{ij} is calculated via (9.3). Moreover, the second optimal solution would be equal to $c(q)F_{ij}$ wherein $c(q)$

is calculated via (9.4). Thus, it suffices to solve the following equations:

$$\frac{1}{2}{K_{ij}}^2 - F_{ij}K_{ij} + \gamma_{ij}|K_{ij}|^q = 0, \tag{9.6}$$

$$K_{ij} + \gamma q \mathbf{sign}(K_{ij})|K_{ij}|^{q-1} - F_{ij} = 0. \tag{9.7}$$

Multiplying (9.7) by K_{ij} and subtracting it from (9.6) we get:

$$K_{ij} = \mathbf{sign}(F_{ij})(2\gamma_{ij}(1-q))^{\frac{1}{2-q}}. \tag{9.8}$$

Combining equations (9.6) and (9.8), γ_{ij} is obtained as expressed by (9.3). Substituting the γ_{ij} derived by (9.3) in (9.8), $K_{ij} = c(q)F_{ij}$ is resulted in which $c(q)$ is calculated via (9.4). For the case that $\gamma > \gamma_{ij}$ holds, g has 1 root and the optimal solution would be $K_{ij} = 0$. In the case that $\gamma < \gamma_{ij}$ holds, g has 3 roots and the optimal solution K_{ij} lies between the two non-zero roots. To find the optimal solution K_{ij}, it suffices to consider the necessary optimality condition $g'(K_{ij}) = 0$ and solve it and choose the solution which lies between the two non-zero roots of g and has the larger absolute value. The X_{ij} is the optimal solution in this case. Thus, proof is complete. □

Remark 35. *It is noteworthy in the case that $\gamma = \gamma_{ij}$ holds, the choice of $K_{ij} = 0$ provides a sparser solution compared to the choice of $K_{ij} = c(q)F_{ij}$. However, it may lead to a poorer performance loss.*

To shed light on proof of Theorem 45, we present a geometrical interpretation via a simple example. To investigate the number of roots of function g, we define the following auxiliary functions:

$$h_1(K_{ij}) := -\frac{1}{2}K_{ij}^2 + F_{ij}K_{ij},\ h_2(K_{ij}) := \gamma|K_{ij}|^q.$$

It can easily be verified that

$$h_1(K_{ij}) + g(K_{ij}) = h_2(K_{ij}),$$

holds, i.e., $g(K_{ij}) = 0$ is satisfied if and only if $h_1(K_{ij}) = h_2(K_{ij})$ holds. It means that, to find the solutions of $g(K_{ij}) = 0$, it suffices to take a look at intersections of plots of functions h_1 and h_2. Based on such a geometrical interpretation and considering the $F_{ij} = 4$ and $q = 0.4$, in Figure 9.1, we geometrically visualize the 3 possible cases ($\gamma = 1.1\gamma_{ij} > \gamma_{ij}$, $\gamma = \gamma_{ij}$, or $\gamma = 0.9\gamma_{ij} < \gamma_{ij}$) in which function g attains 1, 2, or 3 roots, respectively. Utilizing formulas (9.3) and (9.4) and substituting $F_{ij} = 4$ and $q = 0.4$ into those formulas, we get $\gamma_{ij} = 4.8330$ and $c(q) = 0.7500$, respectively. As it is observed in Figure 9.1, when $\gamma = \gamma_{ij}$ holds, the plots of functions h_1 and h_2 are tangent to each other at $K_{ij} = c(q)F_{ij}$. It simply states the geometrical interpretation of satisfaction of $g(K_{ij}) = 0$ and $g'(K_{ij}) = 0$ that is mentioned in proof of Theorem 45. Also, the corresponding set of roots in each of those previously mentioned 3 cases are expressed as follows:

$$\begin{aligned}
&\textit{If } \ \gamma = 1.1\gamma_{ij} > \gamma_{ij}, \quad \textit{then } \ \{0\},\\
&\textit{If } \ \gamma = \gamma_{ij}, \qquad\qquad \textit{then } \ \{0, 3.0000\},\\
&\textit{If } \ \gamma = 0.9\gamma_{ij} < \gamma_{ij}, \quad \textit{then } \ \{0, 1.7225, 4.4462\}.
\end{aligned}$$

In addition, the corresponding plots of function g in each of those previously mentioned 3 cases are visualized in Figure 9.2. Moreover, the corresponding optimal

solutions K for any of those previously mentioned 3 cases are stated as follows:

$$\begin{aligned} &\textit{If } \ \gamma = 1.1\gamma_{ij} > \gamma_{ij}, \quad \textit{then } \ \{0\}, \\ &\textit{If } \ \gamma = \gamma_{ij}, \qquad\qquad \textit{then } \ \{0, 3.0000\}, \\ &\textit{If } \ \gamma = 0.9\gamma_{ij} < \gamma_{ij}, \quad \textit{then } \ \{3.1211\}. \end{aligned}$$

It is remarkable that in the case of $\gamma = 0.9\gamma_{ij} < \gamma_{ij}$, equation $g'(K_{ij}) = 0$ has 2 solutions: $K_{ij} = 0.2821$ and $K_{ij} = 3.1211$. However, as it was previously mentioned in proof of Theorem 45, the optimal solution would be the one which lies between the non-zero roots of g, i.e., 1.7225 and 4.4462. Thus, the optimal solution is $X_{ij} = 3.1211$ as it is shown in Figure 9.2 $\big($and obviously calculated via (9.5)$\big)$.

Also, it should be mentioned that $g'(K_{ij}) = 0$ has 2 non-zero solutions if and only if plots of auxiliary functions

$$h_3(K_{ij}) := F_{ij} - K_{ij} \quad \textit{and} \quad h_4(K_{ij}) := \gamma q |K_{ij}|^{q-1},$$

intersect each other at 2 points $\big($note that $h_3(K_{ij}) + g'(K_{ij}) = h_4(K_{ij})$ holds$\big)$. Equivalently, it suffices to consider the intersection of tangent line to h_4 with vertical axis $K_{ij} = 0$, namely, $(0, T_{ij})$ and impose $|F_{ij}| > T_{ij}$. It can easily be verified that $\gamma \leq \gamma_{ij}$ implies $|F_{ij}| > T_{ij}$, i.e., equivalently, function g' has 2 roots in such a case. The sketch of the proof is as follows: Firstly, T_{ij} is calculated via

$$T_{ij} = (\gamma q)^{\frac{1}{2-q}}(1-q)^{\frac{q-1}{2-q}}(2-q).$$

Then, considering (9.3), we equivalently rewrite $|F_{ij}| > T_{ij}$ as follows:

$$\gamma < \frac{|F_{ij}|^{2-q}(1-q)^{1-q}}{q(2-q)^{2-q}} = \frac{2^{q-1}}{q}\gamma_{ij}. \tag{9.9}$$

Thus, if we can show that

$$h_5(x) := 2^{x-1} - x > 0,$$

holds for all $x \in (0,1)$, then $\gamma \leq \gamma_{ij}$ implies $|F_{ij}| > T_{ij}$. We show that such a condition holds. Since $h_5'(x) = \ln(2)2^{x-1} - 1 < \ln(2)(1) - 1 < 0$ holds $\big($because $2^{x-1} \leq 1$ holds for all $x \in [0,1]\big)$, function h_5 is strictly decreasing in interval $[0,1]$. Thus, since h_5 is strictly decreasing in interval $[0,1]$, then for each $0 < x < 1$, we have $h_5(x) > h_5(1) = 0$, i.e., $h_5(x)$ is positive for all $x \in (0,1)$. Thus, $|F_{ij}| > T_{ij}$ is resulted and proof is complete. Notice that, in addition to γ's satisfying $\gamma \leq \gamma_{ij}$, for all the γ's between γ_{ij} and $\frac{2^{q-1}}{q}\gamma_{ij}$, $|F_{ij}| > T_{ij}$ is implied, i.e., equivalently, function g' has 2 roots in such cases as well as the case that $\gamma \leq \gamma_{ij}$ holds. In other words function g' has 2 roots if and only if (9.9) holds wherein $\frac{2^{q-1}}{q} > 1$ is satisfied.

Figure 9.3 visualizes the implication of $|F_{ij}| > T_{ij}$, (i.e., the case that function g' has 2 roots) from $\gamma \leq \gamma_{ij}$. As it is observed in Figure 9.3, $|F_{ij}| = 4 > T_{ij} = 2.9257$ holds which means that function g' has 2 roots 0.3461 and 3.0000 where the first one is local maximizer and the second one is both local and global minimizer. Also, in such a particular setting, (i.e., $F_{ij} = 4$ and $q = 0.4$), function g' has 2 roots if and only if $\gamma < \gamma_{ij}\frac{2^{q-1}}{q} = 4.8330 \times \frac{2^{0.4-1}}{0.4} = 4.8330 \times 1.6494 = 7.9715$ holds.

Based upon Theorem 45, given a well-performing feedback controller F, we are able to obtain the sparse feedback controller K. The summary of our proposed algorithm is stated by Algorithm 1.

