Estimation and Control of Dynamical Systems with Applications to Multi-Processor Systems

by

Haotian Zhang

A thesis
presented to the University of Waterloo
in fulfillment of the
thesis requirement for the degree of
Doctor of Philosophy
in
Electrical and Computer Engineering

Waterloo, Ontario, Canada, 2016

I hereby declare that I am the sole author of this thesis. This is a true copy of the thesis, including any required final revisions, as accepted by my examiners.

I understand that my thesis may be made electronically available to the public.

Abstract

System and control theory is playing an increasingly important role in the design and analysis of computing systems. This thesis investigates a set of estimation and control problems that are driven by new challenges presented by next-generation Multi-Processor Systems on Chips (MP-SoCs). Specifically, we consider problems related to state norm estimation, state estimation for positive systems, sensor selection, and nonlinear output tracking. Although these problems are motivated by applications to multi-processor systems, the corresponding theory and algorithms are developed for general dynamical systems.

We first study state norm estimation for linear systems with unknown inputs. Specifically, we consider a formulation where the unknown inputs and initial condition of the system are bounded in magnitude, and the objective is to construct an unknown input norm-observer which estimates an upper bound for the norm of the states. This class of problems is motivated by the need to estimate the maximum temperature across a multi-core processor, based on a given model of the thermal dynamics. In order to characterize the existence of the norm observer, we propose a notion of bounded-input-bounded-output-bounded-state (BIBOBS) stability; this concept supplements various system properties, including bounded-input-bounded-output (BIBO) stability, bounded-input-bounded-state (BIBS) stability, and input-output-to-state stability (IOSS). We provide necessary and sufficient conditions on the system matrices under which a linear system is BIBOBS stable, and show that the set of modes of the system with magnitude 1 plays a key role. A construction for the unknown input norm-observer follows as a byproduct.

Then we investigate the state estimation problem for positive linear systems with unknown inputs. This problem is also motivated by the need to monitor the temperature of a multi-processor system and the property of positivity arises due to the physical nature of the thermal model. We extend the concept of strong observability to positive systems and as a negative result, we show that the additional information on positivity does not help in state estimation. Since the states of the system are always positive, negative state estimates are meaningless and the positivity of the observers themselves may be desirable in certain applications. Moreover, positive systems possess certain desired robustness properties. Thus, for positive systems where state estimation with unknown inputs is possible, we provide a linear programming based design procedure for delayed positive observers.

Next we consider the problem of selecting an optimal set of sensors to estimate the states of linear dynamical systems; in the context of multi-core processors, this problem arises due to the need to place thermal sensors in order to perform state estimation. The goal is to choose (at design-time) a subset of sensors (satisfying certain budget constraints) from a given set in order to minimize the trace of the steady state *a priori* or *a posteriori* error covariance produced by a

Kalman filter. We show that the *a priori* and *a posteriori* error covariance-based sensor selection problems are both NP-hard, even under the additional assumption that the system is stable. We then provide bounds on the worst-case performance of sensor selection algorithms based on the system dynamics, and show that certain greedy algorithms are optimal for two classes of systems. However, as a negative result, we show that certain typical objective functions are not submodular or supermodular in general. While this makes it difficult to evaluate the performance of greedy algorithms for sensor selection (outside of certain special cases), we show via simulations that these greedy algorithms perform well in practice.

Finally, we study the output tracking problem for nonlinear systems with constraints. This class of problems arises due to the need to optimize the energy consumption of the CPU-GPU subsystem in multi-processor systems while satisfying certain Quality of Service (QoS) requirements. In order for the system output to track a class of bounded reference signals with limited online computational resources, we propose a sampling-based explicit nonlinear model predictive control (ENMPC) approach, where only a bound on the admissible references is known to the designer *a priori*. The basic idea of sampling-based ENMPC is to sample the state and reference signal space using deterministic sampling and construct the ENMPC by using regression methods. The proposed approach guarantees feasibility and stability for all admissible references and ensures asymptotic convergence to the set-point. Furthermore, robustness through the use of an ancillary controller is added to the nominal ENMPC for a class of nonlinear systems with additive disturbances, where the robust controller keeps the system output close to the desired nominal trajectory.

Acknowledgements

First and foremost, I would like to express my sincerest gratitude to my advisor Professor Shreyas Sundaram for his continuous support and excellent guidance. It has been an honor to be his first student at Waterloo. In the past six years, I have learned a lot from Shreyas, and I could still remember our first meeting outside his office at EIT, from which this fantastic journey started. I could not have imagined having a better advisor and mentor for my graduate study.

I would like to thank Professor Stephen L. Smith for serving as my co-advisor and serving in both my master's and PhD thesis committees. Steve is very knowledgeable and is always able to provide insightful comments and suggestions to my research.

I would like to thank the rest of my thesis committee: Professor Andrew Heunis, Professor Daniel Miller, Professor Soo Jeon, and Professor Mihailo R. Jovanovic. Thank you for your time in reading this thesis and your insightful comments and encouragements. I would also like to thank Professor William W. Melek for representing Soo at the defense and Professor Trefford Simpson for chairing the defense.

