ABSTRACT

Title of dissertation: **CO-DESIGN OF TIME-INVARIANT DYNAMICAL SYSTEMS**

Prasad Vilas Chanekar
Doctor of Philosophy, 2018

Dissertation directed by: Associate Professor Nikhil Chopra, Advisor
Professor Shapour Azarm, Co-Advisor
Department of Mechanical Engineering

Design of a physical system and its controller has significant ramifications on the overall system performance. The traditional approach of first optimizing the physical design and then the controller may lead to sub-optimal solutions. This is due to the interdependence between the physical design and control parameters through the dynamic equations. Recognition of this fact paved the way for investigation into the "Co-Design" research theme wherein the overall system's physical design and control are simultaneously optimized.

Co-design involves simultaneous optimization of the design and the control variables with respect to certain structural property as constraint. The structural property may be in the form of stability, observability or controllability leading to different types of co-design problems. Co-design optimization problems are non-convex optimization problems involving bilinear matrix inequality (BMI) constraints and are NP-hard in general.

In this dissertation, four interrelated research tasks in the area of co-design are

undertaken. In the first research task, a theoretical and computational framework is developed to co-design a class of linear time invariant (LTI) dynamical systems. A novel solution procedure based on an iterative combination of generalized Benders decomposition and gradient projection method is developed guaranteeing convergence to a solution in a finite number of iterations which is within a tolerance bound from the nearest local/global minimum. In the second research task, the sparse and structured static feedback design problem is modeled as a co-design problem. A formulation based on the alternating direction method of multipliers is used to solve the sparse feedback design problem which has given robustness as a constraint. In the third research task, the optimal actuator placement problem is formulated as a co-design problem. The actuator positions are modeled as $0/1-$binary design variables and result in a mixed integer nonlinear programming (MINLP) problem. In the fourth research task, a heuristic procedure to place sensors and design observer is developed for a class of Lipschitz nonlinear systems. The procedure is based on the relation between Lipschitz constant, sensor locations and observer gain.

The vast and diverse application potential of co-design across all engineering branches is the primary motivation and relevance of the research work carried out in this dissertation.

CO-DESIGN OF TIME-INVARIANT DYNAMICAL SYSTEMS

by

Prasad Vilas Chanekar

Dissertation submitted to the Faculty of the Graduate School of the
University of Maryland, College Park in partial fulfillment
of the requirements for the degree of
Doctor of Philosophy
2018

Advisory Committee:
Associate Professor Nikhil Chopra, Chair/Advisor
Professor Shapour Azarm, Co-Advisor
Professor Balakumar Balachandran
Professor Amr Baz
Associate Professor Jin-Oh Hahn
Professor Nuno Martins, Dean's Representative

Preface

कर्मण्येवाधिकारस्ते मा फलेषु कदाचन ।

मा कर्मफलहेतुर्भूर्मा ते सङ्गोऽस्त्वकर्मणि ॥

श्रीमद् भगवद्गीता २–४७

"You have the right to work only, but never to its fruits.

Let not the fruits of action be your motive,

nor let your attachment be to inaction."

Srimad Bhagavad-Gita Chapter 2 Verse 47

Dedication

To,

Family, Almighty

and

His unusual ways which inspire you to persevere HARDER.

Acknowledgment

I would first like to express my deepest gratitude towards God Almighty for providing the wonderful opportunity to study at the Department of Mechanical Engineering, University of Maryland, College Park and undertake this research work.

I also owe my deepest gratitude to my parents, Shri Vilas Gunawantrao Chanekar and Shrimati Sarita Vilas Chanekar and my sisters, Shrimati Prasanna Nandakumar Karlekar and Kumari Sonu Vilas Chanekar, as my academic journey till date is actually the result of their sacrifices and unwavering support. I will forever be indebted to them.

I would like to thank and acknowledge my advisors, Dr. Nikhil Chopra and Dr. Shapour Azarm for their constant help, patience, encouragement, guidance, support and mainly for overlooking my shortcomings. Our interactions over the last five years has made me a better researcher. You made me understand the importance of mathematical rigor and now I am appreciating it. Thank you very much for everything.

I would like to thank all the professors of the institute as well all my teachers till date from whom I have always gained and learned valuable knowledge regarding research, work and life in general.

I would like to thank Dr. Balakumar Balachandran, Dr. Amr Baz, Dr. Jin-Oh Hahn and Dr. Nuno Martins for serving on my dissertation committee and providing me with insightful suggestions.

I would specially like to thank Dr. Ashitava Ghosal and Dr. Pavankumar

Tallapragada from the Indian Institute of Science, Bengaluru, India for the insightful discussions.

I would like to acknowledge the support received from the Office of Naval Research under grant number N000141310160. I would also like to acknowledge the support received from the Petroleum Institute, Khalifa University of Science and Technology, Abu Dhabi, UAE for the Pipeline System Integrity Management Project.

I want to specially thank Tianchen for the insightful discussions on co-design over the last five years. I would also want to thank Bala Kameshwar Poolla of ETH Zürich for the numerous discussions on various topics.

I would like to thank Dr. Fu Lin of United Technologies Research Center for making the implementation of the ADMM algorithm publicly available.

I would like to thank my labmates, Nirupam, Yimeng, Gurtajbir, Bryan, Jeffery, Vipin, Randall, Eliot, Abu, Xiangxue, Eason and Celeste for the joyful memories. I would also like to thank my roommates Jonathan, Tanmoy, Ramakrishna, Shaurya, Deepayan, Aminul and Keshaw who made my stay at Graduate Hills 3412-22 an enjoyable one. I would also want to thank Abhinav, Sayantan, Wrick, Ranchu and Saurabh who made my stay in Maryland a memorable one.

I would like to thank my friends Amit, Harshal, Ameya, Nitin, Gaurav, Sandilya, Vijay, Rama Krishna, Nakul and Nachiket who are always there whenever I need them.

I would like to apologize to all those whom I have unintentionally missed to acknowledge here.

Finally, I would like to thank my brother-in-law Shri Nandakumar Karlekar and nephews Omkar and Abhay who never let things get dull or boring during my India trips.

Last but not the least, I would like to thank my wife Astha for the unconditional love, support and happiness she has brought in my life.

प्रसाद विलास चानेकर

(Prasad Vilas Chanekar)

Table of Contents

List of Figures

List of Abbreviations

ADMM	Alternating Direction Method of Multipliers
ARE	Algebraic Riccati Equation
BMI	Bilinear Matrix Inequality
CPF	Control Proxy Function
FSF	Full-State Feedback
GBD	Generalized Benders Decomposition
GPM	Gradient Projection method
KKT	Karush-Kuhn-Tucker
LBD	Lower Bound
LMI	Linear Matrix Inequality
LQ	Linear Quadratic
LQR	Linear Quadratic Regulator
LTI	Linear Time Invariant
MINLP	Mixed Integer Nonlinear Programming
MISDP	Mixed Integer Semidefinite Programming
SDP	Semidefinite Programming
SOF	Static Output Feedback
SpOF	Sparse Output Feedback
SpSF	Sparse State Feedback
SSOF	Structured Static Output Feedback
UBD	Upper Bound

Chapter 1: Introduction

In this chapter, the motivation and the objective behind this research are discussed. The chapter begins with motivating examples and objective of this dissertation followed by a brief description of the co-design problem and the challenges associated with it. Finally, the research tasks undertaken in this dissertation are briefly described.

1.1 Motivation and Objective

In the traditional system design approach, the design variables (passive components) are optimized first and then the control variables (active components) are optimized. This sequential approach leads to a sub-optimal system in general [3, 4] due to the interdependence of the design and the control variables [5] through the dynamics of the system. This is demonstrated next through a practical engineering system.

Figure 1.1 shows a self-balancing unicycle which is a single wheel personal transporter. A self-balancing unicycle is a battery operated device with a capacity of one rider. Due to its compact size, an unicycle needs less space to operate (park) and as it is battery operated, it causes no pollution. This makes the unicycle a very

popular mode of transport in crowded cities. The unicycle is controlled by using gyroscopes and accelerometers.

(a)

(b)

Figure 1.1: Self-balancing unicycle (a) Unicycle with rider [1]; (b) Unicycle [1].

For analysis and design purposes, an unicycle can be modeled as an inverted pendulum system as shown in Figure 1.2. An inverted pendulum is a naturally unstable system consisting of a cart of mass M which is an abstraction of the body of an unicycle, an inverted pendulum of length l having a mass m attached at the end depicting the height of the unicycle and the rider respectively. The unicycle is driven by a battery operated motor in a "controlled" manner such that a consumption of "minimum" energy is expected. In traditional system design, physical design parameters or passive components like M, l, m are optimized first by taking into account a gross approximation of the system dynamics. Then, based on the optimal values of M, l, m, active component, i.e., the control variable u is designed such that u stabilizes the naturally unstable system. Thus the traditional

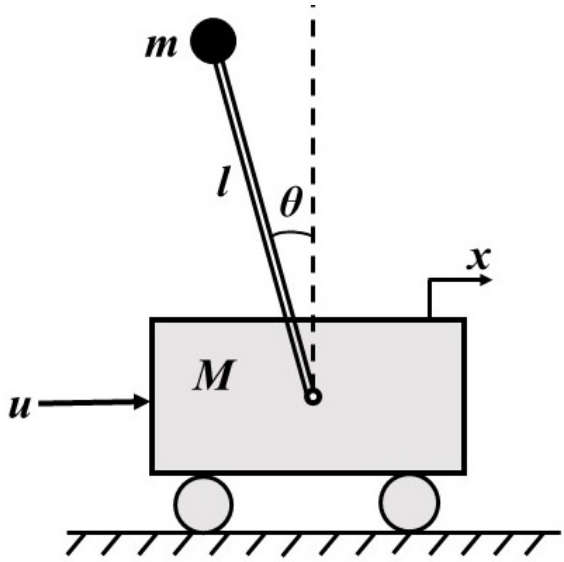


Figure 1.2: An inverted pendulum system.

sequential design process does not exactly take into account the interdependence between M, l, m and u and this might cause the unicycle to be too heavy (or light) leading to a higher consumption of energy in order to maintain stability. Higher energy consumption will require bigger and costlier battery than necessary. To avoid this over (under) design of physical and control parameters, a simultaneous optimization of both is necessary. This need gave rise to the idea of "Co-Design" also known as "co-optimization of design and control".

Co-design has been applied to the design of aerospace structures [6], smart structures [7], electric DC motor [3], mechatronic systems [8], robotic manipulators [9], mechanisms and machine tools [10], chemical process design and control [11]. This wide application of co-design to almost all engineering branches is the basic motivation behind this research.

1.2 Main Topics of the Dissertation

In this section, the main topics of the dissertation are discussed. It is also shown how each topic is a type of co-design problem and can be categorized under the umbrella of "Co-Design".

1.2.1 Co-design Modeling and Optimization

The unicycle discussed in Section 1.1 is modeled to follow linear time invariant (LTI) dynamics as discussed next. Consider the following LTI system,

$$\dot{x} = Ax + Bu, \quad y = Cx, \tag{1.1}$$

where $A \in \mathbb{R}^{n\times n}, B \in \mathbb{R}^{n\times q}, C \in \mathbb{R}^{p\times n}$ are state (system), input and output matrices respectively. $x \in \mathbb{R}^n, y \in \mathbb{R}^p, u \in \mathbb{R}^q$ denote the state vector, output vector and control input of the system respectively. The initial state of x is known and is denoted by x_0 with $x_0 \in \mathbb{R}^n$. The system in (1.1), can be controlled by applying a full-state feedback (FSF) control or a static output feedback (SOF) control as follows,

$$\begin{aligned} &\text{FSF control :} \qquad u = -Kx, \\ &\text{SOF control :} \qquad u = -Ky, \end{aligned} \tag{1.2}$$

where $K \in \mathbb{R}^{q\times n}$ or $K \in \mathbb{R}^{q\times p}$ is the FSF controller gain or SOF controller gain respectively depending on the context. For FSF control, all the state measurements are necessary for control while for SOF control only partial state measurements are required. Let $A = (a_{ij}), B = (b_{ij}), C = (c_{ij}), K = (k_{ij})$. The components

of the matrices A, B, C, K are linear functions of the design variables. Co-design optimization problems are typically multi-objective optimization problems. The objective function of optimization is a linear combination of a convex design objective function and a quadratic control objective function. The control objective function is of the form,

$$f_c \quad \text{or} \quad J := \int_0^\infty \left(y^T Q y + u^T R u\right) dt, \tag{1.3}$$

where Q is a known positive semidefinite matrix and R is a known positive definite matrix. The co-design optimization problem is formulated as,

$$\begin{aligned}
&\min_{d,u} \quad f_d + f_c = f_d(d) + \int_0^\infty \left(y^T Q y + u^T R u\right) dt, \\
&\text{Subject to the constraints,} \\
&\underline{d} \le d \le \overline{d}, \quad g_1(d) \le 0, \quad g_2(d) = 0, \\
&\dot{x} = A(d)\, x + B(d)\, u, \quad y = C(d)\, x, \quad x(0) = x_0, \\
&\text{Designed system should be stable,}
\end{aligned} \tag{1.4}$$

where $d \in \mathbb{R}^{n_d}$ is the design variable and g_1, g_2 are convex design constraints. Matrices A, B, C are functions of the design variable d.

From the unicycle perspective, d represents the design variables M, l, m. u is the control variable which is an abstraction of an effort (force) applied to drive the unicycle in a "regulated" (stabilized) manner. f_d is the design (generally passive) objective function which may be minimizing weight etc., f_c is the control objective which may be the energy supplied by the battery to produce the effort required for driving. g_1, g_2 are constraints on the design variables which can be for example

bounds on M, l, m. It should be noted that constraints on the control variable can be approximated in terms of the design variables. The abstract constraint "Designed system should be stable" ensures the stability of the co-designed system. As f_c is a quadratic objective, linear-quadratic regulator (LQR) which is a FSF controller is used to control the system.

The problem (1.4) is reformulated as,

$$\begin{aligned} &\min_{d,P} \quad f_d(d) + \mathrm{Tr}(P), \\ &\text{Subject to the constraints,} \\ &\underline{d} \leq d \leq \overline{d}, \quad g_1(d) \leq 0, \quad g_2(d) = 0, \quad P \succeq 0, \\ &S(d,P) := A(d)^T P + PA(d) + Q - PB(d) R^{-1} B(d)^T P = 0, \end{aligned} \tag{1.5}$$

where the symmetric matrix P is the control variable and $\mathrm{Tr}(\cdot)$ denotes the trace of the matrix $(\cdot)$. The constraint $S(d,P) = 0$ in (1.5) is an Algebraic Riccati Equation (ARE) [12] constraint and is the mathematical representation of the abstract stability constraint in (1.4). The constraint $S(d,P)$ is a non-convex, nonlinear matrix equality constraint and is a function of design variables d and matrix control variable P. This makes computing a solution to the multi-objective, nonlinear, non-convex co-design optimization problem in (1.3) challenging and is the focus of this topic.

1.2.2 Sparse and Structured Feedback Design

A good control for a dynamical system should provide robust stability against worst case disturbances and model uncertainties while ensuring good performance. It is well-known that $\mathcal{H}_\infty$ control guarantees the necessary robustness and $\mathcal{H}_2$ control

imparts good performance [13,14]. Hence, it is natural for a control system designer to synthesize a control which offers both robustness and good performance. Feedback controllers are of two types: dynamic state feedback and static state feedback. Dynamic feedback control involves controller dynamics making their usage difficult contrary to static feedback which is simple in structure, economically cheap and easy to implement. This makes static feedback a preferred choice of feedback.

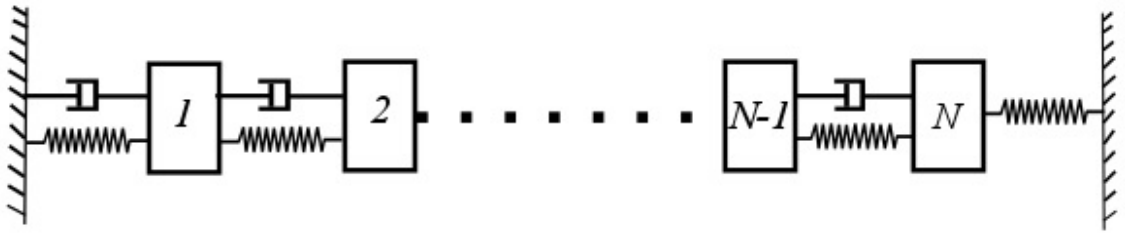


Figure 1.3: A mass-spring-damper chain of N masses

Consider a mass-spring-damper system shown in Figure 1.3 which is an abstraction of structural systems. The dynamics of the said system is described as,

$$\dot{x} = Ax + Bu, \quad y = Cx, \quad u = Ky. \tag{1.6}$$

A, B, C are known system matrices while the feedback controller gain matrix K is unknown. K can be FSF or SOF depending upon C. A feedback controller in a dynamical system can be viewed as a central station which gathers information from "distributed sensors", processes it and generates control action commands for "distributed actuators" [15]. In short a controller is a communication link between sensors and actuators. As the system size increases the static feedback controller becomes complex (dense) causing maintenance and cost issues. Hence, it is desirable to have a sparse or structured feedback. Mathematically, sparse controllers have the

number of 0s (zeros) in the gain matrix as large as possible. For example, for the system in Figure 1.3 with $N = 10$ will have 20 states and can be controlled by a sparse 10×20 FSF controller as shown in Figure 1.4. In Figure 1.4, ✓ indicates a non-zero entry in the gain matrix while a blank space signifies a 0.

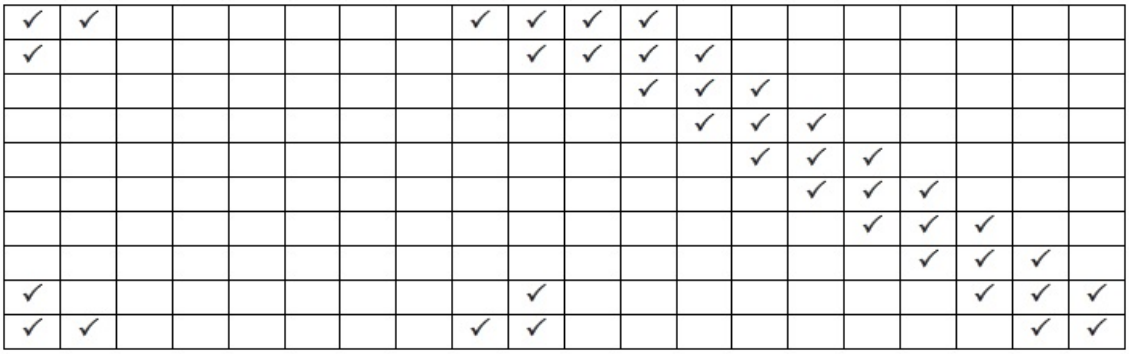
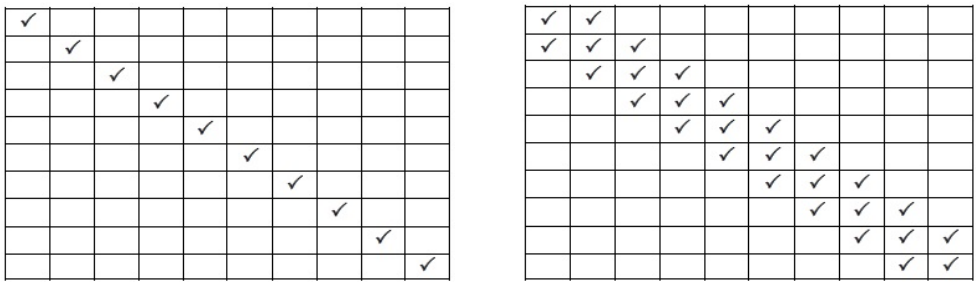


Figure 1.4: A sparse 10×20 FSF controller

Structured controllers are a subset of sparse controller where the non-zero entries have a specific pattern i.e., position of 0s is predefined. For example, for the system in Figure 1.3 with $N = 10$ with only velocity measurements available, structured (diagonal and tridiagonal) 10×10 SOF controllers are shown in Figure 1.5. In practical terms, designing a sparse controller means minimizing the number of communication links between sensors and actuators while in case of structured controller, design is constrained to already present communication links.

As discussed before, a controller should ensure robustness of the system against disturbances/uncertainties along with good performance (consuming minimum energy). In control, $\mathcal{H}_2$ norm is the measure of performance while $\mathcal{H}_\infty$ norm measures robustness [13,14]. Mathematically, the sparse and structured feedback design prob-



(a) (b)

Figure 1.5: 10×10 structured SOF (a) Diagonal, (b) Tridiagonal

lem considering a given robustness level γ is as follows.

$$\begin{aligned} &\min_{K} \quad f_d\left(K\right) + f_c\left(u, \gamma\right), \\ &\text{Subject to the constraints,} \\ &\dot{x} = Ax + Bu, \quad y = Cx, \quad u = Ky, \\ &g\left(K\right), \\ &\text{Designed system should be stable,} \end{aligned} \tag{1.7}$$

where f_d is the design function related to the sparsity of the controller and f_c is the performance measure (abstraction of input energy) dependent upon the controller K and robustness γ . Constraint $g\left(K\right)$ is a design constraint related to the sparsity/structure of the controller. Although only the controller is being optimized, the structure/sparsity of the controller introduces a (design) constraint on the variable K and makes this a co-design problem retaining its challenges. To develop a procedure to design sparse and structured feedback controllers is the aim of this topic.

1.2.3 Optimal Actuator Placement

The state of a dynamical system is driven by actuators based on control input. The ability of any state of a system to get modified (driven) is determined by the controllability of the system. Formally, controllability is the property of a dynamical system which ensures that the system can be steered from any initial state to any final state with the application of an input. The (energy) input is provided to the dynamical system by using one or more actuators in the system. The system controllability as well as energy consumption depends on the placement and number of actuators. In general, computing a minimum number of actuators that ensure controllability of the system is a NP-hard problem [16]. Hence, a practical approach to overcome this difficulty is to place (use) limited number of actuators such that system is controllable. Use of limited number of actuators results in different possible combinations of actuator locations. For example, if the structural system shown in Figure 1.3 can be controlled by two actuators placed on any two available positions, then it can result in multiple possible placement combinations. These multiple combinations may have different energy consumption. This makes the actuator placement problem very important in dynamical system design. In a dynamical system (shown in (1.8)) the input matrix B represents the position of actuators.

Mathematically, the optimal actuator placement problem is written as follows,

$$
\begin{aligned}
&\min_{d} \quad f_c\left(d\right) \\
&\text{Subject to the constraints,} \\
&\dot{x} = Ax + B\left(d\right)u, \\
&B_{ii} \in \{0,1\}, \quad B_{ij} = 0 \quad \text{for} \quad i \neq j, \\
&\mathrm{Tr}\left(B\right) \leq B_{max}, \\
&(A,B) \quad \text{is controllable,}
\end{aligned}
\tag{1.8}
$$

where d is the design variable which depicts the actuator presence ($B_{ii} = 1$) or absence ($B_{ii} = 0$) in the diagonal input matrix B. d is actually a vector of 0's and 1's conveying the value of each diagonal element of B. The value of the input control energy f_c depends upon d. Thus, actuator placement problem is also a type of co-design. To formulate the optimal actuator placement problem and propose a solution procedure is the objective of this topic.

1.2.4 Sensor Placement and Observer Design

In a dynamical system, the output of system y may be purely (some or all) internal states x or a linear combination of some or all the internal states of the system. A sensor in the system measures the output y. The position of sensors in the system is represented in the output matrix C. Information about internal states is required to generate control input to the system. However, the availability of information about the complete internal state depends upon the observability of

the system. Observability is a property of the dynamical system by virtue of which the internal states of a system can be computed from the input-output information [17]. Hence sensors should be placed in the system such that the system is observable. For an observable system, when all the internal states are not known, so an observer is constructed to "estimate" the complete state from input-output information. Consider the following nonlinear system with its observer.

$$\begin{aligned} \dot{x} &= Ax + Bu + \phi(x), \quad y = C(d)\,x, \\ \dot{\hat{x}} &= A\hat{x} + Bu + \phi(\hat{x}) + L(y - C\hat{x}), \end{aligned} \tag{1.9}$$

where d is the design variable similar to the actuator placement problem which depicts the sensor presence ($C_{ii} = 1$) or absence ($C_{ii} = 0$) in the diagonal output matrix C. $\hat{x}$ is an estimate of the real state x, $\phi(x)$ is a Lipschitz nonlinear function [17]. L is the observer gain chosen such that $||x - \hat{x}||_F \longrightarrow 0$ as $t \longrightarrow 0$ where $||(\cdot)||_F$ is the Forbenius norm [18] of $(\cdot)$. The objective is to select design d (place sensors) and L such that the output (control object) is maximized. Thus, sensor placement and observer design can be classified as a co-design problem. The intention of this topic is to formulate a procedure to place sensors and compute linear gain for the observer.

1.3 General Challenges in Co-Design

The co-design optimization problem is modeled as a nonlinear and non-convex optimization problem. The nonlinearity and non-convexity arises due to the system stability constraint which finally transforms into a bilinear matrix inequality

(BMI) [19]. It is well known that the BMI optimization problems are NP-hard in general [20, 21]. Moreover, in sparse and structured feedback design, sparsity and structure constraints on the controller enhance the nonlinearity and non-convexity of the problem. In addition to nonlinearity and non-convexity, sensor and actuator placement problems have $0/1-$integer variables. This results in solving NP-hard mixed-integer nonlinear programming (MINLP) [22] problems to synthesize optimal sensor and actuator locations.

The main topics considered in Section 1.2 and the aforementioned challenges lead to the study of the research components discussed next.

1.4 Research Components in the Dissertation

In this section, an overview of the research problems studied in this dissertation is given.

1. **Co-design Modeling and Optimization:** The objective of this research task is to develop a co-design problem formulation and solution procedure which will lead to optimal solutions with guarantees. The co-design optimization problem is modeled as a BMI optimization problem. A solution procedure consisting of an iterative combination of the generalized Benders decomposition (GBD) [23] and the gradient projection method (GPM) [24] is proposed with provable guarantees. The proposed method is applied to numerical and engineering examples to test its utility. Results show that the proposed approach computes a solution in a finite number of iterations which is with in a

finite provable tolerance from a local/global minimum.

2. **Structured Static Output Feedback Design:** The structured static output feedback (SSOF) problem is formulated as a BMI optimization problem. Similar to the co-design optimization problem, the SSOF design problem is solved using an iterative combination of GBD and GPM. The proposed formulation is applied to design a SSOF controller for an aircraft with favorable results.

3. **Sparse Feedback Design:** Sparse feedback synthesis is posed as an optimization problem with given robustness level as constraint. A scalable solution procedure based on the alternating direction method of multipliers (ADMM) [25] is proposed. The proposed formulation is applied on a class of second order systems which represent a large number of real world applications.

4. **Optimal Actuator Placement:** The optimal actuator placement problem is formulated as a mixed integer BMI optimization problem. By the use of McCormick's relaxation technique [26], the BMI optimization problem is reformulated as an novel equivalent $0/1-$mixed-integer semidefinite programming (MISDP) problem which can be easily solved using the branch-and-bound method [27]. The proposed formulation is applied to a integrator chain system.

5. **Sensor Placement and Observer Design:** The sensor placement and observer design problem for Lipschitz nonlinear systems is set up as an opti-

mization problem. The optimization metric depends upon a relation between Lipschitz constant, sensor positions, observer gain and asymptotic estimation. A heuristic procedure is proposed to solve the optimization. The formulation is then applied to place sensors and design observer for a pipeline without a leak dynamical system.

1.5 Organization of the Dissertation

The remainder of the dissertation is organized as follows: Each of the chapters 2 to 6 is dedicated to the research problems discussed in Section 1.4. Chapter 2 which is to be read first discusses the Co-design Modeling and Optimization Problem. Chapter 3 discusses the SSOF problem. Chapter 4 discusses the Sparse Feedback Design problem. Chapter 5 discusses the Optimal Actuator Placement problem. Chapter 6 discusses the Sensor Placement and Observer Design problem. Each chapter starts with an introduction section which includes a literature review, discusses the challenges involved and contribution in detail. Next, the proposed problem formulation, solution methodology, examples and summarizing remarks follow. Chapter 7 discusses the conclusion of the dissertation work and future directions for further research. Chapters 3 4, 5 and 6 can be read in any order.

Chapter 2: Co-design Modeling and Optimization

This chapter is based on the publications [28, 29].

In this chapter, a novel approach to address the co-design problem for a class of LTI dynamic systems controlled by a LQR feedback is presented. The considered co-design problem is formulated as a non-convex optimization problem with ARE constraint and convex design objective function. Using semi-definite programming (SDP) duality the ARE constraint is reduced into equivalent BMI constraints. This reformulated co-design problem is solved using an iterative algorithm based on the GBD and GPM. The proposed algorithm converges to a solution which is within a specified tolerance from the nearest local minimum (in special cases global minimum) in a finite number of iterations. Necessary and sufficient conditions are developed to test minimality. Three examples are presented to show efficacy of the proposed algorithm.

2.1 Introduction

The wide applicability of co-design has led researchers to concentrate on different strategies to solve the co-design problem. The literature reports on several strategies for solving the co-design optimization problem. Some of these include:

iterative method using optimal control and coupled nonlinear equations [6], co-design based on the coupling property of a system [3, 30, 31], co-design using dynamic optimization technique [32], sequential co-design using Control Proxy Function (CPF) [33]. The general framework of the co-design optimization problem for linear dynamic systems involves system stability conditions. The system stability conditions are reduced to non-convex matrix inequalities which generally have a BMI [19] form, for example in the case of structural systems [34]. The non-convexity in the co-design optimization problem arises from a product of system (physical design) and control parameter matrices. This non-convexity presents several challenges in obtaining an optimal solution to the co-design problem and is the main focus of this research.

A brief overview of the properties of BMIs can be found in [35]. Non-convex matrix inequalities due to the system stability constraints were encountered in a structural co-design optimization problem in [7]. The co-design problem was solved using an iterative procedure which involved convexification of the non-convex matrix inequalities using special convexifying functions. The procedure led to a convergent solution which was not guaranteed to be stationary. A homotopy linearization approach to solve co-design problems involving BMI constraints was studied in [36]. The convergence of the homotopy method to an "acceptable solution" was dependent on the initial value of the variables used at the commencement of the method. A branch-and-bound approach to solve co-design problems involving BMIs was proposed in [35]. A guaranteed globally convergent method to solve co-design problems involving BMIs using dual Lagrangian formulation and exhaustive partitioning of

the space of complicating variables was proposed in [37]. The slow converging nature of the branch-and-bound and exhaustive search approaches limits their application to systems with a small number of decision variables. A convergent gradient-based approach with a local optimality test for the converged stationary solution was discussed in [38]. An iterative approach for co-design by relaxing the BMI constraint into convex sub-problems was studied in [39].

Structurally, it can be shown that BMIs fit well into bilinear optimization problems. GBD is a useful algorithm to solve bilinear optimization problems [23]. A special case of co-design optimization using GBD known as the global optimal joint actuator location and control problem was studied in [40]. An algorithm based on GBD to solve optimization problems with a linear objective function subject to BMI constraints was proposed in [41]. However, the proposed method was limited to the case when the objective function is linear. Further, the authors in [41] conjectured that the algorithm converged to the global optimum. In [28], a BMI formulation for the co-design optimization was proposed without demonstrating convergence to a local/global minimum.

From the aforementioned literature, it can be concluded that the non-convex nonlinear matrix constraints (generally BMIs) are commonly encountered while investigating co-design optimization problems. The non-convex nonlinear matrix constraints typically arise from the stability conditions for the considered system. The current algorithms in the literature for addressing co-design optimization have certain shortcomings. Certain co-design formulations require special functions for optimization (for example, convexifying function in [7], CPF in [31, 33]). Co-design

optimization problems involving BMIs can also be solved using a branch-and-bound approach which requires prior knowledge of control variable bounds [35]. The BMI optimization problems solved using the GBD approach [41] face the shortcoming of being applicable to only linear objective functions. The work in [41] was conjectured to provide a global optimal solution. Moreover, the available BMI formulations and their solution procedures for co-design optimization have issues related to convergence as well as the nature of the computed solution (local/global optimal or sub-optimal) [7, 39, 41]. Works like [3, 30, 31] provided only necessary conditions for local optimality of the co-design optimization problem. The work [40] focused on the global minimum solution of a special co-design problem when the design variables (in linear or nonlinear form) were present only in the input matrix but does not give any insight when the design variables are present in all the system matrices. The co-design optimization procedure developed in [39] required knowledge of the initial system parameters as well as an initial stabilizing control policy which may not always be available and makes the co-design solution dependent on the initial input data. Additionally, the iterative BMI optimization procedures were dependent upon the initialization of the optimization procedure [36]. This presents an additional challenge of finding an appropriate feasible initial design. Additionally, co-design optimization studies like those developed in [39, 40] do not consider a design objective in the optimization problem. Motivated by the aforementioned challenges, the contributions of this chapter are as follows,

1. A new formulation to co-design a class of LTI systems using LQR feedback

controller is proposed where the elements of the system matrices are linear functions of the design variables. By a novel use of SDP duality theory, the optimization problem is reduced to a non-convex problem with a BMI constraint. This formulation has a nonlinear, convex design objective function.

2. The reduced non-convex BMI optimization problem is proved to satisfy the requirements of the GBD procedure. A deterministic algorithm, which iteratively uses the GBD procedure, and a gradient projection method is studied to solve the co-design problem. The proposed algorithm is guaranteed to converge to a solution which is within a specified tolerance from the nearest local minimum (and in special cases global minimum) co-design solution in a finite number of iterations. The proposed solution procedure does not require existence of special functions, prior knowledge of control variable bounds, initial system design parameters, initial stabilizing control policy and is independent of the knowledge of the initial design. It should be noted that all the design parameters appearing linearly in the system matrices can act as design optimization variables.

3. Computationally efficient necessary and sufficient conditions are devised to test the stationarity and local minimality of the converged solution point. These conditions are independent of the control variables, utilize only the design variables, and hence are computationally efficient to handle. Additionally, an upper bound on the specified minimality tolerance is also derived.

4. Conditions under which the co-design problem is convex and has a unique

global minimum solution are established and proved.

5. The novel utilization of SDP duality theory provides a new perspective for handling stabilization/optimization problems involving BMIs.

The remaining chapter is organized as follows: In Section 2.2 the co-design problem is proposed, in Section 2.3 the solution procedure is derived, and in Section 2.4 the co-design algorithm is outlined. In Section 2.5 conditions for convergence and optimality are provided, followed by the computational complexity analysis in Section 2.6. In Section 2.7 three co-design problems are presented, and summarizing remarks are presented in Section 2.8.

2.2 Problem Description

Consider a system with LTI dynamics as follows,

$$\dot{x} = A(d)\,x + B(d)\,u, \quad y = Cx, \tag{2.1}$$

where $A \in \mathbb{R}^{n\times n}, B \in \mathbb{R}^{n\times m}, C \in \mathbb{R}^{q\times n}$, are system matrices, $d \in \mathbb{R}^{n_d}$. $x \in \mathbb{R}^n, y \in \mathbb{R}^q, u \in \mathbb{R}^m$ denote the design, state vector, output vector and control input of the system respectively, n_d is the number of design variables. $A = (a_{ij})$ and $B = (b_{ij})$, where each a_{ij} and b_{ij} are assumed to be linear functions of the design d. The bound on the design is defined by the set $\mathcal{D}_d = \left\{ d \mid \underline{d} \le d \le \overline{d} \right\}$. Let the set $\mathcal{G}_d = \{ d \mid g_1(d) \le 0 \quad \text{and} \quad g_2(d) = 0 \}$. The co-design problem for the system in

(2.1) is stated as follows,

$$\begin{aligned} &\min_{d,u} \quad f_d + f_c = f_d(d) + \int_0^\infty \left(x^T Q x + u^T R u\right) dt, \\ &\text{Subject to the constraints,} \\ &\underline{d} \leq d \leq \overline{d}, \quad g_1(d) \leq 0, \quad g_2(d) = 0, \\ &\dot{x} = A(d)\, x + B(d)\, u, \quad x(0) = x_0, \end{aligned} \tag{2.2}$$

where $x_0 \in \mathbb{R}^n$, $f_d : \mathbb{R}^{n_d} \mapsto \mathbb{R}$, $g_1 : \mathbb{R}^{n_d} \mapsto \mathbb{R}^{n_{g_1}}$, $g_2 : \mathbb{R}^{n_d} \mapsto \mathbb{R}^{n_{g_2}}$, and $Q \succeq 0$, $R \succ 0$ are given real symmetric weight matrices. f_d and f_c are the design and control objective functions respectively. For a matrix X, the notation $X \succeq (\succ)\, 0$ implies X is a positive semidefinite (definite) matrix and X^T denotes the transpose of matrix X. Next, assumptions under which the co-design problem is formulated and solved are listed along with a discussion on their generality, limitations and usage followed by the co-design problem formulation.