Remark 36 (Time Complexity)**.** *In the worst case, the time complexity of the proposed sparsification algorithm (Algorithm 1) is equal to $\mathcal{O}(mns)$ in which s refers to the time complexity of the computation of X_{ij}. It is noteworthy that the time complexity of the method proposed by [33] is equal to $\mathcal{O}(n^3 r)$ in which r refers to the time complexity of the computation of a solution of a nonlinear equation.*

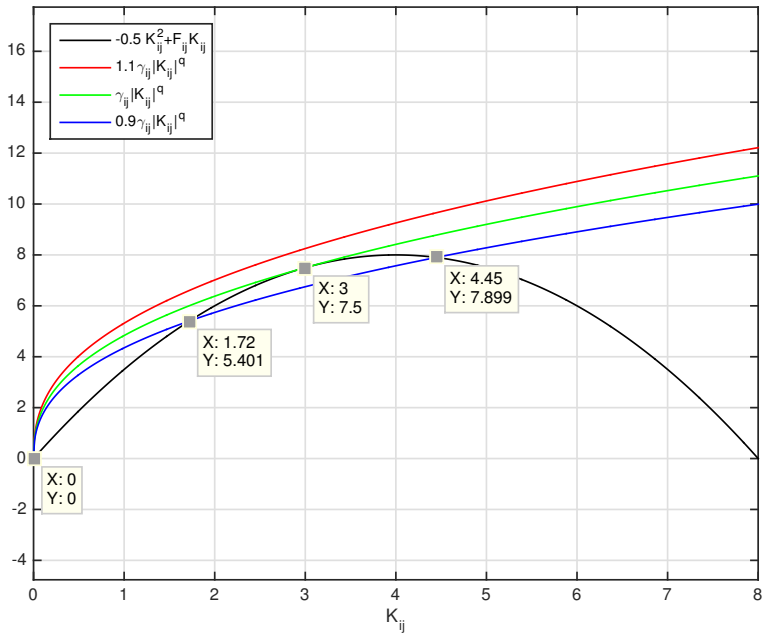


Figure 9.1: Geometrical visualization of 3 possible cases ($\gamma = 1.1\gamma_{ij} > \gamma_{ij}$, $\gamma = \gamma_{ij}$, or $\gamma = 0.9\gamma_{ij} < \gamma_{ij}$) in which function g attains 1, 2, or 3 roots, respectively, in the case of $F_{ij} = 4$ and $q = 0.4$.

9.5 Numerical Simulations

This section is divided into four subsections as follows: (i) Outperforming the Truncation Operator (Operator Associated with Cardinality Minimization); (ii) Relationship Between q and Sparsity-Performance Trade-Off Curves; (iii) Feedback Controller Sparsification for Large-Scale Systems; (iv) Network Sparsification for Large-Scale Networks.

Before proceeding to showcasing the numerical simulations, we define the following measuring quantities:

$$\sigma_D(K) := 100 \times \frac{\|K\|_0}{\|F\|_0}, \ \sigma_P(K) := 100 \times \frac{J(K) - J(F)}{J(F)},$$

in which $J(F)$ and $J(K)$ represent the quadratic performance corresponding to F

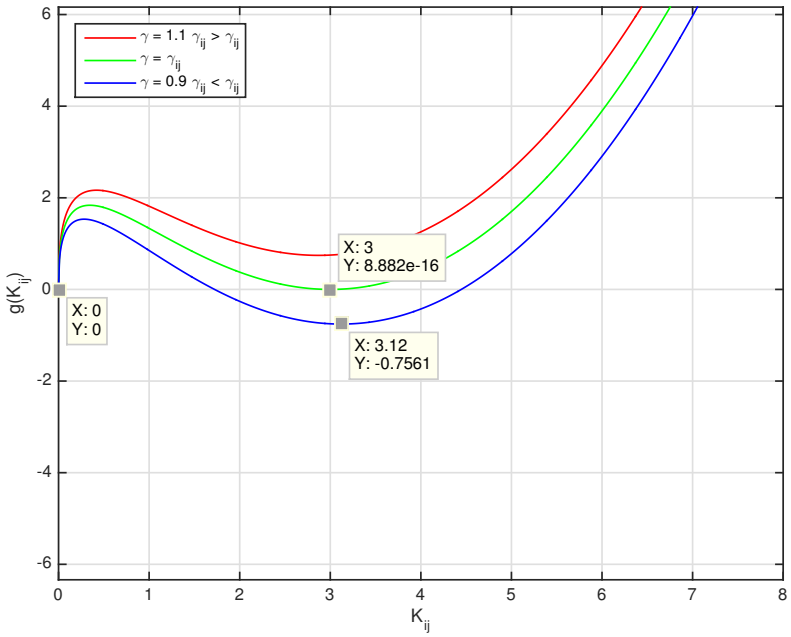


Figure 9.2: Plots of function g in 3 possible cases ($\gamma = 1.1\gamma_{ij} > \gamma_{ij}$, $\gamma = \gamma_{ij}$, or $\gamma = 0.9\gamma_{ij} < \gamma_{ij}$) in the case of $F_{ij} = 4$ and $q = 0.4$.

and K, respectively. Such values can be calculated via the following formulas:

$$J(F) = \mathbf{Tr}(D^T P D), \;\; J(K) = \mathbf{Tr}(D^T L D),$$

wherein P and L symbolize the unique positive definite solutions of the following two Lyapunov equations:

$$(A + BF)^T P + P(A + BF) = Q + F^T R F,$$
$$(A + BK)^T L + L(A + BK) = Q + K^T R K.$$

9.5.1 Outperforming the Truncation Operator (Operator Associated with Cardinality Minimization)

Prior to showing the case of outperforming the truncation operator, we mention that the truncation operator (the operator associated with cardinality minimization) is

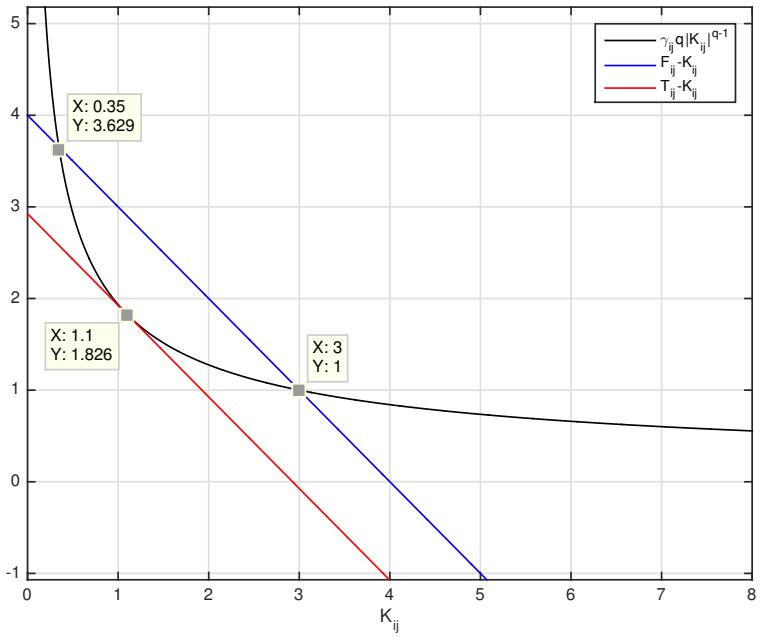


Figure 9.3: Plots of functions h_3, h_4, and the corresponding tangent line to function h_4 in the case of $\gamma = \gamma_{ij} = 4.8330$, $F_{ij} = 4$, and $q = 0.4$.

Algorithm 1: Solution to problem (9.1)

Inputs: A, B, D, F, γ, and q.
For $i = 1 : m$
  For $j = 1 : n$,
    If $F_{ij} = 0$ *then* $K_{ij} = 0$,
    Else
      Calculate γ_{ij} *via* (9.3),
      If $\gamma \geq \gamma_{ij}$ *then* $K_{ij} = 0$,
      Else
      Calculate X_{ij} *via* (9.5) *and* put $K_{ij} = X_{ij}$,
      End
    End
  End
End
Output: K.

acted as follows:

$$\textit{If } \ |F_{ij}| \leq \sqrt{2\gamma}, \quad \textit{then } \ K_{ij} = 0,$$

$$\textit{If } \ |F_{ij}| > \sqrt{2\gamma}, \quad \textit{then } \ K_{ij} = F_{ij}.$$

$J(K)$	$J(K_{\mathrm{T}})$	$\|K\|_0$	$\|K_{\mathrm{T}}\|_0$
168.7350	168.8755	64	64

Table 9.1: Performance/Sparsity quantities for K and K_{T} in the case of 10×10 randomly generated system $\big(J(F) = 151.2711$ and $\|F\|_0 = 100\big)$.

Considering a 10×10 randomly generated system A and a 10×10 randomly generated input matrix B (produced by MATLAB command *randn*), state-weight matrix $Q = I$, input-weight matrix $R = 5I$, $\gamma = 0.0532$, and $q = 0.05$, we observe that the $q \in (0, 1)$ quasi-norm based sparsification method can outperform the well-known truncation operator. The detailed performance/sparsity quantities are shown in Table 9.1. As it is observed, for the same level of sparsity, K and K_{T} attain $\sigma_P(K) = 11.5448$ % and $\sigma_P(K_{\mathrm{T}}) = 11.6376$ %, respectively. Thus, the $q \in (0, 1)$ quasi-norm based sparsification method proposes a sparsified controller which has 0.0929 % performance loss less than the one proposed by well-known truncation operator.