2.2.1 Assumptions

The assumptions for the co-design problem formulation are as follows,

(A1) The system in (2.1) follows LTI dynamics.

(A2) The system utilizes an LQR type of feedback controller. This helps significantly in the analytical treatment of the co-design problem due to the well established theoretical results for LQR control.

(A3) The system in (2.1) is assumed to be stabilizable and detectable in the design set $\mathcal{D}_d$.

(A4) The elements of the system matrices A and B are linear functions of the design variables. Though this assumption helps in synthesizing the final bilinear form of the co-design optimization problem, it limits the application of the proposed formulation. However, by use of suitable algebraic manipulations the required linear property can be realized for certain nonlinear cases (see Example 3).

(A5) The design objective function $f_d(d)$ is a smooth, convex and bounded function of design d. The design constraints $g_1(d)$ are smooth and convex while the design constraints $g_2(d)$ are continuous and linear. This assumption is necessary for the application of the GBD procedure.

Assumptions (A1) and (A2) are not restrictive as general LTI are considered. While addressing the co-design problem for nonlinear dynamical systems would broaden the scope of the work, however, as discussed in Section 2.1, it is an open problem to demonstrate a provably correct approach for the aforementioned LTI co-design problem (2.2). The assumption (A4) that the elements of the system matrices A and B are linear functions of the unknown design variables is satisfied by a large class of practical systems such as structural systems [7], robotic systems [42] etc.

The assumption (A3) that the system should be stabilizable and detectable in the design domain $\mathcal{D}_d$ may not always be realized but enables simplified development of the proposed co-design framework. This assumption also helps in establishing the upper bound on the tolerance from the nearest local/global minimum as explained in the proof of Proposition 2.5.2. The proposed algorithm can also be applied without the assumption (A3). An insight regarding this is provided in the Remark 2.4.1 in

Section 2.4.

The assumption (A5) that the design objective function and the design constraints should be convex enables the computation of the global minimum solution for the current relaxed master problem formulated at each GBD iteration (Section 2.3.3). By dropping the convexity assumption the current relaxed master problem cannot be globally minimized and then no guarantees can be provided on the resultant co-design solution. More often than not, the design constraints are in the form of bounds on the design variables which are convex in nature [7,39]. The smoothness assumption for the design objective function as well as the design constraints helps in computing the bound on the tolerance of the converged solution from the nearest (local/global) minimum. In many real world applications, the co-design objective function consists of only control objective and the unknown design variables are embodied in the constraints of the co-design optimization problem [7, 39, 40].

2.2.2 Co-design Problem Formulation

Consider the control objective function f_c as follows,

$$\begin{aligned} f_c &:= \int_0^\infty \left(x^T Q x + u^T R u\right) dt, \\ \text{Dynamics} &:= \dot{x} = A\left(d\right) x + B\left(d\right) u, \\ \text{Initial State} &:= x\left(0\right) = x_0. \end{aligned} \tag{2.3}$$

As the system is stabilizable and detectable, problem in (2.3) has an unique stabilizing solution [43] which minimizes f_c. The candidate Lyapunov function $V = x^T P x$ where P is a symmetric matrix with $P \succeq 0$ is used to prove the stability of the

system. It is shown that, by using an optimal controller, the LQR control objective is minimized and can be reformulated as [44],

$$f_c = \int_0^\infty \left(x^T Q x + u^T R u\right) dt = x_0^T P x_0, \tag{2.4}$$

where P is the unique stabilizing solution of the ARE,

$$\text{ARE} := A^T P + PA + Q - PBR^{-1}B^T P = 0. \tag{2.5}$$

The full-state optimal feedback controller is given by, $u = -Kx = -R^{-1}B^T Px$, where $K = R^{-1}B^T P$ is the controller gain. The control objective f_c depends on the initial state x_0 of the system which is generally not known a priori. This undesirable difficulty is removed by assuming x_0 to be a random vector with zero mean and unit variance. Instead of minimizing the control objective f_c in (2.4), an average or expected value of the control objective is minimized as follows [44],

$$\mathbb{E}\left[f_c\right] = \mathbb{E}\left[x_0^T P x_0\right] = \text{Tr}\left(P\right), \tag{2.6}$$

where $\mathbb{E}$ is the mathematical expectation operator and $\text{Tr}\left(\cdot\right)$ represents the trace of the matrix $(\cdot)$. Using (2.4), (2.5), and (2.6), the co-design problem (2.2) is formulated as,

$$\begin{aligned} &\min_{d, P \succeq 0} \quad f\left(d, P\right) := f_d\left(d\right) + \text{Tr}\left(P\right), \\ &\text{Subject to the constraints,} \\ &\underline{d} \leq d \leq \overline{d}, \quad g_1\left(d\right) \leq 0, \quad g_2\left(d\right) = 0, \quad P \in \mathcal{S}^n, \\ &A^T\left(d\right) P + PA\left(d\right) + Q - PB\left(d\right) R^{-1} B^T\left(d\right) P = 0, \end{aligned} \tag{2.7}$$

where $\mathcal{S}^n$ is the space of real symmetric matrices.

It is demonstrated next that the problem in (2.7) can be reformulated as an equivalent nonlinear optimization problem stated in (2.8) below,

$$\begin{aligned}
&\min_{d,Z} \quad f(d,Z) := f_d(d) + \mathrm{Tr}(Z_{22}R) + \mathrm{Tr}(Z_{11}Q), \\
&\text{Subject to the constraints,} \\
&\underline{d} \le d \le \overline{d}, \quad g_1(d) \le 0, \quad g_2(d) = 0, \\
&S(d,Z) := I + Z_{12}B^T(d) + B(d)Z_{12}^T + Z_{11}A^T(d) + A(d)Z_{11} = 0, \\
&Z = \begin{pmatrix} Z_{11} & Z_{12} \\ Z_{12}^T & Z_{22} \end{pmatrix} \succeq 0,
\end{aligned} \tag{2.8}$$

where I is an identity matrix of appropriate dimension, $Z \in \mathcal{S}^{n+m}$, $Z_{11} \in \mathcal{S}^n$, $Z_{12} \in \mathbb{R}^{n\times m}$, $Z_{22} \in \mathcal{S}^m$. $\mathcal{S}^{n+m}, \mathcal{S}^n$ and $\mathcal{S}^m$ are the spaces of real symmetric matrices. The functions $f(d,Z)$ and $S(d,Z)$ are continuous and at least twice differentiable. It should be noted that throughout this chapter, the variable Z has the structure defined in (2.8). The main result in this section is proved next.

Theorem 2.2.1. Consider a system having LTI dynamics as in (2.1) $\forall d \in \mathcal{D}_d$. If the system is stabilizable and detectable in the entire set $\mathcal{D}_d$, then the co-design optimization problems in (2.7) and (2.8) are equivalent.

Proof. For a known $d \in \mathcal{D}_d$, the ARE in (2.5) is written as a maximization

problem [45] as,

$$\begin{aligned} &\max_{P\in\mathcal{S}^n} \quad \mathrm{Tr}\left(P\right), \\ &\text{Subject to the constraints,} \\ &S_P\left(P\right) := \begin{pmatrix} A^TP + PA + Q & PB \\ B^TP & R \end{pmatrix} \succeq 0. \end{aligned} \tag{2.9}$$

Using (2.9), the co-design problem in (2.7) is written as,

$$\begin{aligned} &\min_{d} \quad f_d\left(d\right) + \mathrm{Tr}\left(P\right), \\ &\text{Subject to the constraints,} \\ &\underline{d} \leq d \leq \overline{d}, \quad g_1\left(d\right) \leq 0, \quad g_2\left(d\right) = 0, \\ &\max_{P\in\mathcal{S}^n} \quad \mathrm{Tr}\left(P\right), \\ &\qquad \text{Subject to the constraints,} \\ &\qquad \begin{pmatrix} A^T\left(d\right)P + PA\left(d\right) + Q & PB\left(d\right) \\ B^T\left(d\right)P & R \end{pmatrix} \succeq 0. \end{aligned} \tag{2.10}$$

The Lagrangian of the maximization problem (2.9) is,

$$\begin{aligned} \mathcal{L}_P\left(P, Z\right) &= \mathrm{Tr}\left(P\right) + \mathrm{Tr}\left(ZS_P\right), \\ &= \mathrm{Tr}\left(P + Z_{12}B^TP + Z_{12}^TPB + Z_{11}A^TP + Z_{11}PA + Z_{11}Q\right) + \mathrm{Tr}\left(Z_{22}R\right), \end{aligned}$$

where $Z = \begin{pmatrix} Z_{11} & Z_{12} \\ Z_{12}^T & Z_{22} \end{pmatrix} \succeq 0$, is the Lagrange multiplier. Using $\frac{\partial \mathcal{L}_P}{\partial P} = 0$, the dual problem of (2.9) is written as,

$$\begin{aligned} &\min_{Z\succeq 0} \quad \mathrm{Tr}\left(Z_{22}R\right) + \mathrm{Tr}\left(Z_{11}Q\right), \\ &\text{Subject to the constraints,} \\ &I + Z_{12}B^T + BZ_{12}^T + Z_{11}A^T + AZ_{11} = 0. \end{aligned} \tag{2.11}$$

From SDP duality theory [45,46], for stabilizable and detectable LTI systems, strong duality holds between (2.9) and (2.11). Problem (2.10) is then written as,

$$\begin{aligned}
&\min_{d} \quad f_d(d) + v_L(d), \\
&\text{Subject to the constraints,} \\
&\underline{d} \leq d \leq \overline{d}, \quad g_1(d) \leq 0, \quad g_2(d) = 0, \\
&v_L(d) = \min_{Z \succeq 0} \quad \mathrm{Tr}(Z_{22}R) + \mathrm{Tr}(Z_{11}Q) \\
&\qquad \text{Subject to the constraint,} \\
&\qquad I + Z_{12}B^T(d) + B(d)Z_{12}^T + Z_{11}A^T(d) + A(d)Z_{11} = 0.
\end{aligned} \tag{2.12}$$

Using the concept of projection [23,47,48], the problem (2.12) is written as a single level problem as,

$$\begin{aligned}
&\min_{d,Z \succeq 0} \quad f_d(d) + \mathrm{Tr}(Z_{22}R) + \mathrm{Tr}(Z_{11}Q), \\
&\text{Subject to the constraints,} \\
&\underline{d} \leq d \leq \overline{d}, \quad g_1(d) \leq 0, \quad g_2(d) = 0, \\
&I + Z_{12}B^T(d) + B(d)Z_{12}^T + Z_{11}A^T(d) + A(d)Z_{11} = 0,
\end{aligned}$$

which implies equivalence of (2.7) and (2.8).

■

Remark 2.2.1. It should be noted that when the initial state is known, the objective function in (2.7) becomes $f_d(d) + x_0^T P x_0$ and the constraint $S(d,Z)$ in (2.8) reduces to $S(d,Z) := x_0 x_0^T + Z_{12}B^T(d) + B(d)Z_{12}^T + Z_{11}A^T(d) + A(d)Z_{11} = 0$.

2.3 Solution Approach

In this section a solution approach to solve the co-design optimization problem in (2.8) is formulated. As stated earlier in Section 2.2, the elements of the matrices A and B are linear in d and this assumption makes the constraint $S(d, Z)$ in (2.8) bilinear. The problem in (2.8) assumes a special "bilinear" structure which is transformed into a convex form when d or Z is held fixed. GBD [23] is a popular method used to solve the optimization problems involving this kind of special bilinear structure. The GBD procedure used in the co-design algorithm presented in this section along with some of the lemmas is inspired from [23, 48]. The lemmas show fulfillment of the GBD requirements as one of the variables in the co-design problem is a matrix variable.

2.3.1 Comparison of GBD and Co-design Problems

The non-convex co-design optimization problem in (2.8) has a bilinear structure similar to the GBD problem [23]. However, the problem in (2.8) has a matrix variable and this differentiates it from the original GBD problem formulation. The objective function $f(d, Z)$ is convex but the problem as a whole is non-convex due to the presence of the non-convex matrix equality constraint $S(d, Z) = 0$. If $Z = (z_{ij})$, then the matrix constraint $S(d, Z)$ has two types of components. The components which are bilinear functions of d and z_{ij} are called as the complicating constraints. The components which are only linear functions of z_{ij} are called as non-complicating constraints. It is to be noted that the constraint $S(d, Z) = 0$ is linear (convex) when

any one of the $d \in \mathcal{D}_d$ or $Z \in \mathcal{Z}$ is held fixed. Since the set $\mathcal{D}_d$ is the domain set of design variables, it is non-empty, compact and convex. It needs to be demonstrated that Z and its components z_{ij} belong to non-empty, compact and convex sets so that the co-design problem in (2.8) can be cast as a GBD problem. To this end, the following Lemma 2.3.1 is proved.

Lemma 2.3.1. Let $\mathcal{Z}_{ij} = \left\{ z_{ij} \mid z_{ij} \in \mathbb{R}, \quad \underline{z}_{ij} \leq z_{ij} \leq \overline{z}_{ij} \right\}$, $i, j = 1, 2, \ldots, n+m$, and

$\mathcal{Z} = \{ Z \mid Z = (z_{ij}), \quad z_{ij} \in Z_{ij}, \quad Z \in \mathcal{S}^{n+m} \}$. For the sets $\mathcal{Z}$ and set $\mathcal{Z}_{ij} = \left[\underline{z}_{ij}, \overline{z}_{ij}\right] \subset \mathbb{R}$, the following hold,

1. Set $\mathcal{Z}$ is bounded i.e., $\exists \sigma_Z \in \mathbb{R}, 0 < \sigma_Z < \infty$ such that $0 \preceq Z \preceq \sigma_Z I \quad \forall Z \in \mathcal{Z}$.
2. Each set $\mathcal{Z}_{ij}$ is bounded i.e., $|\underline{z}_{ij}| < \infty, \quad |\overline{z}_{ij}| < \infty, \quad \forall i, j$.
3. Sets $\mathcal{Z}_{ij}$ are non-empty, compact and convex.
4. Set $\mathcal{Z}$ is non-empty, compact and convex.

Proof.

1. From the assumptions (A1) – (A5), the system is stabilizable and detectable in the set $\mathcal{D}_d \implies$ solution exists for the ARE in (2.5) $\forall d \in \mathcal{D}_d$. The solution to the ARE can be computed using optimization problems in (2.9) or (2.11) (see proof of Theorem 2.2.1). There exists strong duality property between primal problem in (2.9) and dual problem in (2.11).

   Matrix bounds for the ARE solution P in terms of the matrices A, B, Q, R have been reported in [49]. As A, B, Q, R are bounded matrices for $d \in \mathcal{D}_d \implies P$

is bounded and $\mathrm{Tr}\,(P)$ has a finite positive upper bound. From duality theory [50],

$$\sum_i w_i z_{ii} + \sum_{i \neq j} r_{ij} z_{ij} = \mathrm{Tr}\,(P)\,,$$

where w_i and r_{ij} are the function of components of matrices the A, B, Q, R. This implies $\sum_i z_{ii} < \infty$ and hence,

$$\exists \sigma_Z \in \mathbb{R}, 0 < \sigma_Z < \infty, 0 \preceq Z \preceq \sigma_Z I, \forall Z \in \mathcal{Z}.$$

2. Let σ_i be the eigenvalues of Z, the by using properties of Frobenius norm for square symmetric matrices [18],

$$||Z||_F^2 = \sum_{i,j} |z_{ij}|^2 = \sum_i \sigma_i^2.$$

Part 1 of this lemma implies all eigenvalues of Z are bounded, and hence z_{ij} are bounded. Consequently, $|\underline{z}_{ij}| < \infty, \quad |\overline{z}_{ij}| < \infty, \quad \forall i, j.$

3. Follows from $\mathcal{Z}_{ij} = \left[\underline{z}_{ij}, \overline{z}_{ij}\right] \subset \mathbb{R}$.

4. If the linear system is stabilizable and detectable then the problems in (2.9) and (2.11) are feasible $\forall d \in \mathcal{D}_d$ [46].

   $\implies$ $\mathcal{Z}$ is non-empty.

   Let $\langle d_i \rangle$ be a sequence of $d_i \in \mathcal{D}_d$. For every $d_i \in \mathcal{D}_d$, there exists a $Z_i \in \mathcal{Z}$. As $d_i \to \widehat{d} \in \mathcal{D}_d$, corresponding $Z_i \to \widehat{Z} \in \mathcal{Z}$ $\implies$ $\mathcal{Z}$ is a closed set.

   From Part 1 of this lemma, $0 \preceq Z \preceq \sigma_Z I, \forall Z \in \mathcal{Z}$, hence $\mathcal{Z}$ is bounded. This implies Z is compact.

   As $Z \succeq 0$ $\implies$ $\forall y \in \mathbb{R}^{n+m}$, $y^T Z y \geq 0$.

Let $Z_1, Z_2 \in \mathcal{Z}$ and $\lambda \in [0,1]$,

$$0 \leq y^T \left(\lambda Z_1 + (1-\lambda) Z_2\right) y \leq y^T \left(\lambda \sigma_Z I + (1-\lambda)\sigma_Z I\right) y,$$

$$\implies 0 \preceq \lambda Z_1 + (1-\lambda) Z_2 \preceq \sigma_Z I,$$

$$\implies \lambda Z_1 + (1-\lambda) Z_2 \in \mathcal{Z} \implies \text{Set } \mathcal{Z} \text{ is convex.}$$

■

The algorithm to solve the co-design optimization problem in (2.8) follows a similar iterative pathway as in the GBD procedure and is described in the successive sub-sections.

2.3.2 The Primal Sub-Problem

In case of the co-design optimization problem, the design d is taken as the complicating variable. Since the solution procedure is an iterative process, at the k^{th} GBD iteration for $d = d^k \in \mathcal{D}_d \cap \mathcal{G}_d$, the primal sub-problem is formulated as,

$$\begin{aligned} &\min_{Z \succeq 0} \quad f\left(d^k, Z\right), \\ &\text{Subject to the constraints,} \\ &S\left(d^k, Z\right) = 0. \end{aligned} \tag{2.13}$$

The optimal value of the primal sub-problem in (2.13) at $d = d^k$ represents an upper bound (UBD) for the optimal objective function value of the co-design problem in (2.8) after the k^{th} iteration. It should be noted that design d is always chosen as the complicating variable for the co-design problem because when d is held fixed, the primal sub-problem is always feasible (proved in Lemma 2.3.2 in Section 2.3.3). This

eliminates the infeasible primal problem issue encountered in the GBD procedure [23] for the co-design problem (2.8).

The lower bound (LBD) for the optimal objective function value of the co-design problem in (2.8) is computed by solving the current relaxed master problem presented next.

2.3.3 The Current Relaxed Master Problem

The current relaxed master problem is derived using the dual representation of the primal sub-problem in (2.13) and requires the existence of strong duality for the co-design optimization problem at any fixed $d = d^k$. For this the following Lemma 2.3.2 is proved.

Lemma 2.3.2. Consider the linear system in (2.1) with the co-design optimization problem in (2.8). $\forall \widehat{d} \in \mathcal{D}_d \cap \mathcal{G}_d$, the problem in (2.8) is always feasible and satisfies the conditions of strong duality.

Proof. For any fixed $d = \widehat{d} \in \mathcal{D}_d \cap \mathcal{G}_d$,

$$\min_{Z \succeq 0} \quad f_d\left(\widehat{d}\right) + \mathrm{Tr}\left(Z_{22}R\right) + \mathrm{Tr}\left(Z_{11}Q\right),$$

Subject to the constraints,

$$I + Z_{12}B^T\left(\widehat{d}\right) + B\left(\widehat{d}\right)Z_{12}^T + Z_{11}A^T\left(\widehat{d}\right) + A\left(\widehat{d}\right)Z_{11} = 0.$$

1. From assumption (A3) in Section 2.2.1, $\forall \widehat{d} \in \mathcal{D}_d$, the system in (2.1) is stablizable and detectable $\implies \exists \widehat{Z} \in \mathcal{Z}$ such that $S\left(\widehat{d}, \widehat{Z}\right) = 0$,

   $\implies$ problem in (2.8) is always feasible $\forall \widehat{d} \in \mathcal{D}_d \cap \mathcal{G}_d$.

2. From Theorem 6.2.4 in [51] conditions for strong duality are satisfied.

■

Since strong duality is established in Lemma 2.3.2 at $d = d^k$, the problem in (2.8) can be reformulated as an inner and outer optimization problem using the concept of projection [23, 47, 48] as follows,

$$\begin{aligned}
&\min_{d \in \mathcal{D}_d \cap \mathcal{V}_d} \quad v\left(d\right), \\
&\text{Subject to the constraints,} \\
&v\left(d\right) = \min_{Z \succeq 0} \quad f\left(d, Z\right), \\
&\qquad \text{Subject to the constraints,} \\
&\qquad S\left(d, Z\right) = 0, \\
&g_1\left(d\right) \leq 0, \quad g_2\left(d\right) = 0, \\
&\mathcal{V}_d = \left\{\, d \mid S\left(d, Z\right) = 0 \quad \text{for some} \quad Z \succeq 0 \,\right\}.
\end{aligned} \tag{2.14}$$

The function $v\left(d\right)$ is called as the projected function and the problem (2.14) is called as the projected problem [47]. The solution of the primal problem in (2.13) is identical to the solution of its dual problem due to the existence of strong duality at $d = d^k$ as follows,

$$\begin{aligned}
&\min_{Z \succeq 0} \left\{ f\left(d^k, Z\right) \;\; : \;\; S\left(d^k, Z\right) = 0, \right\} \\
&= \sup_{\lambda \in \mathcal{S}^n} \; \min_{Z \succeq 0} \left\{ \mathcal{L}\left(d^k, Z, \lambda\right) \right\}, \quad \forall d^k \in \mathcal{D}_d \cap \mathcal{G}_d \cap \mathcal{V}_d, \\
&\text{where} \quad \mathcal{L}\left(d^k, Z, \lambda\right) = f\left(d^k, Z\right) + \mathrm{Tr}\left(\lambda^T S\left(d^k, Z\right)\right),
\end{aligned} \tag{2.15}$$

$\mathcal{L}\left(d^k, Z, \lambda\right)$ is the Lagrange function of the primal problem in (2.13) and λ is the Lagrange (optimal) multiplier vector. The inner minimization problem for a general

d can be written using (2.15) as,

$$\begin{aligned} v(d) &= \sup_{\lambda \in \mathcal{S}^n} \min_{Z \succeq 0} \{\mathcal{L}(d, Z, \lambda)\} \\ &= \sup_{\lambda \in \mathcal{S}^n} \min_{Z \succeq 0} \left\{ f(d, Z) + \mathrm{Tr}\left(\lambda^T S(d, Z)\right) \right\}, \quad \forall d \in \mathcal{D}_d \cap \mathcal{G}_d \cap \mathcal{V}_d. \end{aligned} \tag{2.16}$$

From the definition of supremum it follows,

$$v(d) \geq \min_{Z \succeq 0} \left\{ f(d, Z) + \mathrm{Tr}\left(\lambda^T S(d, Z)\right) \right\}, \quad \forall \lambda \in \mathcal{S}^n. \tag{2.17}$$

From Lemma 2.3.2 the inner minimization problem is always feasible. Using the dual representation in (2.17), the equivalent of problem (2.14) can be written as,

$$\begin{aligned} &\min_{d \in \mathcal{D}_d \cap \mathcal{V}_d} \quad v(d), \\ &\text{Subject to the constraints,} \\ &v(d) \geq \min_{Z \succeq 0} \left\{ f(d, Z) + \mathrm{Tr}\left(\lambda^T S(d, Z)\right) \right\}, \quad \forall \lambda \in \mathcal{S}^n, \\ &g_1(d) \leq 0, \quad g_2(d) = 0. \end{aligned} \tag{2.18}$$

The optimization problem in (2.18) is called as the master problem which is still difficult to solve due to the presence of the constraint $d \in \mathcal{D}_d \cap \mathcal{V}_d$. The most obvious strategy to overcome this difficulty is to ignore or relax the constraint $d \in \mathcal{D}_d \cap \mathcal{V}_d$ which leads to the following relaxed problem,

$$\begin{aligned} &\min_{d \in \mathcal{D}_d, \mu_B \in \mathbb{R}} \quad \mu_B, \\ &\text{Subject to the constraints,} \\ &\mu_B \geq \min_{Z \succeq 0} \left\{ f(d, Z) + \mathrm{Tr}\left(\lambda^T S(d, Z)\right) \right\}, \quad \forall \lambda \in \mathcal{S}^n, \\ &g_1(d) \leq 0, \quad g_2(d) = 0. \end{aligned} \tag{2.19}$$

The formulation of the current relaxed master problem is based on the Lagrangian formulations. The Lagrangian formulation after k^{th} GBD iteration and for any

$d \in \mathcal{D}_d \cap \mathcal{G}_d$ where Z^k, λ^k is obtained by solving the sub-problem (2.13) for $d = d^k$ is defined as,

$$\mathcal{L}\left(d, Z^k, \lambda^k\right) = f\left(d, Z^k\right) + \mathrm{Tr}\left({\lambda^k}^T S\left(d, Z^k\right)\right). \tag{2.20}$$

The local linear support function for the k^{th} GBD iteration is formulated as follows,

$$\mathcal{L}\left(d, Z^k, \lambda^k\right)\Big|_{d^k}^{lin} = \mathcal{L}\left(d^k, Z^k, \lambda^k\right) + \nabla_d \mathcal{L}^T\left(d, Z^k, \lambda^k\right)\Big|_{d^k}\left(d - d^k\right). \tag{2.21}$$

The outer approximation used in the current relaxed master problem is actually a set of local linear support functions cumulatively constructed on the objective function using the Lagrangian formulations in (2.20). It should be noted that the linear support functions are also called as optimality cuts. The current relaxed master problem at the k^{th} GBD iteration is stated as,

$$\begin{aligned}
&\min_{d \in \mathcal{D}_d, \mu_B \in \mathbb{R}} \quad \mu_B, \\
&\text{Subject to the constraints,} \\
&\mu_B \geq \mathcal{L}\left(d, Z^j, \lambda^j\right)\Big|_{d^j}^{lin}, \quad \forall j = 1, 2, \ldots, k, \\
&\mu_B \geq \mathcal{L}\left(d, Z^k, \lambda^k\right)\Big|_{d^k}^{lin}, \quad \forall \left(d^k, Z^k, \lambda^k\right) \in \Lambda_v^{k_n} \\
&g_1(d) \leq 0, \quad g_2(d) = 0.
\end{aligned} \tag{2.22}$$

The current relaxed master problem in (2.22) for any GBD iteration is a convex problem. k_n is the GBD restart number and $\Lambda_v^{k_n}$ is the set of valid optimality cuts whose significance is explained in Section 2.3.6. The constraint $\mu_B \geq \mathcal{L}\left(d, Z^j, \lambda^j\right)\Big|_{d^j}^{lin}, \quad \forall j = 1, 2, \ldots, k,$ in (2.22) refers to the accumulation of linear support functions approximating the objective function until the k^{th} GBD iteration.

At the end of the k^{th} GBD iteration, the optimal value of the current relaxed master problem represents the LBD for the optimal value of the co-design optimization problem in (2.8). It should be noted that, at each $(k+1)^{th}$ GBD iteration of the current relaxed master problem in (2.22) one constraint is added to the constraints present in the previous k^{th} GBD iteration thereby reducing the size of the feasible region. Hence, $\mu_B^k \leq \mu_B^{k+1}$ i.e., the sequence $\langle \mu_B^k \rangle$ is a non-decreasing sequence. Here $\langle (\cdot) \rangle$ represents the sequence of $(\cdot)$.

2.3.4 Validity of the Outer Approximation

The outer approximation defined in (2.21) is a valid under-estimator for the objective function value $f(d, Z)$ or equivalently of $v(d)$ as defined in (2.14) if it globally underestimates $v(d)$ for any feasible $d \in \mathcal{D}_d \cap \mathcal{G}_d$ [47]. For any convex function, its first-order approximation is its global under-estimator as well as the linear support function [50]. For the existence of valid global linear support functions for $v(d)$ using Lagrangian in (2.20), it is necessary to prove that $v(d)$ is a continuous and convex function. One condition for $v(d)$ to be convex is proved in Theorem 2 below.

Theorem 2.3.1. Let $M = BR^{-1}B^T$ and A has no design variables. If the elements of M are linear functions of the design $d \in \mathcal{D}_d \cap \mathcal{V}_d$ then the projected function $v(d)$ defined in (2.14) is a convex function in the design d.

Proof. The function $v(d)$ is defined in (2.14) as $v(d) : \mathcal{D}_d \cap \mathcal{V}_d \mapsto \mathbb{R}$. The assumption (A3) makes the set $\mathcal{D}_d \cap \mathcal{V}_d$ convex.

$\implies$ domain of $v\left(d\right)$ is convex.

From assumption (A3) and [45, 46], for any $d \in \mathcal{D}_d \cap \mathcal{V}_d$, $v\left(d\right)$ is the solution of the ARE at d. From [52], the solution of the ARE in (2.5) is a convex function of matrix M. As elements of M are linear functions of $d \in \mathcal{D}_d \cap \mathcal{V}_d$ $\implies$ solution of the ARE is convex in d.

$\implies$ $v\left(d\right)$ is a convex function in the design variable $d \in \mathcal{D}_d \cap \mathcal{V}_d$.

■

Two properties related to $v\left(d\right)$ when it is a convex function are proved in Lemma 2.3.3 and Lemma 2.3.4. These properties help in proving the convergence of the proposed algorithm to a solution which is within a specified bound from the global minimum solution for the convex case.

Lemma 2.3.3. If the projected function $v\left(d\right)$ defined in (2.14) is a convex function of design d then $\mathcal{L}\left(d, Z^j, \lambda^j\right)\Big|_{d^j}^{lin}$ defined in (2.21) is the global underestimating linear support function of $v\left(d\right)$ at any point $d^k \in \mathcal{D}_d \cap \mathcal{V}_d$.

Proof. From (2.14) and (2.15),

$$v\left(d^k\right) = \mathcal{L}\left(d^k, Z^k, \lambda^k\right) = f\left(d^k, Z^k\right) + \mathrm{Tr}\left(\lambda^{k^T} S\left(d^k, Z^k\right)\right),$$

where Z^k, λ^k is obtained by solving the sub-problem (2.13) for $d = d^k$. For fixed Z^k, λ^k, the function $f\left(d, Z^k\right)$ is convex in d and $\mathrm{Tr}\left(\lambda^{k^T} S\left(d, Z^k\right)\right)$ is linear in d. Also, from Theorem 2.3.1, $v\left(d\right)$ is convex in d. Hence, $\mathcal{L}\left(d, Z^j, \lambda^j\right)\Big|_{d^j}^{lin}$ is the globally underestimating linear support function of $v\left(d\right)$ at any point $d^k \in \mathcal{D}_d \cap \mathcal{V}_d$ [50].

■

Lemma 2.3.4. When the projected function $v(d)$ defined in (2.14) is a convex function of design d, the following holds,

1. If $Z = Z^k$ is the solution of the optimization problem (2.13) at $d = d^k$ and $\lambda = \lambda^k$ is the corresponding Lagrangian multiplier then,
   $v\left(d^k\right) = f\left(d^k, Z^k\right) = \mathcal{L}\left(d^k, Z^k, \lambda^k\right) \geq \mathcal{L}\left(d^k, Z, \lambda\right)\Big|_d^{lin}$.

2. At the optimal point $\left(\widehat{d}, \widehat{Z}, \widehat{\lambda}\right)$,
   $f\left(\widehat{d}, \widehat{Z}\right) = v\left(\widehat{d}\right) = \mathcal{L}\left(\widehat{d}, \widehat{Z}, \widehat{\lambda}\right) = \mathcal{L}\left(\widehat{d}, \widehat{Z}, \widehat{\lambda}\right)\Big|_{\widehat{d}}^{lin}$.

Proof.

1. From (2.16), (2.20) and Lemma 2.3.2, $\forall d^k \in \mathcal{D}_d \cap \mathcal{G}_d$

$$v\left(d^k\right) = f\left(d^k, Z^k\right) = \mathcal{L}\left(d^k, Z^k, \lambda^k\right).$$

   From Lemma 2.3.3, $\mathcal{L}(d, Z, \lambda)$ is a convex function in d when Z, λ are held fixed, so from the properties of convex functions [50],

$$\mathcal{L}\left(d^k, Z, \lambda\right) \geq \mathcal{L}(d, Z, \lambda) + \nabla_d \mathcal{L}^T(d, Z, \lambda)\Big|_d \left(d^k - d\right),$$
$$v\left(d^k\right) = f\left(d^k, Z^k\right) = \mathcal{L}\left(d^k, Z^k, \lambda^k\right) \geq \mathcal{L}\left(d^k, Z, \lambda\right)\Big|_d^{lin}.$$

2. At the optimal point $\left(\widehat{d}, \widehat{Z}, \widehat{\lambda}\right)$,

$$\nabla_d \mathcal{L}\left(d, \widehat{Z}, \widehat{\lambda}\right)\Big|_{\widehat{d}} = 0,$$
$$v\left(\widehat{d}\right) = f\left(\widehat{d}, \widehat{Z}\right) = \mathcal{L}\left(\widehat{d}, \widehat{Z}, \widehat{\lambda}\right) = \mathcal{L}\left(\widehat{d}, \widehat{Z}, \widehat{\lambda}\right)\Big|_{\widehat{d}}^{lin}.$$

■

For the case when the design variables are present in matrices A and B, the function $v(d)$ is in general non-convex. When the function $v(d)$ is non-convex, the

GBD procedure may not even converge to a stationary point [47]. In this case, the GBD procedure needs to be modified suitably to compute a solution which is within a specified tolerance from the nearest local minimum. The theoretical development for this is presented in the subsequent sequel.

2.3.5 Necessary and Sufficient Conditions for Minimum

The GBD procedure may converge to a design point which may not even be stationary. To conclude if the converged design point is a stationary point, the following proposition is stated.

Proposition 2.3.1. Let the GBD procedure converge to the design point $d^* \in \mathcal{D}_d \cap \mathcal{G}_d$ and Z^*, λ^* are computed from the primal sub-problem (2.13), then d^* is a stationary point for the co-design problem in (2.8) if $\nabla_d f\left(d^*, Z^*\right) + \nabla_d \mathrm{Tr}\left(\lambda^{*T} S\left(d^*, Z^*\right)\right) + \beta_1^T \nabla_d g_1\left(d^*\right) + \beta_2^T \nabla_d g_2\left(d^*\right) + \delta_u - \delta_l = 0$ and $\beta_1^T g_1\left(d^*\right) + \beta_2^T g_2\left(d^*\right) + \delta_l^T\left(\underline{d} - d^*\right) + \delta_u^T\left(d^* - \overline{d}\right) = 0$ for some $\beta_1 \in \mathbb{R}^{n_{g_1}}, \beta_1 \geq 0, \beta_2 \in \mathbb{R}^{n_{g_2}}, \delta_l \in \mathbb{R}^{n_d}, \delta_l \geq 0, \delta_u \in \mathbb{R}^{n_d}, \delta_u \geq 0$.