9.5.2 Relationship Between q and Sparsity-Performance Trade-Off Curves

The main goal of this subsection is to assess the relationship between q and sparsity-performance trade-off curves. Directed by such an attitude, let us consider a 100×100 randomly generated system A and a 100×100 randomly generated input matrix B along with state-weight matrix $Q = 4I$ and input-weight matrix $R = 2I$ and 9 equidistant values of $q \in (0, 1)$, (i.e., 0.1, 0.2, $\cdots$, 0.9), and then, visualize such trade-off curves for those varying values. Also, for each q, we consider 100 logarithmically scaled γ between $\gamma_{\min}$ and $\gamma_{\max}$ where

$$\gamma_{\min} := \underset{i,j}{\text{minimize}}\ \gamma_{ij},\ \ \gamma_{\max} := \underset{i,j}{\text{maximize}}\ \gamma_{ij}.$$

The sparsity-performance trade-off curves are visualized in Figure 9.4. The zoomed versions of such curves are visualized in Figures 9.5, 9.6, and 9.7. The more detailed illustration of data visualized in Figures 9.5, 9.6, and 9.7 is partially provided by Tables 9.2, 9.3, and 9.4, respectively. Observing the data illustrated in Tables 9.2, 9.3, and 9.4, we see that as q decreases, for a fixed level of sparsity, the obtained performance loss $\sigma_P(K)$ decreases accordingly. In other words, as value of q gets close to 0, i.e., truncation operator (operator associated with cardinality minimization), the sparse controller with better sparsity-performance balance is obtained. Also, similarly, as Figures 9.4, 9.5, 9.6, and 9.7 demonstrate, the sparsity-performance trade-off curve with lower q lies below the sparsity-performance trade-off curve with higher q, that is, the lower q we have, the better sparsity-performance balance is struck.

Remark 37. *It is worth mentioning that to obtain a sparse K, we must have $\gamma \geq \gamma_{\min}$. Also, If $\gamma > \gamma_{\max}$ holds, then $K = 0$ is resulted which is not stabilizing for an unstable system A. That is the reason we assume that γ varies from $\gamma_{\min}$ to $\gamma_{\max}$ in our numerical simulations. In addition, based on our numerical simulations, we observe that there exists a $\gamma^{critical}$ between $\gamma_{\min}$ and $\gamma_{\max}$ such that for all $\gamma \in [\gamma_{\min}, \gamma^{critical})$, we get a stabilizing K and for all $\gamma \in [\gamma^{critical}, \gamma_{\max}]$ we get a destabilizing K. Definitely, such an invalid destabilizing ones are excluded in plotting Figure 9.4.*

9.5.3 Feedback Controller Sparsification for Large-Scale Systems

To show the effectiveness of capability of our proposed method in the case of applying to the large-scale systems, we consider two classes of systems: (i) Randomly Generated Systems; (ii) Sub-Exponentially Spatially Decaying Systems.

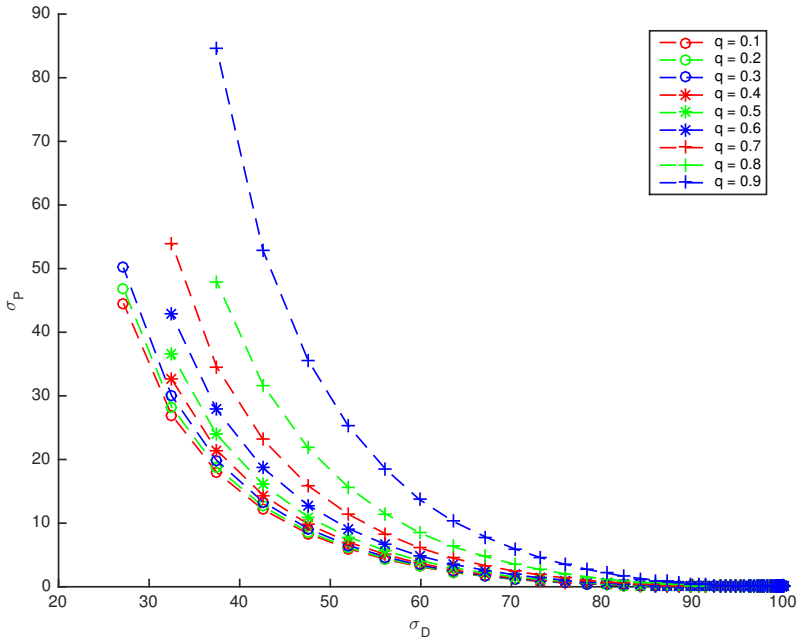


Figure 9.4: Sparsity-Performance trade-off curves for varying values of q. We set the following visualization rules: (i) $q^{\text{Red}} < q^{\text{Green}} < q^{\text{Blue}}$ (ii) $q^{\text{Circle}} < q^{\text{Asterisk}} < q^{\text{Plus}}$, wherein each superscript refers to the corresponding color or sign associated with q. (For the 100×100 randomly generated system).

q	$\sigma_P(K)\big(\sigma_D(K) = 37.45\big)$	$\sigma_P(K)\big(\sigma_D(K) = 42.57\big)$
0.1	18.0053	12.1427
0.2	18.7679	12.6295
0.3	19.8644	13.3451
0.4	21.4750	14.4124
0.5	23.9205	16.0469
0.6	27.8241	18.6587
0.7	34.5709	23.1265
0.8	47.9838	31.7213
0.9	84.5283	52.8658

Table 9.2: Performance quantities for fixed values of sparsity quantities around $\sigma_D = 40$ and varying values of q. (For the 100×100 randomly generated system).

Randomly Generated Systems

Let us consider a $10,000 \times 10,000$ randomly generated system A and a $10,000 \times 10,000$ randomly generated input matrix B. Also, for state-weight and input-weight matrices, we choose $Q = I$ and $R = 5I$, respectively. Furthermore, $\gamma = 7.3410 \times 10^{-5}$ and

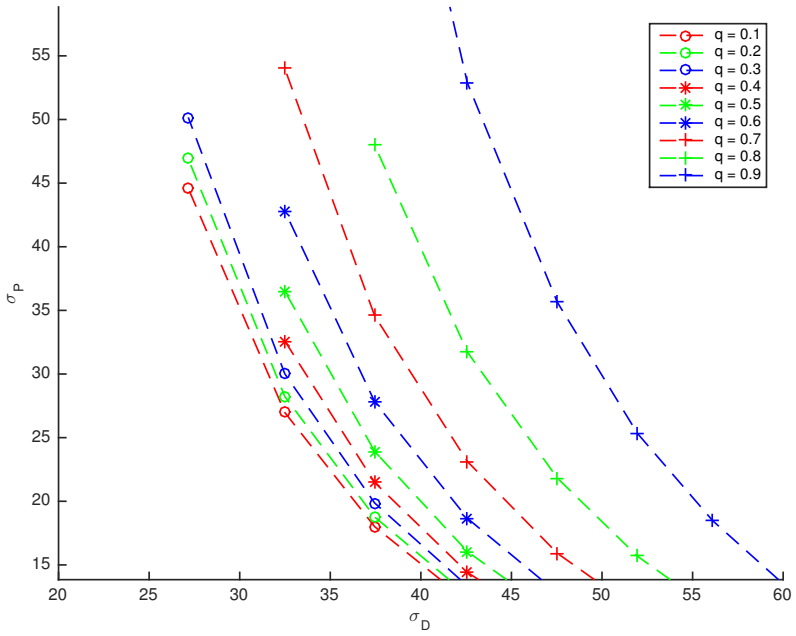


Figure 9.5: Sparsity-Performance trade-off curves around $\sigma_D = 40$ for varying values of q. We set the following visualization rules: (i) $q^{\text{Red}} < q^{\text{Green}} < q^{\text{Blue}}$ (ii) $q^{\text{Circle}} < q^{\text{Asterisk}} < q^{\text{Plus}}$, wherein each superscript refers to the corresponding color or sign associated with q. (For the 100×100 randomly generated system).

q	$\sigma_P(K)\big(\sigma_D(K) = 59.82\big)$	$\sigma_P(K)\big(\sigma_D(K) = 63.52\big)$
0.1	3.1581	2.2763
0.2	3.2606	2.3469
0.3	3.4292	2.4676
0.4	3.7013	2.6670
0.5	4.1416	2.9954
0.6	4.8703	3.5459
0.7	6.1326	4.5078
0.8	8.5056	6.3239
0.9	13.7439	10.3145

Table 9.3: Performance quantities for fixed values of sparsity quantities around $\sigma_D = 60$ and varying values of q. (For the 100×100 randomly generated system).

$q = 0.005$ are assumed. The performance/sparsity quantities of the corresponding sparse feedback controller is illustrated in Table 9.5. According to the data presented in Table 9.5, 68.228982 % of elements of feedback controller K is sparsified while having the 76.0190 % performance loss.

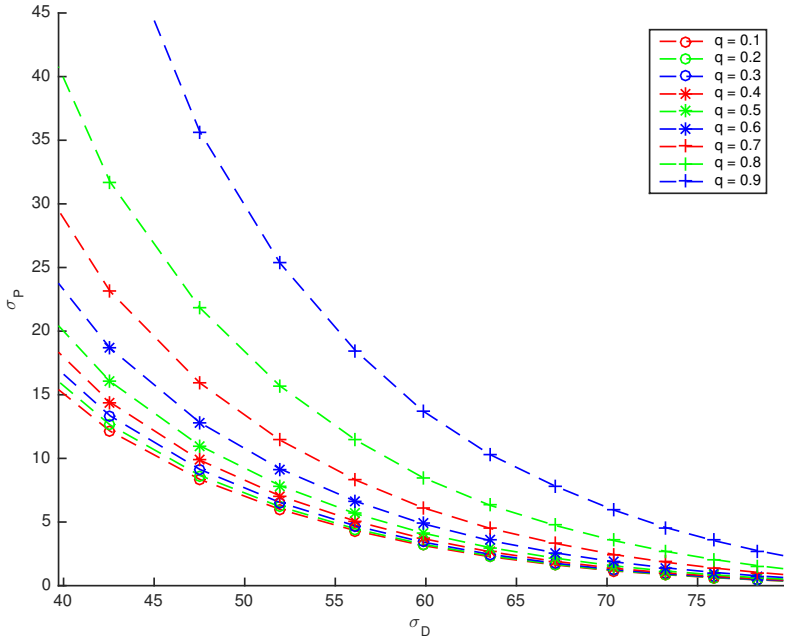


Figure 9.6: Sparsity-Performance trade-off curves around $\sigma_D = 60$ for varying values of q. We set the following visualization rules: (i) $q^{\text{Red}} < q^{\text{Green}} < q^{\text{Blue}}$ (ii) $q^{\text{Circle}} < q^{\text{Asterisk}} < q^{\text{Plus}}$, wherein each superscript refers to the corresponding color or sign associated with q. (For the 100×100 randomly generated system).

q	$\sigma_P(K)\big(\sigma_D(K) = 78.31\big)$	$\sigma_P(K)\big(\sigma_D(K) = 80.54\big)$
0.1	0.4539	0.3263
0.2	0.4670	0.3358
0.3	0.4930	0.3552
0.4	0.5404	0.3909
0.5	0.6239	0.4549
0.6	0.7720	0.5695
0.7	1.0428	0.7815
0.8	1.5722	1.1998
0.9	2.7503	2.1380

Table 9.4: Performance quantities for fixed values of sparsity quantities around $\sigma_D = 80$ and varying values of q. (For the 100×100 randomly generated system).

The sparsity pattern of the first 100×100 sub-block of K is visualized in Figure 9.8.

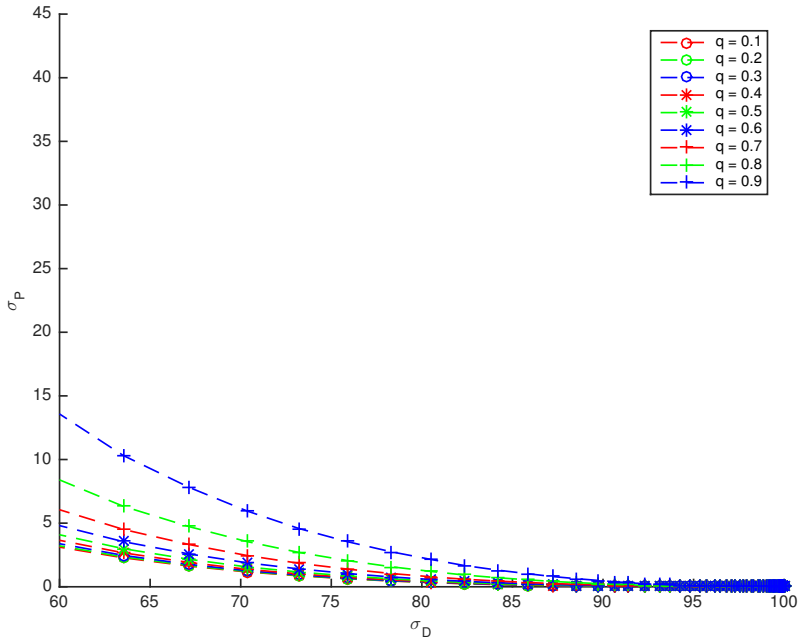


Figure 9.7: Sparsity-Performance trade-off curves around $\sigma_D = 80$ for varying values of q. We set the following visualization rules: (i) $q^{\text{Red}} < q^{\text{Green}} < q^{\text{Blue}}$ (ii) $q^{\text{Circle}} < q^{\text{Asterisk}} < q^{\text{Plus}}$, wherein each superscript refers to the corresponding color or sign associated with q. (For the 100×100 randomly generated system).

$\sigma_P(K)$	$\sigma_D(K)$	$J(K)$	$\|K\|_0$
76.0190	31.771018	6.2137×10^6	$31,771,018$

Table 9.5: Performance/Sparsity quantities for K in the case of $10,000 \times 10,000$ randomly generated system $\big(J(F) = 3.5302 \times 10^6$ and $\|F\|_0 = 10^8\big)$.

Sub-Exponentially Spatially Decaying Systems

Likewise the previous case, let us consider a $10,000 \times 10,000$ sub-exponentially spatially decaying system A and a $10,000 \times 10,000$ input matrix B. It is noteworthy that the ij^{th} element of sub-exponentially spatially decaying system M is defined as follows:

$$M_{ij} = c\mathcal{M}_{ij}e^{-\alpha|i-j|^\beta},$$

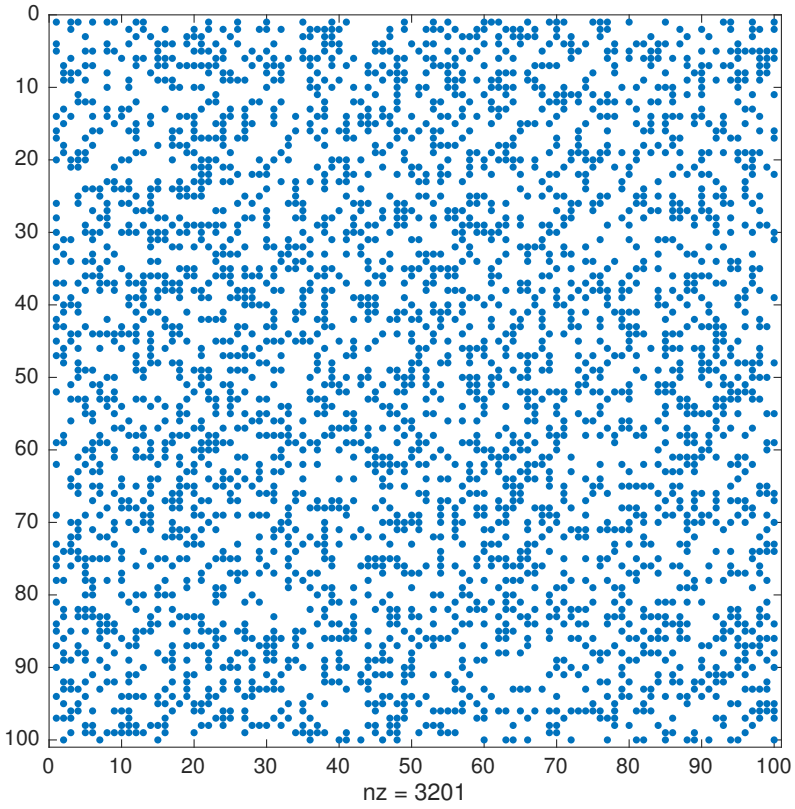


Figure 9.8: Sparsity pattern of the first 100×100 sub-block of K in the case of $10,000 \times 10,000$ randomly generated system (Blue dots represent the non-zero elements and "nz" denotes the number of non-zero elements of the first 100×100 sub-block of K).

where $\mathcal{M}_{ij} \in \mathcal{N}(0,1)$, c is a fixed positive scalar, α determines the band-width of matrix M, and β specifies the rate of spatially decaying in such a matrix. For state-weight and input-weight matrices, we choose $Q = 5I$ and $R = I$, respectively. In addition, $\alpha_A = \alpha_B = 0.25$, $\beta_A = \beta_B = 0.5$, $c = 10$, $\gamma = 3.2052 \times 10^{-6}$, and $q = 0.005$ are assumed. The performance/sparsity quantities of the corresponding sparse feedback controller is illustrated in Table 9.6. On the basis of the data presented in Table 9.5, 89.552029 % of elements of feedback controller K is sparsified while having the 0.0079 % performance loss.

The sparsity pattern of K is visualized in Figure 9.9.

$\sigma_P(K)$	$\sigma_D(K)$	$J(K)$	$\|K\|_0$
0.0079	10.447971	8.4083×10^5	$10,447,971$

Table 9.6: Performance/Sparsity quantities for K in the case of $10,000 \times 10,000$ sub-exponentially spatially decaying system $\big(J(F) = 8.4077 \times 10^5$ and $\|F\|_0 = 10^8\big)$.

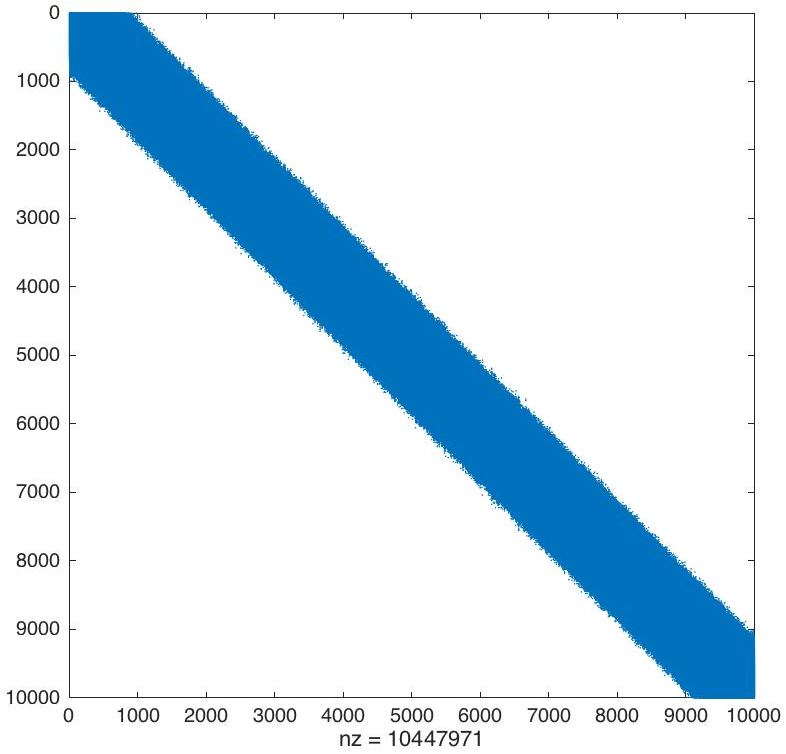


Figure 9.9: Sparsity pattern of K in the case of $10,000 \times 10,000$ sub-exponentially spatially decaying system (Blue dots represent the non-zero elements and "nz" denotes the number of non-zero elements of K).