Proof. The Lagrangian of the problem in (2.8) can be written as,

$$\mathcal{L}_v\left(d, \beta_1, \beta_2, \lambda, Z\right) = f\left(d, Z\right) + \mathrm{Tr}\left(\lambda^T S\left(d, Z\right)\right) + \beta_1^T g_1\left(d\right) + \beta_2^T g_2\left(d\right) - \mathrm{Tr}\left(\lambda_Z^T Z\right) + \delta_l^T\left(\underline{d} - d\right) + \delta_u^T\left(d - \overline{d}\right),$$

where $\beta_1 \in \mathbb{R}^{n_{g_1}}, \beta_1 \geq 0, \beta_2 \in \mathbb{R}^{n_{g_2}}, \delta_l \in \mathbb{R}^{n_d}, \delta_l \geq 0, \delta_u \in \mathbb{R}^{n_d}, \delta_u \geq 0, \lambda_Z \in \mathcal{S}^n, \lambda_Z \succeq 0$ are the Lagrangian dual variables. Taking gradient of $\mathcal{L}_v\left(d, \beta_1, \beta_2, \lambda, Z\right)$ with respect to d at (d^*, λ^*, Z^*) and applying Karush-Kuhn-Tucker (KKT) conditions [51] gives the required result. It should be noted that the KKT conditions related to Z

are always satisfied due to the assumption (A3).

■

The computed stationary point d^* may not be a minimum. To conclude if the stationary point computed using GBD procedure is a local minimum, the following proposition is utilized.

Proposition 2.3.2. The stationary point $d^* \in \mathcal{D}_d \cap \mathcal{G}_d$ computed using the GBD procedure, is a local minimum point for the co-design problem in (2.8), if $H(d^*) = \nabla_d^2 f_d(d^*) + \nabla_d^2 \mathrm{Tr}(P(d^*)) + \nabla_d^2 \left(\beta_1^T g_1(d^*)\right) \succ 0$.

Proof. $\mathcal{L}_v(d, \beta_1, \beta_2, \lambda, Z)$ is as defined in proof of Proposition 2.3.1. For each $d \in \mathcal{D}_d$, corresponding λ, Z are computed from (2.13) and corresponding P is computed from the ARE in (2.5). From Theorem 2.2.1,

$$f(d, Z) + \lambda S(d, Z,) = f(d, P) = f_d(d) + \mathrm{Tr}(P).$$

$\mathcal{L}_v(d, \beta_1, \beta_2, \lambda, Z)$, can be rewritten using $f(d, P)$. Here, $\lambda_Z = 0$ as $Z \succ 0$ and P is a function of d through the ARE in (5). The result follows from the sufficient condition for a local minimum at the at a stationary design point d^* of (2.8) is [51],

■

While the computation of $\nabla_d^2 f_d(d^*)$ and $\nabla_d^2 \left(\beta_1^T g_1(d^*)\right)$ is straightforward, $\nabla_d^2 \mathrm{Tr}(P(d^*))$ is computed from the parametric sensitivity of the ARE [53] as follows.

Let $\frac{\partial X}{\partial d_i} = X_i$, $\frac{\partial^2 X}{\partial d_i \partial d_j} = X_{ij}$, $X^* = X(d^*)$ and $M = BR^{-1}B^T$. P^* is the solution of the ARE, $A^T P + PA + Q - PMP = 0$ at the design point d^*. Differentiating the

ARE with respect to the design vector component d_i at the design point d^* gives,

$$\begin{aligned} A_c^{*T} P_i + P_i A_c^* + \Theta = 0, \\ A_c^* = A^* - M^* P^*, \\ \Theta = A_i^{*T} P^* + P^* A_i^* - P^* M_i^* P^*. \end{aligned} \tag{2.23}$$

Differentiating (2.23) with respect to the design vector component d_j at the design point d^* gives

$$\begin{aligned} & A_c^{*T} P_{ij} + P_{ij} A_c^* + \Upsilon = 0, \\ & \Upsilon = A_i^{*T} P_j^* + A_j^{*T} P_i^* + P_j^* A_i^* + P_i^* A_j^* - P_i^* M_j^* P^* - P_i^* M^* P_j^* \\ & \qquad - P_j^* M_i^* P^* - P^* M_{ij}^* P^* - P^* M_i^* P_j^* - P_j^* M^* P_i^* - P^* M_j^* P_i^*. \end{aligned} \tag{2.24}$$

where, P_i^*, P_j^* are computed from (2.23).

$$H_{ij}\left(d^*\right) = \frac{\partial^2 f_d\left(d^*\right)}{\partial d_i \partial d_j} + \frac{\partial^2 \left(\beta_1^T g_1\left(d^*\right)\right)}{\partial d_i \partial d_j} + \mathrm{Tr}\left(P_{ij}\left(d^*\right)\right). \tag{2.25}$$

Remark 2.3.3. In practice the necessary conditions are implemented as

$$\left| \left\{ \nabla_d f\left(d^*, Z^*\right) + \nabla_d \mathrm{Tr}\left(\lambda^{*T} S\left(d^*, Z^*\right)\right) + \beta_1^T \nabla_d g_1\left(d^*\right) + \beta_2^T \nabla_d g_2\left(d^*\right) + \delta_u - \delta_l \right\}_i \right| \leq \varepsilon_S$$

where $|(\cdot)|$ is the absolute value of $(\cdot)$, $\{\cdot\}_i$ is the i^{th} component of the vector $\{\cdot\}$ and $0 < \varepsilon_S < 1$ is a small predefined constant.

Remark 2.3.4. In practice while implementing the sufficient condition it should be ensured that the Hessian matrix $H\left(d^*\right)$ is away from singularity.

Remark 2.3.5. If $x_0^T P x_0$ is used instead of $\mathrm{Tr}\left(P\right)$ in the co-design objective function in (2.7) then $x_0^T P_{ij}\left(d^*\right) x_0$ should be used instead of $\mathrm{Tr}\left(P_{ij}\left(d^*\right)\right)$ in (2.25).

If the GBD procedure converges to a non-optimal design point d^* then the GBD procedure needs to be restarted from a new (better) design point d^{**} such

that $f\left(d^{**}, P^{**}\right) < f\left(d^{*}, P^{*}\right)$. The procedure to compute the new design point d^{**} is explained in the next subsection.

2.3.6 New Design Point and Valid Optimality Cuts

The new design point is computed using the GPM for linear and nonlinear constrained optimization problems. The method is well explained in Chapter 5 of [24] and hence the details are skipped due to space considerations.

At the non-minimum stationary point d^{*}, let N be the matrix whose columns are linearly independent gradients of active inequality and equality constraints from the constraint set $\left\{g_1\left(d\right) \leq 0, g_2\left(d\right) = 0, \underline{d} \leq d \leq \overline{d}\right\}$. Here, $N \in \mathbb{R}^{n_d \times p_N}$, where p_N is the number of linearly independent gradients of active constraints at d^{*}. The new design point d^{**} is given by,

$$\begin{aligned} d^{**} &= d^{*} - \alpha_r \left[I - N\left(N^T N\right)^{-1} N^T\right] \frac{\nabla_d f^{*}}{||\nabla_d f^{*}||}, \\ \nabla_d f^{*} &= \nabla_d f_d\left(d\right)\Big|_{d=d^{*}} + \nabla_d \mathrm{Tr}\left(P\left(d\right)\right)\Big|_{d=d^{*}}, \end{aligned} \tag{2.26}$$

where $\alpha_r > 0$ is a predefined step-size. $\nabla_d \mathrm{Tr}\left(P\left(d\right)\right)$ can be computed using (2.23).

If d^{*} is a saddle point or a maximum point, then d^{**} is computed as,

$$\begin{aligned} d^{**} &= \underset{d \in \mathcal{D}_d \cap \mathcal{G}_d}{\arg\min} \left\{f^{*}_{-1}, f^{*}_{1}, \ldots, f^{*}_{-i}, f^{*}_{i}, \ldots, f^{*}_{-p}, f^{*}_{p}\right\}, \\ f^{*}_{-i} &= f\left(d^{*} - \alpha_{rs} e_i\right), \qquad f^{*}_{i} = f\left(d^{*} + \alpha_{rs} e_i\right), \end{aligned} \tag{2.27}$$

where $\alpha_{rs} > 0$ is a predefined step-size and $e_i \in \mathbb{R}^{n_d \times 1}$ is a unit vector with 1 as the i^{th} component and rest all components 0.

The GBD procedure is restarted from the new design point d^{**}. Some of the linear support functions, (d^i, Z^i, λ^i), generated from the previous GBD procedure

iterations can be retained after the restart of GBD procedure at d^{**}. These retained linear support functions are called as valid optimality cuts. The valid optimality cuts are re-utilized in the current relaxed master problem (2.22) to accelerate the GBD procedure by decreasing the design search space. The set of valid optimality cuts are selected using (2.20) and (2.21) as follows,

$$\begin{aligned} & f\left(d^{**}, P\left(d^{**}\right)\right) \geq \mathcal{L}\left(d^{**}, Z^k, \lambda^k\right)\Big|_{d^k}^{lin}, \\ & k = 1, 2, \ldots, N_{op}, \end{aligned} \tag{2.28}$$

where N_{op} are the total number of valid optimality cuts generated from the previous GBD procedures. The set of valid optimality cuts at the GBD restart number k^n is defined by $\Lambda_v^{k_n}$ as follows,

$$\begin{aligned} \Lambda_v^{k_n} = \Big\{ \left(d^i, Z^i, \lambda^i\right) \mid \left(d^i, Z^i, \lambda^i\right) \in \Lambda_v^{k_n - 1} \quad \text{and} \\ \left(d^i, Z^i, \lambda^i\right) \text{satisfy (2.28) for } d^{**} \Big\}, k_n \geq 1. \end{aligned} \tag{2.29}$$

It should be noted that at the start of the co-design optimization procedure $k_n = 0$, $\Lambda_v^{k_n}$ does not exist. At the first restart $k_n = 1$, Λ_v^0 is the set of all the optimality cuts generated when the GBD procedure was used for the first time.

Remark 2.3.6. The search space for the current relaxed master problem (2.22) can be increased by modifying the condition in (2.28) as $f\left(d^{**}, P\left(d^{**}\right)\right) - \varepsilon_V \geq \mathcal{L}\left(d^{**}, Z^k, \lambda^k\right)\Big|_{d^k}^{lin}$, where, $\varepsilon_V > 0$ is a predefined constant. It should be noted that the GBD procedure can also be restarted at d^{**} without the use of valid optimality cuts.

Remark 2.3.7. In this work the GPM for linear and nonlinear constrained optimization problems is used to compute the new design point d^{**}. Instead, other

computational methods, for example, the feasible directions method [24] can also be utilized to compute d^{**}.

2.4 Co-design Optimization Algorithm

The steps of the algorithm to solve the co-design optimization problem are given below. The GBD procedure convergence parameter is $\varepsilon > 0$.

1. Set LBD $= -\infty$, UBD $= +\infty, \varepsilon, \varepsilon_S, \varepsilon_V, \alpha_r, \alpha_{rs}$ and GBD restart number $k_n = 0$. Starting design point $d^1 = d^s$. Here $d^s \in \mathcal{D}_d \cap \mathcal{G}_d$ is any known feasible design point.

2. Set GBD iteration number $k = 1, d^k = d^1$.

3. Solve the primal problem in (2.13) for fixed $d = d^k$ to compute d^k, Z^k and λ^k. Store the output and if $f\left(d^k, Z^k\right) <$ UBD then set UBD $= f\left(d^k, Z^k\right), d^* = d^k$.

4. Solve the current relaxed master problem in (2.22) to compute μ_B and d. Set LBD $= \mu_B, k = k + 1, d^k = d$.

5. Check the convergence of bounds using LBD $\geq$ UBD $- \varepsilon$. If convergence is achieved then continue to Step 6 otherwise go to Step 3.

6. If convergence is achieved at Step 5, then test the stationarity condition using ε_S and Section 2.3.5 for d^*. If d^* is stationary, test sufficient condition for local minimum.

   (a) If d^* is a non-stationary point or non-minimum stationary point then,

$k_n = k_n + 1$, compute new design point d^{**} and update set $\Lambda_v^{k_n}$ using $\alpha_r, \alpha_{rs}, \varepsilon_V$ and Section 2.3.6. Go to Step 2 and with input $d^1 = d^{**}$ and $\Lambda_v^{k_n}$.

(b) If d^* fulfills sufficient condition for local minimum go to Step 7.

7. $f^{opt} = \text{UBD}, d^{opt} = d^*$. Using the synthesized design d^{opt} compute the solution matrix P in (2.5). The full-state feedback controller gain is computed using, $K = R^{-1}B^T P$.

Remark 2.4.1. The above algorithm can also be applied without the assumption (A3), that is, without assuming system (2.1) to be stabilizable and detectable in the design space $\mathcal{D}_d$. If the assumption (A3) is neglected then there may be design points computed from the current relaxed master problem in Section 2.3.3 for which the system (2.1) is not stabilizable. For such design points from Lemma 2.3.2, the primal problem (2.13) will be infeasible. In such situations an infeasible primal problem can be constructed to generate infeasiblity cuts which are added as constraints in the current relaxed master problem in Section 2.3.3 [23, 48]. It should be noted that for $d^1 = d^s$ in the proposed algorithm, the system (2.1) should be stabilizable.

In the next section, convergence and optimality properties of the algorithm are analyzed.

2.5 Convergence and Optimality Analysis

In [23, 48] convergence and optimality analysis for the GBD formulation was done for the case of algebraic variables. In the case of the co-design problem (2.8),

one of the variables is a matrix variable. Hence, suitable modifications are made in the original lemmas in [23, 48] so that the convergence proof in [23, 48] is directly applicable to the co-design problem. The optimality proof is not trivial due to the presence of the matrix variable and is given in Proposition 2.5.2. For the convergence proof, the upper semi-continuity of the set of optimal solutions of the primal sub-problem in (2.13) for fixed d and the uniform boundedness of the set of optimal (Lagrangian) multipliers is required. Consequently, the following Lemma 2.5.1 is proved first.

Lemma 2.5.1. Let $f(d, Z)$ and $S(d, Z)$ be continuous on $\mathcal{D}_d \times \mathcal{Z}$. Let for fixed d, $\mathcal{P}(d)$ be the set of optimal solutions of the primal sub-problem in (2.13) and $\mathcal{U}\left(d^k\right)$ be the set of corresponding optimal multipliers, then for fixed $d = d^k \in \mathcal{D}_d$ at the k^{th} GBD iteration the following hold,

1. If $d^k \to \widehat{d}$ and $\lambda^k \to \widehat{\lambda}$ then $\widehat{\lambda} \in \mathcal{U}\left(\widehat{d}\right)$.
2. If $d^k \to \widehat{d}$ then $\mathcal{P}(d)$ is upper semi-continuous at $\widehat{d}$.
3. If for $\widehat{d} \in \mathcal{D}_d$, $\exists \widehat{Z} \in \mathcal{Z}$ such that $S\left(\widehat{d}, \widehat{Z}\right) = 0$, then the set $\mathcal{U}(d)$ is uniformly bounded in some neighborhood of $\widehat{d}$.

Proof. The proof is constructed using Lemma 2.1 in [23], Lemma 7.2 and Lemma 7.3 in [48].

■

Define,

$$\nabla_d \mathcal{L}_v\left(d, Z, \lambda, \beta_1, \beta_2, \delta_l, \delta_u\right) = \nabla_d f\left(d, Z\right) + \nabla_d \mathrm{Tr}\left(\lambda^T S\left(d, Z\right)\right) + \beta_1^T \nabla_d g_1\left(d\right) + \beta_2^T \nabla_d g_2\left(d\right) + \delta_u - \delta_l, \tag{2.30}$$

$$CC\left(d, \beta_1, \beta_2, \delta_l, \delta_u\right) = \beta_1^T g_1\left(d\right) + \beta_2^T g_2\left(d\right) + \delta_l^T\left(\underline{d} - d\right) + \delta_u^T\left(d - \overline{d}\right).$$

Let the co-design algorithm converge to d^* when the following convergence conditions are fulfilled,

$$\begin{aligned} &\left|\left\{\nabla_d \mathcal{L}_v\left(d^*, Z^*, \lambda^*, \beta_1^*, \beta_2^*, \delta_l^*, \delta_u^*\right)\right\}_i\right| \leq \varepsilon_S, \quad \text{for} \quad i = 1, 2, \ldots, n_d, \\ &CC\left(d^*, \beta_1^*, \beta_2^*, \delta_l^*, \delta_u^*\right) = 0, \\ &\nabla_d^2 f_d\left(d^*\right) + \nabla_d^2 \mathrm{Tr}\left(P\left(d^*\right)\right) + \nabla_d^2\left(\beta_1^T g_1\left(d^*\right)\right) \succ 0, \end{aligned} \tag{2.31}$$

where n_d is the number of design variables. Let $\widehat{d}$ be the nearest minimum design point to d^* such that,

$$\begin{aligned} &\left\{\nabla_d \mathcal{L}_v\left(\widehat{d}, \widehat{Z}, \widehat{\lambda}, \widehat{\beta}_1, \widehat{\beta}_2, \widehat{\delta}_l, \widehat{\delta}_u\right)\right\}_i = 0, \quad \text{for} \quad i = 1, 2, \ldots, n_d, \\ &CC\left(\widehat{d}, \widehat{\beta}_1, \widehat{\beta}_2, \widehat{\delta}_l, \widehat{\delta}_u\right) = 0, \\ &\nabla_d^2 f_d\left(\widehat{d}\right) + \nabla_d^2 \mathrm{Tr}\left(P\left(\widehat{d}\right)\right) + \nabla_d^2\left(\beta_1^T g_1\left(\widehat{d}\right)\right) \succ 0. \end{aligned} \tag{2.32}$$

Next, propositions guaranteeing convergence and optimality are presented.

2.5.1 Convergence and Optimality Proofs

Proposition 2.5.1. Consider the linear system in (2.1) and co-design optimization problem in (2.8) with the assumptions (A1) – (A5). The proposed algorithm in Section 2.4 when applied to the problem in (2.8) converges in a finite number of iterations for $0 < \varepsilon_S < 1, \varepsilon > 0$.

Proof. The co-design optimization algorithm is an iterative combination of gradient projection method and GBD procedure. The proof for convergence of GBD procedure in a finite number of iterations for $\varepsilon > 0$ between restarts is constructed from the Theorem 2.5 in [23] and Theorem 7.1 in [48] with suitable modifications. The GPM is solved only once before each GBD restart to compute a new (better) design point. The new (better) design point has a lower objective function value. The aggregate algorithm results in computing a non-increasing sequence of objective function values at each step of the algorithm till the stopping criterion described in Section 2.3.5 is fulfilled for $0 < \varepsilon_S < 1$. Also, the objective function has a finite lower bound. Hence, the co-design optimization algorithm converges in a finite number of iterations for $0 < \varepsilon_S < 1, \varepsilon > 0$.

■

Proposition 2.5.2. Consider the linear system in (2.1) and co-design optimization problem in (2.8) with the assumptions (A1) – (A5). The proposed algorithm in Section 2.4 when applied to the problem in (2.8),

1. converges to a solution within a specified GBD tolerance $\varepsilon > 0$ from the unique global minimum when the projected function $v(d)$ is convex.

2. converges to a solution within a specified GBD tolerance $\varepsilon > 0$ from the unique global minimum when $M = BR^{-1}B^T$ is a linear function of d and A is a constant matrix having no design variables.

3. converges to a solution within a tolerance of $\sqrt{n_d}\varepsilon_S||\overline{d} - \underline{d}||$ from the nearest local minimum when the projected function $v(d)$ is non-convex.

Proof.

1. Let the algorithm in Section 2.4 converge to $(d^* \in \mathcal{D}_d, \mu_B^* \in \mathbb{R})$ after k iterations where d^* is a solution of the co-design problem within a specified GBD tolerance $\varepsilon > 0$ from the global minimum $\implies (d^*, \mu_B^*)$ is the solution of the current relaxed master problem in (2.22) after k GBD iterations,

$$\begin{aligned} &\mu_B^* = \min_{d \in \mathcal{D}_d, \mu_B \in \mathbb{R}} \quad \mu_B, \\ &\text{Subject to the constraints,} \\ &\mu_B \geq \mathcal{L}\left(d, Z^j, \lambda^j\right)\Big|_{d^j}^{lin}, \quad \forall j = 1, 2, \ldots, k, \\ &g_1(d) \leq 0, \quad g_2(d) = 0. \end{aligned} \tag{2.33}$$

Corresponding values of primal variable and optimal multiplier are Z^* and λ^* respectively. From Lemma 2.3.4,

$$\begin{aligned} &\mathcal{L}(d^*, Z^*, \lambda^*) = f(d^*, Z^*) = v(d^*), \\ &\text{with} \quad \mu_B^* \geq f(d^*, Z^*) - \varepsilon, \quad \text{and} \quad f(d^*, Z^*) - f\left(\widehat{d}, \widehat{Z}\right) \leq \varepsilon, \end{aligned}$$

where $\left(\widehat{d}, \widehat{Z}\right)$ is the global optimal solution. Assume d^* is not the solution within a specified tolerance from the global minimum. This implies,

$$f(d^*, Z^*) - f\left(\widehat{d}, \widehat{Z}\right) > \varepsilon \quad \text{and} \quad \exists d^{**} \in \mathcal{D}_d, \mu_B^{**} \in \mathbb{R}, Z^{**} \in \mathcal{Z}, \lambda^{**} \in \mathbb{R}^{n_c}, \tag{2.34}$$

such that,

$$\begin{aligned} &f(d^{**}, Z^{**}) - f\left(\widehat{d}, \widehat{Z}\right) \leq \varepsilon, \\ &\mu_B^{**} = \mathcal{L}(d^{**}, Z^{**}, \lambda^{**}) = f(d^{**}, Z^{**}), \\ &g_1(d^{**}) \leq 0, \quad g_2(d^{**}) = 0, \\ &\mu_B^{**} < \mu_B^*. \end{aligned} \tag{2.35}$$

As $v(d)$ is given to be convex then from Lemma 2.3.4,

$$\mu_B^{**} \geq \mathcal{L}(d^{**}, Z, \lambda)\Big|_d^{lin}. \tag{2.36}$$

Now, (2.36) is true for any (d^j, Z^j, λ^j), with, $j = 1, 2, \ldots, k$ implies,

$$\mu_B^{**} \geq \mathcal{L}\left(d^{**}, Z^j, \lambda^j\right)\Big|_{d^j}^{lin} \quad \text{for} \quad j = 1, 2, \ldots, k. \tag{2.37}$$

From (2.37), (d^{**}, μ_B^{**}) lies in the feasible set of convex optimization problem (2.33),

$$\implies \mu_B^{**} \geq \mu_B^{*}. \tag{2.38}$$

Above inequality (2.38) contradicts the assumption made in (2.34) and (2.35). Hence, $f(d^*, Z^*)$ is a solution within a specified GBD tolerance ε from the global minimum.

2. Follows from Theorem 2.3.1 and Part 1 of this Proposition.

3. When $v(d)$ is a non-convex function, the GBD procedure in the co-design optimization algorithm may converge to a non-minimum design point. This is tested by necessary and sufficient conditions for a local minimum derived in the Section 2.3.5 using ε_S. If the converged design point is not a local minimum, then the GBD procedure is restarted at a new (better) design point computed using the GPM as explained in Section 2.3.6. If the algorithm performs multiple restarts then, $f^0 > f^1 > \ldots > f^i > \ldots > f^{k_n}$ where f^i is the function value at the GBD restart number i, and k_n is the current GBD restart number. The sequence $\{f^i\}$ is a decreasing sequence and f has a finite

lower bound. At each instance when the GBD procedure converges, necessary and sufficient conditions for a local minimum (as defined using ε_S) are tested. From Proposition 2.5.1, the algorithm converges to a $d^* \in \mathcal{D}_d$. Let $\mathcal{L}_v$ be as defined in proof of Proposition 2.3.2 using P. The convergence condition is,

$$\left|\left\{\nabla_d \mathcal{L}_v\right\}_i\right| \leq \varepsilon_S \quad \text{for} \quad i = 1, 2, \ldots, n_d,$$
$$\implies \left|\left|\nabla_d \mathcal{L}_v\right|\right| \leq \sqrt{n_d}\varepsilon_S,$$

where $||\cdot||$ represents the norm of a vector. As (2.1) is stabilizable and detectable, $\mathrm{Tr}\,(P)$ is an analytic function of design d for all $d \in \mathcal{D}_d$ [54]. This makes $\mathcal{L}_v\,(d)$ a smooth function of d and $\nabla_d \mathcal{L}_v$ locally Lipschitz around $\widehat{d}$ [55]. Let $\widehat{d}$ be the nearest local minimum to d^*. Using $\nabla_d \mathcal{L}_v\left(\widehat{d}\right) = 0$, the following is true,

$$\left|\left|\nabla_d \mathcal{L}_v\left(\widehat{d} + t_L\left(d^* - \widehat{d}\right)\right) - \nabla_d \mathcal{L}_v\left(\widehat{d}\right)\right|\right|$$
$$\leq L_c t_L \left|\left|d^* - \widehat{d}\right|\right|, \forall t_L \in [0,1]\,, L_c = \frac{\sqrt{n_d}\varepsilon_S}{\left|\left|d^* - \widehat{d}\right|\right|}.$$

Using the Descent Lemma [56],

$$\mathcal{L}_v\,(d^*) \leq \mathcal{L}_v\left(\widehat{d}\right) + \left(d^* - \widehat{d}\right)^T \nabla_d \mathcal{L}_v\left(\widehat{d}\right) + \frac{L_c}{2}||d^* - \widehat{d}||^2. \tag{2.39}$$

Using (2.31) the following can be written,

$$\begin{aligned} \mathcal{L}_v\,(d^*) &= f\,(d^*, Z^*) = f\,(d^*, P^*)\,, \\ \mathcal{L}_v\left(\widehat{d}\right) &= f\left(\widehat{d}, \widehat{Z}\right) = f\left(\widehat{d}, \widehat{P}\right), \end{aligned} \tag{2.40}$$

where Z and P are computed using (2.13) and (2.5) for the corresponding d. Substituting L_c and (2.40) in (2.39) gives,

$$f\,(d^*, P^*) \leq f\left(\widehat{d}, \widehat{P}\right) + \frac{\sqrt{n_d}\varepsilon_S}{2}||d^* - \widehat{d}||,$$
$$\implies f\,(d^*, P^*) - f\left(\widehat{d}, \widehat{P}\right) \leq \sqrt{n_d}\varepsilon_S||\overline{d} - \underline{d}||.$$

■

Remark 2.5.3. $\varepsilon_S \longrightarrow 0, \varepsilon \longrightarrow 0$ results in $f\left(d^*, Z^*\right) \longrightarrow f\left(\widehat{d}, \widehat{Z}\right)$. From extensive simulations it was observed that $0.1 \leq \varepsilon_S \leq 0.001$ and $0 < \varepsilon < 1$ are good choices for ε_S and ε respectively. The lower value of ε increases the number of GBD iterations while the lower value of ε_S may increase the number of GBD restarts.

In the next section, a discussion on the computational complexity of the algorithm is presented.

2.6 Computational Complexity Analysis

The co-design optimization algorithm described earlier has three main parts, the GBD procedure, the testing of necessary/sufficient conditions at the convergence point, and if required, the evaluation of a new design point. The GBD procedure consists of iteratively solving the primal sub-problem and the current relaxed master problem. The primal subproblem (2.13) is an SDP with martix variable Z of dimension $(n+m) \times (n+m)$. The worst case complexity of solving the primal subproblem for each GBD iteration is $\mathcal{O}\left((n+m)^{6.5}\right)$ using interior-point methods [57]. For a known d, due to the existence of strong duality, instead of solving (2.13), the problem (2.9) can be solved. The matrix variable P has dimension $n \times n$ and the variable Z is the dual variable of the constraint $S_P\left(P\right)$ in (2.9). This brings down the worst case complexity to $\mathcal{O}\left(n^{6.5}\right)$. The current relaxed master problem is always convex with no matrix variable and can be solved quickly. The testing of necessary and sufficient condition for optimality, as per Section 2.3.5, requires solving of

multiple Lyapunov equations. The complexity of solving Lyapunov equations with a $n \times n$ matrix variable is $\mathcal{O}(n^3)$ [58]. The last step of finding a new design point as described in Section 2.3.6 involves computing inverse of the matrix $N^T N$. The matrix $N^T N$ is a square matrix of dimension p_N. The complexity for this matrix inversion is $\mathcal{O}(p_N^3)$.

From the aforementioned discussion, the slowest step of the co-design optimization algorithm is computation of the solution of the primal sub-problem SDP. Hence, an estimate of the number of times the primal SDP with a $n \times n$ matrix variable is solved in the worst case can act as a complexity measure for the algorithm in Section 2.4. In general, an expression stating the GBD complexity is not known in the literature. GBD procedure is similar to a cutting plane algorithm. An estimate for the worst case complexity of a cutting plane procedure similar to GBD is known to be $\mathcal{O}^*(n_d^2 \varepsilon^{-2})$ [59], where the $\mathcal{O}^*$ means the lower order terms are ignored. The worst case complexity of the gradient descent method for an unconstrained non-convex optimization to compute a $\sqrt{n_d}\varepsilon_S$-stationary solution is known to be $\mathcal{O}\left(n_d^{-1}\varepsilon_S^{-2}\right)$ [60]. Assuming no improvement for the constrained case for the gradient projection method, in the worst case the number of times the primal SDP with a $n \times n$ matrix variable needs to be solved is approximately $\mathcal{O}\left(n_d^{-1}\varepsilon_S^{-2}\right) \cdot \mathcal{O}^*(n_d^2 \varepsilon^{-2})$. When there is no GBD restart, the complexity is approximately $\mathcal{O}^*(n_d^2 \varepsilon^{-2})$. Though the aforementioned complexity bounds are very conservative, the algorithm converges much faster in practice.

In the next section, three examples are presented to demonstrate the utility of the proposed co-design approach.

2.7 Examples

The first example demonstrates the proposed co-design algorithm in detail while the second and third examples demonstrate Proposition 2.5.2. The software MATLAB [61], with packages YALMIP [62] and SeDuMi-1.3 [63] is used for computation. YALMIP solves the primal problem in step 3 while 'fmincon' in MATLAB solves the current relaxed master problem in step 4 of the algorithm proposed in Section 2.4.

2.7.1 Example 1: Numerical Example

Consider a linear dynamic system with system matrices and other relevant information is,

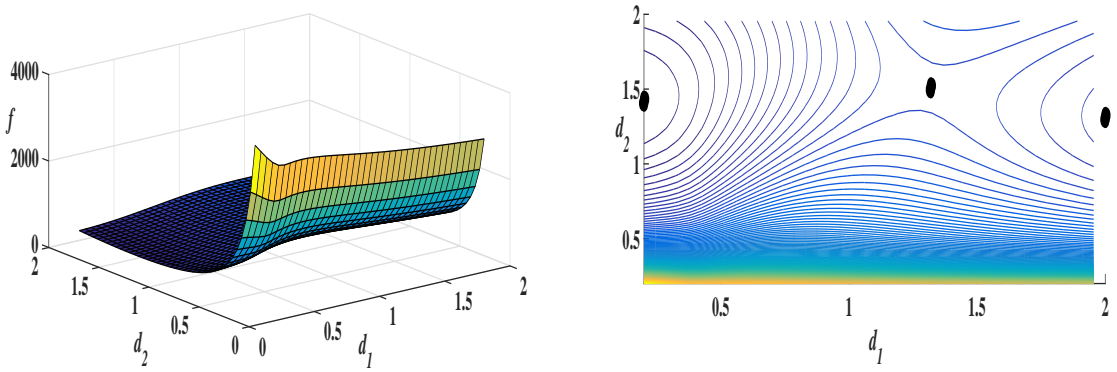
$$A = \begin{pmatrix} 0 & 1 & 0 & 0 \\ -25 & 16d_1 & 6 & -8 \\ 0 & -1 & 0 & 1 \\ 0 & -8 & -6 & 8 \end{pmatrix}, \quad B = \begin{pmatrix} 0 & 0 \\ d_1 & 0 \\ 0 & 0 \\ 0 & -d_2 \end{pmatrix},$$

$$C = \begin{pmatrix} 1 & 1 & 1 & 1 \end{pmatrix}, \quad Q = C^T C, \quad R = I_2,$$

$$d = \begin{pmatrix} d_1 & d_2 \end{pmatrix}^T, f_d(d) = (d_1 - 10)^2 + 100(d_2 - 0.8)^2,$$

$$0.2 \leq d_1, d_2 \leq 2, \quad f(d, P) = f_d(d) + \mathrm{Tr}(P).$$

The surface and contour plots of the objective function $f(d, P)$ in the given design domain are shown in Figure 2.1. The function $f(d, P)$ as well as its pro-



(a) (b)

Figure 2.1: Objective function $f(d, P)$, (a) Surface plot, (b) Contour plot.

jection on the design domain $v(d)$ is non-convex. The function f has three stationary points as shown in the Figure 2.1b with black dots namely, a global minimum at $(0.2 \quad 1.4212)^T$ with $f = 237.95$, a saddle point at $(1.3207 \quad 1.5050)^T$ with $f = 571.69$ and a local minimum at $(2 \quad 1.3144)^T$ with $f = 495.73$. The co-design optimization problem was formulated as in (2.8), and the co-design optimization algorithm proposed in Section 2.4 was applied for different starting design points d^s. The predefined parameters are, $\varepsilon = 0.01, \varepsilon_S = 0.1, \varepsilon_V = 15, \alpha_r = \alpha_{rs} = 0.2$. For $d^s = (1.3 \quad 1.4)^T$, the local minimum was obtained after 5 GBD restarts, for $d^s = (0.5 \quad 1.25)^T$, the global minimum was obtained after 1 GBD restart and for $d^s = (1.3207 \quad 1.5050)^T$, that is, starting at the saddle point, the local minimum was obtained after 2 GBD restarts. Table 2.1 shows the number of GBD iterations at each GBD restart required to reach the local minimum when the co-design optimization algorithm was applied for $d^s = (1.3 \quad 1.4)^T$. In the case when $\varepsilon_S = 0.01$ and all other parameters unchanged, the same local minimum was reached in 7 GBD

restarts. It should be noted that at each restart new design point was computed using the GPM. The system was also designed following the traditional sequen-

GBD restart number	0	1	2	3	4	5
GBD iterations	4	12	5	4	2	1

Table 2.1: GBD iteration at each GBD restart for $\varepsilon_S = 0.1, d^s = (1.3 \quad 1.4)^T$.

tial design methodology. By traditional sequential design methodology, first the system's design function f_d is optimized to compute the optimal design d^o without taking into consideration the control constraints. Now, using this computed optimal design d^o, the LQR controller is synthesized. Using the traditional design procedure, $d^o = (2 \quad 0.81)^T$ and $f = f_d + f_c = 573$, which is worse than the co-design solution computed using the proposed algorithm. For $n_d = 2, \varepsilon = 0.01, \varepsilon_S = 0.1$, from Section 2.6, the worst case computations of the primal SDP are $\mathcal{O}(50) \cdot \mathcal{O}^*(10^4)$. From Table 2.1, for the proposed algorithm in Section 2.4, the total number of computations of the primal SDP are 28 which is much less than the worst case complexity.

2.7.2 Example 2: Satellite Attitude Control

Attitude or orientation about the pitch-roll-yaw axes of earth pointing satellites on circular orbits is controlled by using two pairs of thrust jets [64]. This example studies the co-design of the roll-yaw orientation system. The design parameter is the orientation of the roll-yaw thrust jet pair in the roll-yaw plane controlled

by optimal feedback gains. This co-design problem was previously studied in [40] to obtain a global co-design solution. The roll-yaw linearized dynamic system and various parameters describing the system are [40, 64],

$$A = \begin{pmatrix} 0 & 1 & 0 & 0 \\ -4k_x & 0 & 0 & k_x - 1 \\ 0 & 0 & 0 & 1 \\ 0 & k_y + 1 & k_y & 0 \end{pmatrix}, \quad B = \begin{pmatrix} 0 \\ \frac{I_z}{I_x} \cos\alpha \\ 0 \\ \frac{I_z}{I_y} \cos\beta \end{pmatrix},$$

$$C = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{pmatrix}, \quad Q = C^T C, \quad R = 1,$$

$$x = \begin{pmatrix} \phi & \dot{\phi} & \psi & \dot{\psi} \end{pmatrix}^T, \quad d_1 = \cos\alpha, \quad d_2 = \cos\beta,$$

$$k_x = \frac{I_x - I_y}{I_x}, \quad k_y = \frac{I_z - I_x}{I_y}, \quad \frac{I_z}{I_x} = \frac{7}{6}, \quad \frac{I_x}{I_y} = 3,$$

where $\alpha, \beta,$ denote the directions of the thrust jets in the roll-yaw plane which produce the normalized control u. The parameters I_x, I_y and I_z denote the principal moments of inertia about the roll, yaw and pitch axes respectively, while k_x and k_y are the characteristic satellite constants. The following geometrical constraint is also imposed,

$$\begin{aligned} &\cos^2\alpha + \cos^2\beta = 1, \quad \alpha, \beta \in \left[0, \frac{\pi}{2}\right], \\ \implies &d_1^2 + d_2^2 = 1, \quad d_1, d_2 \in [0, 1]. \end{aligned} \tag{2.41}$$

There is no design objective hence $f_d = 0$. The design variable $d = \begin{pmatrix} d_1 & d_2 \end{pmatrix}^T$ appears linearly in the input matrix B. The eigenvalues of the open loop system are,

$$\text{Eig}\,(A) = (-1.89j, 1.89j, -0.69, 0.69).$$

The presence of one eigenvalue with positive real part makes the open loop system unstable. The system was co-designed to make the closed loop system stable with design objective being optimal placement (orientation) of the actuator thrust jets. The geometrical constraint in (2.41) is a non-convex constraint which makes the approach proposed in this work inapplicable. The non-convex constraint is reformulated as follows,

$$\begin{aligned} &X - dd^T = 0, \quad X \succeq 0, \quad X \in \mathcal{S}^2, \\ &\mathrm{Tr}\,(X) = 1. \end{aligned} \tag{2.42}$$

Relaxing $X - dd^T = 0$ as $X - dd^T \succeq 0$ and applying Schur complement procedure [50], the relaxed convex constraints are as follows,

$$\begin{aligned} &\begin{pmatrix} X & d \\ d^T & 1 \end{pmatrix} \succeq 0, \quad X \succeq 0, \quad X \in \mathcal{S}^2, \\ &\mathrm{Tr}\,(X) = 1. \end{aligned} \tag{2.43}$$

In this case, X becomes an additional design variable. The co-design problem is formulated as follows,

$$\begin{aligned} &\min_{d_1, d_2, X, Z} \quad \mathrm{Tr}\,(Z_{11}Q) + \mathrm{Tr}\,(Z_{22}R)\,, \\ &\text{Subject to the constraints,} \\ &0 \leq d_1 \leq 1, \quad 0 \leq d_2 \leq 1, \\ &I + Z_{12}B^T + BZ_{12}^T + Z_{11}A^T + AZ_{11} = 0, \\ &\begin{pmatrix} X & d \\ d^T & 1 \end{pmatrix} \succeq 0, \quad \mathrm{Tr}\,(X) = 1, \\ &Z \succeq 0, \quad Z \in \mathcal{S}^5, \quad X \succeq 0, \quad X \in \mathcal{S}^2. \end{aligned} \tag{2.44}$$

ε	10^{-1}	10^{-2}	10^{-3}	10^{-4}	10^{-5}
α	39.94°	39.94°	39.29°	39.29°	39.14°
f	4.2169	4.2169	4.2157	4.2157	4.2156
GBD itrs.	5	5	8	9	10

Table 2.2: Comparison of results for various values of ε (GBD itrs.: GBD iterations).

The convergence parameter value was set as $\varepsilon = 10^{-5}$. The starting point was taken to be $d_1^s = \cos 89.42°$. It should be noted that in this problem $M = BR^{-1}B^T$ can be written as a linear function of X. Hence the algorithm should converge to the global minimum solution within a specified tolerance justifying Proposition 2.5.2. The algorithm in Section 2.4 converged to the global minimum of the co-design problem in (2.44) in 10 GBD iterations and 0 (zero) GBD restarts. The exhaustive search results were obtained by dividing the design set of $\alpha \in \left[0, \frac{\pi}{2}\right]$ in 200 parts.

	RG	PA $\varepsilon = 10^{-5}$	ES
f^{opt}	4.2159	4.2156	4.2156
α	38.74°	39.14°	39.14°

Table 2.3: Comparison of results from various methods (RG: Ref. [40], PA: Proposed Algorithm, ES: Exhaustive Search).

The synthesized controller is $K = (-0.1481 \quad 0.4959 \quad 1.2395 \quad 1.0384)^T$ and it sta-

bilizes the co-designed system. The advancement of UBD/LBD is shown in Figure 2.2. Table 2.2 shows the comparison of results computed for different values of the

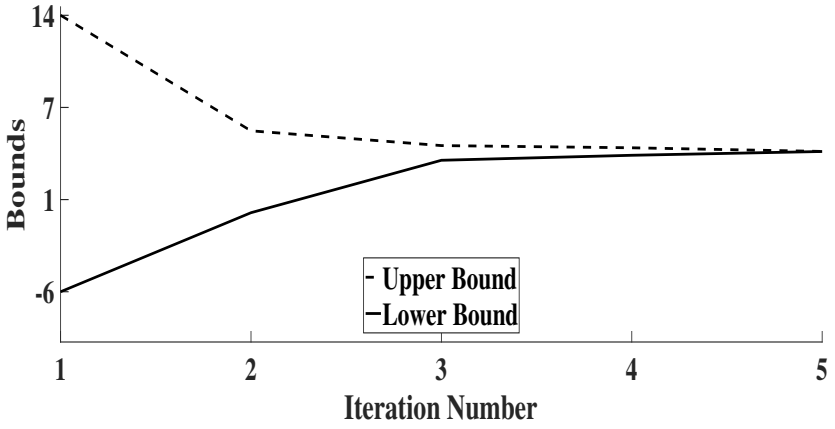


Figure 2.2: Advancement of UBD/LBD.

convergence parameter ε. From Table 2.2, as ε decreases the number of GBD iterations increase but there is not much improvement in the optimal value of the objective function. Table 2.3 shows comparison of the solution obtained using the proposed algorithm with [40] and exhaustive search solutions. This example also shows the effective application of the convex relaxation in (2.43) to the non-convex constraint in (2.42). It should be noted that the synthesized co-design resulted in a stable closed loop system. The method to compute the global minimum solution of the co-design actuator location problem presented in [40] only applies when the design parameters (in linear or nonlinear form) are present only in the input matrix B of the linear dynamics in (2.1). Also, the co-design optimization problem presented in [40] had no design objective function. The proposed algorithm ensures convergence to a solution within a provable tolerance from the nearest local minimum and applies to the co-design problem with the design variables appearing in linear form

in the matrices A and B of the linear dynamics in (2.1). The proposed algorithm also handles the case of the presence of the design objective in the objective function of the co-design optimization problem. From Table 2.2, it is evident that the number of primal SDP computations for the proposed algorithm in Section 2.4 is much less than the worst case primal SDP computations $\mathcal{O}^*\left(n_d^2\varepsilon^{-2}\right)$ calculated for the respective ε.

2.7.3 Example 3: Load Positioning System

The load positioning system co-design optimization was studied in Section 4 of [39] and Section V-A of [38]. It should be noted that the original system had system matrices as nonlinear functions of the design variables. By suitable change of variables, the system matrices were reformulated as linear functions of the design variables. The proposed algorithm was applied to the load positioning system from the same starting design point as in [39] and $\varepsilon_S = 0.05, \varepsilon = 10^{-4}$. The co-design optimization algorithm converged to $d^* = \begin{pmatrix} 1 & 0.0667 & 0.4 & 0.0097 \end{pmatrix}^T$, $f^{opt} = 169.5734$ in just 6 GBD iterations and 0 (zero) GBD restarts while the computation method used in [39] converged to an objective function value of 169.5836 in 24 iterations. Here the total number iterations refer to the total number of SDP computations with a 4×4 matrix variable. The worst case SDP computations are $\mathcal{O}^*\left(10^9\right)$. The solution d^* also satisfied the necessary and sufficient conditions for a local minimum as per Section 2.3.5. An exhaustive search done by dividing the design domain into $20 \times 20 \times 20 \times 20$ grid points computed the minimum objective function value

$f^{es} = 169.5579$. Now, $\sqrt{n_d}\varepsilon_S||\overline{d} - \underline{d}|| = 0.1149$ and $f^{opt} - f^{es} = 0.0115$. This implies $f^{opt} - f^{es} < \sqrt{n_d}\varepsilon_S||\overline{d} - \underline{d}||$ as described in Proposition 2.5.2.

2.8 Summary

In this chapter, a new formulation for the optimal co-design of a class of LTI dynamic systems controlled by a LQR feedback has been presented. The design objective in the proposed formulation is convex and the elements of the system matrices are linear functions of the design variables. With the effective use of SDP duality properties, the co-design problem is reformulated into an equivalent bilinear and non-convex optimization problem. An iterative algorithm based on GBD and GPM has been proposed to solve the optimization problem. The algorithm is guaranteed to converge in a finite number of iterations to a solution which is within a tolerance bound from the nearest local/global minimum. To implement the algorithm, neither the existence of special functions nor the prior knowledge of the control variable bounds is required. The algorithm is also independent of the knowledge of the initial design or the initial stabilizing control variables. Moreover, conditions for obtaining a global minimum of the co-design problem within a specified tolerance are provided. It should be noted that the reformulation of the original co-design problem demonstrates the utility of duality in semidefinite programming which paves a way for the use of known optimization methods. The utility of the proposed algorithm has been demonstrated by three examples.

Chapter 3: Structured Static Output Feedback Design

This chapter is based on the publication [65].

Conventional full-state feedback requires information about all the states to control the system. In many applications either information about all states is not available or not required to control the system. In those cases a SOF controller is used to control the system. SSOF is a class of controllers for which controller gain structure can be predefined.

In this chapter, a new design procedure is proposed for a class of linear time invariant systems controlled by SSOF controllers. The SSOF synthesis problem is posed as an optimization problem with a Lyapunov equation like constraint which is quadratic in gain variables. The problem is reformulated as an optimization problem with a Bilinear Matrix Inequality constraint. An iterative combination of GBD and GPM is used to solve the proposed design problem. Necessary and sufficient conditions are derived to test the minimality of the computed solution. Finally, the proposed formulation is demonstrated through an engineering example.

3.1 Introduction

Conventional full-state feedback control requires utilization of full state information. In many applications, this may not be feasible and hence SOF control provides a useful alternative for control design. SOF finds wide applications in control of structural systems [66], vehicle control [67], flight control [68], network control [69] where the network can be an electric power grid or communication network, etc. In SOF, it is possible to select the controller gain structure a priori. This formulation is known as SSOF control and is the focus of the proposed work in this chapter.

SOF due to its simplicity has been an interesting topic for designers since the 1970s. One of the earliest works uses SOF design for closed-loop pole assignment [70]. SOF stabilization using decision methods has been studied in [71]. A brief survey of SOF is presented in [72]. Stability, convexity and performance analysis of SOF problems is presented in [73, 74]. Linear quadratic sub-optimal SOF control is studied in [75]. Conventional SOF design methods of synthesizing controllers, for example using Algebraic Riccati equation, do not offer designers the flexibility to choose the controller gain structure. Here, choosing the structure of the controller gain implies presetting certain elements of the controller gain matrix to zero.

A survey of SSOF design is presented in [76]. SSOF design problem imposes certain constraints on the controller gain variables and such problems are known to be NP-hard [20, 21]. SSOF design using a gradient based iterative method while penalizing the elements of gain matrix is studied in [77]. The NP-hardness of the

SSOF design problem is due to the BMI [21] constraints. The BMI constraints are involved in the SSOF design problem due to the system stability constraints. Various methods such as, convexifying potential function [78], linearization algorithm [79], non-smooth techniques [80], Linear Matrix Inequality (LMI) relaxation methods [81] have been proposed to solve the SSOF design problem. The gain structure is usually a predefined sparse pattern and is synthesized using augmented Lagrangian approach for the state feedback case in [82]. Rank constrained optimization for synthesizing structured controllers is utilized in [83–85]. SSOF problems are BMI optimization problems which are solved using branch-and-bound method [35, 86], path-following method [36], GBD [23] in [41]. The aforementioned methods have at least one of the following shortcomings: requirement of convexifying functions, knowledge of bounds on all unknown variables, dependence on penalty constant, convergence and optimality guarantees, and dependence on the initial design.

In this work, a SSOF design procedure is proposed for a class of LTI systems with stabilizing SSOF controller gain. The problem is formulated as an optimization problem with an integral Linear Quadratic (LQ) objective function and a Lyapunov equation [87] like constraint which is quadratic in gain variables. With the use of Schur complement procedure [50] and projection technique [23], the problem lends itself into an optimization problem with linear objective function and a BMI constraint. An iterative combination of GBD and GPM [24] is used to compute the solution. Necessary and sufficient conditions are derived to test the stationarity and local minimality of the converged solution. Though it has been observed that the proposed algorithm converges at least to a local minimum solution, the proof guar-

anteeing the convergence has not been established yet. The proposed formulation when applied to a practical example previously studied in [77] resulted in a globally minimum solution.

The chapter is organized as follows, Section 3.2 describes the problem in detail, Section 3.3 describes the theoretical aspects of the solution procedure, Section 3.4 describes the optimization algorithm. In Section 3.5, an example is presented validating the proposed design procedure and Section 3.6 presents a summary of the chapter.

3.2 Problem Description

Consider a system with LTI dynamics as follows,

$$\dot{x} = Ax + Bu, \qquad y = Cx, \qquad x(0) = x_0, \tag{3.1}$$

where $A \in \mathbb{R}^{n\times n}, B \in \mathbb{R}^{n\times q}, C \in \mathbb{R}^{p\times n}$ are state (system), input and output matrices respectively. $x \in \mathbb{R}^n, y \in \mathbb{R}^p, u \in \mathbb{R}^q$ denote the state vector, output vector and control input of the system respectively. The initial state of x is known and is denoted by x_0. The output feedback control is of the form,

$$u = -Ky, \tag{3.2}$$

where $K \in \mathbb{R}^{q\times p}$ is the stabilizing SSOF controller gain for the LTI system (A, B, C). K belongs to the set $\mathcal{K} = \left\{ K \mid \underline{K} \leq K \leq \overline{K} \right\}$. It should be noted that the structure of each gain matrix $K \in \mathcal{K}$ is predefined. Set $\mathcal{K}$ is non-empty, continuous, compact and convex. When C is an identity matrix then K is a full state feedback. The gain

K is chosen such that it minimizes the control energy,

$$J := \int_0^\infty \left(x^T Q x + u^T R u\right) dt, \tag{3.3}$$

where $Q \succeq 0$, $R \succ 0$ are known real symmetric weight matrices. For a matrix Y, the notation $Y \succeq (\succ) 0$ implies Y is a positive semidefinite (definite) matrix and Y^T denotes the transpose of matrix Y. The optimal static output feedback design problem is stated as,

$$\begin{aligned} &\min_K \quad \int_0^\infty \left(x^T Q x + u^T R u\right) dt \\ &\text{Subject to the constraints,} \\ &\underline{K} \leq K \leq \overline{K}, \\ &\dot{x} = Ax + Bu, \quad y = Cx, \quad x(0) = x_0, \\ &u = -Ky, \quad A_c = A - BKC \quad \text{is Hurwitz.} \end{aligned} \tag{3.4}$$

The following assumptions are made for the stabilizing SSOF design problem,

(A1) Output matrix C is full row rank.

(A2) Pair (A, B) is stabilizable.

(A3) Pair $\left(\sqrt{Q}, A\right)$ is detectable.

(A4) System (A, B, C) is output stabilizable i.e., there exists a $K \in \mathcal{K}$ such that $(A - BKC)$ is Hurwitz.

Using the following well known result [77],

$$\begin{aligned} J &= \int_0^\infty \left(x^T Q x + u^T R u\right) dt = x_0^T P x_0, \\ &= \mathrm{Tr}\left(P x_0 x_0^T\right) = \mathrm{Tr}\left(P X_0\right), \end{aligned} \tag{3.5}$$

where $X_0 = x_0 x_0^T$ and $P \succeq 0, P \in \mathcal{S}^n$ is the unique solution of the Lyapunov like equation (LE),

$$\begin{aligned} \text{LE} &:= A_c^T P + P A_c + Q + C^T K^T R K C = 0, \\ A_c &= A - BKC, \end{aligned} \tag{3.6}$$

and $\mathcal{S}^n$ is the space of real symmetric matrices. Here, A_c is Hurwitz and $\text{Tr}\,(\cdot)$ represents the trace of matrix $(\cdot)$. If x_0 is an uniformly distributed random variable with zero mean and unit variance, then $J = \mathbb{E}\left[\text{Tr}\,(PX_0)\right] = \text{Tr}\,(P)$ i.e., $X_0 = I$ in (3.5) where I is an identity matrix of appropriate dimension. The problem in (3.4) is rewritten as,

$$\begin{aligned} &\min_{K,P} \quad \text{Tr}\,(PX_0) \\ &\text{Subject to the constraints,} \\ &\underline{K} \leq K \leq \overline{K}, \quad X_0 = x_0 x_0^T, \\ &\text{LE} = A_c^T P + P A_c + Q + C^T K^T R K C = 0, \\ &A_c = A - BKC, \quad P \succeq 0, \quad P \in \mathcal{S}^n. \end{aligned} \tag{3.7}$$

The problem in (3.7) is non-convex due to the constraint LE which is nonlinear and non-convex in the unknown variables. The solution to the Lyapunov like constraint LE is computed by solving a SDP problem [50] which is explained in the following Lemma 3.2.1,

Lemma 3.2.1. Consider the LTI system in (3.1). If $K \in \mathcal{K}$ is a known stabilizing SSOF controller gain of the system, then the solution $P \succeq 0, P \in \mathcal{S}^n$ to the equation LE in (3.6) is computed by solving a SDP problem.

Proof. As $K \in \mathcal{K}$ is a known stabilizing output feedback controller gain of the system, this implies $A_c = A - BKC$ is Hurwitz. From results in Chapter 6 of [87], the solution to the equation LE in (3.6) is the solution of the optimization problem given as follows,

$$\begin{aligned} &\min_{P} \quad \mathrm{Tr}\,(PX_0) \\ &\text{Subject to the constraints,} \\ &\mathrm{LE1} := -A_c^T P - PA_c - Q - C^T K^T RKC \succeq 0, \\ &P \succeq 0, \quad P \in \mathcal{S}^n, \quad A_c = A - BKC. \end{aligned} \tag{3.8}$$

As $R \succ 0$, applying Schur complement procedure [50] to the constraint LE1 in (3.8) gives,

$$\begin{aligned} &\min_{P} \quad \mathrm{Tr}\,(PX_0) \\ &\text{Subject to the constraints,} \\ &\begin{pmatrix} -A_c^T P - PA_c - Q & C^T K^T \\ KC & R^{-1} \end{pmatrix} \succeq 0, \\ &P \succeq 0, \quad P \in \mathcal{S}^n, \quad A_c = A - BKC, \quad X_0 = x_0 x_0^T. \end{aligned} \tag{3.9}$$

The above optimization problem (3.9) is a SDP problem [50] in the variable P.

■

The nonlinear design optimization problem in (3.7) is converted into an optimization problem with a BMI [19] constraint using Lemma 3.2.1 and is demonstrated in Theorem 3.2.1 next,

Theorem 3.2.1. Consider the LTI system in (3.1). If the assumptions (A1)-(A4) hold, then the optimal SSOF design problem stated in (3.7) is reformulated as an

equivalent optimization problem with a BMI constraint.

Proof. As the assumptions (A1)-(A4) hold, this implies a stabilizing output feedback controller gain $K \in \mathcal{K}$ exists such that $A_c = A - BKC$ is Hurwitz. For a stabilizing controller gain $K \in \mathcal{K}$, substituting (3.9) from Lemma 3.2.1 for the constraint LE in (3.7) gives,

$$
\begin{aligned}
&\min_{K} \quad v_K(K) \\
&\text{Subject to the constraints,} \\
&\underline{K} \leq K \leq \overline{K}, \\
&v_K(K) = \min_{P} \quad \mathrm{Tr}(PX_0) \\
&\qquad \text{Subject to the constraints,} \\
&\qquad \begin{pmatrix} -A_c^T P - PA_c - Q & C^T K^T \\ KC & R^{-1} \end{pmatrix} \succeq 0, \\
&\qquad P \succeq 0, \quad P \in \mathcal{S}^n, \\
&\qquad A_c = A - BKC, \quad X_0 = x_0 x_0^T.
\end{aligned}
\tag{3.10}
$$

Using the concept of projection [23, 47, 48], the problem (3.10) is written as an equivalent single level problem as,

$$
\begin{aligned}
&\min_{K,P} \quad J(P) = \mathrm{Tr}(PX_0) \\
&\text{Subject to the constraints,} \\
&\underline{K} \leq K \leq \overline{K}, \quad X_0 = x_0 x_0^T, \quad A_c = A - BKC, \\
&\mathrm{S}(K,P) := \begin{pmatrix} -A_c^T P - PA_c - Q & C^T K^T \\ KC & R^{-1} \end{pmatrix} \succeq 0, \\
&P \succeq 0, \quad P \in \mathcal{S}^n.
\end{aligned}
\tag{3.11}
$$

The optimization problem in (3.11) has a linear objective function $J(P)$ and a BMI constraint $\mathrm{S}(K,P)$. The constraint $\mathrm{S}(K,P)$ has bilinear matrix products between K and P. Hence, optimization problem (3.11) is equivalent to (3.7).

■

An important property of formulations (3.7), (3.9) and (3.11) is proved in the Theorem 3.2.2 next,

Theorem 3.2.2. Consider the LTI system in (3.1). If $K = \widehat{K} \in \mathcal{K}$ is a non-stabilizing controller gain, then optimization problems (3.7), (3.9) and (3.11) are infeasible for $K = \widehat{K} \in \mathcal{K}$.

Proof. For known $K = \widehat{K} \in \mathcal{K}$, problems (3.9) and (3.11) are equivalent. For non-stabilizing controller gain $K = \widehat{K} \in \mathcal{K}$, no $P \succeq 0, P \in \mathcal{S}^n$ exists such that the constraint LE in (3.7) is fulfilled. This implies infeasibility of the optimization problems (3.7), (3.9) and (3.11) for $K = \widehat{K} \in \mathcal{K}$.

■

3.3 Solution Procedure

In this section, the solution procedure to solve the SSOF design optimization problem (3.11) is presented. The problem in (3.11) is a non-convex optimization problem with linear objective function $J(P)$ and one non-convex BMI constraint $\mathrm{S}(K, P)$. The problem (3.11) assumes a special "bilinear" structure for which GBD [23] is used to compute the solution. When K is held constant then (3.11) transforms into a convex SDP problem. Here, K is called as the "complicating" variable while P is called as the non-complicating variable. The BMI constraint $\mathrm{S}(K, P)$ is called as the complicating constraint. The theoretical details about GBD are explained in [23] and are skipped due to space limitation. The GBD procedure proceeds by first formulating the feasible (or infeasible) primal sub-problem and later the master problem. The feasible primal sub-problem provides the UBD while the master problem gives the LBD of (3.11). The GBD procedure iterates between primal subproblem and master problems till UBD and LBD are sufficiently close as per a predefined convergence criterion. However, it is known that the GBD may or may not converge to a stationary point [47]. Hence, a local (or global) solution may not be obtained by using GBD. To overcome this difficulty, the GBD is restarted from a new design point. In this section, first the GBD procedure details are given in Sections 3.3.1, 3.3.2 and 3.3.3. Section 3.3.4 states the necessary and sufficient conditions for minimum. Section 3.3.5 presents the procedure for computation of a new design point required to restart the GBD.

3.3.1 Feasible Primal Sub-Problem

For a stabilizing SSOF controller gain $K^i \in \mathcal{K}$ the feasible primal sub-problem is formulated as,

$$\begin{aligned} &\min_{P} \quad \mathrm{Tr}\left(PX_0\right) \\ &\text{Subject to the constraints,} \\ &X_0 = x_0 x_0^T, \quad A_c^i = A - BK^iC, \\ &\mathrm{S}\left(K^i, P\right) \succeq 0, \quad P \succeq 0, \quad P \in \mathcal{S}^n. \end{aligned} \tag{3.12}$$

The optimal value of (3.12) is the UBD of (3.11). Let $Z^i \succeq 0, Z^i \in \mathcal{S}^{n+q}$ be the Lagrange dual variable of the constraint $\mathrm{S}\left(K^i, P\right) \succeq 0$ and P^i the computed primal variable.

3.3.2 Infeasible Primal Sub-Problem

From Theorem 3.2.2, when $K = K_1^j \in \mathcal{K}$ is non-stabilizing, then (3.11) is infeasible. The infeasible primal sub-problem for GBD at $K = K_1^j$ is formulated as follows,

$$\begin{aligned} &\min_{P_1, \rho_S} \quad \rho_S \\ &\text{Subject to the constraints,} \\ &A_c^j = A - BK_1^jC, \quad \rho_S \geq 0, \\ &\mathrm{S}\left(K_1^j, P_1\right) + \rho_S I \succeq 0, \quad P_1 - \rho_P I \succeq 0, \quad P_1 \in \mathcal{S}^n. \end{aligned} \tag{3.13}$$

The constant $\rho_P > 0$ is known and it ensures $P_1 \succ 0$. Let $Z_1^j \succeq 0, Z_1^j \in \mathcal{S}^{n+q}$ be the Lagrange dual variable of the constraint $\mathrm{S}\left(K_1^j, P_1\right) + \rho_S I \succeq 0$ and P_1^j the computed primal variable.

3.3.3 Master Problem

The master problem is formulated using P, P_1, Z, Z_1 which are computed by solving the feasible and infeasible primal sub-problems. The master problem computes K and the LBD for (3.11). To construct the master problem, the feasible primal problem (3.12) must fulfill conditions for strong duality which are discussed in Lemma 3.3.1.

Lemma 3.3.1. Consider the LTI system in (3.1) with assumptions (A1)-(A4). For a stabilizing gain $\widehat{K} \in \mathcal{K}$, (3.12) fulfills conditions of strong duality.

Proof. As $\widehat{K} \in \mathcal{K}$ is a stabilizing gain, this implies $\exists P \succ 0$ such that $\mathrm{S}\left(\widehat{K}, P\right) \succ 0$. Also, problem (3.12) is convex. From Theorem 6.2.4 in [51], (3.12) fulfills conditions of strong duality.

■

Next, Lagrangians for the feasible and infeasible primal sub-problems are constructed. The Lagrangian for the feasible primal sub-problem in (3.12) for stabilizing K is,

$$\mathcal{L}\left(K, P, Z\right) = \mathrm{Tr}\left(PX_0\right) - \mathrm{Tr}\left(Z\mathrm{S}\left(K, P\right)\right). \tag{3.14}$$

The Lagrangian for the infeasible primal sub-problem in (3.12) for non-stabilizing K_1 is,

$$\mathcal{L}_1\left(K_1, P_1, Z_1\right) = -\mathrm{Tr}\left(Z_1\mathrm{S}\left(K_1, P_1\right)\right). \tag{3.15}$$

The master problem after the N_G^{th} GBD iteration is given as,

$$\begin{aligned}
&\min_{\mu,K} \quad \mu \\
&\text{Subject to the constraints,} \\
&\underline{K} \leq K \leq \overline{K}, \quad \mu \in \mathbb{R}, \quad X_0 = x_0 x_0^T, \\
&\mathcal{L}\left(K, P^i, Z^i\right) \leq \mu, \qquad i = 1, \ldots, N_f, \\
&\mathcal{L}_1\left(K, P_1^j, Z_1^j\right) \leq 0, \qquad j = 1, \ldots, N_{if},
\end{aligned} \tag{3.16}$$

where N_f and N_{if} are the number of stabilizing and non-stabilizing $K \in \mathcal{K}$ respectively computed till the N_G^{th} GBD iteration. The master problem is convex and after each GBD iteration feasible space reduces hence $\mu^{N_G} = \text{LBD}$ forms a non-decreasing sequence. The GBD converges to a stabilizing (K^*, P^*) when $\text{LBD} \geq \text{UBD} - \varepsilon_G$, where $\varepsilon_G > 0$ is a predefined GBD convergence criterion. Next, tests to check the stationarity and local minimality of K^* are presented.

3.3.4 Necessary and Sufficient Conditions for Minimum

The necessary conditions for K^* to be a local minimum point of problem in (3.11) are derived next. Let $k = \text{vec}\left(K\right), k^* = \text{vec}\left(K^*\right)$, where $\text{vec}\left(\cdot\right)$ is the vector representation of the matrix $(\cdot)$. The lower and upper bounds of k are $\underline{k}$ and $\overline{k}$ respectively. The conditions to determine if K^* is a stationary point are as follows,

Proposition 3.3.1. Let the GBD procedure converge to the point $K^* \in \mathcal{K}$ and let Z^*, P^* be computed from the feasible primal sub-problem (3.12). Then K^* is a stationary point for the problem in (3.11), if $-\text{Tr}\left(Z^* \nabla_k \text{S}\left(K^*, P^*\right)\right) + \delta_u - \delta_l = 0$, $\text{Tr}\left(Z^* \text{S}\left(K^*, P^*\right)\right) = 0$, and $\delta_l^T\left(\underline{k} - k\right) + \delta_u^T\left(k - \overline{k}\right) = 0$ for some $\delta_l \in \mathbb{R}^{qp}, \delta_l \geq 0, \delta_u \in$

$\mathbb{R}^{qp}, \delta_u \geq 0$ and $k = \text{vec}(K)$.

Proof. The Lagrangian for the problem (3.11) is,

$$\mathcal{L}_v(K,P) = \text{Tr}(PX_0) - \text{Tr}(Z\text{S}(K,P)) + \delta_l^T(\underline{k} - k) + \delta_u^T\left(k - \overline{k}\right) - \text{Tr}(\lambda_P P), \tag{3.17}$$

where $\delta_l \in \mathbb{R}^{qp}, \delta_l \geq 0, \delta_u \in \mathbb{R}^{qp}, \delta_u \geq 0, \lambda_P \in \mathcal{S}^n, \lambda_P \succeq 0$ are the Lagrangian dual variables. Taking gradient with respect to k at (K^*, P^*) and applying KKT conditions [51] gives the required result.

■

As K^* is a stabilizing SSOF controller gain, the KKT conditions corresponding to the variable P are automatically fulfilled. In practice, the stationarity conditions are implemented as $\left|\left(-\text{Tr}(Z^*\nabla_k\text{S}(K^*,P^*)) + \delta_u - \delta_l\right)_i\right| \leq \varepsilon_S$, $\text{Tr}(Z^*\text{S}(K^*,P^*)) = 0$ and $\delta_l^T(\underline{k} - k) + \delta_u^T\left(k - \overline{k}\right) = 0$ where $|(\cdot)|$ is the absolute value of $(\cdot)$, $(\cdot)_i$ is the i^{th} component of the vector $(\cdot)$ and $0 < \varepsilon_S < 1$ is a small predefined constant. To test the local minimality of the stationary point K^* the following condition is applied,

Proposition 3.3.2. If $H(K^*) = \nabla_k^2 J(P(K^*)) = \text{Tr}(\nabla_k^2 P(K^*) X_0) \succ 0$, then the stationary point $K^* \in \mathcal{K}$ computed using the GBD procedure is a local minimum point of (3.11).

Proof. P is a function of K through the Lyapunov like equation LE in (3.6). From theory of optimality [51], the condition for K^* to be a local minimum point is $H(K^*) = \nabla_k^2 J(P(K^*)) = \text{Tr}(\nabla_k^2 P(K^*) X_0) \succ 0$.

■

The procedure to compute $H(K^*)$ is presented next. Let $k = \text{vec}(K)$, $\frac{\partial Y}{\partial k_i} =$

$k_i, \frac{\partial^2 Y}{\partial k_i \partial k_j} = Y_{ij}, Y^* = Y(K^*)$. As P^* is the solution of LE in (3.6) at $K = K^*$ and K_i is always a constant, differentiation of LE with respect to component k_i of k at K^* gives,

$$\begin{aligned} &A_c^{*T} P_i + P_i A_c^* + \Theta = 0, \\ &\Theta = A_{ci}^T P^* + P^* A_{ci} + M_i^* + M_i^{*T}, \\ &M_i^* = C^T K_i^T R K^* C, \quad A_{ci} = -B K_i C. \end{aligned} \tag{3.18}$$

Differentiation of (3.18) with respect to component k_j of k at K^* gives,

$$\begin{aligned} &A_c^{*T} P_{ij} + P_{ij} A_c^* + \Upsilon + M_{ij} + M_{ij}^T = 0, \\ &\Upsilon = A_{cj}^T P_i^* + P_i^* A_{cj} + A_{ci}^T P_j^* + P_j^* A_{ci} \\ &M_{ij} = C^T K_i^T R K_j C, \\ &A_{ci} = -B K_i C, \quad A_{cj} = -B K_j C, \end{aligned} \tag{3.19}$$

where P_i, P_j are computed from (3.18). The components of $H(K^*)$ are computed as,

$$H_{ij}(K^*) = \mathrm{Tr}\left(P_{ij}(K^*) X_0\right). \tag{3.20}$$

The above equations (3.18) and (3.19) are Lyapunov like equations and can be easily solved. If the point K^* is not an local minimum, then GBD is restarted from a new design point K^{**} such that $\mathrm{Tr}(P^{**} X_0) < \mathrm{Tr}(P^* X_0)$.

3.3.5 New Design Point

The new design point is computed using the GPM for constrained optimization problems. The details of the method are skipped due to space limitation and the reader is referred to Chapter 5 of [24] for additional details. Let W be a matrix

whose columns are the linearly independent gradients of active constraints from the constraint set $\left\{\underline{k} \leq k \leq \overline{k}\right\}$ at the point k^*. If p_W are the number of linearly independent gradients of active constraints at k^*, then $W \in \mathbb{R}^{qp \times p_W}$. The new design point $k^{**} = \text{vec}\left(K^{**}\right)$ is given by,

$$\begin{aligned} k^{**} &= k^* - \alpha_r \left[I - W\left(W^T W\right)^{-1} W^T\right] \frac{\nabla_k J^*}{||\nabla_k J^*||}, \\ \nabla_k J^* &= \nabla_k \left(\text{Tr}\left(P\left(K^*\right) X_0\right)\right), \end{aligned} \tag{3.21}$$

where $\alpha_r > 0$ is a predefined step-size and $\nabla_k \left(\text{Tr}\left(P\left(K^*\right)\right)\right)$ can be computed using (3.18). If K^* is a saddle point, then $k^{**} = \text{vec}\left(K^{**}\right)$ is computed using,

$$\begin{aligned} k^{**} &= \underset{\underline{k} \leq k \leq \overline{k}}{\arg\min} \left\{J^*_{-1}, J^*_1, \ldots, J^*_{-i}, J^*_i, \ldots, J^*_{-qp}, f^*_{qp}\right\} \\ J^*_{-i} &= J\left(P\left(k^* - \alpha_{rs} e_i\right)\right), J^*_i = J\left(P\left(k^* + \alpha_{rs} e_i\right)\right), \end{aligned} \tag{3.22}$$

where, $\alpha_{rs} > 0$ is a predefined step-size and $e_i \in \mathbb{R}^{qp}$ is the standard basis. The values of α_r, α_{rs} should be such that the new k^{**} lies in the feasible domain and the resultant K^{**} is stabilizing. In the next section, the design optimization algorithm is presented.

3.4 Design Optimization Algorithm

The steps to solve the design optimization problem (3.11) are as follows,

1. Set LBD $= -\infty$, UBD $= +\infty, \rho_P, \varepsilon_G, \varepsilon_S, \alpha_r, \alpha_{rs}$ and GBD restart number $N_R = 0$. Let $N_f = 0$ and $N_{if} = 0$ be the indices for stabilizing and non-stabilizing K respectively. Set SSOF controller gain matrix bounds $\underline{K}, \overline{K}$. Input any feasible starting stabilizing design point $K^1 = K^s$ such that $K^s \in \mathcal{K}$.

2. Set GBD iteration number $N_G = 1, K^{N_G} = K^1$.

3. Test if K^{N_G} is stabilizing or non-stabilizing by examining if the Hurwitz property holds for $A_c^{N_G} = A - BK^{N_G}C$ or not.