9.5.4 Network Sparsification for Large-Scale Networks

One of advantages of our proposed method is that, by a slight modification, it can easily be translated to a network sparsification method which is applicable to the large-scale networks. Indeed, to sparsify a network, (i.e., Laplacian $\mathcal{L}$), it suffices to sparsify the corresponding adjacency matrix, (i.e., $\mathcal{A}$) of the corresponding underlying graph. In other words, to obtain the sparsified network $\hat{\mathcal{L}}$ out of a given network $\mathcal{L}$,

$\sigma_P(\hat{\mathcal{L}})$	$\sigma_D(\hat{\mathcal{A}})$	$J(\hat{\mathcal{L}})$	$\|\hat{\mathcal{A}}\|_0$
70.7598	42.5389	0.8526	$424,964$

Table 9.7: Performance/Sparsity quantities for $\hat{\mathcal{L}}$ in the case of $1,000 \times 1,000$ randomly generated undirected network $\big(J(\mathcal{L}) = 0.4993$ and $\|\mathcal{A}\|_0 = 999,000\big)$.

we consider the following modified version of problem (9.1):

$$\underset{\hat{\mathcal{A}}}{\text{minimize}} \ \frac{1}{2}\|\hat{\mathcal{A}} - \mathcal{A}\|_F^2 + \gamma\|\hat{\mathcal{A}}\|_q^q, \tag{9.10}$$

in which $\hat{\mathcal{A}}$ and $\mathcal{A}$ denote the corresponding adjacency matrices of Laplacians $\hat{\mathcal{L}}$ and $\mathcal{L}$, respectively. Let us assume that a $1,000 \times 1,000$ randomly generated undirected network $\mathcal{L}$ is given. Moreover, $\gamma = 0.5822$ and $q = 0.01$ are chosen. Applying the proposed sparsification method, the performance/sparsity quantities associated with the obtained $\hat{\mathcal{L}}$ are illustrated in Table 9.7.

As it is observed, at the expense of 70.7598 % performance loss, 57.4611 % of links is sparsified. The graph representations of subgraphs consisting of the first 50 nodes of $\mathcal{L}$ and $\hat{\mathcal{L}}$ and their corresponding links are visualized in Figure 9.10.

Comparing Figures 9.10(a) and 9.10(b), verifies that the subgraph corresponding to $\hat{\mathcal{L}}$ has less link than the one corresponding to $\mathcal{L}$.

Remark 38. *It is noteworthy that the sparsity/performance quantities for the case of networks are defined as follows:*

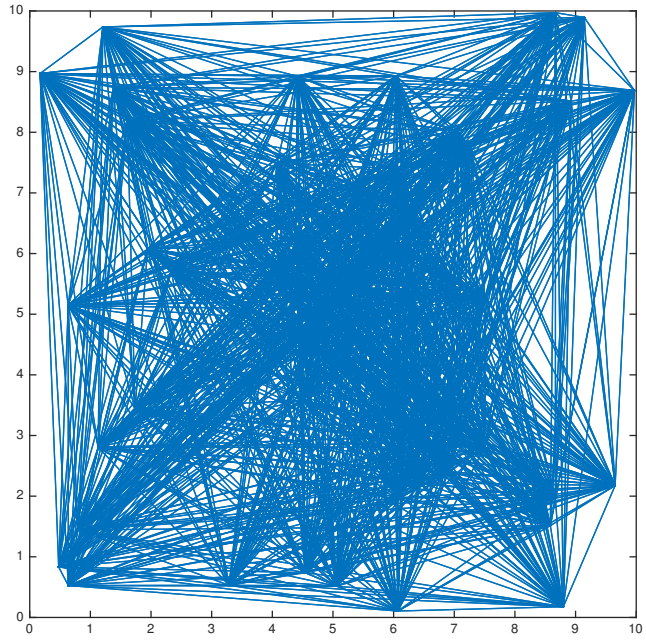
$$\sigma_D(\hat{\mathcal{A}}) := 100 \times \frac{\|\hat{\mathcal{A}}\|_0}{\|\mathcal{A}\|_0}, \ \sigma_P(\hat{\mathcal{L}}) := 100 \times \frac{J(\hat{\mathcal{L}}) - J(\mathcal{L})}{J(\mathcal{L})},$$

wherein $J(\mathcal{L})$ and $J(\hat{\mathcal{L}})$ are calculated via the following formulas:

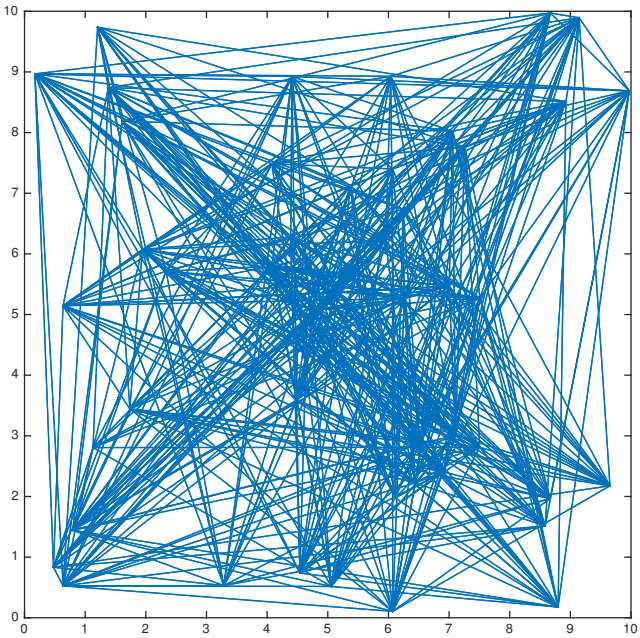
$$J(\mathcal{L}) = \frac{1}{2}\mathbf{Tr}\big((\mathcal{L} + \frac{\mathbf{1}\mathbf{1}^T}{n})^{-1} - \frac{\mathbf{1}\mathbf{1}^T}{n}\big), \ J(\hat{\mathcal{L}}) = \frac{1}{2}\mathbf{Tr}\big((\hat{\mathcal{L}} + \frac{\mathbf{1}\mathbf{1}^T}{n})^{-1} - \frac{\mathbf{1}\mathbf{1}^T}{n}\big).$$

9.6 Conclusion

On the basis of notion of $q \in (0, 1)$ quasi-norms, a sparsification procedure is presented which is importantly applicable to large-scale systems. Sparsity-performance trade-off related to such sparsified feedback controllers are assessed via large-scale and medium-size systems. It is observed that in some cases, $q \in (0, 1)$ quasi-norm based method can outperform the truncation based cardinality minimization. Also, as an interesting observation, it is seen that, as q decreases, the corresponding sparsity-performance trade-off behavior is improved. However, our proposed procedure has two main drawbacks: (i) because of its heuristic nature the closed-loop stability is not guaranteed (ii) the sparsification rate is limited.A remaining problem which can be regarded as a future direction, is the improvement on the sparsification rate.



(a)



(b)

Figure 9.10: (a) Graph representation of subgraph consisting of the first 50 nodes of $\mathcal{L}$ and its corresponding links (b) Graph representation of subgraph consisting of the first 50 nodes of $\hat{\mathcal{L}}$ and its corresponding links. (For the $1,000 \times 1,000$ randomly generated undirected network).

Chapter 10

Conclusions and Future Directions

After formulating the sparsity-promoting optimal controller design problems for the class of LTI systems, various optimization tools are employed to tackle such non-convex and generally NP-hard problems. Some of such optimization tools are as follows: bi-linear rank penalty technique, ℓ_1-regularization, SDP, nonlinear optimization, and quasi-norm minimization. Particularly, in the case of feedback controller sparsification for large-scale systems, two helpful ideas are utilized: (i) non-fragility (ii) quasi-norms. Merging the ideas from spatial sparsity and temporal sparsity, self-triggered sparse optimal control (SSOC) design is proposed in which sparsity is improved both in time and space simultaneously. Throughout the dissertation, extensive numerical simulations confirm the following main observations: a fundamental trade-off exists between density level of feedback controller and its corresponding performance loss, parametric uncertainty and time-delay have negative impacts on sparsification process, a fundamental trade-off exists between non-fragility and sparsity level of feedback controller, and in the case of sparsification via quasi-norms, the smaller q, the sparser solution is achieved. Some problems still remain open: improving the quality of density-performance trade-off curves via decreasing the sub-optimality level of our proposed solutions, improving the limited sparsification rate in

the case of large-scale systems, developing the computationally distributed version of our proposed methods, and applying the proposed methods to real-world applications in an experimental way.

Bibliography

[1] Lin, F., Fardad, M. & Jovanovic, M. R. Design of optimal sparse feedback gains via the alternating direction method of multipliers. *IEEE Transactions on Automatic Control* **58**, 2426–2431 (2013).

[2] Rotkowitz, M. & Lall, S. Decentralized control information structures preserved under feedback. In *Proceedings of the IEEE 41st Conference on Decision and Control*, vol. 1, 569–575 (2002).

[3] Rotkowitz, M. & Lall, S. A characterization of convex problems in decentralized control. *IEEE Transactions on Automatic Control* **50**, 1984–1996 (2005).

[4] Fardad, M., Lin, F. & Jovanovic, M. R. Sparsity-promoting optimal control for a class of distributed systems. In *Proceedings of the American Control Conference*, 2050–2055 (2011).

[5] Dhingra, N., Lin, F., Fardad, M. & Jovanović, M. R. On identifying sparse representations of consensus networks. *IFAC Proceedings Volumes* **45**, 305–310 (2012).