   (a) If K^{N_G} is stabilizing. Solve the feasible primal problem (3.12) in Section 3.3.1 for fixed $K = K_{N_G}$ to compute K^{N_G}, P^{N_G} and Z^{N_G}. Store $K^{N_f} = K^{N_G}, P^{N_f} = P^{N_G}, Z^{N_f} = Z^{N_G}$ and set $N_f = N_f + 1$. If $J\left(P^{N_G}\right) <$ UBD, then set UBD $= J\left(P^{N_G}\right), K^* = K^{N_G}$.

   (b) If K^{N_G} is non-stabilizing. Solve the infeasible primal problem (3.13) in Section 3.3.2 for fixed $K_1 = K_{N_G}$ to compute $K_1^{N_G}, P_1^{N_G}$ and $Z_1^{N_G}$. Store $K_1^{N_{if}} = K^{N_G}, P^{N_{if}} = P^{N_G}, Z^{N_{if}} = Z^{N_G}$ and set $N_{if} = N_{if} + 1$.

4. Solve the master problem in (3.16).

   (a) If master problem is feasible. Compute μ and K. Set $N_G = N_G + 1$, LBD $= \mu, K^{N_G+1} = K$. Check the convergence of bounds using LBD $>$ UBD $- \varepsilon_G$. If convergence is achieved, then continue to Step 5 with $K = K^*$ otherwise go to Step 2.

   (b) If master problem is infeasible go to Step 5 with $K = K^*$.

5. For $K = K^*$ test the stationarity condition using ε_S and Section 3.3.4 for K^*. If K^* is stationary, test for local minimality.

   (a) If K^* is a non-stationary point or non-minimal stationary point, then, $N_R = N_R + 1$. Compute new stabilizing design point K^{**} using α_g, α_{rs} and Section 3.3.5. Go to Step 2 and with input $K^1 = K^{**}$.

(b) If K^* is a local minimum point go to Step 6.

6. When local minimum is achieved $J^{opt} = \text{UBD}, K^{opt} = K^*$. Using the synthesized design K^{opt} compute the solution matrix P^{opt} in (3.6).

In the next section, an engineering example justifying the proposed design approach is presented.

3.5 Example

Consider the lateral axis model of L-1011 aircraft at cruise flight condition controlled by SSOF control and studied in [68,77]. The details of the dynamic model can be obtained from [68,77]. The matrices $A \in \mathbb{R}^{7\times7}, B \in \mathbb{R}^{7\times2}, C \in \mathbb{R}^{4\times7}$ are state (system), input and output matrices respectively. The vectors $x \in \mathbb{R}^7, y \in \mathbb{R}^4, u \in \mathbb{R}^2$ are state, output and input vectors respectively. Due to the special dynamics of the L-1011 system, the SSOF controller gain $K \in \mathbb{R}^{2\times4}$ has the following structure,

$$K = \begin{pmatrix} k_{11} & 0 & k_{13} & 0 \\ 0 & k_{22} & 0 & k_{24} \end{pmatrix}. \tag{3.23}$$

The SSOF design optimization problem is formulated as in (3.11) and design optimization algorithm in Section 3.4 is applied. The values of input parameters for the

algorithm are,

$$\begin{aligned}
&\rho_P = 5, \quad \varepsilon_G = 0.1, \quad \varepsilon_S = 0.5, \\
&\alpha_r = 0.2, \quad \alpha_{rs} = 0.2, \quad X_0 = I, \\
&-7 \leq k_y \leq 7, \quad y = 11, 22, 13, 24, \\
&K^s = \begin{pmatrix} -6 & 0 & 0 & 0 \\ 0 & 0 & 0 & -1 \end{pmatrix}, \\
&Q = diag\{ \begin{pmatrix} 1 & 1 & 30 & 30 & 5 & 5 & 1 \end{pmatrix}^T \}, \\
&R = diag\{ \begin{pmatrix} 1 & 1 \end{pmatrix}^T \},
\end{aligned} \tag{3.24}$$

where k_y is the non-zero SSOF controller gain matrix entry. The algorithm in Section 3.4 converges to the optimal after 4 GBD restarts. The GBD iterations for each GBD restart are tabulated in Table 3.1.

GBD restart number	0	1	2	3	4
GBD iterations	19	3	3	19	42

Table 3.1: GBD iterations for each GBD restart

The optimal stabilizing SSOF controller gain and the objective function value computed are,

$$\begin{aligned}
K^{opt} &= \begin{pmatrix} -2.77 & 0 & 3.25 & 0 \\ 0 & -5.31 & 0 & -6 \end{pmatrix}, \\
J^{opt} &= \mathrm{Tr}\left(P^{opt} X_0\right) = 142.8.
\end{aligned} \tag{3.25}$$

The global minimum solution computed by exhaustively searching the design space $\mathcal{K}$ is $J^{global} = 142.76$. Thus, the stabilizing SSOF controller gain synthesized using

the algorithm in Section 3.4 converges to a global minimum solution within practical error limits.

3.6 Summary

In this chapter, a new formulation for the design of optimal stabilizing SSOF controller gain for LTI systems is presented. The original design problem has a quadratic constraint in the controller gain variables is reformulated into an equivalent optimization problem with a BMI constraint. The equivalent problem with BMI constraint is then solved using an iterative combination of GBD and GPM procedures. Necessary and sufficient conditions are derived to test the stationarity and minimality of the computed solution. Though it has been observed that the proposed algorithm leads at least to a local minimum solution, the proof guaranteeing it is not included in this work. The proposed formulation is applied to the design of a SSOF controller of an engineering system previously studied in [68, 77]. Application of the design optimization algorithm in Section 3.4 results in a globally minimum solution for the engineering problem which was verified by an exhaustive search procedure.

It has been observed during simulations that constants $\rho_P, \varepsilon_G, \varepsilon_S, \alpha_r, \alpha_{rs}$ influence the execution time of the algorithm proposed in Section 3.4. Small value of ε_G increase the number of GBD iterations. Small values of $\rho_P, \varepsilon_S, \alpha_r, \alpha_{rs}$ increase the number of GBD restarts.

Chapter 4: Sparse Feedback Design

A controller of a dynamical system should impart robustness to the system with good performance. Along with robustness and good performance, sparsity of the controller is also important as sparse feedback controller implies less complicated system.

In this chapter, a scalable sparse feedback controller design procedure is presented for the mixed $\mathcal{H}_2/\mathcal{H}_\infty$ control problem. Sparse controller design procedure for a class of second order systems such as structural systems and power systems with collocated sensors, actuators and disturbances is also proposed. The proposed formulation is justified with examples.

4.1 Introduction

The mixed $\mathcal{H}_2/\mathcal{H}_\infty$ control problem was first introduced in [88] where a necessary condition for minimizing an upper bound of the $\mathcal{H}_2$ norm with a $\mathcal{H}_\infty$ norm constraint was proposed. [89–91] solved the mixed $\mathcal{H}_2/\mathcal{H}_\infty$ control problem as a dual of the problem in [88] giving a sufficient condition for minimization. In [92], it was proved that the necessary condition in [88] is also sufficient and the sufficient condition in [89–91] is also necessary. In [93], it was demonstrated that for $\mathcal{H}_\infty$ control,

the static state feedback control delivers the same performance as the dynamic state feedback control. Dynamic feedback control involves controller dynamics making their use difficult as compared to the static feedback which is simple in structure, economically cheap and easy to implement. This makes static feedback a preferred choice for mixed $\mathcal{H}_2/\mathcal{H}_\infty$ control. Approaches in literature to design the static state and output feedback for mixed $\mathcal{H}_2/\mathcal{H}_\infty$ are discussed next. Convex optimization based approaches involving linear/bilinear matrix inequalities (LMIs/BMIs) were used to design static mixed $\mathcal{H}_2/\mathcal{H}_\infty$ controllers in [13, 94, 95, 95–99]. Other approaches include non-smooth optimization [14], Riccati equation based suboptimal approach [100], successive over-bounding of quadratic terms approach [101], exterior point approach [102]. An upper bound expression for the mixed $\mathcal{H}_\infty$ norm for collocated structural systems was derived in [66]. Although static feedback controllers have advantages, their centralized structure has drawbacks when the size of the system becomes large. A controller in a dynamical system can be viewed as a central station which gathers information from "distributed sensors", processes it and generates control action commands for "distributed actuators" [15]. When the size of the system is large, the centralized controller structure involves a lot of interconnections (for information transmission) which increases the system's complexity and creates maintenance issues [103]. This makes the prospect of having sparse controllers desirable. It should be noted that here sparse controller means that the number of interconnections between sensors and actuators should be as less as possible. Mathematically, a sparse controller signifies that the number of zeros in the gain matrix should be as large as possible.

Sparse and structured (full state or output) feedback controller design has been a very popular research problem due to its vast and diverse application scope as well as its difficult nature (known to be NP-hard [20]). Structured feedback is a subset of sparse feedback in which the positions of zeros in the gain matrix are predefined. For introducing sparsity in the controller gain matrix, ℓ_1-minimization framework is generally used [104]. Convex optimization based approaches used to design sparse and structured controllers are: quadratic invariance property for synthesizing sparse controllers [105, 106], Youla (like) parameterization based approaches [107, 108], rank constraint relaxation approach [83, 84], convexifying potential function approach [78], convex relaxation approach using polyhedral Lyapunov functions [109], structured dynamic output feedback controllers for a given $\mathcal{H}_\infty$ performance [110]. Convex approaches like [111–113] require the optimization variable to have a block diagonal structure. A scalable convex optimization based approach with block diagonal requirement and using chordal decomposition was presented in [114]. A convex optimization based globally convergent approach to synthesize $\mathcal{H}_2$ norm sparse controllers was proposed in the recent work [103]. An ADMM [25] based approach to design $\mathcal{H}_2$ norm sparse and structured controllers was used in [82, 104].

From the aforementioned literature survey, it can be concluded that the sparse mixed $\mathcal{H}_2/\mathcal{H}_\infty$ control is an important research problem. Although, convex optimization based methods are popular for sparse controller synthesis, their computational performance deteriorates as the size of the system increases. Also, scalable convex approaches require optimization variables to have a special block diagonal

structure. Approaches in literature for sparse and structured controller design are mainly concentrated for $\mathcal{H}_2$ control with little consideration to the $\mathcal{H}_\infty$ performance. Keeping in view the aforementioned shortcomings, the contributions of this chapter are as follows.

1. The ADMM based ℓ_1-minimization framework is adapted to design sparse mixed $\mathcal{H}_2/\mathcal{H}_\infty$ control feedback. The $\mathcal{H}_\infty$ performance is included in the constraint of the minimization problem. The proposed approach is scalable and is guaranteed to converge to a stationary solution under certain conditions.

2. Sparse feedback design procedure for mixed $\mathcal{H}_2/\mathcal{H}_\infty$ control for a class of second order system having collocated actuators, sensors and disturbances is presented. The class of system consists of structural systems and linearized power system/network swing equation. The proposed formulation also includes a method to select an initial stabilizing controller.

3. For the linearized swing equation, a computationally fast heuristic procedure which uses only matrix operations to design sparse (diagonal) output feedback controllers is proposed.

This chapter is organized as follows: Section 4.2 introduces the mixed $\mathcal{H}_2/\mathcal{H}_\infty$ control problem, Section 4.3 describes the sparse feedback controller design problem followed by Section 4.4 explaining the design procedure. Section 4.5 presents comments on the implementation of the design procedure, Section 4.6 studies a class of second order systems with its properties, Section 4.7 presents examples justifying the proposed formulation followed by summarizing remarks in Section 4.8.

4.2 Mixed $\mathcal{H}_2/\mathcal{H}_\infty$ Control

In this section, the mixed H_2/H_∞ control problem is described.

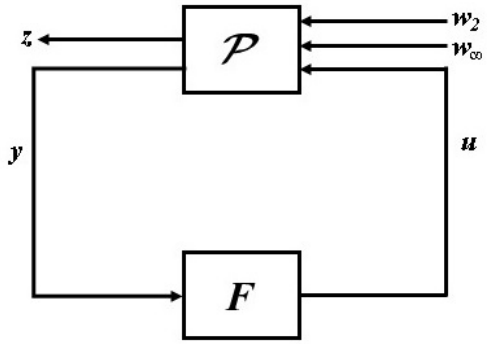


Figure 4.1: Plant $\mathcal{P}$ with controller F

Consider the plant $\mathcal{P}$ along with its controller F as shown in Figure 4.1. The plant $\mathcal{P}$ follows LTI dynamics and F is a static (full state or output) feedback controller. Mathematically, the plant-controller system is represented as,

$$\mathcal{P}: \quad \begin{pmatrix} \dot{x} \\ \hline z \\ y \end{pmatrix} = \left(\begin{array}{c|ccc} A & B_2 & B_\infty & B \\ \hline C_z & 0 & 0 & D_z \\ C & 0 & 0 & 0 \end{array}\right) \begin{pmatrix} x \\ \hline w_2 \\ w_\infty \\ u \end{pmatrix}, \tag{4.1}$$

$$u = Fy,$$

where $x \in \mathbb{R}^{n_x}$ is the state, $y \in \mathbb{R}^{n_y}$ is the measured output, $u \in \mathbb{R}^{n_u}$ is the control input and $w_2 \in \mathbb{R}^{n_{w_2}}, w_\infty \in \mathbb{R}^{n_{w_\infty}}$ are exogenous inputs. $A, B, C, B_2, B_\infty, C_z, D_z$ are system matrices of appropriate dimension. Static output feedback gain matrix F is also of appropriate dimension. When $C = I$, where I is an identity matrix, then F is a full state feedback gain. It should be noted that, throughout this

chapter, I denotes an identity matrix of appropriate dimension as required. z is the performance vector. The system in (4.1) has the following assumptions,

(A1) $(A, B), (A, B_2), (A, B_\infty)$ are stabilizable.

(A2) $(A, C), (A, C_z)$ are detectable.

(A3) $D_z^T \begin{pmatrix} D_z & C_z \end{pmatrix} = \begin{pmatrix} R_z & 0 \end{pmatrix}, R_z \succ 0.$

Here, matrix $X \succ (\succeq) 0$ means X is positive definite (semidefinite) matrix and X^T denotes the transpose of the matrix X.

The mixed $\mathcal{H}_2/\mathcal{H}_\infty$ control problem is stated for a given $\mathcal{H}_\infty$ norm $\gamma > 0$ as follows [14],

$$\begin{aligned} &\min_{F} \quad ||\mathcal{T}_{w_2 \to z}||_{\mathcal{H}_2}, \\ &\text{Subject to the constraints,} \\ &||\mathcal{T}_{w_\infty \to z}||_{\mathcal{H}_\infty} \leq \gamma, \\ &\text{System dynamics in (4.1),} \\ &(A + BFC) \quad \text{is Hurwitz.} \end{aligned} \tag{4.2}$$

Here, $\mathcal{T}_{w_{(.)} \to z}$ is the closed-loop transfer function for $w_{(.)} \to z$ and is a function of F. $||\cdot||_{\mathcal{H}_2}$ and $||\cdot||_{\mathcal{H}_\infty}$ denote the $\mathcal{H}_2$ and $\mathcal{H}_\infty$ norms [12] respectively. The problem in (4.2) is challenging mainly due to the non-smoothness of the $\mathcal{H}_\infty$ constraint. Instead of solving the difficult problem in (4.2), its suboptimal version [97, 100] is solved. The performance metric for the mixed $\mathcal{H}_2/\mathcal{H}_\infty$ is derived next.

4.2.1 Optimization Metric

The $\mathcal{H}_2$ performance of the system is measured by the squared $\mathcal{H}_2$ norm. Let F be a stabilizing static feedback controller such that $\widetilde{A} = A + BFC$ is Hurwitz. It is known that [97],

$$||\mathcal{T}_{w_2 \to z}||_{\mathcal{H}_2}^2 = \text{Tr}\left(B_2^T X_2 B_2\right), \tag{4.3}$$

where $\text{Tr}\,(\cdot)$ is trace of the matrix $(\cdot)$ and $X_2 \succ 0$ is the solution of the Lyapunov equation,

$$\begin{aligned} &\widetilde{A}^T X_2 + X_2 \widetilde{A} + \widetilde{C}_z^T \widetilde{C}_z = 0, \\ &\text{where} \quad \widetilde{C}_z = C_z + D_z F C. \end{aligned} \tag{4.4}$$

Also from [89, 97],

$$\text{Tr}\left(B_2^T X_2 B_2\right) \leq J\,(F) = \text{Tr}\left(B_2^T X_\infty B_2\right), \tag{4.5}$$

where $X_\infty \succ 0$ is the solution of the Riccati equation for a known $\mathcal{H}_\infty$ norm $\gamma > 0$,

$$\widetilde{A}^T X_\infty + X_\infty \widetilde{A} + \gamma^{-2} X_\infty B_\infty B_\infty^T X_\infty + \widetilde{C}_z^T \widetilde{C}_z = 0. \tag{4.6}$$

Hence to include the effect of γ, the objective $J\,(F)$ is minimized. Next, the formulation for the sparse feedback design problem is presented.

4.3 Sparse Feedback Controller Design Problem

In this section, the problem formulation for the sparse feedback design problem is presented. The sparse feedback design problem is designed on the lines of $\mathcal{H}_2$

sparse feedback problem in [104] as follows.

$$\begin{aligned} \min_{F} \quad & J(F) + \lambda g(F), \\ & g(F) = \sum_{i,j} W_{ij}|F_{ij}|. \end{aligned} \tag{4.7}$$

Here $\lambda \geq 0$ is a sparsity promoting constant with larger value of λ resulting in a sparser F. The weights $W_{ij} \geq 0$ where $(\cdot)_{ij}$ is the element in the $(i,j)^{th}$ position of the matrix $(\cdot)$. The weighted ℓ_1-norm, $\sum_{i,j} W_{ij}|F_{ij}|$ is the convex relaxation of the cardinality of F. When $W_{ij} = \frac{1}{|F_{ij}|}, F_{ij} \neq 0$ and $W_{ij} = \frac{1}{\varepsilon_W}, F_{ij} = 0, 0 < \varepsilon_W \ll 1$, the ℓ_1-norm coincides with the cardinality of F. The objective function in (4.7) is the sum of two functions. With a new variable G, the problem (4.7) is rewritten as follows.

$$\begin{aligned} \min_{F,G} \quad & J(F) + \lambda \sum_{i,j} W_{ij}|G_{ij}|, \\ & \text{Subject to the constraints,} \\ & F - G = 0. \end{aligned} \tag{4.8}$$

The augmented Lagrangian of (4.8) is,

$$\mathcal{L}_\rho(F, G, \Lambda) = J(F) + \lambda \sum_{i,j} W_{ij}|G_{ij}| + \mathrm{Tr}\left(\Lambda^T (F - G)\right) + \frac{\rho}{2}||F - G||_F^2, \tag{4.9}$$

where $||\cdot||_F$ is the Frobenius norm [18]. The ADMM solution procedure to solve the optimization problem (4.8) is discussed next.

4.4 Sparse Feedback Design Procedure

In this section the ADMM based sparse feedback algorithm presented in [104] is adapted to design sparse feedback controllers for mixed $\mathcal{H}_2/\mathcal{H}_\infty$ control. A preliminary property required in the design algorithm is proved first.

Lemma 4.4.1. For a given $\gamma > 0$, let F be a stabilizing gain such that $\widetilde{A} = A+BFC$ and $\widetilde{A}_\gamma = \widetilde{A} + \gamma^{-2} B_\infty B_\infty^T X_\infty$ are Hurwitz, $\widetilde{C}_z = C_z + D_z FC$ then,

$$\nabla_F J = 2\left[D_z^T \widetilde{C}_z + B^T X_\infty\right] Y C^T, \tag{4.10}$$

where $\nabla_F(\cdot)$ is the gradient of $(\cdot)$ with respect to F and $Y \succ 0$, $X_\infty \succeq 0$ are the solutions of,

$$\widetilde{A}^T X_\infty + X_\infty \widetilde{A} + \gamma^{-2} X_\infty B_\infty B_\infty^T X_\infty + \widetilde{C}_z^T \widetilde{C}_z = 0, \tag{4.11a}$$

$$\widetilde{A}_\gamma Y + Y \widetilde{A}_\gamma^T + B_2 B_2^T = 0. \tag{4.11b}$$

Proof. From [115], the derivative $J'dF$ where dF is the differential of F is,

$$J'dF = \mathrm{Tr}\left(\nabla_F J^T dF\right) = \mathrm{Tr}\left(B_2 B_2^T X'_\infty dF\right). \tag{4.12}$$

Differentiating (4.11b) with respect to F gives,

$$\begin{aligned} \widetilde{A}_\gamma^T X'_\infty dF + X'_\infty dF \widetilde{A}_\gamma + (BdFC)^T X_\infty + X_\infty (BdFC) \\ + (D_z dFC)^T \widetilde{C}_z + \widetilde{C}_z^T (D_z dFC) = 0. \end{aligned} \tag{4.13}$$

Multiplying (4.11b) by $X_\infty dF$ and (4.13) by Y. Subtracting new (4.13) from new (4.11b), taking trace and using (4.12) gives the required result (4.10).

■

Next subsections present the steps involved in the design procedure.

4.4.1 Initialization Step

Select the weight matrix W to be a matrix of ones of the size same as F. Set λ and γ. Choose initial stabilizing F^0, $G^0 = F^0$, $\Lambda^0 = 0$. Set convergence constants $\varepsilon_{FG}, \varepsilon_{GG}, \varepsilon_F$, ADMM iteration $k = 0$.

4.4.2 $F-$Minimization Step

The $F-$minimization involves solving the following problem,

$$F^{k+1} = \arg\min_{F} \quad \mathcal{L}_{\rho}\left(F, G^{k}, \Lambda^{k}\right). \tag{4.14}$$

The problem (4.14) is reformulated using $U^{k} = G^{k} - \left(\frac{1}{\rho}\right)\Lambda^{k}$ and completing the squares with respect to F as,

$$\begin{aligned} F^{k+1} &= \arg\min_{F} \quad \phi\left(F\right), \\ \phi\left(F\right) &= J\left(F\right) + \frac{\rho}{2}||F - U^{k}||_{F}^{2}. \end{aligned} \tag{4.15}$$

The problem (4.15) is solved using a modified version of Anderson-Moore algorithm [116] as follows.

1. Set $i = 0$, stabilizing $F_i = F^k$.

2. Using F_i, compute $X_{\infty i}$ from (4.11a).

3. Using F_i and $X_{\infty i}$ compute Y_i from (4.11b).

4. The necessary optimality condition of (4.14) using (4.10) and Assumption (A3) is $\nabla_F \phi = \nabla_F J + \rho\left(F - U^k\right) = 0$ which yields the following Sylvester equation,

$$2\left(R_z F C + B^T X_{\infty}\right) Y C^T + \rho\left(F - U^k\right) = 0. \tag{4.16}$$

   Using F_i and $X_{\infty i}$ and Y_i, compute $\overline{F}_i$.

5. Form $\widetilde{F}_i = \overline{F}_i - F_i$. Compute $F_{temp} = F^i + s_i \widetilde{F}_i$, where step-size s_i is calculated using Armijo rule [56, 104].

6. If $||\nabla_F \phi\left(F_i\right)||_F < \varepsilon_F$ then $F^{k+1} = F_i$, end $F-$minimization and go to $G-$minimization in Section 4.4.3. Else, $i = i + 1$ and $F_i = F_{temp}$ go to Step 2.

4.4.3 $G-$Minimization Step

The $G-$minimization involves solving the following problem,

$$G^{k+1} = \arg\min_{G} \quad \mathcal{L}_{\rho}\left(F^{k+1}, G, \Lambda^{k}\right). \tag{4.17}$$

The problem (4.17) is reformulated using $V^{k} = F^{k+1} + \left(\frac{1}{\rho}\right)\Lambda^{k}$ and completing the squares with respect to G as,

$$\begin{aligned} G^{k+1} &= \arg\min_{G} \quad \phi\left(G\right), \\ \phi\left(G\right) &= \lambda \sum_{i,j} W_{ij}|G_{ij}| + \frac{\rho}{2}||G - V^{k}||_{F}^{2}. \end{aligned} \tag{4.18}$$

The function $\phi\left(G\right)$ in (4.18) is rewritten as $\phi\left(G\right) = \sum_{i,j}\left(\lambda W_{ij}|G_{ij}| + \frac{\rho}{2}\left(G_{ij} - V_{ij}^{k}\right)^{2}\right)$ and the component-wise unique solution is [25, 104],

$$G_{ij}^{k+1} = \begin{cases} \left(1 - \frac{\lambda}{\rho}\frac{W_{ij}}{|V_{ij}|}\right)V_{ij}, & \text{if} \quad |V_{ij}| > \frac{\lambda}{\rho}W_{ij}, \\ 0, & \text{if} \quad |V_{ij}| \leq \frac{\lambda}{\rho}W_{ij}. \end{cases} \tag{4.19}$$

4.4.4 $\Lambda-$Update Step

The value of Λ is updated as,

$$\Lambda^{k+1} = \Lambda^{k} + \rho\left(F^{k+1} - G^{k+1}\right). \tag{4.20}$$

4.4.5 Convergence Testing Step

If $||F^{k+1} - G^{k+1}||_{F} \leq \varepsilon_{FG}$ and $||G^{k+1} - G^{k}||_{F} \leq \varepsilon_{GG}$, then ADMM convergence is achieved. Optimal values of unknowns are $F^{opt} = F^{k}, G^{opt} = G^{k}, \Lambda^{opt} = \Lambda^{k}$ and go

to next step in Section 4.4.6. Else, $k = k+1$ and go to ADMM step in Section 4.4.2 with F^k, G^k, Λ^k.

4.4.6 Reweighing Step

The ADMM procedure steps are from Section 4.4.2 to Section 4.4.5. For improving the sparsity of F, the ADMM procedure should be restarted by readjusting (reweighing) the weight matrix W with respect to F^{opt} as follows,

$$W_{ij} = \frac{1}{\left|F_{ij}^{opt}\right| + \varepsilon_W}, \quad 0 < \varepsilon_W \ll 1. \tag{4.21}$$

The ADMM procedure is restarted with the new weight matrix W as per (4.21) and $k = 0, F^0 = F^{opt}, G^0 = G^{opt}, \Lambda^0 = \Lambda^{opt}$. From computational experience $\varepsilon_W = 10^{-3}$ and repeating the ADMM procedure with new weights for 5 times gives good results.

4.5 Comments on Algorithm Implementation

Some points to be considered while implementing the sparse controller design algorithm are described in this section.

4.5.1 Selection of Initial Stabilizing F

The success of the algorithm depends upon availability of the initial stabilizing F. When full-state feedback controller is used, the initial stabilizing F is the standard LQR gain. When F is a output feedback controller, then the initial stabilizing F is computed using method described in [117]. It should be noted that

computing an initial stabilizing output feedback F is itself a challenging task and may not always be possible.

4.5.2 Selection of γ

The value of γ is selected with respect to the initial stabilizing F. For a stable closed loop system (in Hurwitz sense i.e all the eigenvalues of $\widetilde{A}$ have negative real parts), the analytical expressions for computing the lower bound (γ_{lb}) and upper bound (γ_{ub}) of γ are provided in [118]. Any γ satisfying $0 < \gamma_{lb} \leq \gamma \leq \gamma_{ub}$ is the $\mathcal{H}_\infty$ norm if the Hamiltonian matrix H for the ARE in (4.6) [118],

$$H = \begin{pmatrix} \widetilde{A} & \gamma^{-2} B_\infty B_\infty^T \\ -\widetilde{C}_z^T \widetilde{C}_z & -\widetilde{A}^T \end{pmatrix} \tag{4.22}$$

has no purely imaginary eigenvalues.

4.5.3 Convergence

The convergence of the sparse controller design algorithm depends on the selection of γ and sparsity parameter λ.

Proposition 4.5.1. For a given γ, λ, if the sparse controller design ADMM based algorithm described in Section 4.4 converges, then it converges to a critical point of (4.7).

Proof. Convergence of ADMM to (F^*, G^*, Λ^*) implies $F^* - G^* = 0$. F^* minimizes (4.14) implies $\nabla_F J(F^*) + \Lambda^* = 0$. G^* minimizes (4.17) implies $\lambda \partial g(G^*) - \Lambda^* = 0$. Here, ∂g is the sub-gradient of convex function g. Above discussion along with

$F^* - G^* = 0$ implies $\nabla_F J(F^*) + \lambda \partial g(F^*) = 0$ which is the necessary condition for optimality of (4.7). Hence, (F^*, G^*, Λ^*) is the critical point of (4.7).

■

4.5.4 Complexity of Algorithm

The ADMM algorithm described in Section 4.4 uses AREs or Lyapunov equations which can be solved cheaply with a computational complexity of $\mathcal{O}(n^3)$ [58] rather than the expensive semidefinite programming methods with complexity $\mathcal{O}(n^6)$ [119]. Thus, repeatedly solving the sparse design problem for different values of input parameters becomes a feasible task for the designer and will result in sparser controllers with acceptable γ.

4.5.5 Miscellaneous Points

The algorithm breaks down when γ is too small or λ is too large. In such situations, adjusting values of γ and λ will ensure smooth path for the ADMM procedure to achieve convergence with acceptable γ and sparsity.

4.5.6 Structured Feedback Design

Structured feedback for mixed $\mathcal{H}_2/\mathcal{H}_\infty$ control can be designed using Lemma 4.4.1 and method described in [82].

Next, sparse controller design procedure for a class of second order systems is

discussed.

4.6 Class of Second Order Systems

In this section, sparse controller design for a class of a second order dynamical systems with special properties is studied. The class of systems has dynamics as follows,

$$\begin{aligned} &M\ddot{\theta} + D\dot{\theta} + L\theta = B_u u + B_w w, \\ &y = C_y\dot{\theta}, \quad z = C_z \begin{pmatrix} \theta \\ \\ \dot{\theta} \end{pmatrix} + D_z u, \quad u = Fy. \end{aligned} \tag{4.23}$$

The dynamics in (4.23) is followed by several practical systems such as structural systems and linearized power system/network swing equation. M, D, L are inertia, damping and stiffness or graph Laplacian matrices respectively. The dynamics in (4.23) is rewritten as in (4.1) as,

$$\begin{aligned} &x = \begin{pmatrix} \theta & \dot{\theta} \end{pmatrix}^T, \quad w_2 = w_\infty = w, \\ &A = \begin{pmatrix} 0 & I \\ \\ -M^{-1}L & -M^{-1}D \end{pmatrix}, \quad B = \begin{pmatrix} 0 \\ \\ M^{-1}B_u \end{pmatrix}, \\ &B_2 = B_\infty = \begin{pmatrix} 0 \\ \\ M^{-1}B_w \end{pmatrix}, \quad C = \begin{pmatrix} 0 & C_y \end{pmatrix}. \end{aligned} \tag{4.24}$$

The matrix D_z should satisfies the assumption (A3) in Section 4.2. It is also assumed that the sensors, actuators as well as disturbances are collocated i.e.,

$$C_z = \begin{pmatrix} 0 & C_{z0} \\ 0 & 0 \end{pmatrix}, \quad D_z = D_z, \tag{4.25}$$
$$B_u = B_w = C_y^T = C_{z0}^T.$$

Due to specific properties, both structural systems and linearized swing equation are separately discussed next. Let $\alpha_{max}(Z)$ be the largest eigenvalue of the symmetric matrix Z. Matrix $Z^\dagger$ be the Moore-Penrose generalized inverse [18] of matrix Z. For a given real matrix Z, the matrix $Z^\perp$ is the orthogonal complement such that $Z^\perp Z = 0$ and $Z^\perp Z^{\perp T} \succ 0$.

4.6.1 Structural Systems

For structural systems (4.23), M, D, L are symmetric positive definite matrices. Two properties from [66] related to structural systems are stated below.

Theorem 4.6.1. [66] Consider a structural system (4.23) and its state space representation (4.24) with assumptions (4.25) and $D_z = 0$. The system has an open loop (i.e. $u = 0$) $\mathcal{H}_\infty$ norm γ_0 which satisfies $\gamma_0 < \gamma_{max} = \alpha_{max}\left(B_u^T D^{-1} B_u\right)$.

Theorem 4.6.2. [66] Consider a structural system (4.23) and its state space representation (4.24) with assumptions (4.25) and $D_z = 0$. For any $\gamma_0 < \gamma_{max}$ there exists a symmetric matrices $\widetilde{F}_0$ and F_0 as follows.

1. If B_u is square and invertible, then

$$F_0 \succ \widetilde{F}_0 = \frac{1}{\gamma_0} I - B_u^{-1} D B_u^{-1^T}. \tag{4.26}$$

2. If $B_u B_u^T$ is singular, then

$$F_0 \succ \widetilde{F}_0 = B_u^{\dagger} \left[D B_u^{\perp T} \left(B_u^{\perp} D B_u^{\perp T} \right)^{-1} B_u^{\perp} D - D + \frac{1}{\gamma_0} B_u B_u^T \right] B_u^{\dagger T}. \tag{4.27}$$

The F_0 computed above is such that for $F = -F_0$, the matrix $(A + BFC)$ is Hurwitz and the closed loop system has an $\mathcal{H}_\infty$ norm less than γ_0.

Remark 4.6.1. As B_u is a full column rank matrix with dimension $n_x \times n_u$, its Moore-Penrose generalized inverse is $B_u^{\dagger} = \left(B_u^T B_u\right)^{-1} B_u^T$. The orthogonal complement of B_u which is $B_u^{\perp}$ is computed using the singular value decomposition (SVD) [18] of B_u. The SVD of $B_u = \left(P_{n_u} | P_{n_x - n_u} \right) \begin{pmatrix} \Sigma_{n_u} \\ \hline 0 \end{pmatrix} Q_{n_u}^T$. Then $B_u^{\perp} = P_{n_x - n_u}^T$, where $P_{n_x - n_u}$ is a matrix of dimension $n_x \times (n_x - n_u)$.

When $D_z \neq 0$, then $\widetilde{F}_0$ computed as in Theorem 4.6.2 results in a closed loop system with $\mathcal{H}_\infty$ norm greater than γ_0 as proved in Proposition 4.6.2 next.

Proposition 4.6.2. Consider the structural systems (4.23) and (4.24) with assumptions (4.25). Let $\widetilde{F}_0$ be computed for given $\gamma_0 > 0$ and $D_z = 0$ as in Theorem 4.6.2. If $D_z \neq 0$, then the closed loop $\mathcal{H}_\infty$ norm of the system is less than γ with $\gamma > \gamma_0$.

Proof. Let for the closed loop system (4.23) and (4.24), $\widetilde{A} = A + B\widetilde{F}_0 C$ with $D_z = 0$. $\widetilde{A}$ is stable. The closed loop system has an $\mathcal{H}_\infty$ norm less than γ_0 and satisfies the bounded real lemma (BRL) [87] for a symmetric Lyapunov matrix

$P = \begin{pmatrix} L & 0 \\ 0 & M \end{pmatrix} \succ 0$ as follows,

$$\begin{pmatrix} \widetilde{A}^T P + P\widetilde{A} & PB_\infty & C_z^T \\ B_\infty^T P & -\gamma_0 I & 0 \\ C_z & 0 & -\gamma_0 I \end{pmatrix} \preceq 0. \tag{4.28}$$

Let $\widetilde{C}_z = C_z + D_z \widetilde{F}_0 C$ and $\Theta = \widetilde{A}^T P + P\widetilde{A}$ then,

$$\begin{pmatrix} \Theta & PB_\infty & \widetilde{C}_z^T \\ B_\infty^T P & -\gamma I & 0 \\ \widetilde{C}_z & 0 & -\gamma I \end{pmatrix} = \underbrace{\begin{pmatrix} \Theta & PB_\infty & C_z^T \\ B_\infty^T P & -\gamma I & 0 \\ C_z & 0 & -\gamma I \end{pmatrix}}_{BRL_1} + \underbrace{\begin{pmatrix} 0 & 0 & (D_z F_0 C)^T \\ 0 & 0 & 0 \\ D_z F_0 C & 0 & 0 \end{pmatrix}}_{BRL_2}. \tag{4.29}$$

From (4.28), $BRL_1 \preceq 0$ for $\gamma = \gamma_0$ and BRL_2 is a symmetric real indefinite matrix with 0 (zero) trace. Then, $BRL_1 + BRL_2 \preceq 0$ for $\gamma > \gamma_0$. Hence for $D_z \neq 0$, the closed loop $\mathcal{H}_\infty$ norm of the system is less than γ with $\gamma > \gamma_0$.

■

Remark 4.6.3. From Theorem 4.6.2 for $F_0 = \widetilde{F}_0 + c_f I$ with $c_f > 0$ and $D_z = 0$, the closed loop system will have an $\mathcal{H}_\infty$ norm less than γ_0. When $D_z \neq 0$, suitable values of c_f and γ are selected using Section 4.5.2 and then the sparse feedback design procedure is applied. To get a better design, the design procedure is repeated for various c_f, γ.

Next, properties of the linearized swing equation are discussed.

4.6.2 Linearized Swing Equation

For the linearized swing equation (4.23), M, D are diagonal positive definite matrices. The Laplacian matrix L represents the network interactions and is symmetric positive semidefinite i.e. $L = L^T \succeq 0$. Let n_θ be the dimension of the vector θ in (4.23), $\mathbb{1}_{n_\theta}$ be a vector of 1's with dimension n_θ, $\mathbb{0}$ be a vector of 0's of appropriate dimension as required and $\eta_0 = \begin{pmatrix} \mathbb{1}_{n_\theta} \\ \mathbb{0} \end{pmatrix} \in \mathbb{R}^{2n_\theta}$. L has a 0 (zero) eigenvalue with the corresponding eigenvector $\mathbb{1}_{n_\theta}$ i.e. $L\mathbb{1}_{n_\theta} = \mathbb{0}$. The matrix A is not Hurwitz but marginally stable i.e. A has one 0 (zero) eigenvalue corresponding to the marginally stable mode η_0. The remaining eigenvalues of A are stable having negative real parts. Next lemma will show that the marginally stable mode η_0 is not observable.

Lemma 4.6.1. Consider the linearized swing equation (4.23) and (4.24) with assumptions (4.25). If $C\eta_0 = \mathbb{0}$ then the marginally stable mode η_0 corresponding to the eigenvalue 0 is not observable.

Proof. As $A\eta_0 = \mathbb{0}$, $C\eta_0 = \mathbb{0}$ implies from [12] the required result.

■

The unobservable mode η_0 does not appear in the performance output z and hence, the system has a finite $\mathcal{H}_2$ norm as shown in [120]. If the marginally stable mode is eliminated then the remaining system is stable and has a finite $\mathcal{H}_\infty$ norm. To eliminate the marginally stable mode, the system needs to be transformed into a basis orthogonal to η_0. Consider the SVD of matrix A, $A = U_A \Lambda_A V_A^T$. V_A is a

$2n_\theta \times 2n_\theta$ matrix with the first $2n_\theta - 1$ column vectors orthogonal η_0. Let, Φ be the $2n_\theta \times (2n_\theta - 1)$ transformation matrix composed of the first $2n_\theta - 1$ column vectors of the matrix V_A such that,

$$\Phi^T \eta_0 = \mathbb{0}, \quad \Phi^T \Phi = I, \quad \Phi\Phi^T = I - \frac{1}{n_\theta}\eta_0\eta_0^T. \tag{4.30}$$

The new state vector in the transformed basis is $\xi = \Phi^T x \in \mathbb{R}^{2n_\theta - 1}$. The system matrices are transformed as,

$$\begin{aligned} &\mathcal{A} = \Phi^T A \Phi, \quad \mathcal{B} = \Phi^T B = \mathcal{B}_\infty, \\ &\mathcal{C} = C\Phi, \quad \mathcal{C}_z = C_z\Phi, \quad \mathcal{D}_z = D_z. \end{aligned} \tag{4.31}$$

The transformed system $(\mathcal{A}, \mathcal{B}, \mathcal{C}, \mathcal{B}_\infty, \mathcal{C}_z, \mathcal{D}_z)$ is stable. The Proposition 4.6.4 presented next shows that the Theorem 4.6.1, Theorem 4.6.2 and Proposition 4.6.2 in Section 4.6.1 derived for structural systems are also applicable for the transformed linearized swing equation.

Proposition 4.6.4. Theorem 4.6.1, Theorem 4.6.2 and Proposition 4.6.2 in Section 4.6.1 derived for structural systems are also applicable for the transformed linearized swing equation $(\mathcal{A}, \mathcal{B}, \mathcal{C}, \mathcal{B}_\infty, \mathcal{C}_z, \mathcal{D}_z)$.

Proof. Consider the Lyapunov matrix $\widetilde{P} = \Phi^T P \Phi$ with $P = \begin{pmatrix} L & 0 \\ 0 & M \end{pmatrix}$. $P \succeq 0$ but $\widetilde{P} \succ 0$ as the transformed basis does not contain the marginally stable mode η_0. As the transformed system $(\mathcal{A}, \mathcal{B}, \mathcal{C}, \mathcal{B}_\infty, \mathcal{C}_z, \mathcal{D}_z)$ is stable then from BRL [87] $\exists \gamma > 0$

such that for $\mathcal{D}_z = 0, u = 0$,

$$\mathcal{M} = \begin{pmatrix} \mathcal{A}^T\widetilde{P} + \widetilde{P}\mathcal{A} & \widetilde{P}\mathcal{B}_\infty & \mathcal{C}_z^T \\ \mathcal{B}_\infty^T\widetilde{P} & -\gamma I & 0 \\ \mathcal{C}_z & 0 & -\gamma I \end{pmatrix} \preceq 0. \tag{4.32}$$

Using (4.30) and (4.31),

$$\mathcal{M} = \Pi^T \underbrace{\begin{pmatrix} A^TP + PA & PB_\infty & C_z^T \\ B_\infty^TP & -\gamma I & 0 \\ C_z & 0 & -\gamma I \end{pmatrix}}_{BRL_P} \Pi \preceq 0, \quad \text{where} \quad \Pi = \begin{pmatrix} \Phi & 0 & 0 \\ 0 & I & 0 \\ 0 & 0 & I \end{pmatrix}. \tag{4.33}$$

For $\mathcal{M} \preceq 0$ requires $BRL_P \preceq 0$. The required result now follows from [66] and Section 4.6.1.

■

For $B_u = I$, for the swing equation with $D_z = 0$, F_0 computed using Theorem 4.6.1 is a sparse diagonal stabilizing controller gain. But when $D_z \neq 0$, the sparse (diagonal) feedback can be designed using only matrix operations as shown in Example 4.7.2 in Section 4.7.

Next, examples are presented justifying the sparse controller design design procedure.

4.7 Examples

In this section two examples justifying the proposed mixed $\mathcal{H}_2/\mathcal{H}_\infty$ sparse controller design procedure are presented. The first example is a structural system

while the second example is a linearized swing equation for a power network adapted from [120].

4.7.1 Structural System Example

Consider a chain of mass-spring-damper blocks of length $N = 50$. Each block has values of mass, stiffness and damping coefficient to be unity. This leads to a second order system similar to (4.23). For this case inertia matrix $M = I$ where I is a 50×50 identity matrix. The damping matrix D and stiffness matrix L are 50×50 symmetric tridiagonal matrices with 2 at each main diagonal place and -1 at each first sub- and super-diagonal places. $B_u = B_\infty = B_2 = C_{z0} = I$, $D_z = \begin{pmatrix} 0 \\ \sqrt{10}I \end{pmatrix}$. The state vector is of dimension 100 and the state matrix is of dimension 100×100. The system is controlled by a sparse state feedback (SpSF) and a sparse output feedback (SpOF) control. The $\mathcal{H}_\infty$ norm is selected as $\gamma = 8$. The analysis and comparison for the SpSF and SpOF cases is presented next.

4.7.1.1 Sparse Full State Feedback

In this case the complete state vector is measured which implies C is an identity matrix of dimension 100. The initial stabilizing F i.e F_{SF}^0 is the LQR feedback computed from the following,

$$\begin{aligned} &A^T X + XA + C_z^T C_z - XB \left(D_z^T D_z\right)^{-1} B^T X = 0, \\ &F_{SF}^0 = -\left(D_z^T D_z\right)^{-1} B^T X. \end{aligned} \tag{4.34}$$

The dimension of F_{SF} is 50×100 which means F has 5000 unknown variables and $\lambda = 0.00248$. The structure of synthesized F is as shown in Figure 4.2 with the dots representing non-zero values.

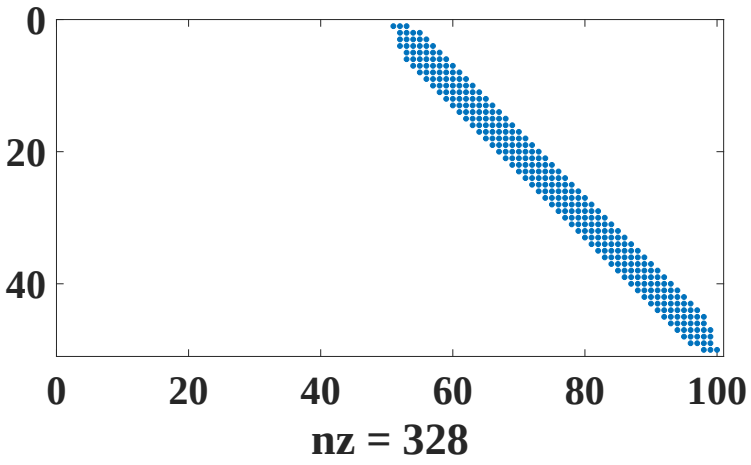


Figure 4.2: Designed sparse F_{SF}

4.7.1.2 Sparse Output Feedback

In this case only the velocity components of the state are measured which implies $C = \begin{pmatrix} 0 & I \end{pmatrix}$ of dimension 50×100. The initial stabilizing F i.e F_{OF}^0 is computed as per Proposition 4.6.2 with $c_f = 4$. The dimension of F_{OF} is 50×50 which means F has 2500 unknown variables and $\lambda = 0.00285$. The structure of synthesized F is as shown in Figure 4.3 with the dots representing non-zero values.

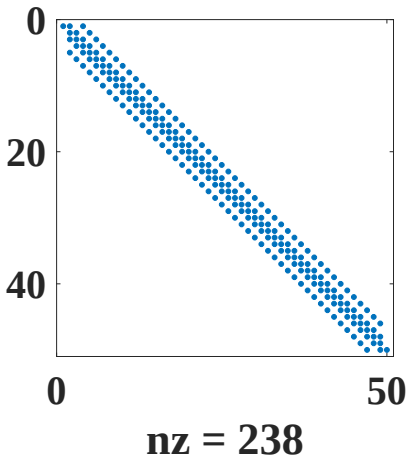


Figure 4.3: Designed sparse F_{OF}

Parameter	Full State Feedback	Output Feedback
λ	0.00248	0.00285
No. of unknowns	5000	2500
Initial non-zero F_{ij}	5000	148
Optimal non-zero F_{ij}	328	238
Initial J^0	35.12	410.52
Optimal J	35.31	35.53
Initial squared $\mathcal{H}_2$ norm	33.64	398.5
Optimal squared $\mathcal{H}_2$ norm	33.81	34.01
Execution time (CPU seconds)	26.48	60.69

Table 4.1: Sparse full state and output feedback comparison

4.7.1.3 Comparison of Results

The proposed algorithm is programmed in MATLAB [61] on a 64-bit laptop computer with core i7-7500 processor and 16 GB RAM. The Table 4.1 shows the comparison of results for sparse full state and output feedback controllers. It is interesting to note from Figure 4.2 that the state feedback only uses the velocity measurements. This is because in the performance index z, the weight matrix C_z only multiplies the velocity components of the state. The final designed controllers for both full state and output feedback are dependent on the choice of the initial stabilizing F. In case of full state feedback, the initial and optimal squared $\mathcal{H}_2$ norms are almost the same. For the output feedback case, although the initial stabilizing F_0 computed using Proposition 4.6.2 is sparse and tridiagonal with initial 94.08% sparsity but the system has a large squared $\mathcal{H}_2$ norm. The optimal F has 90.48% sparsity with squared $\mathcal{H}_2$ norm less than one-tenth of the initial. Thus by sacrificing 3.6% sparsity, the performance of the system has improved more than 10 times. From Table 4.1, it is evident that the execution time of the design procedure is very less and this facilitates the application of the procedure for different values of c_f, γ, λ in order to obtain a better F.

4.7.2 Linearized Swing Equation Example

In this section, a 12 bus three-area power network adapted from [120] is studied to show the efficacy of Proposition 4.6.4. The matrices M, D, K of dimension 9×9

are as follows.

$$M = \begin{pmatrix} 37.24 & 12.41 & 7.22 & 35.38 & 35.38 & 12.73 & 37.24 & 37.24 & 37.24 \end{pmatrix}^T,$$

$$D = \begin{pmatrix} 1.95 & 3.34 & 2.71 & 2.29 & 1.11 & 4.77 & 2.23 & 2.23 & 7.50 \end{pmatrix}^T,$$

$$L = \begin{pmatrix} 20.22 & -7.82 & -12.40 & 0 & 0 & 0 & 0 & 0 & 0 \\ -7.82 & 38.82 & -31.00 & 0 & 0 & 0 & 0 & 0 & 0 \\ -12.40 & -31.00 & 58.89 & 0 & 0 & -7.81 & 0 & 0 & -7.68 \\ 0 & 0 & 0 & 20.55 & -7.81 & -12.74 & 0 & 0 & 0 \\ 0 & 0 & 0 & -7.81 & 40.48 & -32.67 & 0 & 0 & 0 \\ 0 & 0 & -7.81 & -12.74 & -32.67 & 61.67 & 0 & 0 & -8.45 \\ 0 & 0 & 0 & 0 & 0 & 0 & 19.79 & -7.80 & -11.99 \\ 0 & 0 & 0 & 0 & 0 & 0 & -7.80 & 38.37 & -30.57 \\ 0 & 0 & -7.68 & 0 & 0 & -8.45 & -11.99 & -30.57 & 58.69 \end{pmatrix}.$$

$B_u = B_\infty = B_2 = C_{z0} = I$ where I is a 9×9 identity matrix. The system matrix A is of dimension 18×18. For $D_z = 0$, by using Proposition 4.6.4, γ_{max} and F_0 for $\gamma_0 < \gamma_{max}$ are computed by using matrix operations. When $D_z \neq 0$, from Proposition 4.6.4, the new closed loop $\gamma > \gamma_0$ and satisfies $0 < \gamma_{lb} < \gamma < \gamma_{ub}$. γ_{lb}, γ_{ub} are computed as per Section 4.5.2 for $\widetilde{C}_z = C_z + D_z F_0 C$. The new stabilizing feedback gain $F = F_0 + c_f I$ is computed by adjusting γ and c_f. As F_0 is a sparse diagonal matrix, the resulting F is also diagonal. For the considered example, with $D_z = 0$ gives $\gamma_{max} = 0.9$ and F_0 is computed for $\gamma_0 = 0.8$ as per Proposition

4.6.4. When $D_z = \begin{pmatrix} 0 \\ 2I \end{pmatrix}$ where I is a 9×9 identity matrix, $\gamma_{lb} = 2.1, \gamma_{ub} = 70.5$. The optimal F is computed by adjusting values of γ and c_f The optimal values are: $\gamma = 2.2$ and $c_f = 1.8$. Since only matrix operations, are utilized, the design procedure although heuristic is very fast which makes it practically very useful. This justifies Proposition 4.6.4. To further optimize the performance, a structured feedback design problem as per Section 4.5.6 can be solved using the computed F as initialization and diagonal structure as constraint.

4.8 Summary

In this chapter, an ADMM based computationally fast and scalable procedure to design sparse mixed $\mathcal{H}_2/\mathcal{H}_\infty$ feedback controllers has been presented. The scalability of the design procedure facilitates its usage for large scale practical applications. Sparse feedback design for a class of second order systems such as structural systems and power systems/networks which are widely applied in real world has also been described. While selection of initial stabilizing controller still remains a challenge for the general sparse output feedback problem, it has been successfully resolved for the class of second order systems described in this chapter.

Chapter 5: Optimal Actuator Placement

This chapter is based on the publication [121].

A completely controllable linear dynamical system can be steered from any given initial state to any specified final state with an application of input control energy. The input control energy is provided through a combination of actuators. It is desirable to have a limited number of actuators, which also presents the possibility of multiple actuator combinations that render the system completely controllable. Hence, the optimal actuator placement problem very important in system design. Previous studies have been mainly focused on solving the optimal actuator placement problem using greedy heuristic methods which can provide a sub-optimal solution.

In this chapter, the optimal actuator placement problem is presented as a $0/1-$MISDP problem, and is solved using the branch-and-bound procedure [27]. The problem formulation can be applied to both stable and unstable systems, and the solution procedure does not require an initial controllable actuator combination (starting point). Numerical simulations performed on two examples yield the global optimal solution for the optimal actuator placement problem.

5.1 Introduction

Actuators are utilized to control many complex systems such as biological networks [122, 123], social networks [124], electrical smart grids [125] and traffic system networks [126].

The concept of controllability was first introduced by Kalman [127] ,and since then it has been researched extensively. The work [127] answered the question whether the system is completely controllable or not, and sheds some light regarding the "quality" [64] of a completely controllable system. The quality of a completely controllable can be neglected if there exists a unique combination of actuators. If multiple combinations of actuators are able to completely control a certain system, then it is useful to order the various actuator combinations with respect to a "metric" so that non-optimal selection of actuators can be avoided.

The actuators require "control energy" to actuate (control) the system, and this concept can be utilized for ordering the potential actuator combinations. One of the earliest work on optimally selecting the actuator combination based on the control energy perspective was reported in [128]. Three physical measures, based on the control energy concept, were proposed in [64] to quantify the quality of a completely controllable system. The three measurable quantities are determinant, trace and the maximal eigenvalue of the inverse characteristic controllability grammian [64]. In the recent work [129], two more energy based controllability criteria were specified, namely, trace and minimum eigenvalue of the controllability grammian. In [130], it has been demonstrated that the trace of grammian as a metric may

not automatically ensure controllability and can lead to a poor choice of actuator combination.

An actuator is a source of input control energy consumption. The optimal actuator placement problem involves selection of an actuator combination using a limited number of actuators such that the system is completely controllable and consumes minimum energy. This problem was shown to be NP-hard [16]. Actuator placement for linear systems based on $\mathcal{H}_2$ and $\mathcal{H}_\infty$ optimization was presented in [131]. An iterative procedure based on the ADMM [25] procedure was presented in [132]. A framework for structural input and control configuration selection to achieve structural controllability was presented in [133].

It has been proved that the control energy is equivalent to the trace of the inverse of controllability grammian metric and is a supermodular function [2, 129]. This supermodularity property is used to solve the actuator placement problem using a greedy heuristic procedure [134] to obtain a solution which is provably close to the optimal solution. Actuator placement problem was solved as a leader selection problem in network consensus dynamics in [135] where the leader states act as control inputs. The solution procedure in [135] is based on a greedy heuristic procedure which used supermodularity property of the mean square error of link noise. In [16], the actuator placement problem was solved using greedy heuristic algorithm which maximizes the rank increase of the controllability matrix. In [2], the actuator placement problem was solved iteratively by using supermodularity. The available formulations for the optimal actuator placement problems are solved using procedures based on greedy heuristic algorithm. The greedy procedure does

not guarantee global optimality of the obtained solution. In the related work [129], authors assume a controllable system to place additional actuators to minimize the input control energy metric using the greedy heuristic procedure. This assumption of an initially controllable system introduces sub-optimality in the solution. In all the aforementioned formulations, it was assumed that trace of inverse of the controllability gramian is supermodular. In [136], it was proved that trace of inverse of the controllability gramian is not supermodular. Hence, no guarantee can be established on the computed solution.

In this work, a novel formulation to solve the optimal actuator placement problem for a linear system is presented. The nonlinear mixed integer optimal actuator placement problem is reformulated into an equivalent novel linear $0/1-$MISDP problem using a convex relaxation technique. The proposed formulation is applied to both stable and unstable chain of integrators [2] (with states ≤ 25). The aforementioned examples, when solved using a branch-and-bound procedure, leads to the global optimal actuator placement solution which is verified by exhaustive search procedure, and is also compared with the greedy heuristic procedure. It should be noted that no theoretical guarantee regarding the optimality or global nature of the computed solution has been provided in this work.

The chapter is organized as follows: In section 5.2, the formulation of the problem is developed which is followed by the solution procedure in Section 5.3. In Section 5.4, numerical examples using the proposed approach are studied, and a summary of the chapter is presented in Section 5.5.

5.2 Problem Formulation

Consider the continuous time linear system,

$$\begin{aligned} \dot{x}(t) &= Ax(t) + Bu(t), \\ t &\in \mathbb{R}, \quad t \geq 0, \end{aligned} \tag{5.1}$$

where $A = (A_{ij}) \in \mathbb{R}^{n\times n}, B = (B_{ij}) \in \mathbb{R}^{n\times n}$, are system matrices, $x \in \mathbb{R}^{n\times 1}, x_0 \in \mathbb{R}^{n\times 1}, u \in \mathbb{R}^{n\times 1}$ denote the state vector, initial state vector and control input of the system respectively. The matrix B is a 0/1-diagonal matrix, i.e., $B_{ii} \in \{0, 1\}$ and $B_{ij} = 0, \quad \forall i \neq j$. The domain of B is denoted by $\mathcal{D}_B$. If $B_{ii} = 1 \implies$ the state x_i receives input, while $B_{ii} = 0 \implies$ the state x_i does not receive any input. For a matrix X, the notation $X \succeq (\succ) 0$ implies X is a positive semidefinite (definite) matrix and X^T denotes the transpose of matrix X. $\mathcal{S}^n$ denotes the space of $n \times n$ real symmetric matrices.

For the pair (A, B) in (5.1) the controllability grammian W_c, for any finite $t_1 \in \mathbb{R}$, $t_1 > 0$ and initial condition $x(0) = x_0$, $x_0 \in \mathbb{R}^{n\times 1}$ can be written as [2, 137],

$$W_c(t_1) = \int_0^{t_1} e^{At} B B^T e^{A^T t} dt. \tag{5.2}$$

The control energy for the system in (5.1) is,

$$\text{Control Energy} = \min_{u, x(0)=x_0} \int_0^{t_1} u^T(t)\, u(t)\, dt. \tag{5.3}$$

As $t_1 \to \infty$, $W_c(t_1) \to W_c$ the control energy is related to the controllability grammian as [137],

$$\text{Control Energy} = \min_{u, x_0} \int_0^{\infty} u^T(t)\, u(t)\, dt = x_0^T W_c^{-1} x_0. \tag{5.4}$$

If x_0 is assumed to be a random vector uniformly distributed on a unit sphere with zero mean and unit variance, then the average control energy is,

$$\text{Average Control Energy} = \text{Tr}\left(W_c^{-1}\right), \tag{5.5}$$

where $\text{Tr}(\cdot)$ represents the trace of $(\cdot)$.

The controllability grammian W_c can be computed as [138],

Case 1: If A is Hurwitz, then W_c is the unique solution of the equation,

$$AW_c + W_cA^T + BB^T = 0. \tag{5.6}$$

Case 2: If A is not Hurwitz but the pair (A, B) is stabilizable, then X is the stabilizing solution to,

$$XA + A^TX - XBB^TX = 0. \tag{5.7}$$

Let $F = -B^TX$, then W_c is the solution to,

$$(A + BF)W_c + W_c(A + BF)^T + BB^T = 0. \tag{5.8}$$

If A is stable then $X = 0$ and (5.8) reduces to (5.6).

Another objective of actuator placement is the use of a limited number of actuators to control the system. Taking into consideration minimum control energy

and limited number of actuators the optimal actuator placement problem is,

$$\begin{aligned}
&\min_{u,B\in\mathcal{D}_B} \quad \int_0^\infty u^T(t)\,u(t)\,dt \\
&\text{Subject to the constraints,} \\
&\dot{x}(t) = Ax(t) + Bu(t), \\
&x(0) = x_0, \\
&\mathrm{Tr}(B) \le B_{max}, \\
&(A,B) \quad \text{is controllable,}
\end{aligned} \tag{5.9}$$

where B_{max} is the upper limit on the number of actuators to be used.

Using the average control energy, the optimal actuator placement problem, independent of the initial state and time with $\widehat{W}_c = W_c^{-1}$, can be formulated as follows,

$$\begin{aligned}
&\min_{B\in\mathcal{D}_B, X\succeq 0, \widehat{W}_c\succeq 0} \quad \mathrm{Tr}\left(\widehat{W}_c\right) \\
&\text{Subject to the constraints,} \\
&\mathrm{Tr}(B) \le B_{max}, \\
&XA + A^TX - XBB^TX = 0, \\
&\widehat{W}_c\left(A - BB^TX\right) + \left(A - BB^TX\right)^T\widehat{W}_c + \widehat{W}_cBB^T\widehat{W}_c = 0.
\end{aligned} \tag{5.10}$$

The optimization problem in (5.10) has a linear objective and nonlinear equality constraints. The above problem can be reformulated as an optimization problem with linear objective and bilinear constraints. To this end, the following lemma is proved.

Lemma 5.2.1. Consider the time invariant linear system in (5.1) where A may or may not be Hurwitz. If the pair (A, B) is stabilizable, then

1. $\exists X \succeq 0, X \in \mathcal{S}^n$ such that $XA + A^TX - XBB^TX = 0$,

2. $\exists F \in \mathbb{R}^{n\times n}, W_c \succeq 0, W_c \in \mathcal{S}^n$ such that $F = -B^TX$ and $(A+BF)W_c + W_c(A+BF)^T + BB^T = 0$,

3. $\exists \widehat{W}_c \succeq 0, \widehat{W}_c \in \mathcal{S}^n$ such that $\widehat{W}_c = W_c^{-1}$ and $\widehat{W}_c(A+BF) + (A+BF)^T\widehat{W}_c + \widehat{W}_cBB^T\widehat{W}_c = 0$,

4. The pair $\left(X, \widehat{W}_c\right)$ is the solution of the optimization problem

$$\begin{aligned} &\max_{X,\widehat{W}_c} \quad \mathrm{Tr}\left(\widehat{W}_c\right) \\ &\text{Subject to the constraints,} \\ &\begin{pmatrix} A^TX + XA & XB \\ B^TX & I \end{pmatrix} \succeq 0, \\ &\begin{pmatrix} A^TX_W + X_WA & -X_WB \\ -B^TX_W & I \end{pmatrix} \succeq 0, \\ &X_W = X - \widehat{W}_c, \\ &X \in \mathcal{S}^n, \quad \widehat{W}_c \in \mathcal{S}^n. \end{aligned} \tag{5.11}$$

Proof. Part 1 and 2 have been proved in Theorem 2 [138].

3. Since W_c satisfying (5.8) exists and is finite then W_c is invertible and $\widehat{W}_c = W_c^{-1} \succeq 0$ exists with $W_c\widehat{W}_c = I$. Pre-multiplying and post-multiplying (5.8) by $\widehat{W}_c$ gives,

$$\widehat{W}_c(A+BF) + (A+BF)^T\widehat{W}_c + \widehat{W}_cBB^T\widehat{W}_c = 0. \tag{5.12}$$

4. Substituting $F = -B^T X$ in (5.12) and using (5.7) gives,

$$\begin{aligned} &\widehat{W}_c A + A^T \widehat{W}_c - \widehat{W}_c B B^T X - X B B^T \widehat{W}_c \\ &\qquad + \widehat{W}_c B B^T \widehat{W}_c = 0, \\ &A^T \left(\widehat{W}_c - X\right) + \left(\widehat{W}_c - X\right) A \\ &\qquad + \left(\widehat{W}_c - X\right) B B^T \left(\widehat{W}_c - X\right) = 0. \end{aligned} \tag{5.13}$$

As (A, B) is stabilizable, from part 1 a unique $X \succeq 0$ exists which is the solution of the ARE in (5.7). Hence, unique $F = -B^T X$ and corresponding unique $\widehat{W}_c \succeq 0$ exist such that $(A + BF)$ is Hurwitz. As (5.7) and (5.13) are AREs, $X, \widehat{W}_c$ are solutions of the optimization problem [45] given by,

$$\max_{X, \widehat{W}_c} \quad \mathrm{Tr}\left(\widehat{W}_c - X + X\right) = \mathrm{Tr}\left(\widehat{W}_c\right)$$

Subject to the constraints,

$$\begin{aligned} \mathrm{S}_X &:= \begin{pmatrix} A^T X + X A & X B \\ B^T X & I \end{pmatrix} \succeq 0, \\ \mathrm{S}_{XW} &:= \begin{pmatrix} A^T X_W + X_W A & -X_W B \\ -B^T X_W & I \end{pmatrix} \succeq 0, \\ X_W &= X - \widehat{W}_c, \\ X &\in \mathcal{S}^n, \quad \widehat{W}_c \in \mathcal{S}^n. \end{aligned} \tag{5.14}$$

■

Using the above Lemma 1, the problem in (5.10) can be written as an equivalent

problem in (5.15) below. This is proved in Theorem 1.

$$
\begin{aligned}
&\min_{B\in\mathcal{D}_B,T,Z} \quad \mathrm{Tr}\,(Z_{22}) + \mathrm{Tr}\,(T_{22}) \\
&\text{Subject to the constraints,} \\
&\mathrm{Tr}\,(B) \leq B_{max}, \\
&\mathrm{S}_Z\,(B,Z) := AZ_{11} + Z_{11}A^T + BZ_{12}^T + Z_{12}B^T + I = 0, \\
&\mathrm{S}_T\,(B,T) := I - AT_{11} - T_{11}A^T + BT_{12}^T + T_{12}B^T = 0, \\
&Z = \begin{pmatrix} Z_{11} & Z_{12} \\ Z_{12}^T & Z_{22} \end{pmatrix} \succeq 0, \quad T = \begin{pmatrix} T_{11} & T_{12} \\ T_{12}^T & T_{22} \end{pmatrix} \succeq 0 \\
&Z \in \mathcal{S}^{2n}, \quad T \in \mathcal{S}^{2n}.
\end{aligned}
\tag{5.15}
$$

Remark 5.2.1. It should be noted that when A is known to be Hurwitz, $X = 0$ in (5.10) and $Z = 0$ in (5.15).

Remark 5.2.2. In the remainder of the chapter, it is understood that the symmetric matrices Z and T have the same structure as defined in (5.15).

Theorem 5.2.1. Consider the linear system in (5.1). If the pair (A, B) is stabilizable for $B \in \mathcal{D}_B$, then the actuator design optimization problem in (5.10) is equivalent to the optimization problem in (5.15).

Proof. Using part 4 of Lemma 1, the optimal actuator placement problem can be

reformulated as,

$$\min_{B\in\mathcal{D}_B} \quad v_L(B)$$

Subject to the constraints,

$$\mathrm{Tr}(B) \leq B_{max},$$

$$v_L(B) = \max_{X,\widehat{W}_c} \quad \mathrm{Tr}\left(\widehat{W}_c\right)$$

Subject to the constraints,

$$\begin{pmatrix} A^TX + XA & XB \\ B^TX & I \end{pmatrix} \succeq 0, \tag{5.16}$$

$$\begin{pmatrix} A^TX_W + X_WA & -X_WB \\ -B^TX_W & I \end{pmatrix} \succeq 0,$$

$$X_W = X - \widehat{W}_c,$$

$$X \in \mathcal{S}^n, \quad \widehat{W}_c \in \mathcal{S}^n.$$

The Lagrangian of the problem in (5.14) is,

$$\begin{aligned}\mathcal{L}(B,Z,T) &= \mathrm{Tr}\left(\widehat{W}_c\right) + \mathrm{Tr}(Z\mathrm{S}_X) + \mathrm{Tr}(T\mathrm{S}_{XW}),\\ &= \mathrm{Tr}\left(\widehat{W}_c\right) + \mathrm{Tr}\left(Z_{11}\left(A^TX + XA\right)\right) + \mathrm{Tr}\left(Z_{12}B^TX + Z_{12}^TXB + Z_{22}I\right)\\ &\quad + \mathrm{Tr}\left(T_{11}\left(A^TX_W + X_WA\right)\right) + \mathrm{Tr}\left(T_{22}I - T_{12}B^TX_W + T_{12}^TX_WB\right),\end{aligned} \tag{5.17}$$

where $Z = \begin{pmatrix} Z_{11} & Z_{12} \\ Z_{12}^T & Z_{22} \end{pmatrix} \succeq 0, T = \begin{pmatrix} T_{11} & T_{12} \\ T_{12}^T & T_{22} \end{pmatrix} \succeq 0$ are Lagrange multipliers. Using

$\frac{\partial \mathcal{L}}{\partial X} = 0, \frac{\partial \mathcal{L}}{\partial \widehat{W}_c} = 0$, the dual problem of (5.14) can be written as,

$$\begin{aligned} &\min_{Z \succeq 0, T \succeq 0} \quad \mathrm{Tr}\left(Z_{22}\right) + \mathrm{Tr}\left(T_{22}\right) \\ &\text{Subject to the constraints,} \\ &AZ_{11} + Z_{11}A^T + BZ_{12}^T + Z_{12}B^T + I = 0, \\ &I - AT_{11} - T_{11}A^T + BT_{12}^T + T_{12}B^T = 0. \end{aligned} \tag{5.18}$$

When $B = \widehat{B} \in \mathcal{D}_B$ is held fixed, then problem in (5.14) is a convex problem. Since (A, B) is stabilizable, from results in Section IV of [46], $\exists X \succ 0, \widehat{W}_c \succ 0$ such that $\mathrm{S}_X\left(X\right) \succ 0$ and $\mathrm{S}_W\left(X, \widehat{W}_c\right) \succ 0$. This leads to the existence of strong duality between (5.14) and (5.18) [50, 51]. The problem can then be written as,

$$\begin{aligned} &\min_{B \in \mathcal{D}_B} \quad v_L\left(B\right) \\ &\text{Subject to the constraints,} \\ &\mathrm{Tr}\left(B\right) \leq B_{max}, \\ &v_L\left(B\right) = \min_{Z \succeq 0, T \succeq 0} \quad \mathrm{Tr}\left(Z_{22}\right) + \mathrm{Tr}\left(T_{22}\right) \\ &\qquad \text{Subject to the constraints,} \\ &\qquad AZ_{11} + Z_{11}A^T + BZ_{12}^T + Z_{12}B^T + I = 0, \\ &\qquad I - AT_{11} - T_{11}A^T + BT_{12}^T + T_{12}B^T = 0. \end{aligned} \tag{5.19}$$

Using the concept of projection [48], the problem (5.19) can be written as a single

level problem,

$$\begin{aligned}
&\min_{B\in\mathcal{D}_B, T\succeq 0, Z\succeq 0} \quad \mathrm{Tr}\,(Z_{22}) + \mathrm{Tr}\,(T_{22}) \\
&\text{Subject to the constraints,} \\
&\mathrm{Tr}\,(B) \leq B_{max}, \\
&\mathrm{S}_Z\,(B,Z) := AZ_{11} + Z_{11}A^T + BZ_{12}^T + Z_{12}B^T + I = 0, \\
&\mathrm{S}_T\,(B,T) := I - AT_{11} - T_{11}A^T + BT_{12}^T + T_{12}B^T = 0.
\end{aligned} \tag{5.20}$$

■

The following two propositions prove important properties of the formulations in (5.10) and (5.15).

Proposition 5.2.3. Consider the linear system in (5.1) such that $B = \widehat{B} \in \mathcal{D}_B$ is known. If the optimization problems in (5.10) is infeasible, then problem (5.15) is also infeasible and vice versa.

Proof. From Theorem 1, problems (5.10) and (5.15) are equivalent. Hence, infeasibility of one implies infeasibility of the other.

■

Proposition 5.2.4. Consider the linear system in (5.1). For $B = \widehat{B} \in \mathcal{D}_B$ the optimization problem in (5.15) is infeasible if and only if the pair $\left(A, \widehat{B}\right)$ is uncontrollable.

Proof. When $B = \widehat{B} \in \mathcal{D}_B$ the pair $\left(A, \widehat{B}\right)$ is uncontrollable $\Longleftrightarrow$ the AREs in (5.10) as well as $\mathrm{S}_Z\,(B,Z)$ and $\mathrm{S}_T\,(B,T)$ in (5.15) have no positive semidefinite solution leading to infeasible optimization problem (5.15).