[6] Lin, F., Fardad, M. & Jovanović, M. R. Sparse feedback synthesis via the Alternating Direction Method of Multipliers. In *Proceedings of the American Control Conference*, 4765–4770 (2012).

[7] Motee, N. & Sun, Q. Measuring sparsity in spatially interconnected systems. In *IEEE 52nd Annual Conference on Decision and Control*, 1520–1525 (2013).

[8] Jovanovic, M. R. & Lin, F. Sparse quadratic regulator. In *European Control Conference (ECC)*, 1047–1052 (2013).

[9] Wytock, M. & Kolter, J. Z. A fast algorithm for sparse controller design. *arXiv preprint arXiv:1312.4892* (2013).

[10] Polyak, B., Khlebnikov, M. & Shcherbakov, P. An LMI approach to structured sparse feedback design in linear control systems. In *Proceeding of the European Control Conference*, 833–838 (2013).

[11] Nagahara, M., Quevedo, D. E. & Ostergaard, J. Sparse packetized predictive control for networked control over erasure channels. *IEEE Transactions on Automatic Control* **59**, 1899–1905 (2014).

[12] Kong, H., Goodwin, G. C. & Seron, M. M. A cost-effective sparse communication strategy for networked linear control systems: an svd-based approach. *International Journal of Robust and Nonlinear Control* **25**, 2223–2240 (2015).

[13] Wang, Y., Lopez, J. & Sznaier, M. Sparse static output feedback controller design via convex optimization. In *IEEE 53rd Annual Conference on Decision and Control (CDC)*, 376–381 (2014).

[14] Motee, N. & Sun, Q. Sparsity measures for spatially decaying systems. In *Proceedings of the American Control Conference*, 5459–5464 (2014).

[15] Arastoo, R., Motee, N. & Kothare, M. V. Optimal sparse output feedback control design: a rank constrained optimization approach. *arXiv preprint arXiv:1412.8236* (2014).

[16] Wu, X. & Jovanović, M. R. Sparsity-promoting optimal control of consensus and synchronization networks. In *American Control Conference*, 2936–2941 (2014).

[17] Fardad, M., Lin, F. & Jovanović, M. R. Design of optimal sparse interconnection graphs for synchronization of oscillator networks. *IEEE Transactions on Automatic Control* **59**, 2457–2462 (2014).

[18] Fardad, M. & Jovanovic, M. R. On the design of optimal structured and sparse feedback gains via sequential convex programming. In *American Control Conference*, 2426–2431 (2014).

[19] Sabău, Ş. & Martins, N. C. Youla-like parametrizations subject to qi subspace constraints. *IEEE transactions on Automatic Control* **59**, 1411–1422 (2014).

[20] Bahavarnia, M. Sparse linear-quadratic feedback design using affine approximation. *arXiv preprint arXiv:1507.08592* (2015).

[21] Arastoo, R., Bahavarnia, M., Kothare, M. & Motee, N. Output feedback controller sparsification via $\mathcal{H}_2$-approximation. *IFAC-PapersOnLine* **48**, 112–117 (2015).

[22] Wang, Y., Lopez, J. & Sznaier, M. A convex optimization approach to synthesizing sparse dynamic output feedback controllers. *IFAC-PapersOnLine* **48**, 95–100 (2015).

[23] Siami, M. & Motee, N. Network sparsification with guaranteed systemic performance measures. *IFAC-PapersOnLine* **48**, 246–251 (2015).

[24] Lian, F., Duel-Hallen, A. & Chakrabortty, A. Sparsity-constrained games and distributed optimization with applications to wide-area control of power systems. *arXiv preprint arXiv:1605.00620* (2016).

[25] Hassan-Moghaddam, S., Dhingra, N. K. & Jovanović, M. R. Topology identification of undirected consensus networks via sparse inverse covariance estimation. In *IEEE 55th Conference on Decision and Control (CDC)*, 4624–4629 (2016).

[26] Bahavarnia, M. & Motee, N. Periodic time-triggered sparse linear quadratic controller design. In *54th Annual Allerton Conference on Communication, Control, and Computing (Allerton)*, 1060–1067 (2016).

[27] Arastoo, R., Bahavarnia, M., Kothare, M. V. & Motee, N. Closed-loop feedback sparsification under parametric uncertainties. In *IEEE 55th Conference on Decision and Control (CDC)*, 123–128 (2016).

[28] Arastoo, R., GhaedSharaf, Y., Kothare, M. V. & Motee, N. Optimal state feedback controllers with strict row sparsity constraints. In *American Control Conference (ACC)*, 1948–1953 (2016).

[29] Jovanović, M. R. & Dhingra, N. K. Controller architectures: Tradeoffs between performance and structure. *European Journal of Control* **30**, 76–91 (2016).

[30] Motee, N. & Sun, Q. Sparsity and spatial localization measures for spatially distributed systems. *SIAM Journal on Control and Optimization* **55**, 200–235 (2017).

[31] Dhingra, N. K. & Jovanović, M. R. A method of multipliers algorithm for sparsity-promoting optimal control. In *American Control Conference (ACC)*, 1942–1947 (2016).

[32] Bahavarnia, M. & Motee, N. Sparse memoryless lqr design for uncertain linear time-delay systems. *IFAC-PapersOnLine* **50**, 10395–10400 (2017).

[33] Bahavarnia, M., Somarakis, C. & Motee, N. State feedback controller sparsification via a notion of non-fragility. In *IEEE 56th Annual Conference on Decision and Control (CDC)*, 4205–4210 (2017).

[34] Bahavarnia, M., Tabuada, P., Somarakis, C. & Motee, N. Improving sparsity in time and space via self-triggered sparse optimal controllers. In *IEEE 56th Annual Conference on Decision and Control (CDC)*, 4199–4204 (2017).

[35] Bahavarnia, M. & Motee, N. Row-column sparse linear quadratic controller design via bi-linear rank penalty technique and non-fragility notion. In *25th Mediterranean Conference on Control and Automation (MED)*, 1165–1169 (2017).

[36] Martensson, K. & Rantzer, A. A scalable method for continuous-time distributed control synthesis. In *American Control Conference (ACC)*, 6308–6313 (2012).

[37] Chizeck, H. J., Willsky, A. S. & Castanon, D. Discrete-time markovian-jump linear quadratic optimal control. *International Journal of Control* **43**, 213–231 (1986).

[38] Lavaei, J. Optimal decentralized control problem as a rank-constrained optimization. In *51st Annual Allerton Conference on Communication, Control, and Computing (Allerton)*, 39–45 (2013).

[39] Fazelnia, G., Madani, R. & Lavaei, J. Convex relaxation for optimal distributed control problem. In *IEEE 53rd Conference on Decision and Control*, 896–903 (2014).

[40] Fattahi, S., Fazelnia, G. & Lavaei, J. Transformation of optimal centralized controllers into near-global static distributed controllers. In *IEEE 54th Conference on Decision and Control (CDC)*, 4915–4922 (2015).

[41] Bahavarnia, M. Feedback controller sparsification via quasi-norms. *Submitted to IEEE 57th Annual Conference on Decision and Control (CDC)* (2018).

[42] Motee, N. & Jadbabaie, A. Optimal control of spatially distributed systems. *IEEE Transactions on Automatic Control* **53**, 1616–1629 (2008).

[43] Velasco, M., Fuertes, J. & Marti, P. The self triggered task model for real-time control systems. In *Work-in-Progress Session of the 24th IEEE Real-Time Systems Symposium (RTSS03)*, vol. 384 (2003).

[44] Tabuada, P. Event-triggered real-time scheduling of stabilizing control tasks. *IEEE Transactions on Automatic Control* **52**, 1680–1685 (2007).

[45] Lemmon, M., Chantem, T., Hu, X. S. & Zyskowski, M. On self-triggered full-information $\mathcal{H}_\infty$ controllers. In *International Workshop on Hybrid Systems: Computation and Control*, 371–384 (2007).

[46] Wang, X. & Lemmon, M. State based self-triggered feedback control systems with $\mathcal{L}_2$ stability. In *17th IFAC world congress* (2008).

[47] Anta, A. & Tabuada, P. Self-triggered stabilization of homogeneous control systems. In *American Control Conference*, 4129–4134 (2008).

[48] Mazo, M. & Tabuada, P. On event-triggered and self-triggered control over sensor/actuator networks. In *IEEE 47th Conference on Decision and Control (CDC)*, 435–440 (2008).

[49] Mazo, M., Anta, A. & Tabuada, P. On self-triggered control for linear systems: Guarantees and complexity. In *European Control Conference (ECC)*, 3767–3772 (2009).

[50] Mazo, M. & Tabuada, P. Input-to-state stability of self-triggered control systems. In *Proceedings of the IEEE 48th Conference on Decision and Control held jointly with the 28th Chinese Control Conference. CDC/CCC*, 928–933 (2009).

[51] Wang, X. & Lemmon, M. D. Self-triggered feedback control systems with finite-gain stability. *IEEE Transactions on Automatic Control* **54**, 452–467 (2009).

[52] Anta, A. & Tabuada, P. To sample or not to sample: Self-triggered control for nonlinear systems. *IEEE Transactions on Automatic Control* **55**, 2030–2042 (2010).

[53] Camacho, A. *et al.* Self-triggered networked control systems: An experimental case study. In *IEEE International Conference on Industrial Technology (ICIT)*, 123–128 (2010).

[54] Mazo, M., Anta, A. & Tabuada, P. An iss self-triggered implementation of linear controllers. *Automatica* **46**, 1310–1314 (2010).