■

5.3 Solution Approach

The optimal actuator placement problem in (5.15) has bilinear terms in the constraints $S_Z(B,Z)$ and $S_T(B,T)$. In $S_Z(B,Z)$ the bilinear product is between B and Z_{12}, and in $S_T(B,T)$ the bilinear product is between B and T_{12}. The bilinear product is relaxed into linear inequalities using McCormick's relaxation [26, 139] as follows,

$$\begin{aligned} &\text{Let,} \quad P_Y = BY, \quad \text{where,} \quad Y = Z_{12}^T, T_{12}^T, \\ &P_Y = (p_{ij}), \quad B = (b_{ij}), \quad Y = (y_{ij}). \end{aligned} \tag{5.21}$$

It is assumed that $Y \in \left[Y^L, Y^U\right]$ and $B \in \{0,1\}$. Since B is a diagonal matrix, each entry of P_Y, i.e, p_{ij}, will be of the form $b_{ij}y_{ij}$. The general term of the product p_{ij} can be relaxed (dropping subscripts ij for brevity) as,

$$\begin{aligned} p &= by, \\ p &\geq b^L y + b y^L - b^L y^L, \\ p &\geq b^U y + b y^U - b^U y^U, \\ p &\leq b^L y + b y^U - b^L y^U, \\ p &\leq b^U y + b y^L - b^U y^L. \end{aligned} \tag{5.22}$$

For $b^L = 0$ and $b^U = 1$, the constraints are reduced as,

$$
\begin{aligned}
& p = by, \\
& R_Y\left(B, Y, P_Y\right) = \begin{cases} by^L - p \leq 0, \\ y + y^U\left(b-1\right) - p \leq 0, \\ p - by^U \leq 0, \\ p - y - y^L\left(b-1\right) \leq 0. \end{cases}
\end{aligned} \tag{5.23}
$$

The general optimal actuator design problem can then be stated as,

$$
\begin{aligned}
& \min_{B \in \mathcal{D}_B, T \succeq 0, Z \succeq 0, P_Z, P_T} \quad \mathrm{Tr}\left(Z_{22}\right) + \mathrm{Tr}\left(T_{22}\right) \\
& \text{Subject to the constraints,} \\
& \mathrm{Tr}\left(B\right) \leq B_{max}, \\
& SP_Z\left(B, Z, P_Z\right) := AZ_{11} + Z_{11}A^T + P_Z + P_Z^T + I = 0, \\
& SP_T\left(B, T, P_T\right) := I - AT_{11} - T_{11}A^T + P_T + P_T^T = 0, \\
& R_Z\left(B, Z, P_Z\right) \leq 0, \\
& R_T\left(B, T, P_T\right) \leq 0, \\
& Z_{12} \in \left[Z_{12}^L, Z_{12}^U\right], \quad T_{12} \in \left[T_{12}^L, T_{12}^U\right],
\end{aligned} \tag{5.24}
$$

where $R_Z\left(B, Z, P_Z\right)$ and $R_T\left(B, T, P_T\right)$ represent vectors of corresponding McCormick's relaxation. The reformulation of the nonlinear optimization problem in (5.15) into an equivalent linear optimization problem in (5.24) is proved in Theorem 2.

Theorem 5.3.1. For the linear system in (5.1), the optimal actuator placement problems in (5.15) and (5.24) are equivalent.

Proof. Consider the general bilinear product term in constraints $SP_Z(B, Z, P_Z)$ and $SP_T(B, T, P_T)$ in (5.24) as $p = by$ as in (5.23). Now for any feasible value of $B \in \mathcal{D}_B$, $b = 0$ or $b = 1$.

Case 1: $b = 0$,

$$R_Y(B, Y, P_Y) = \begin{cases} p = 0 \\ \\ y^L \leq y \leq y^U. \end{cases} \tag{5.25}$$

Case 2: $b = 1$,

$$R_Y(B, Y, P_Y) = \begin{cases} p = y \\ \\ y^L \leq y \leq y^U. \end{cases} \tag{5.26}$$

From (5.25) and (5.26), it can be inferred that at any feasible $B \in \mathcal{D}_B$,

$$\begin{aligned} S_Z(B, Z) &= \mathrm{SP}_Z(B, Z, P_Z), \\ S_T(B, T) &= \mathrm{SP}_T(B, T, P_T). \end{aligned} \tag{5.27}$$

$\Longrightarrow$ Problems (5.15) and (5.24) are equivalent.

■

The optimal actuator design problem in (5.24) is a $0/1-$MISDP problem. The objective function and constraints are linear. The problem can be solved using branch-and-bound procedure [140].

Remark 5.3.1. It should be noted that the above derived results are also applicable for any general $0/1 - B$ matrix.

5.4 Examples

In this section, the proposed formulation is applied on two examples, namely a stable integrator chain and an unstable integrator chain [2]. The proposed formulation is tested on systems with increasing number of states/nodes. The stable integrator chain system has -1 at each main diagonal entry and 1 at each first sub-diagonal entry of the system matrix. Rest of the elements have value 0. An example of a 5-node integrator chain is shown in Figure 5.1.

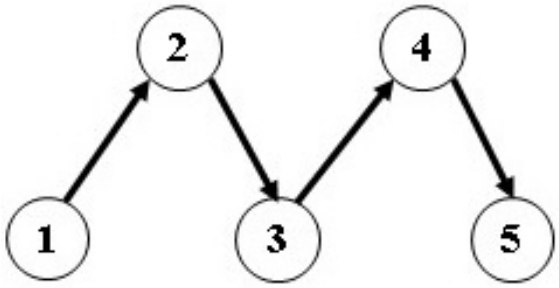


Figure 5.1: 5-node integrator chain [2]

The system matrix for the 5-node integrator chain is as follows,

$$A = \begin{pmatrix} -1 & 0 & 0 & 0 & 0 \\ 1 & -1 & 0 & 0 & 0 \\ 0 & 1 & -1 & 0 & 0 \\ 0 & 0 & 1 & -1 & 0 \\ 0 & 0 & 0 & 1 & -1 \end{pmatrix}. \tag{5.28}$$

The maximum number of actuators are $B_{max} = \lceil \frac{n}{4} \rceil$, where, n is the number of states/nodes and $\lceil \cdot \rceil$ is the ceiling function. Here, $5 \leq n \leq 25$ and n is

an odd number. The unstable integrator chain system is formed by introducing 1 in the system matrix A at the position $A(2,2)$. It is assumed that $Z_{12} \in 15300\,[-ones(n,n), ones(n,n)]$ and $T_{12} \in 15300\,[-ones(n,n), ones(n,n)]$. Branch-and-bound procedure is implemented to solve the optimal actuator placement problem using software MATLAB R2016a [61], with package YALMIP [62]. The simulation is carried on a computer with Intel core i7-4770 CPU @3.40 GHz processor and 8 GB of RAM. An upper bound of 2000 iteration is set for the branch-and-bound procedure of YALMIP. For $n > 19$ the algorithm stops after 2000 iterations.

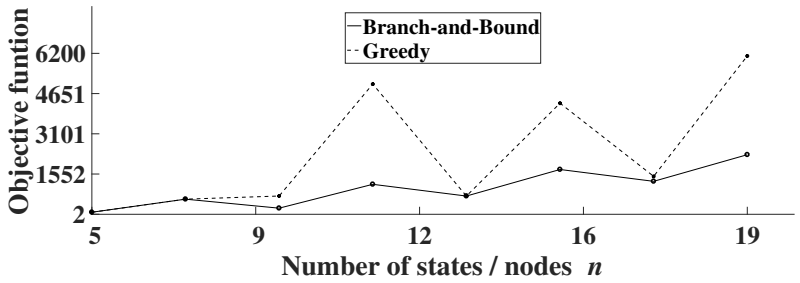


Figure 5.2: Comparison of branch-and-bound and greedy procedures for stable system

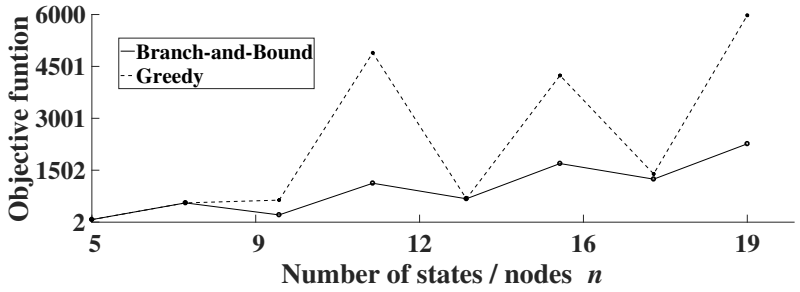


Figure 5.3: Comparison of branch-and-bound and greedy procedures for unstable system

The branch-and-bound procedure, when applied to optimization problem in

(5.24), computes the global optimal actuator combination for both stable and unstable systems. For $n \leq 17$, the global optimal actuator placement combination is also computed by exhaustively searching all possible actuator combinations. The results of the exhaustive search were found to be identical to the results obtained by the proposed formulation. Heuristic greedy procedure [129,134] is also implemented to solve the optimal actuator placement problem. The comparison of results of the greedy procedure and branch-and-bound procedures, for both stable and unstable systems, are shown in Figure 5.2 and Figure 5.3.

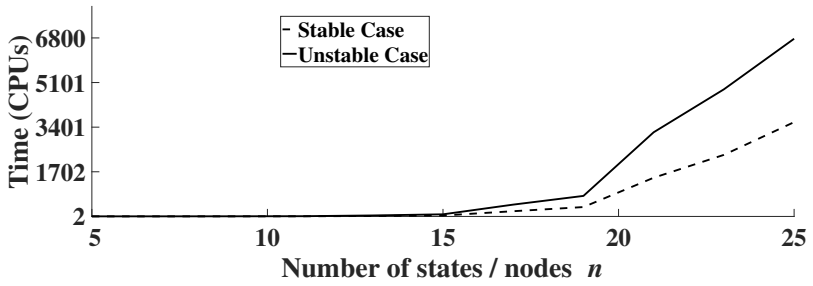


Figure 5.4: Time complexity for stable and unstable system

5.5 Summary

This chapter presents a new formulation for the optimal actuator placement problem with constraints on the number of actuators. The original time dependent problem is converted into a time independent algebraic optimization problem with mixed integer bilinear matrix equality constraints. Finally with application of McCormick's relaxation, the bilinear problem was converted into a $0/1-$MISDP which can be solved using branch-and-bound procedure. The utility of the proposed

formulation is justified by applying it to design actuator placement for one stable system and one unstable system. The proposed formulation, along with the application of branch-and-bound procedure, yields a global optimal solution. This was confirmed by an exhaustive search over all possible actuator combinations. However, it is to be noted that no theoretical guarantee regarding the optimality/global nature of the computed solution has been provided in this chapter. Furthermore, no initial assumption of controllable starting actuator combination is required to initiate the branch-and-bound procedure. In the examples considered in this chapter, the proposed formulation along with branch-and-bound procedure outperforms the greedy heuristic procedure which is evident from Figure 5.2 and Figure 5.3. For a stable integrator system with $n = 5$ and $B_{max} = 2$, the optimal actuator placement obtained by using the proposed formulation is same as that presented in [2]. The jumps in the objective function value as seen in Figure 5.2 and Figure 5.3 are due to $\lceil \frac{n}{4} \rceil$ actuators controlling n and $n + 2$ nodes. For example, $n = 5$ and $n = 7$ nodes are controlled by 2 but $n = 9$ is controlled by 3 actuators. Hence this sudden jump in objective function value has been observed.

Chapter 6: Sensor Placement and Observer Design

A dynamic system is effectively regulated when the control applied is a reaction to the changes happening in the system. The information about the changes in some states of the system is generally measured as an output by a limited number of expensive sensors. The unmeasured states are then estimated by an observer using the information about the input and output. Thus, the placement of sensors and observer design are interrelated and together they influence the estimate of the complete state.

In this chapter, the sensor placement and observer design problem is studied for a class of nonlinear systems in which the nonlinearity is at least locally Lipschitz. A relation between Lipschitz constant, sensor positions, observer gain and asymptotic estimation is derived using results from the literature. A heuristic procedure based on the derived relation is proposed to place sensors and compute observer gain. The proposed formulation and solution procedure is justified using a nonlinear dynamic model of pipeline flow without a leak.

6.1 Introduction

In general, sensor placement and observer design are treated as two separate design problems. The sensor placement involves computing locations at which sensors are placed and often lead to set function optimization problems [141]. Traditional approaches to place sensors are mainly focused on maximizing the observability of the system [142]. For LTI systems, observability for a given sensor location combination is easily verified using Lyapunov equation [12]. In [129], it has been proved that the trace of an observability matrix is a submodular set function of the sensor locations. Using the submodularity property, scalable greedy approach [141] is used to place sensors in LTI systems [129]. Controller design and observer design are dual problems [143]. Hence, once sensors are placed, the observer can be easily designed using well-known control design techniques.

The above mentioned techniques for sensor placement and observer design are valid for LTI systems. In general, real world systems are often distributed and have nonlinear dynamics. Usual procedure to place sensors in distributed systems is by using discretization methods. The distributed system is first discretized into several nodes and then the nonlinear dynamics can be written into state space form with the node variables serving as state variables. Now, sensors are placed at (some of) these nodes for measurement purpose and then methods like Kalman Filtering [143] are used to estimate the complete state of the system. As the sensors to measure such distributed processes are often expensive, it is desirable to place limited number of sensors and then use effective state estimation techniques to compute the complete

state.

Some distributed nonlinear processes like the pipeline fluid flow without a leak may be represented by discretized nonlinear Lipschitz dynamics (refer Section 6.3). In [144], it has been proved that the states of (at least locally) Lipschitz nonlinear systems can be asymptotically estimated by an nonlinear observer with a linear gain.

Keeping in view the aforementioned requirements, in this chapter the sensor placement and observer design is formulated as an optimization problem for a class of Lipschitz nonlinear systems. A relationship between sensor locations, observer gain, Lipshitz constant and asymptotic estimation is derived using results in the literature. The derived relationship is used as an optimization metric and a heuristic optimization procedure is developed to place sensors and design observer gain. The proposed procedure is then applied to place sensors and design observer for a pipeline flow without a leak dynamic system.

This chapter is organized as follows: In Section 6.2, sensor placement, observer design problem and solution procedure is described. In Section 6.3, pipeline flow without a leak dynamic model is studied followed by an example in Section 6.4. A summary of the chapter in presented in Section 6.5.

6.2 Sensor Placement and Observer Design for Nonlinear Dynamics

Consider the nonlinear system dynamics as follows.

$$\dot{x} = Ax + Bu + g(x), \quad y = Cx. \tag{6.1}$$

where $A \in \mathbb{R}^{n\times n}, B \in \mathbb{R}^{n\times m}, B \in \mathbb{R}^{n\times n}$ are system matrices, $x \in \mathbb{R}^n, u \in \mathbb{R}^m, y \in \mathbb{R}^n$ denote the state vector, control input and the output vector of the system respectively. The nonlinear function $g(x) : \mathbb{R}^n \mapsto \mathbb{R}^n$ is a Lipschitz nonlinear function with a Lipschitz constant γ and is the nonlinear component of the dynamics. The matrix C is a 0/1-diagonal output matrix, i.e., $C_{ii} \in \{0, 1\}$ and $C_{ij} = 0, \quad \forall i \neq j$. Here C_{ij} is the element at the $(i, j)^{th}$ position in the matrix C. The domain of C is denoted by $\mathcal{D}_C$. If $C_{ii} \neq 0$ implies that the output y_i is measured, while $C_{ii} = 0$ implies that the output y_i is not measured. For a matrix X, the notation $X \succeq (\succ) 0$ implies X is a positive semidefinite (definite) matrix and X^T denotes the transpose of matrix X. $\mathcal{S}^n$ denotes the space of $n \times n$ real symmetric matrices. The observer (estimator) for the system (6.1) is,

$$\dot{\hat{x}} = A\hat{x} + Bu + g(\hat{x}) + L(y - C\hat{x}), \tag{6.2}$$

where $\hat{x}$ is the estimate of x and L is the observer gain. The estimation error dynamics is,

$$\dot{e} = (A - LC)e + [g(x) - g(\hat{x})], \quad e = x - \hat{x}. \tag{6.3}$$

The objective of sensor placement is to place sensors such that the system (A, C) is observable. The observer (gain) is designed such that the error dynamics (6.3) is asymptotically stable. For nonlinear systems of type (6.1), the asymptotic stability of the observer error dynamics is ensured by a result derived from Theorem 2 and Theorem 5 from [144]. Let $\lambda_i(\{\cdot\})$ be the i^{th} eigenvalue of $\{\cdot\}$, $\mathrm{Re}(\{\cdot\})$ be the real part of $\{\cdot\}$ and $|\{\cdot\}|$ be the absolute value of $\{\cdot\}$. Let $(A - LC) = T\Lambda T^{-1}$, then $K_2(T)$ is the condition number of T. I be an identity matrix of appropriate

dimension as required and $j = \sqrt{-1}$.

Proposition 6.2.1. Consider Lipschitz nonlinear dynamics (6.1) where $g(x)$ is a Lipschitz nonlinear function with Lipschitz constant γ and the pair (A, C) is observable. Let (6.2) be the observer of (6.1). If the observer gain L is chosen such that $(A - LC)$ is Hurwitz and $\gamma < \frac{\min\limits_{1 \leq i \leq n} \left| \mathrm{Re}\left(\lambda_i(A-LC)\right) \right|}{K_2(T)}$ then the error dynamics in (6.3) is asymptotically stable.

Proof. From Theorem 5 of [144], if $\gamma < \frac{\min\limits_{1 \leq i \leq n} \left| \mathrm{Re}\left(\lambda_i(A-LC)\right) \right|}{K_2(T)}$ then $\min\limits_{\omega \geq 0} \quad \lambda_{min}\left(A - LC - \omega j I\right) > \gamma$. From Theorem 2 of [144], if (A, C) is observable and L is such that $(A - LC)$ is Hurwitz and $\min\limits_{\omega \geq 0} \quad \lambda_{min}\left(A - LC - \omega j I\right) > \gamma$ then the observer error dynamics (6.3) is asymptotically stable.

■

The Proposition 6.2.1 established a relation between sensor positions (C), observer gain (L), Lipschitz constant (γ) and asymptotic stability of the observer error dynamics. Computing minimum number of sensors such that a system is observable is a NP-hard problem [145]. Hence, having an upper bound on the number of sensors used is a practically useful solution. A heuristic procedure to place sensors and design observer is presented next.

6.2.1 Sensor Placement and Observer Design Procedure

The conditions: (A, C) observable and placing eigenvalues of $(A - LC)$ into the far left half-plane, are not enough to ensure the asymptotic stability of the error dynamics (6.3). Along with the aforementioned conditions, eigenvectors of

$(A - LC)$ also need to be well-conditioned.

Let V be the set of locations available for placing sensors, C_{max} is the upper bound on the number of sensors, $S \subset V$ is a set of location of sensors. S contains locations of diagonal C with value 1 i.e., $C_{ii} = 1$ if and only if $i \in S$. $S_{ini} \subset V$ is the initial set of locations of sensors such that (A, C^{ini}) is observable. The objective of sensor placement and observer design is to select C (i.e., S) and L such that $f_S(S) = \frac{\min_{1 \leq i \leq n} \left| \mathrm{Re}\left(\lambda_i(A-LC)\right) \right|}{K_2(T)} > \gamma$. To accommodate larger values of γ, the value of f_S should be as large as possible. A heuristic procedure to place sensors and design observer is as follows.

1. Set n, $V = \{1, 2, 3, \ldots, n\}$. Set $j = 0$, $S^0 = S_{ini} = \{1\}$ or $S^0 = S_{ini} = \{n\}$.

2. $e_j^* = arg \max_{e_j \in V \setminus S^{j-1}} f_S(S^{j-1} \cup \{e_j\})$, where $f_S(S^{j-1} \cup \{e_j\}) = \frac{\min_{1 \leq i \leq 2N-1} \left| \mathrm{Re}\left(\lambda_i\left(A - L_r^j C^j\right)\right) \right|}{K_2(T^j)}$. C^j is constructed from $S^{j-1} \cup \{e_j\}$. As, $\left(A^T, C^{jT}\right)$ is controllable, L_r^{jT} is the LQR gain [143] computed using the '*care*' function of MATLAB [61] with $Q = R = I$ where I is the identity matrix of appropriate dimension. $S^j = S^{j-1} \cup \{e_j^*\}$. If $j < C_{max} - 1$, do $j = j + 1$ and repeat Step 2 else $S^G = S^j$ is the optimal sensor placement combination set and go to Step 3.

3. Construct C using S^G from Step 2. L is computed form the following opti-

mization problem.

$$\max_{L\in[\underline{L},\overline{L}]} \quad \frac{\min\limits_{1\le i\le 2N-1}\left|\mathrm{Re}\big(\lambda_i\left(A-LC\right)\big)\right|}{K_2\left(T\right)},$$

$$\begin{aligned} &\text{Subject to constraints,} \\ &\mathrm{Re}\big(\lambda_i\left(A-LC\right)\big)<0, \quad i=1,2,\ldots,2N-1, \\ &(A-LC)=T\Lambda T^{-1}. \end{aligned} \tag{6.4}$$

Problem (6.4) is solved using the '*patternsearch*' function in MATLAB [61] where initial solution L^0 is the LQR gain computed as in Step 2.

Next, the sensor placement and observer design algorithm is applied to a pipeline flow case study.

6.3 Case Study: Pipeline Flow

Figure 6.1 shows the model of a pipeline system. It is assumed that there are no convective changes in velocity, the fluid has constant density and the pipeline has constant cross-sectional area. The general dynamic equation for flow through a pipeline without a leak is as follows [146, 147],

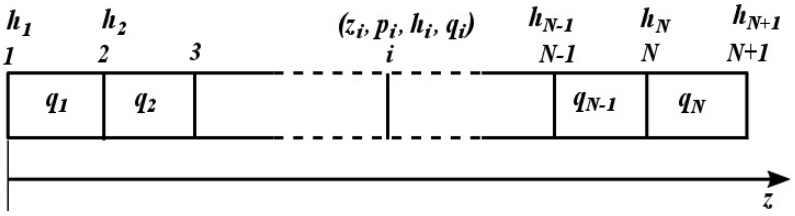


Figure 6.1: Pipeline model

$$\begin{aligned} &\frac{\partial q}{\partial t}+\frac{A_r}{\rho}\frac{\partial p}{\partial z}+\frac{f_r q|q|}{2d_r A_r}=0, \\ &\frac{\partial p}{\partial t}+\frac{\rho b^2}{A_r}\frac{\partial q}{\partial z}=0, \end{aligned} \tag{6.5}$$

where p is pressure of the fluid, q is flow rate of the fluid, ρ is density of the fluid, b is isothermic speed of sound, A_r is cross-section area of the pipeline, d_r is inner diameter of the pipeline, f_r is friction coefficient between fluid and the pipeline, L_p is length of the pipeline and z is spatial direction along the length of the pipeline with $z \in [0, L_p], t \geq 0$. It is assumed that the pipeline is always parallel to the ground so that gravitational effects are neglected. The term $q|q|$ accounts for the reverse flows. Neglecting reverse flow effects, the dynamic model is written as,

$$\begin{aligned} \frac{\partial q}{\partial t} &= -\frac{A_r}{\rho}\frac{\partial p}{\partial z} - \frac{f_r q^2}{2 d_r A_r}, \\ \frac{\partial p}{\partial t} &= -\frac{\rho b^2}{A_r}\frac{\partial q}{\partial z}. \end{aligned} \tag{6.6}$$

If the pressure is represented by the pressure head $h = \frac{p}{\rho g}$, the dynamics (6.6) is rewritten as,

$$\begin{aligned} \frac{\partial q}{\partial t} &= -k_1\frac{\partial h}{\partial z} - k_3 q^2, \\ \frac{\partial h}{\partial t} &= -k_2\frac{\partial q}{\partial z}, \end{aligned} \tag{6.7}$$

where $k_1 = g A_r, k_2 = \frac{b^2}{g A_r}, k_3 = \frac{f_r}{2 d_r A_r}$ and g is the gravitational acceleration. The dynamic model in (6.7) is discretized into N sections along the z-axis using a finite difference method as follows [146, 147],

$$\begin{aligned} \frac{\partial q}{\partial z} &\approx \frac{q_{i+1} - q_i}{\triangle z_i}, \\ \frac{\partial h}{\partial z} &\approx \frac{h_{i+1} - h_i}{\triangle z_i}, \\ \triangle z_i &= z_{i+1} - z_i, \quad L_p = z_{N+1} - z_1. \end{aligned} \tag{6.8}$$

The discretized flow dynamics are,

$$\begin{aligned} \dot{q}_i &= a_1 (h_i - h_{i+1}) - k_3 q_i^2, \quad i = 1, 2, 3, \ldots, N, \\ \dot{h}_{i+1} &= a_2 (q_i - q_{i+1}), \quad i = 1, 2, 3, \ldots, N-1, \end{aligned} \tag{6.9}$$

where $a_1 = \frac{k_1}{\triangle z_i}, a_2 = \frac{k_2}{\triangle z_i}, \triangle z_i = \frac{L_p}{N}$. The dynamic model in (6.9) is nonlinear. The inlet and outlet pressure heads are h_1 and h_{N+1} respectively. The input to the dynamic system in (6.9) is $u = \begin{pmatrix} h_1 & h_{N+1} \end{pmatrix}^T$. The state variables for the system are pressure head and flow rate variables at discretization points $i = 1, 2, 3, \ldots, N$. The state vector $x \in \mathbb{R}^{2N-1}$ is written as,

$$x = \begin{pmatrix} q_1 & h_2 & q_2 & h_3 & \ldots & h_N & q_N \end{pmatrix}^T. \tag{6.10}$$

At steady state, $\frac{\partial q}{\partial t} = 0, \overline{q}_i = \overline{q}$, the steady state pressure head as,

$$h_i = h_1 - \frac{k_3 \overline{q}^2}{k_1} z_i. \tag{6.11}$$

For a given $\overline{q}$, the values of h_1 and h_{N+1} can be appropriately selected. It should be noted that N sections of the pipeline lead to $N+1$ discretization points and $2N-1$ state variables. The pipeline flow dynamics is easily transformed as in (6.1). For example when $N = 3$,

$$\begin{aligned} x &= \begin{pmatrix} q_1 & h_2 & q_2 & h_3 & q_3 \end{pmatrix}^T, \\ A &= \begin{pmatrix} 0 & -a_1 & 0 & 0 & 0 \\ a_2 & 0 & -a_2 & 0 & 0 \\ 0 & a_1 & 0 & -a_1 & 0 \\ 0 & 0 & a_2 & 0 & -a_2 \\ 0 & 0 & 0 & a_1 & 0 \end{pmatrix}, g(x) = \begin{pmatrix} -k_3 q_1^2 \\ 0 \\ -k_3 q_2^2 \\ 0 \\ -k_3 q_3^2 \end{pmatrix}. \end{aligned} \tag{6.12}$$

For the pipeline flow dynamics as C is a diagonal matrix, the output consists of flow rate and pressure measurements. State variable q_1 is the inlet flow rate and q_N is the outlet flow rate. Next lemma shows that a sensor at the inlet or outlet makes the system (A, C) observable.

Lemma 6.3.1. Let C_i be a diagonal matrix with 1 at the i^{th} diagonal position and rest all elements are zeros. Then for the pipeline flow dynamics (6.9) represented as a system in (6.1), the pair (A, C_i) is observable when $i = 1$ or $i = 2N - 1$.

Proof. As, C_i has only one non-zero element, it is treated as a row vector which will result in a observability matrix $\mathcal{OM} \in \mathbb{R}^{2N-1 \times 2N-1}$ [12] as,

$$\mathcal{OM} = \begin{pmatrix} C_i \\ C_i A \\ \vdots \\ C_i A^{2N-2} \end{pmatrix}. \tag{6.13}$$

For the case $i = 1$, the matrix C_i is taken as a $2N - 1$ length row vector with 1 as the 1^{st} element. Constructing $\mathcal{OM}$ results in a lower triangular matrix under the diagonal joining $(1, 1)$ and $(2N - 1, 2N - 1)$. The diagonal is given by,

$$\begin{aligned} &\mathcal{OM}_{11} = 1, \quad \mathcal{OM}_{11} = -a_1, \\ &\mathcal{OM}_{ii} = \begin{cases} -a_2 \mathcal{OM}_{i-1,i-1} & \text{for} \quad i \geq 3 \quad \text{and} \quad i \quad \text{is odd}, \\ -a_1 \mathcal{OM}_{i-1,i-1} & \text{for} \quad i \geq 4 \quad \text{and} \quad i \quad \text{is even}. \end{cases} \end{aligned} \tag{6.14}$$

$\mathcal{OM}$ is a lower triangular matrix with independent columns. For the case when $i = 2N-1$, $\mathcal{OM}$ is a lower triangular matrix below the diagonal joining $(1, 2N - 1)$ and $(2N - 1, 1)$ and independent columns. In both cases, rank $(\mathcal{OM}) = 2N - 1$ as $a_1 \neq 0$ and $a_2 \neq 0$. Hence, the pair (A, C_i) is observable when $i = 1$ or $i = 2N - 1$.

■

As a bounded input is applied to the pipeline system and no reverse flows are assumed, the discretized nonlinear dynamics (6.9) has a bounded operating

region. Hence the states are bounded. Next corollary shows how Proposition 6.2.1 is applicable to the pipeline system.

Corollary 6.3.1. Consider the pipeline flow dynamics (6.9) and its representation (6.1) with q_{max} as the maximum allowable flow rate in the pipeline. If C, L can be computed such that (A, C) is observable and $2k_3 q_{max} \leq \frac{\min\limits_{1\leq i\leq 2N-1}\left|\mathrm{Re}\left(\lambda_i(A-LC)\right)\right|}{K_2(T)}$ then the observer error dynamics in (6.3) is asymptotically stable.

Proof. As the operating region for the pipeline system is bounded, it is easily observed that the nonlinear function $g(x)$ is locally Lipschitz with $\gamma = 2k_3 q_{max}$. Rest of result follows now.

■

For the pipeline system, the diagonal output matrix C represents the positions of the flow rate and pressure sensors. Thus, the algorithm in Section 6.2.1 not only computes where to place the sensor but also informs the designer what type of sensor to use. In general, placing poles and having a well-conditioned eigenstructure for $(A - LC)$ is a challenging task. Hence, the algorithm in Section 6.2.1 tries to compute a value of $\frac{\min\limits_{1\leq i\leq 2N-1}\left|\mathrm{Re}\left(\lambda_i(A-LC)\right)\right|}{K_2(T)}$ close to $2k_3 q_{max}$. Next, an example is presented to demonstrate the proposed sensor placement and observer design approach.

6.4 Example

Consider an oil pipeline system with the parameters stated below.

$$
\begin{aligned}
&L_p = 10\ km = 10000\ m, \quad \rho = 870\ \frac{kg}{m^3} \quad g = 9.81\ \frac{m}{s^2}, \quad b = 1400\ \frac{m}{s}, \\
&f_r = 0.027, \quad d_r = 1\ m, \quad A_r = \frac{\pi d_r^2}{4}, \quad h_{in} = 104.55\ m, \quad h_{out} = 100\ m, \\
&\overline{q} = 0.45\ \frac{m^3}{s} = 245,000\ \frac{barrels}{day}, \quad q_{max} = 0.5\ \frac{m^3}{s} = 272,000\ \frac{barrels}{day}, \\
&N = 10, \quad n = 2N - 1 = 19, \quad C_{max} = 5, \\
&-10 \leq L_{ij} \leq 10, \quad 1 \leq i \leq n, \quad 1 \leq j \leq C_{max}.
\end{aligned}
\tag{6.15}
$$

The pipeline system stated in (6.15) has $N = 10$ sections, $N+1 = 11$ discretization points and $n = 2N - 1 = 19$ state variables. The dynamics were constructed as in (6.9) and (6.1). The sensor placement and observer design was done as stated in Section 6.2.1 with $S^0 = S_{ini} = \{1\}$. Sensor placement is also done by linearizing the dynamics and using greedy procedure [141] with observability grammian [12] as an optimization metric. The observer gain L^{greedy} is computed as the LQR gain as in Step 2 of the procedure in Section 6.2.1. The optimal sensor placement arrangements for the nonlinear and linearized cases are tabulated in Table 6.1. The initial state for the system (6.9) was taken to be 0 for the flow rate variables and h_{out} for the pressure variables. The initial state for the observer (for both nonlinear and linearized cases) was a 0 vector. The evolution of the Frobenius norm [18] of the estimation error for the linearized and the nonlinear cases is shown in Figure 6.2. The simulation is performed using MATLAB [61] in the time interval $[0, 30]$ seconds. The output of procedure in Section 6.2.1 leads to: $K_2(T) = 258$, $\min\limits_{1 \leq i \leq 2N-1} \left| \mathrm{Re}\big(\lambda_i (A_0 - LC)\big) \right| =$

Point	1	2	3	4	5	6	7	8	9	10
ST-Nonlinear	F	—	—	—	P	P	F	—	—	F
ST-Linearized	F	—	—	—	P	P	P	—	—	F

Table 6.1: Type of sensor at each discretization point for nonlinear and linearized cases. ST: Sensor Type, P: Pressure head sensor, F: Flow rate sensor.

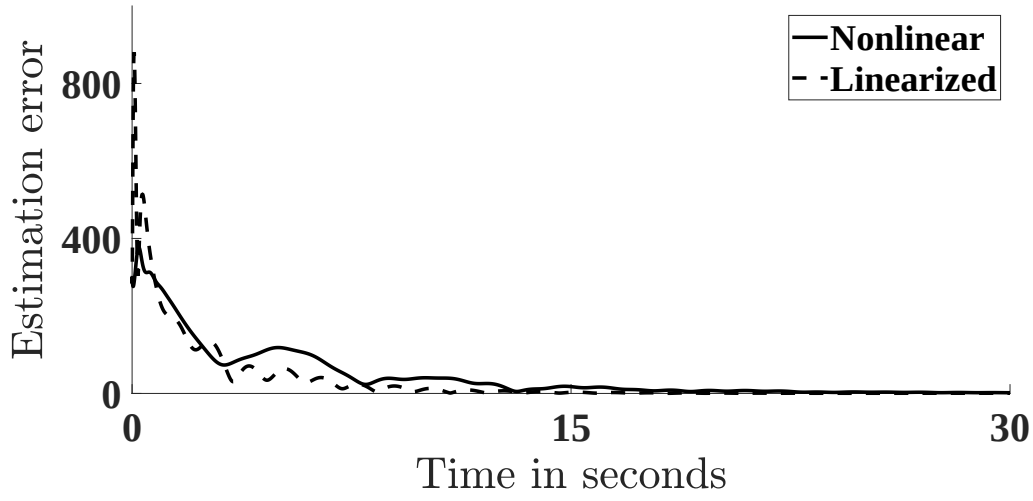


Figure 6.2: Evolution of estimation error with time for nonlinear and linearized cases

0.095 and $\frac{\min\limits_{1\leq i\leq 2N-1}\left|\mathrm{Re}\left(\lambda_i(A_0-LC)\right)\right|}{K_2(T)} = 0.00038$. The Lipschitz constant $2k_3q_{max} = 0.0172 > 0.00038$ which violates Corollary 6.3.1 in Section 6.2, but the estimation error converges to zero as observed in Figure 6.2. The maximum magnitude of the gain matrix element is 10. For the linearized case, the maximum magnitude of the LQR observer gain matrix element is 190. The estimation error for the linearized case also converges to zero but the maximum gain component magnitude for the linearized case is 19 times that of the nonlinear case.

6.5 Summary

In this chapter, a heuristic procedure is proposed to place sensors and design observer for a class of Lipschitz nonlinear dynamic systems. The synthesis procedure is based on a relation between sensor locations, observer gain, Lipschitz constant and asymptotic stability of the observer error dynamics. The proposed procedure was applied to a pipeline line flow without a leak dynamic model to compute the optimal sensor locations/types and observer gain. The developed procedure was demonstrated for an oil pipeline example.

Chapter 7: Conclusions, Contributions and Future Research Directions

In this chapter, conclusions and contributions of this dissertation are discussed along with future directions for further research.

7.1 Conclusions

This dissertation tried to address an important engineering problem related to design of dynamical systems. The advent of cyber physical systems (CPS) and depletion of traditional energy resources has led to the development of engineering systems like smart buildings, smart structures, smart grids, smart (electric) cars etc. These novel systems are expected to have higher degree of autonomy, robustness, energy efficiency with guarantees on stability which can be achieved by a system optimal design. As co-design involves the optimization of interdependent (physical) design and control (including observer) variables, it leads to the system optimal design. In co-design, the physical design variables can be the physical parameters of the system or the actuator/sensor locations or the controller structure/sparsity.