[55] Chen, T. & Francis, B. A. *Optimal sampled-data control systems* (Springer Science & Business Media, 2012).

[56] Heemels, W., Johansson, K. H. & Tabuada, P. An introduction to event-triggered and self-triggered control. In *CDC*, 3270–3285 (2012).

[57] Durand, S., Guerrero-Castellanos, J.-F. & Lozano-Leal, R. Self-triggered control for the stabilization of linear systems. In *CCE*, 1–6 (2012).

[58] Nowzari, C. & Cortés, J. Self-triggered coordination of robotic networks for optimal deployment. *Automatica* **48**, 1077–1087 (2012).

[59] Gommans, T., Antunes, D., Donkers, T., Tabuada, P. & Heemels, M. Self-triggered linear quadratic control. *Automatica* **50**, 1279–1287 (2014).

[60] Brunner, F., Heemels, W. & Allgower, F. Robust self-triggered mpc for constrained linear systems. In *European Control Conference (ECC)*, 472–477 (2014).

[61] Santos, C., Mazo, M. & Espinosa, F. Adaptive self-triggered control of a remotely operated p3-dx robot: Simulation and experimentation. *Robotics and Autonomous Systems* **62**, 847–854 (2014).

[62] Souza, M., Deaecto, G. S., Geromel, J. C. & Daafouz, J. Self-triggered linear quadratic networked control. *Optimal Control Applications and Methods* **35**, 524–538 (2014).

[63] Almeida, J., Silvestre, C. & Pascoal, A. M. Self-triggered output feedback control of linear plants in the presence of unknown disturbances. *IEEE Transactions on Automatic Control* **59**, 3040–3045 (2014).

[64] Antunes, D. & Heemels, W. Rollout event-triggered control: Beyond periodic control performance. *IEEE Transactions on Automatic Control* **59**, 3296–3311 (2014).

[65] Nowzari, C. & Cortés, J. Self-triggered and team-triggered control of networked cyber-physical systems. *Event-Based Control and Signal Processing* (2015).

[66] Bamieh, B., Paganini, F. & Dahleh, M. A. Distributed control of spatially invariant systems. *IEEE Transactions on Automatic Control* **47**, 1091–1107 (2002).

[67] Bamieh, B. & Voulgaris, P. G. A convex characterization of distributed control problems in spatially invariant systems with communication constraints. *Systems & Control Letters* **54**, 575–583 (2005).

[68] Lin, F., Fardad, M. & Jovanović, M. R. Design of optimal sparse feedback gains via the Alternating Direction Method of Multipliers. *IEEE Transactions on Automatic Control* **58**, 2426–2431 (2013).

[69] Bernstein, D. S., Haddad, W. M. & Nett, C. N. Minimal complexity control law synthesis, part 2: Problem solution via $\mathcal{H}_2/\mathcal{H}_\infty$ optimal static output feedback. In *American Control Conference*, 2506–2511 (1989).

[70] Leibfritz, F. An lmi-based algorithm for designing suboptimal static $\mathcal{H}_2/\mathcal{H}_\infty$ output feedback controllers. *SIAM Journal on Control and Optimization* **39**, 1711–1735 (2001).

[71] Peaucelle, D. & Arzelier, D. An iterative method for mixed $\mathcal{H}_2/\mathcal{H}_\infty$ synthesis via static output-feedback. In *IEEE Conference on Decision and Control*, 3464–3469 (2001).

[72] Gao, Y. & Sun, D. A majorized penalty approach for calibrating rank constrained correlation matrix problems. *Preprint available at* ***http://www.math.nus.edu.sg/£\sim£matsundf/MajorPen.pdf*** (2010).

[73] Mishra, B., Meyer, G., Bonnabel, S. & Sepulchre, R. Fixed-rank matrix factorizations and riemannian low-rank optimization. *CoRR* **abs/1209.0430** (2012).

[74] Zhou, G., Huang, W., Gallivan, K. A., Van Dooren, P. & Absil, P. A. Rank-constrained optimization: A riemannian manifold approach. Tech. Rep. UCL-INMA-2015.02, U.C.Louvain (2015).

[75] Candes, E., Wakin, M. & Boyd, S. Enhancing sparsity by reweighted ℓ_1 minimization. *Journal of Fourier Analysis and Applications* **14**, 877–905 (2008).

[76] Boyd, S., Parikh, N., Chu, E., Peleato, B. & Eckstein, J. Distributed optimization and statistical learning via the Alternating Direction Method of Multipliers. *Foundations and Trends in Machine Learning* **3**, 1–124 (2011).

[77] Delgado, R. A., Agüero, J. C. & Goodwin, G. C. A rank-constrained optimization approach: Application to factor analysis. *IFAC Proceedings Volumes* **47**, 10373–10378 (2014).

[78] Gorski, J., Pfeuffer, F. & Klamroth, K. Biconvex sets and optimization with biconvex functions: a survey and extensions. *Mathematical Methods of Operations Research* **66**, 373–407 (2007).

[79] Gadewadikar, J., Lewis, F. L., Subbarao, K., Peng, K. & Chen, B. M. H-infinity static output-feedback control for rotorcraft. *Journal of Intelligent and Robotic Systems* **54**, 629–646 (2009).

[80] Wu, X., Dörfler, F. & Jovanović, M. R. Input-output analysis and decentralized optimal control of inter-area oscillations in power systems. *IEEE Transactions on Power Systems* **31**, 2434–2444 (2016).

[81] Bamieh, B., Pearson, J. B., Francis, B. A. & Tannenbaum, A. A lifting technique for linear periodic systems with applications to sampled-data control. *Systems & Control Letters* **17**, 79–88 (1991).

[82] Bamieh, B., Pearson Jr, J. B. *et al.* A general framework for linear periodic systems with applications to $\mathcal{H}^\infty$/sampled-data control. *IEEE Transactions on Automatic Control,* **37**, 418–435 (1992).

[83] Chen, T., Francis, B. *et al.* $\mathcal{H}_2$-optimal sampled-data control. *IEEE Transactions on Automatic Control* **36**, 387–397 (1991).

[84] Sukumar, S. & Chatterjee, D. A jammers perspective of reachability and lq optimal control. *Automatica* **70**, 295–302 (2016).

[85] Wang, X. & Lemmon, M. D. Self-triggering under state-independent disturbances. *IEEE Transactions on Automatic Control* **55**, 1494–1500 (2010).

[86] Kalbat, A., Madani, R., Fazelnia, G. & Lavaei, J. Efficient convex relaxation for stochastic optimal distributed control problem. In *52nd Annual Allerton Conference on Communication, Control, and Computing*, 589–596 (2014).

[87] Wang, Y.-S. & Matni, N. Localized lqg optimal control for large-scale systems. In *American Control Conference (ACC)*, 1954–1961 (2016).

[88] Wang, Y. S., Matni, N. & Doyle, J. C. Localized lqr control with actuator regularization. In *American Control Conference (ACC)*, 5205–5212 (2016).

[89] Fattahi, S. & Lavaei, J. Theoretical guarantees for the design of near globally optimal static distributed controllers. In *54th Annual Allerton Conference on Communication, Control, and Computing (Allerton)*, 582–589 (2016).

[90] Xie, L., Fu, M. & de Souza, C. E. $\mathcal{H}_\infty$ control and quadratic stabilization of systems with parameter uncertainty via output feedback. *IEEE Transactions on Automatic Control* **37**, 1253–1256 (1992).

[91] Xie, L. & de Souza, C. E. Robust $\mathcal{H}\infty$ control for linear systems with norm-bounded time-varying uncertainty. *IEEE Transactions on Automatic Control* **37**, 1188–1191 (1992).

[92] Petersen, I. R. A stabilization algorithm for a class of uncertain linear systems. *Syst. Control Lett.* **8**, 351–357 (1987).

[93] Petersen, I. R. Stabilization of an uncertain linear system in which uncertain

parameters enter into the input matrix. *SIAM J. Control Optim.* **26**, 1257–1264 (1988).

[94] Khargonekar, P., Petersen, I. & Zhou, K. Robust stabilization of uncertain linear systems: quadratic stabilizability and $\mathcal{H}_\infty$; control theory. *IEEE Transactions on Automatic Control* **35**, 356–361 (1990).

[95] Yu, H. & Lau, V. K. N. Rank-constrained Schur-convex optimization with multiple trace/log-det constraints. *IEEE Transacation on Signal Processing* **59**, 304–314 (2011).

[96] Shalev-Shwartz, S., Gonen, A. & Shamir, O. Large-scale convex minimization with a low-rank constraint. *CoRR* **abs/1106.1622** (2011).

[97] Kulkarni, V. V. *et al.* Gene regulatory network modeling using literature curated and high throughput data. *Systems and Synthetic Biology* **6**, 69–77 (2012).

[98] Eller, D., Aggarwal, J. & Banks, H. Optimal control of linear time-delay systems. *IEEE Transactions on Automatic Control* **14**, 678–687 (1969).

[99] Mori, T. Criteria for asymptotic stability of linear time-delay systems. *IEEE Transactions on Automatic Control* **30**, 158–161 (1985).

[100] Moheimani, S. R. & Petersen, I. Optimal quadratic guaranteed cost control of a class of uncertain time-delay systems. In *IEEE 34th Conference on Decision and Control*, 1513–1518 (1995).

[101] Yu, L. & Chu, J. An lmi approach to guaranteed cost control of linear uncertain time-delay systems. *Automatica* **35**, 1155–1159 (1999).

[102] Gu, K. An integral inequality in the stability problem of time-delay systems. In *IEEE 39th Conference on Decision and Control*, 2805–2810 (2000).

[103] Fridman, E. & Shaked, U. A descriptor system approach to $\mathcal{H}_\infty$ control of linear time-delay systems. *IEEE Transactions on Automatic Control* **47**, 253–270 (2002).

[104] Fridman, E. & Shaked, U. An improved stabilization method for linear time-delay systems. *IEEE Transactions on Automatic Control* **47**, 1931–1937 (2002).

[105] Richard, J.-P. Time-delay systems: an overview of some recent advances and open problems. *Automatica* **39**, 1667–1694 (2003).

[106] Gu, K., Chen, J. & Kharitonov, V. L. *Stability of time-delay systems* (Springer Science & Business Media, 2003).

[107] Gouaisbaut, F. & Peaucelle, D. Delay-dependent stability analysis of linear time delay systems. *IFAC Proceedings Volumes* **39**, 54–59 (2006).

[108] Shyu, K.-K. & Yan, J.-J. Robust stability of uncertain time-delay systems and its stabilization by variable structure control. *International Journal of Control* **57**, 237–246 (1993).

[109] Park, P. *et al.* A delay-dependent stability criterion for systems with uncertain time-invariant delays. *IEEE Transactions on Automatic Control* **44**, 876–877 (1999).

[110] Lin, C., Wang, Q.-G. & Lee, T. H. A less conservative robust stability test for linear uncertain time-delay systems. *IEEE Transactions on Automatic Control* **51**, 87–91 (2006).

[111] Boyd, S., El Ghaoui, L., Feron, E. & Balakrishnan, V. *Linear matrix inequalities in system and control theory*, vol. 15 (SIAM, 1994).

[112] Grant, M., Boyd, S. & Ye, Y. Cvx: Matlab software for disciplined convex programming (2008).

[113] Keel, L. & Bhattacharyya, S. P. Robust, fragile, or optimal? *IEEE Transactions onAutomatic Control* **42**, 1098–1105 (1997).

[114] Jadbabaie, A., Abdallah, C. T., Famularo, D. & Dorato, P. Robust, non-fragile and optimal controller design via linear matrix inequalities. In *Proceedings of the American Control Conference*, vol. 5, 2842–2846 (1998).

[115] Corrado, J. R. & Haddad, W. M. Static output feedback controllers for systems with parametric uncertainty and controller gain variation. In *Proceedings of the American Control Conference*, vol. 2, 915–919 (1999).

[116] Famularo, D., Dorato, P., Abdallah, C. T., Haddad, W. M. & Jadbabaie, A. Robust non-fragile lq controllers: the static state feedback case. *International Journal of control* **73**, 159–165 (2000).

[117] Takahashi, R. H., Dutra, D. A., Palhares, R. M. & Peres, P. L. On robust non-fragile static state-feedback controller synthesis. In *Proceedings of the IEEE 39th Conference on Decision and Control*, vol. 5, 4909–4914 (2000).

[118] Yang, G.-H. & Wang, J. L. Non-fragile $\mathcal{H}_\infty$ control for linear systems with multiplicative controller gain variations. *Automatica* **37**, 727–737 (2001).

[119] Du, H., Lam, J. & Sze, K. Y. Non-fragile output feedback $\mathcal{H}_\infty$ vehicle suspension control using genetic algorithm. *Engineering Applications of Artificial Intelligence* **16**, 667–680 (2003).

[120] Du, H., Lam, J. & Sze, K. Y. Non-fragile $\mathcal{H}_\infty$ vibration control for uncertain structural systems. *Journal of Sound and Vibration* **273**, 1031–1045 (2004).

[121] Park, J. H. Robust non-fragile control for uncertain discrete-delay large-scale systems with a class of controller gain variations. *Applied Mathematics and Computation* **149**, 147–164 (2004).

[122] Peaucelle, D. & Arzelier, D. Ellipsoidal sets for resilient and robust static output-feedback. *IEEE Transactions on Automatic Control* **50**, 899–904 (2005).

[123] Lien, C.-H. $\mathcal{H}_\infty$ non-fragile observer-based controls of dynamical systems via lmi optimization approach. *Chaos, Solitons & Fractals* **34**, 428–436 (2007).

[124] Bahavarnia, M. & Tavazoei, M. S. A new view to ziegler–nichols step response tuning method: Analytic non-fragility justification. *Journal of Process Control* **23**, 23–33 (2013).

[125] Van Loan, C. How near is a stable matrix to an unstable matrix? Tech. Rep., Cornell University (1984).

[126] Trefethen, L. N. & Bau III, D. *Numerical linear algebra*, vol. 50 (Siam, 1997).

[127] Rautert, T. & Sachs, E. W. Computational design of optimal output feedback controllers. *SIAM Journal on Optimization* **7**, 837–852 (1997).

[128] Nagahara, M., Quevedo, D. E. & Nešić, D. Maximum hands-off control: a paradigm of control effort minimization. *IEEE Transactions on Automatic Control* **61**, 735–747 (2016).

[129] Zhang, F. *The Schur complement and its applications*, vol. 4 (Springer Science & Business Media, 2006).

[130] Chung, T.-S. & Wu, C.-J. A computationally efficient numerical algorithm for the minimum-time control problem of continuous systems. *Automatica* **28**, 841–847 (1992).

[131] Mosek, A. The mosek optimization software. *Online at http://www. mosek. com* **54**, 2–1 (2010).

[132] Matni, N. & Chandrasekaran, V. Regularization for design. *IEEE Transactions on Automatic Control* **61**, 3991–4006 (2016).

[133] Fattahi, S. & Lavaei, J. Theoretical guarantees for the design of near globally optimal static distributed controllers. In *54th Annual Allerton Conference on Communication, Control, and Computing (Allerton)*, 582–589 (2016).

[134] Fazelnia, G., Madani, R., Kalbat, A. & Lavaei, J. Convex relaxation for optimal distributed control problems. *IEEE Transactions on Automatic Control* **62**, 206–221 (2017).

[135] Moghaddam, S. H. & Jovanovic, M. R. Topology design for stochastically-forced consensus networks. *IEEE Transactions on Control of Network Systems* (2017).

[136] Ito, K. & Kunisch, K. A variational approach to sparsity optimization based on lagrange multiplier theory. *Inverse problems* **30**, 015001 (2013).

Vita

MirSaleh Bahavarnia was born in Tabriz, Iran on August 11, 1990. In 2007, he was awarded Bronze Medal of Iran National Math Olympiad. In 2008, he was ranked 378th among 320,000 participants in the nationwide university entrance exam. He was one of the proposers of the problems of International Mathematical Olympiad Shortlist, Bremen, Germany in 2009. He received his B.Sc. degree in Electrical Engineering (Control) and Minor certification in Math from Sharif University of Technology, Tehran, Iran in 2013. He graduates from Lehigh University with Ph.D. degree in Mechanical Engineering (Control), Bethlehem, PA, USA in 2018. During his Ph.D. studies, he was honored to receive Dean's Doctoral Assistantship, Research Assistantship, Teaching Assistantship, Rossin Doctoral Fellowship, and Departmental Graduate Assistantship awards.

Publications

Published

1. Bahavarnia, M., Somarakis, C. & Motee, N. State feedback controller sparsification via a notion of non-fragility. In IEEE 56th Annual Conference on Decision and Control (CDC), 4205–4210 (2017).

2. Bahavarnia, M., Tabuada, P., Somarakis, C. & Motee, N. Improving sparsity in time and space via self-triggered sparse optimal controllers. In IEEE 56th Annual Conference on Decision and Control (CDC), 4199–4204 (2017).

3. Mousavi, H. K., Somarakis, C., Bahavarnia, M. & Motee, N. Performance bounds and optimal design of randomly switching linear consensus networks. In American Control Conference (ACC), 4347–4352 (2017).

4. Bahavarnia, M. & Motee, N. Sparse memoryless lqr design for uncertain linear time-delay systems. IFAC-PapersOnLine, 50(1), 10395–10400 (2017).

5. Bahavarnia, M. & Motee, N. Row-column sparse linear quadratic controller design via bi-linear rank penalty technique and non-fragility notion. In 25th Mediterranean Conference on Control and Automation (MED), 1165–1169 (2017).

6. Arastoo, R., Bahavarnia, M., Kothare, M. V. & Motee, N. Closed-loop feedback sparsification under parametric uncertainties. In IEEE 55th Annual Conference on Decision and Control (CDC), 123–128 (2016).

7. Bahavarnia, M. & Motee, N. Periodic time-triggered sparse linear quadratic controller design. In 54th Annual Allerton Conference on Communication, Control, and Computing (Allerton), 1060–1067 (2016).

8. Arastoo, R., Bahavarnia, M., Kothare, M. & Motee, N. Output feedback controller sparsification via $\mathcal{H}_2$-approximation. IFAC-PapersOnLine 48(22), 112–117 (2015).

9. Bahavarnia, M., Tavazoei, M. S. & Mesbahi, A. Non-fragile tuning of fractional-order PD controllers for IPD-modelled processes. IFAC Proceedings Volumes 46(1), 361–366 (2013).

10. Bahavarnia, M. & Tavazoei, M. S. A new view to Ziegler-Nichols step response tuning method: Analytic non-fragility justification. Journal of Process Control 23, 23–33 (2013).

11. Tavazoei, M. S. & Bahavarnia, M. Fractional-order models and overshooting step responses. In 4th IFAC Workshop on Fractional Differentiation and Its Applications (2010).

Accepted

1. Bahavarnia, M. Structured sparse approximation of a linear consensus network. 14th IEEE International Conference on Control & Automation (2018).

Submitted

1. Bahavarnia, M. Feedback controller sparsification via quasi-norms. Submitted to IEEE 57th Annual Conference on Decision and Control (CDC) (2018).