In Chapter 2, the co-design problem was formulated as a nonlinear, non-convex optimization problem with an ARE constraint for a class of LTI systems. These LTI

systems have design variables in the systems matrices in linear form and are controlled by LQR control. The goal of co-design was to optimize the design objective and input energy using the interplay between design and control variables. The optimization aimed to yield a stable co-designed system. By the use of SDP duality, the co-design problem was reformulated as a BMI optimization problem. This BMI optimization problem was shown to conform with the GBD structure. An iterative solution procedure consisting of GBD and GPM was proposed. The procedure converged to a solution in a finite number of iterations which is with in a tolerance bound from the nearest local/global minimum. The tolerance bound was proved to be a function of the number of design variables, design bound and user defined optimality tolerance criterion. The convergence of the GBD+GPM iterative procedure in a finite iterations was also proved. Tests to examine the stationarity and local minimum nature of the convergence point were formulated using the KKT conditions. A condition establishing convexity of the co-design problem leading to a global solution was also established. The proposed formulation was successfully applied to a numerical example to demonstrate the GBD+GPM solution procedure. The formulation was also applied to two engineering examples from the literature. The results justified the utility of the proposed formulation and solution procedure.

In Chapter 3, the SSOF design problem was shown to conform with the co-design BMI optimization formulation. The objective of the SSOF co-design was to design a controller with predefined structure such that optimal controller stabilizes the system and consumes less input energy. The constraints on the controller structure played the role of design variable constraints. The GBD+GPM solution

procedure with certain modifications was proposed to solve the SSOF optimization problem. Tests based on the KKT conditions to test the stationarity and local minimum nature of the computed solution were derived. The proposed formulation and solution procedure was applied to design a SSOF controller for a L-1011 aircraft. Exhaustive search of the design space indicated that the SSOF controller solution computed by the GBD+GPM procedure is a near global solution.

In Chapter 4, the sparse feedback design problem was solved as a mixed $\mathcal{H}_2/\mathcal{H}_\infty$ co-design optimization problem. The purpose of sparse feedback design was to make the controller less complex, make the system consume less input energy (good performance) and satisfy a given robustness criterion towards disturbances and uncertainties. The sparsity requirement on the controller structure acted as a design objective function. The given robustness level appeared as a constraint in the optimization problem. A scalable solution procedure based on ADMM which was used to design $\mathcal{H}_2$ sparse controllers was adapted to optimize the sparse mixed $\mathcal{H}_2/\mathcal{H}_\infty$ controllers. The proposed formulation was applied to a class of second order systems which are widely applied in real world namely: structural systems and linearized power system/network swing equation. Based on the results in the literature for the class of second order systems, a method to select robustness level and initial stabilizing controller was also presented.

In Chapter 5, the optimal actuator placement problem was formulated as a co-design optimization problem. The aim of optimal actuator placement was to place limited number of actuators such that the system consumes minimum input energy. The locations of the actuators represented by $0/1-$integer variables were

taken as design variables. The problem was formulated for both Hurwitz and non-Hurwitz LTI systems. The original problem consisted of ARE constraints which were reformulated into BMI constraints by using SDP duality. By the use of McCormick's relaxation, the BMI optimization problem was again reformulated into an equivalent $0/1-$MISDP. This $0/1-$MISDP was solved using the branch-and-bound procedure. The proposed formulation was applied to place actuators in a system consisting of a chain of integrators which let to a global minimum solution confirmed by exhaustive search of the design domain.

In Chapter 6, the sensor placement and observer design problem for a class of Lipschitz nonlinear systems was framed as a co-design optimization problem. The motivation behind sensor placement and observer design is to estimate the complete state accurately using limited number of output sensor measurements and nonlinear observer with linear gain matrix. The location of the sensors were represented in the system by $0/1-$integer variables and are the design variables. The class of nonlinear system have Lischitz continuous nonlinearity. Using existing results, a relation between Lipschitz constant, sensor positions, observer gain and asymptotic estimation was derived and used as an optimization metric. A heuristic solution procedure was devised to compute sensor locations and linear gain of the observer. The proposed formulation was applied on a Lipschitz nonlinear pipeline flow without a leak system.

Next, the contributions of the dissertation are listed.

7.2 Contributions of the Dissertation

In this section, the contribution made by the dissertation towards each of the topics discussed in Section 1.2 are briefly stated.

1. **Co-design Modeling and Optimization:** The co-design problem is modeled as a BMI optimization problem. An iterative solution procedure based on GBD and GPM is developed to compute a solution within a provable tolerance bound from the nearest local/global minimum in a finite number of iterations. Condition on the convexity of the co-design problem is likewise derived. (**Chapter 2.**)

2. **Sparse and Structured Feedback Design:** The SSOF design problem is formulated as a BMI optimization problem. An iterative procedure based on GBD and GPM is developed to synthesize the SSOF controller. The ADMM procedure is adapted to solve the sparse feedback design problem with a given robustness constraint. A detailed study of designing sparse controllers for a class of second systems which includes practically applied structural systems and linearized power system/network swing equation is also carried out. (**Chapter 3** and **Chapter 4.**)

3. **Optimal Actuator Placement:** The optimal actuator placement problem is first formulated as a $0/1-$MINLP. By using relaxation methods, the MINLP is then reformulated as a novel equivalent $0/1-$MISDP problem. The MISDP is now solved using the branch-and-bound procedure. (**Chapter 5.**)

4. **Sensor Placement and Observer Design:** An optimization metric dependent the relation between Lipschitz constant, sensor positions, observer gain and asymptotic estimation is proposed to compute sensor locations and design observer. A heuristic procedure to place sensors and design observer is developed. (**Chapter 6.**)

Some insights regarding further research are discussed next.

7.3 Future Directions for Further Research

Although the exploration of the research problems in this dissertation has been fruitful, some issues if addressed will enhance the utility of the work in the dissertation. These unanswered issues are listed as future directions for further research stated next according to the topics in Section 1.2.

1. **Co-design Modeling and Optimization:** In this dissertation, LQR type of controller is used for stabilization with independent design and control optimization objective functions. The natural extension will be to extend the developed co-design formulation to general controllers without the assumption of stabilizability and detectability. The case of co-design with design variables appearing as nonlinear functions in the system matrices and having optimization objective as a mixed function of design and control variables can be considered. The inclusion of the practically important constraint of control input saturation in the developed co-design optimization framework can also be explored.

2. **Sparse and Structured Feedback Design:** In this dissertation, the sparse and structured feedback design is done without taking into account the effect of physical design variables. The study of the effect of physical design parameters on the controller sparsity can be interesting. In addition, extending or modifying the scalable ADMM based sparse feedback design method to include robustness level as an optimization variable is also open for exploration. The inclusion of "feedthrough" from exogenous inputs to the performance output with noise in sensor measurements can also be considered.

3. **Optimal Actuator Placement:** In this dissertation, the $0/1-$MISDP optimal actuator placement problem is solved using the branch-and-bound approach which has poor scalability. Hence, using a scalable relaxation based solution procedure to solve the $0/1-$MISDP optimal actuator placement problem and provide guarantees on the computed solution can be considered. The study of the effect of actuator locations on the robustness level of the system is also a possible extension.

4. **Sensor Placement and Observer Design:** In this dissertation, a heuristic sensor placement and observer design procedure is proposed and applied on a pipeline without a leak system case study. Hence, to develop a deterministic methodology for sensor placement and observer design along with guarantees on the computed solution can be considered. From the perspective of the pipeline case study, leak detection and localization using the proposed sensor placement and observer design procedure can be an interesting investigation.

For this a leak can be considered as an additional nonlinearity.

Bibliography

[1] https://www.thegreenhead.com/2012/11/sbu-v3-self-balancing-unicycle.php.

[2] V. Tzoumas, M. A. Rahimian, G. J. Pappas, and A. Jadbabaie. Minimal actuator placement with bounds on control effort. *IEEE Transactions on Control of Network Systems*, 3(1):67–78, March 2016.

[3] Hosam K Fathy, Julie Reyer, Panos Y Papalambros, and A G Ulsoy. On the coupling between the plant and controller optimization problems. *Proceedings of the American Control Conference (ACC)*, 3:1864–1869, 2001.

[4] Achille Messac. Control-structure integrated design with closed-form design metrics using physical programming. *AIAA Journal*, 36(5):855–864, 1998.

[5] R. E. Skelton. Model error concepts in control design. *International Journal of Control*, 49(5):1725–1753, 1989.

[6] Arthur L Hale, WE Dahl, and J LISOWSKI. Optimal simultaneous structural and control design of maneuvering flexible spacecraft. *Journal of Guidance, Control, and Dynamics*, 8(1):86–93, 1985.

[7] J. F. Camino, M. C. de Oliveira, and R. E. Skelton. Convexifying linear matrix inequality methods for integrating structure and control design. *Journal of Structural Engineering*, 129(7):978–988, 2003.

[8] Q Li, WJ Zhang, and Li Chen. Design for control-a concurrent engineering approach for mechatronic systems design. *IEEE/ASME transactions on mechatronics*, 6(2):161–169, 2001.

[9] Thambirajah Ravichandran, David Wang, and Glenn Heppler. Simultaneous plant-controller design optimization of a two-link planar manipulator. *Mechatronics*, 16(3–4):233 – 242, 2006.

[10] J.M. Rodriguez-Fortun, J. Orus, J. Alfonso, J.R. Sierra, F. Buil, F. Rotella, and J.A. Castellanos. Model-based mechanical and control design of a three-axis platform. *Mechatronics*, 22(7):958–969, 2012.

[11] L. A. Ricardez Sandoval, H. M. Budman, and P. L. Douglas. Simultaneous design and control of processes under uncertainty: A robust modelling approach. *Journal of Process Control*, 18(7–8):735 – 752, 2008.

[12] Kemin Zhou, John Comstock Doyle, Keith Glover, et al. *Robust and optimal control*, volume 40. Prentice hall New Jersey, 1996.

[13] Denis Arzelier and Dimitri Peaucelle. An iterative method for mixed H_2/H_∞ synthesis via static output-feedback. In *Decision and Control, 2002, Proceedings of the 41st IEEE Conference on*, volume 3, pages 3464–3469. IEEE, 2002.

[14] Pierre Apkarian, Dominikus Noll, and Aude Rondepierre. Mixed H_2/H_∞ control via nonsmooth optimization. *SIAM Journal on Control and Optimization*, 47(3):1516–1546, 2008.

[15] Ahmadreza Argha, Li Li, and Steven W Su. Design of H_2 (H_∞)-based optimal structured and sparse static output feedback gains. *Journal of the Franklin Institute*, 354(10):4156–4178, 2017.

[16] A. Olshevsky. Minimal controllability problems. *IEEE Transactions on Control of Network Systems*, 1(3):249–258, Sept 2014.

[17] Hassan K Khalil. Noninear systems. *Prentice-Hall, New Jersey*, 2(5):5–1, 1996.

[18] Carl D Meyer. *Matrix analysis and applied linear algebra*, volume 71. Siam, 2000.

[19] Jeremy G VanAntwerp and Richard D Braatz. A tutorial on linear and bilinear matrix inequalities. *Journal of Process Control*, 10(4):363–385, 2000.

[20] Vincent Blondel and John N Tsitsiklis. Np-hardness of some linear control design problems. *SIAM Journal on Control and Optimization*, 35(6):2118–2127, 1997.

[21] Onur Toker and Hitay Ozbay. On the np-hardness of solving bilinear matrix inequalities and simultaneous stabilization with static output feedback. *Proceedings of the American Control Conference (ACC)*, 4:2525–2526, 1995.

[22] Samuel Burer and Adam N Letchford. Non-convex mixed-integer nonlinear programming: A survey. *Surveys in Operations Research and Management Science*, 17(2):97–106, 2012.

[23] Arthur M Geoffrion. Generalized benders decomposition. *Journal of optimization theory and applications*, 10(4):237–260, 1972.

[24] Raphael T Haftka and Zafer Gürdal. *Elements of structural optimization*, volume 11. Springer Science & Business Media, 2012.

[25] Stephen Boyd, Neal Parikh, Eric Chu, Borja Peleato, Jonathan Eckstein, et al. Distributed optimization and statistical learning via the alternating direction method of multipliers. *Foundations and Trends® in Machine learning*, 3(1):1–122, 2011.

[26] Garth P. McCormick. Computability of global solutions to factorable nonconvex programs: Part I — convex underestimating problems. *Mathematical Programming*, 10(1):147–175, 1976.

[27] Christodoulos A Floudas. *Nonlinear and mixed-integer optimization: fundamentals and applications*. Oxford University Press, 1995.

[28] Prasad Vilas Chanekar, Nikhil Chopra, and Shapour Azarm. A new formulation for co-design of linear systems with system matrices having affine design variables. In *Control Conference (ICC), 2016 Indian*, pages 507–513. IEEE, 2016.

[29] Prasad Vilas Chanekar, Nikhil Chopra, and Shapour Azarm. Co-design of linear systems using generalized benders decomposition. *Automatica*, 89:180–193, 2018.

[30] Rakesh Patil, Zoran Filipi, and Hosam Fathy. Computationally efficient combined plant design and controller optimization using a coupling measure. *Journal of Mechanical Design*, 134(7):071008, 2012.

[31] Diane L Peters, PY Papalambros, and AG Ulsoy. Control proxy functions for sequential design and control optimization. *Journal of Mechanical Design*, 133(9):091007, 2011.

[32] James T Allison, Tinghao Guo, and Zhi Han. Co-design of an active suspension using simultaneous dynamic optimization. *Journal of Mechanical Design*, 136(8):081003, 2014.

[33] Diane L. Peters, Panos Y. Papalambros, and A. Galip Ulsoy. Sequential co-design of an artifact and its controller via control proxy functions. *Mechatronics*, 23(4):409 – 418, 2013.

[34] H. Tanaka and T. Sugie. General framework and bmi formulae for simultaneous design of structure and control systems. *Proceedings of the IEEE Conference on Decision and Control (CDC)*, 1:773–778, Dec 1997.

[35] Keat-Choon Goh, Michael G Safonov, and George P Papavassilopoulos. Global optimization for the biaffine matrix inequality problem. *Journal of global optimization*, 7(4):365–380, 1995.

[36] Arash Hassibi, Jonathan How, and Stephen Boyd. A path-following method for solving bmi problems in control. *Proceedings of the American Control Conference (ACC)*, 2:1385–1389, 1999.

[37] H. D. Tuan, P. Apkarian, and Y. Nakashima. A new lagrangian dual global optimization algorithm for solving bilinear matrix inequalities. *International Journal of Robust and Nonlinear Control*, 10(7):561–578, 2000.

[38] Yuh-Shyang Wang and Yebin Wang. A gradient-based approach for optimal plant controller co-design. *Proceedings of the American Control Conference (ACC)*, pages 3249–3254, July 2015.

[39] Yu Jiang, Yebin Wang, Scott A Bortoff, and Zhong-Ping Jiang. An iterative approach to the optimal co-design of linear control systems. *International Journal of Control*, 89(4):680–690, 2016.

[40] J.C. Geromel. Convex analysis and global optimization of joint actuator location and control problems. *Automatic Control, IEEE Transactions on*, 34(7):711–720, Jul 1989.

[41] Eric Beran, Lieven Vandenberghe, and Stephen Boyd. A global bmi algorithm based on the generalized benders decomposition. *Control Conference (ECC), 1997 European*, pages 3741–3746, 1997.

[42] Mark W Spong, Seth Hutchinson, and Mathukumalli Vidyasagar. *Robot modeling and control*, volume 3. Wiley New York, 2006.

[43] Vladimir Kučera. On nonnegative definite solutions to matrix quadratic equations. *Automatica*, 8(4):413 – 423, 1972.

[44] W. S. Levine and M. Athans. On the determination of the optimal constant output feedback gains for linear multivariable systems. *Automatic Control, IEEE Transactions on*, 15(1):44–48, Feb 1970.

[45] David D Yao, Shuzhong Zhang, and Xun Yu Zhou. Stochastic linear-quadratic control via semidefinite programming. *SIAM Journal on Control and Optimization*, 40(3):801–823, 2001.

[46] V. Balakrishnan and L. Vandenberghe. Semidefinite programming duality and linear time-invariant systems. *IEEE Transactions on Automatic Control*, 48(1):30–41, Jan 2003.

[47] N V Sahinidis and Ignacio E Grossmann. Convergence properties of generalized benders decomposition. *Computers & Chemical Engineering*, 15(7):481–491, 1991.

[48] C. A. Floudas and V. Visweswaran. Primal-relaxed dual global optimization approach. *Journal of Optimization Theory and Applications*, 78(2):187–225, 1993.

[49] Chien-Hua Lee. Matrix bounds of the solutions of the continuous and discrete riccati equations–a unified approach. *International Journal of Control*, 76(6):635–642, 2003.

[50] Stephen Boyd and Lieven Vandenberghe. *Convex Optimization*. Cambridge University Press, New York, NY, USA, 2004.

[51] Mokhtar S Bazaraa, Hanif D Sherali, and Chitharanjan M Shetty. *Nonlinear programming: theory and algorithms*. John Wiley & Sons, 2013.

[52] Gerhard Freiling and Vlad Ionescu. Monotonicity and convexity properties of matrix riccati equations. *IMA Journal of Mathematical Control and Information*, 18(1):61–72, 2001.

[53] Bruce K Colburn and Larry R White. Computational considerations for a spacecraft attitude control system employing control moment gyros. *Journal of Spacecraft and Rockets*, 14(1):45–53, 1977.

[54] D Delchamps. Analytic feedback control and the algebraic riccati equation. *IEEE Transactions on Automatic Control*, 29(11):1031–1033, 1984.

[55] Jonathan Borwein and Adrian S Lewis. *Convex analysis and nonlinear optimization: theory and examples*. Springer Science & Business Media, 2010.

[56] Dimitri P Bertsekas. *Nonlinear programming*. Athena scientific Belmont, 1999.

[57] Peng Wang, Chunhua Shen, and Anton Van Den Hengel. A fast semidefinite approach to solving binary quadratic problems. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1312–1319, 2013.

[58] Alexander Lanzon, Yantao Feng, Brian DO Anderson, and Michael Rotkowitz. Computing the positive stabilizing solution to algebraic riccati equations with an indefinite quadratic term via a recursive method. *IEEE Transactions on automatic control*, 53(10):2280–2291, 2008.

[59] Ragnar Wallin, Chung-Yao Kao, and Anders Hansson. A cutting plane method for solving kyp-sdps. *Automatica*, 44(2):418–429, 2008.

[60] Coralia Cartis, Nicholas IM Gould, and Ph L Toint. On the complexity of steepest descent, newton's and regularized newton's methods for nonconvex unconstrained optimization problems. *Siam journal on optimization*, 20(6):2833–2852, 2010.

[61] MATLAB. *version 9.4.0 (R2018a)*. The MathWorks Inc., Natick, Massachusetts, 2018.

[62] J. Löfberg. Yalmip : A toolbox for modeling and optimization in MATLAB. In *Proceedings of the CACSD Conference*, Taipei, Taiwan, 2004.

[63] Jos F. Sturm. Using sedumi 1.02, a matlab toolbox for optimization over symmetric cones. *Optimization Methods and Software*, 11(1-4):625–653, 1999.

[64] PC Müller and HI Weber. Analysis and optimization of certain qualities of controllability and observability for linear dynamical systems. *Automatica*, 8(3):237–246, 1972.

[65] Prasad Vilas Chanekar, Nikhil Chopra, and Shapour Azarm. Optimal structured static output feedback design using generalized benders decomposition. In *Decision and Control (CDC), 2017 IEEE 56th Annual Conference on*, pages 4819–4824. IEEE, 2017.

[66] Yuanqiang Bai and Karolos M Grigoriadis. H_∞ collocated control of structural systems: An analytical bound approach. *Journal of guidance, control, and dynamics*, 28(5):850–853, 2005.

[67] Sabrina Aouaouda, Mohammed Chadli, and Hamid-Reza Karimi. Robust static output-feedback controller design against sensor failure for vehicle dynamics. *IET Control Theory & Applications*, 8(9):728–737, 2014.

[68] AN Andry, EY Shapiro, and JC Chung. Eigenstructure assignment for linear systems. *IEEE transactions on aerospace and electronic systems*, (5):711–729, 1983.

[69] Luis A Montestruque and Panos J Antsaklis. State and output feedback control in model-based networked control systems. *Proceedings of the IEEE Conference on Decision and Control (CDC)*, 2:1620–1625, 2002.

[70] E Davison and S Chow. An algorithm for the assignment of closed-loop poles using output feedback in large linear multivariable systems. *IEEE Transactions on Automatic Control*, 18(1):74–75, 1973.

[71] Brian Anderson, N Bose, and El Jury. Output feedback stabilization and related problems-solution via decision methods. *IEEE Transactions on Automatic control*, 20(1):53–66, 1975.

[72] Vassilis L Syrmos, Chaouki T Abdallah, Peter Dorato, and Karolos Grigoriadis. Static output feedback : a survey. *Automatica*, 33(2):125–137, 1997.

[73] JC Geromel, PLD Peres, and SR Souza. Convex analysis of output feedback control problems: Robust stability and performance. *IEEE Transactions on Automatic Control*, 41(7):997–1003, 1996.

[74] Jose C Geromel, CC De Souza, and RE Skelton. Static output feedback controllers: Stability and convexity. *IEEE Transactions on Automatic Control*, 43(1):120–125, 1998.

[75] T Iwasaki, RE Skelton, and JC Geromel. Linear quadratic suboptimal control with static output feedback. *Systems & Control Letters*, 23(6):421–430, 1994.

[76] Mahdieh S Sadabadi and Dimitri Peaucelle. From static output feedback to structured robust static output feedback: A survey. *Annual Reviews in Control*, 42:11–26, 2016.

[77] DD Moerder and A Calise. Convergence of a numerical algorithm for calculating optimal output feedback gains. *IEEE Transactions on Automatic Control*, 30(9):900–903, 1985.

[78] MC De Oliveira, JF Camino, and RE Skelton. A convexifying algorithm for the design of structured linear controllers. *Proceedings of the IEEE Conference on Decision and Control (CDC)*, 3:2781–2786, 2000.

[79] Jeongheon Han and Robert E Skelton. An lmi optimization approach for structured linear controllers. *Proceedings of the IEEE Conference on Decision and Control (CDC)*, 5:5143–5148, 2003.

[80] Dominikus Noll, Pierre Apkarian, and Vincent Bompart. Nonsmooth structured control design. *IFAC Proceedings Volumes*, 40(13):357–362, 2007.

[81] Didier Henrion and J-B Lasserre. Convergent relaxations of polynomial matrix inequalities and static output feedback. *IEEE Transactions on Automatic Control*, 51(2):192–202, 2006.

[82] Fu Lin, Makan Fardad, and Mihailo R Jovanovic. Augmented lagrangian approach to design of structured optimal state feedback gains. *IEEE Transactions on Automatic Control*, 56(12):2923–2929, 2011.

[83] Javad Lavaei. Optimal decentralized control problem as a rank-constrained optimization. *Communication, Control, and Computing (Allerton), 2013 51st Annual Allerton Conference on*, pages 39–45, 2013.

[84] Reza Arastoo, Nader Motee, and Mayuresh V Kothare. Optimal sparse output feedback control design: a rank constrained optimization approach. *arXiv preprint arXiv:1412.8236*, 2014.

[85] Y Wang, J Lopez, and Mario Sznaier. Sparse static output feedback controller design via convex optimization. *Proceedings of the IEEE Conference on Decision and Control (CDC)*, pages 376–381, 2014.

[86] Michihiro Kawanishi, Toshiharu Sugie, and Hiroshi Kanki. Bmi global optimization based on branch and bound method taking account of the property of local minima. *Proceedings of the IEEE Conference on Decision and Control (CDC)*, 1:781–786, 1997.

[87] Stephen Boyd, Laurent El Ghaoui, Eric Feron, and Venkataramanan Balakrishnan. *Linear matrix inequalities in system and control theory*. SIAM, 1994.

[88] Dennis S Bernstein and Wassim M Haddad. LQG control with an H_∞ performance bound: a riccati equation approach. *IEEE Transactions on Automatic Control*, 34(3):293–305, 1989.

[89] John Doyle, Kemin Zhou, and Bobby Bodenheimer. Optimal control with mixed H_2 and H_∞ performance objectives. In *American Control Conference, 1989*, pages 2065–2070. IEEE, 1989.

[90] Kemin Zhou, Keith Glover, Bobby Bodenheimer, and John Doyle. Mixed H_2 and H_∞ performance objectives. I. robust performance analysis. *IEEE Transactions on automatic control*, 39(8):1564–1574, 1994.

[91] John Doyle, Kemin Zhou, Keith Glover, and Bobby Bodenheimer. Mixed H_2 and H_∞ performance objectives. II. optimal control. *IEEE Transactions on Automatic Control*, 39(8):1575–1587, 1994.

[92] H-H Yeh, Siva S Banda, and B-C Chang. Necessary and sufficient conditions for mixed H_2 and H_∞ optimal control. *IEEE Transactions on Automatic Control*, 37(3):355–358, 1992.

[93] Pramod P Khargonekar, Ian R Petersen, and Mario A Rotea. H_∞-optimal control with state-feedback. *IEEE Transactions on Automatic Control*, 33(8):786–788, 1988.

[94] Pramod P Khargonekar and Mario A Rotea. MixedH_2/H_∞ control: a convex optimization approach. *IEEE Transactions on Automatic Control*, 36(7):824–837, 1991.

[95] Carsten W Scherer. From mixed to multi-objective control. In *Decision and Control, 1999. Proceedings of the 38th IEEE Conference on*, volume 4, pages 3621–3626. IEEE, 1999.

[96] Eric Ostertag. An improved path-following method for mixed H_2/H_∞ controller design. *IEEE Transactions on Automatic Control*, 53(8):1967–1971, 2008.

[97] Friedemann Leibfritz. An lmi-based algorithm for designing suboptimal static H_2/H_∞ output feedback controllers. *SIAM journal on Control and Optimization*, 39(6):1711–1735, 2001.

[98] JC Geromel, PLD Peres, and SR de Souza. A convex approach to the mixed H_2/H_∞ control problem for discrete-time uncertain systems. *SIAM Journal on Control and Optimization*, 33(6):1816–1833, 1995.

[99] B Halder and T Kailath. Lmi based design of mixed H_2/H_∞ controllers: the state feedback case. In *American Control Conference, 1999. Proceedings of the 1999*, volume 3, pages 1866–1870. IEEE, 1999.

[100] Jen-Te Yu. A new static output feedback approach to the suboptimal mixed H_2/H_∞ problem. *International Journal of Robust and Nonlinear Control*, 14(12):1023–1034, 2004.

[101] Takashi Shimomura and Takao Fujii. Multiobjective control design via successive over-bounding of quadratic terms. In *Decision and Control, 2000. Proceedings of the 39th IEEE Conference on*, volume 3, pages 2763–2768. IEEE, 2000.

[102] Yuki Imanishi, Yasushi Kami, and Eitaku Nobuyama. An exterior-point approach to the mixed H_2/H_∞ control problem. In *Decision and Control (CDC), 2015 IEEE 54th Annual Conference on*, pages 1978–1982. IEEE, 2015.

[103] Maryam Babazadeh and Amin Nobakhti. Sparsity promotion in state feedback controller design. *IEEE Transactions on Automatic Control*, 62(8):4066–4072, 2017.

[104] Fu Lin, Makan Fardad, and Mihailo R Jovanović. Design of optimal sparse feedback gains via the alternating direction method of multipliers. *IEEE Transactions on Automatic Control*, 58(9):2426–2431, 2013.

[105] Michael Rotkowitz and Sanjay Lall. A characterization of convex problems in decentralized control. *IEEE Transactions on Automatic Control*, 50(12):1984–1996, 2005.

[106] Michael C Rotkowitz and Nuno C Martins. On the nearest quadratically invariant information constraint. *IEEE Transactions on Automatic Control*, 57(5):1314–1319, 2012.

[107] Şerban Sabău and Nuno C Martins. Youla-like parametrizations subject to qi subspace constraints. *IEEE transactions on Automatic Control*, 59(6):1411–1422, 2014.

[108] Xin Qi, Murti V Salapaka, Petros G Voulgaris, and Mustafa Khammash. Structured optimal and robust control with multiple criteria: A convex solution. *IEEE Transactions on Automatic Control*, 49(10):1623–1640, 2004.

[109] Yin Wang, Jose A Lopez, and Mario Sznaier. Convex optimization approaches to information structured decentralized control. *IEEE Transactions on Automatic Control*, 2018.

[110] Simone Schuler, Ping Li, James Lam, and Frank Allgöwer. Design of structured dynamic output-feedback controllers for interconnected systems. *International Journal of Control*, 84(12):2081–2091, 2011.

[111] AI Zečević and DD Šiljak. Control design with arbitrary information structure constraints. *Automatica*, 44(10):2642–2647, 2008.

[112] Josep Rubió-Massegú, Josep M Rossell, Hamid Reza Karimi, and Francisco Palacios-Quinonero. Static output-feedback control under information structure constraints. *Automatica*, 49(1):313–316, 2013.

[113] Francisco Palacios-Quiñonero, Josep Rubió-Massegú, Josep M Rossell, and Hamid Reza Karimi. Feasibility issues in static output-feedback controller design with application to structural vibration control. *Journal of the Franklin Institute*, 351(1):139–155, 2014.

[114] Yang Zheng, Richard P Mason, and Antonis Papachristodoulou. Scalable design of structured controllers using chordal decomposition. *IEEE Transactions on Automatic Control*, 63(3):752–767, 2018.

[115] Tankred Rautert and Ekkehard W Sachs. Computational design of optimal output feedback controllers. *SIAM Journal on Optimization*, 7(3):837–852, 1997.

[116] Brian D O Anderson and John B Moore. *Linear optimal control*. Prentice-Hall, New Jersey, 1971.

[117] Jen-Te Yu. A convergent algorithm for computing stabilizing static output feedback gains. *IEEE Transactions on Automatic Control*, 49(12):2271–2275, 2004.

[118] Stephen Boyd, Venkatarmanan Balakrishnan, and Pierre Kabamba. A bisection method for computing the h_∞ norm of a transfer matrix and related problems. *Mathematics of Control, Signals and Systems*, 2(3):207–219, 1989.

[119] Lieven Vandenberghe, V Ragu Balakrishnan, Ragnar Wallin, Anders Hansson, and Tae Roh. Interior-point algorithms for semidefinite programming problems derived from the kyp lemma. In *Positive polynomials in control*, pages 195–238. Springer, 2005.

[120] Bala Kameshwar Poolla, Saverio Bolognani, and Florian Dörfler. Optimal placement of virtual inertia in power grids. *IEEE Transactions on Automatic Control*, 62(12):6209–6220, 2017.

[121] Prasad Vilas Chanekar, Nikhil Chopra, and Shapour Azarm. Optimal actuator placement for linear systems with limited number of actuators. In *American Control Conference (ACC), 2017*, pages 334–339. IEEE, 2017.

[122] Gábor Orosz, Jeff Moehlis, and Richard M. Murray. Controlling biological networks by time-delayed signals. *Philosophical Transactions of the Royal Society of London A: Mathematical, Physical and Engineering Sciences*, 368(1911):439–454, 2009.

[123] Indika Rajapakse, Mark Groudine, and Mehran Mesbahi. What can systems theory of networks offer to biology? *PLoS Comput Biol*, 8(6):e1002543, 2012.

[124] A. Khanafer and T. Başar. Information spread in networks: Control, games, and equilibria. *Information Theory and Applications Workshop (ITA), 2014*, pages 1–10, Feb 2014.

[125] Lijun Chen, Na Li, Libin Jiang, and Steven H Low. Optimal demand response: problem formulation and deterministic case. *Control and optimization methods for electric smart grids*, pages 63–85, 2012.

[126] R. Claes, T. Holvoet, and D. Weyns. A decentralized approach for anticipatory vehicle routing using delegate multiagent systems. *IEEE Transactions on Intelligent Transportation Systems*, 12(2):364–373, June 2011.

[127] Rudolf Emil Kalman. On the general theory of control systems. *IRE Transactions on Automatic Control*, 4(3):110–110, 1959.

[128] C. D. Johnson. Optimization of a certain quality of complete controllability and observability for linear dynamical systems. *Journal of Basic Engineering*, 91(2):228–237, 1969.

[129] T. H. Summers, F. L. Cortesi, and J. Lygeros. On submodularity and controllability in complex dynamical networks. *IEEE Transactions on Control of Network Systems*, 3(1):91–101, March 2016.

[130] F. Pasqualetti, S. Zampieri, and F. Bullo. Controllability metrics, limitations and algorithms for complex networks. *IEEE Transactions on Control of Network Systems*, 1(1):40–52, March 2014.

[131] U. Münz, M. Pfister, and P. Wolfrum. Sensor and actuator placement for linear systems based on H_2 and H_∞ optimization. *IEEE Transactions on Automatic Control*, 59(11):2984–2989, Nov 2014.

[132] Neil K Dhingra, Mihailo R Jovanović, and Zhi-Quan Luo. An admm algorithm for optimal sensor and actuator selection. *Proceedings of the IEEE Conference on Decision and Control*, pages 4039–4044, 2014.

[133] S. Pequito, S. Kar, and A. P. Aguiar. A framework for structural input/output and control configuration selection in large-scale systems. *IEEE Transactions on Automatic Control*, 61(2):303–318, Feb 2016.

[134] G. L. Nemhauser, L. A. Wolsey, and M. L. Fisher. An analysis of approximations for maximizing submodular set functions—i. *Mathematical Programming*, 14(1):265–294, 1978.

[135] A. Clark, L. Bushnell, and R. Poovendran. A supermodular optimization framework for leader selection under link noise in linear multi-agent systems. *IEEE Transactions on Automatic Control*, 59(2):283–296, Feb 2014.

[136] Alexander Olshevsky. On (non) supermodularity of average control energy. *IEEE Transactions on Control of Network Systems*, 2017.

[137] Didier Georges. The use of observability and controllability gramians or functions for optimal sensor and actuator location in finite-dimensional systems. *Proceedings of the IEEE Conference on Decision and Control*, 4:3319–3324, 1995.

[138] Kemin Zhou, Gregory Salomon, and Eva Wu. Balanced realization and model reduction for unstable systems. *International Journal of Robust and Nonlinear Control*, 9(3):183–198, 1999.

[139] Leo Liberti. *Reformulation and convex relaxation techniques for global optimization*. PhD thesis, Springer, 2004.

[140] Govind Menon, M. Nabil, and Sridharakumar Narasimhan. Branch and bound algorithm for optimal sensor network design. *IFAC Proceedings Volumes*, 46(32):690 – 695, 2013.

[141] Andreas Krause, Jure Leskovec, Carlos Guestrin, Jeanne VanBriesen, and Christos Faloutsos. Efficient sensor placement optimization for securing large water distribution networks. *Journal of Water Resources Planning and Management*, 134(6):516–526, 2008.

[142] Kiran D'souza and Bogdan I Epureanu. Sensor placement for damage detection in nonlinear systems using system augmentations. *AIAA journal*, 46(10):2434–2442, 2008.

[143] Bernard Friedland. *Control system design: an introduction to state-space methods*. Courier Corporation, 2012.

[144] Rajesh Rajamani. Observers for lipschitz nonlinear systems. *IEEE transactions on Automatic Control*, 43(3):397–401, 1998.

[145] Xiaofei Liu, Sérgio Pequito, Soummya Kar, Yilin Mo, Bruno Sinopoli, and A Aguiar. Minimum robust sensor placement for large scale linear time-invariant systems: a structured systems approach. In *4th IFAC Workshop on Distributed Estimation and Control in Networked Systems (NecSys)*, pages 417–424, 2013.

[146] Cristina Verde. Minimal order nonlinear observer for leak detection. *Journal of Dynamic Systems, Measurement, and Control*, 126(3):467–472, 2004.

[147] Lizeth Torres, Gildas Besancon, and Cristina Verde. Leak detection using parameter identification. *IFAC Proceedings Volumes*, 45(20):910–915, 2012.