I would like to thank Dr. Raid Ayoub and Dr. Michael Kishinevsky from Intel Cooperation. Thank you for proposing interesting problems which motivated the results in this thesis and for the fruitful discussions. I would also like to thank my collaborators Dr. Ankush Chakrabarty and Professor Greg Buzzard. It is my pleasure to work with so many great minds.

I am grateful to all my friends at Waterloo. In particular, I would like to thank Dr. Qinghua Shen, Ebrahim Moradi Shahrivar, Ashish Ranjan Hota, Thilan Costa, Elaheh Fata, Mohammad Pirani, Yuze Huang, Dr. Bo Zhu, Fei Wang, Chunyan Nan, Lin Tian, Tianhan Zhang, Dan Xie, and Dr. Abdul Rehman. You made my time at Waterloo so enjoyable.

I would like to thank my parents for their love and support. Their encouragements are always the power for me to keep going.

Last but not least, I would like to thank my fiancee Peng. PhD is a long journey, but life is longer. I am thankful that I could continue this journey with you.

The materials in this thesis are based on works supported by the Strategic CAD Labs, Intel Cooperation.

Dedicated to my parents and my fiancee.

Table of Contents

List of Tables **x**

List of Figures **xi**

1 Introduction **1**

1.1 State Estimation 1

1.2 Sensor Selection 4

1.3 Output Tracking 6

1.4 Organization and Contributions 8

1.5 Notation and Terminology 10

1.5.1 Sets 10

1.5.2 Matrices 11

1.5.3 Functions and Signals 11

2 State Norm Estimation for Linear Dynamical Systems **12**

2.1 Introduction 12

2.2 Background: State Estimation 13

2.2.1 Linear System Estimation 14

2.2.2 Set-membership Filtering 16

2.3 Unknown Input Norm-observers for Linear Systems with Unknown Inputs 17

2.4 BIBOBS Stability 18

2.5 A Discussion of BIBOBS Stability 30
2.5.1 Related Linear System Stability Notions 30
2.5.2 Interpretation in Nonlinear Setting 32
2.6 Illustrative Examples 33
2.6.1 Illustration of Condition (i) in Theorem 5 33
2.6.2 Illustration of Condition (iii) in Theorem 5 34
2.6.3 Construction of An Unknown Input Norm Observer 35
2.7 Summary 37

3 State Estimation for Positive Linear Dynamical Systems 38
3.1 Introduction 38
3.2 Background: Positive Systems 39
3.3 Strong Observability of Positive Systems 40
3.4 Positive Observers with Unknown Inputs 42
3.5 Summary 46

4 Sensor Selection for Linear Dynamical Systems 47
4.1 Introduction 47
4.2 Background: Sensor Scheduling 48
4.3 Problem Formulation 51
4.4 Complexity of the Priori and Posteriori KFSS Problems 53
4.5 Upper Bounds on the Performance of Sensor Selection Algorithms 57
4.5.1 Upper Bound for $r(\Sigma)$ 57
4.5.2 Upper Bound for $r(\Sigma^*)$ 61
4.6 Greedy Algorithms 63
4.6.1 Optimality of Greedy Algorithms for Two Classes of Systems 64
4.6.2 Lack of Submodularity of the Cost Functions 67
4.6.3 Approximated KFSS Problem 70

4.7 Simulation . . . 72
4.7.1 Performance Evaluation . . . 72
4.7.2 Complexity Analysis . . . 76
4.8 Summary . . . 76

5 Output Tracking for Nonlinear Dynamical Systems 78
5.1 Introduction . . . 78
5.2 Background: Model Predictive Control . . . 79
5.3 Problem Formulation . . . 82
5.4 Sampling based Nominal ENMPC . . . 83
5.4.1 Sampling Scheme . . . 83
5.4.2 ENMPC Design . . . 84
5.5 Feasibility and Stability Analysis . . . 86
5.6 Extension To Robust ENMPC . . . 90
5.7 Simulation . . . 96
5.7.1 2-D System . . . 97
5.7.2 CPU-GPU Queueing System . . . 99
5.8 Summary . . . 102

6 Conclusions and Future Research 103
6.1 Conclusions . . . 103
6.2 Future Research . . . 104

References 106

List of Tables

2.1 Comparison of different notions of stability. 32

4.1 Performance comparison of different algorithms over randomly generated stable systems. 74

4.2 Performance comparison of different algorithms over randomly generated unstable systems. 75

5.1 Complexity comparison of different NMPC approaches. 99

List of Figures

1.1 Temperature estimation for multi-processor systems. 2

1.2 Sensor selection for multi-processor systems. 4

1.3 The CPU-GPU queueing system model proposed in [59]. 6

2.1 Illustration of the alarm system. 13

2.2 Venn diagram for the classes of linear systems that are BIBO stable, BIBS stable and BIBOBS stable. 32

2.3 Performance of the unknown input norm observer (2.14). 36

4.1 Complexity comparison of different algorithms. 77

5.1 Illustration of the robust ENMPC controller. 91

5.2 Performance of the sampling based nominal ENMPC controller and the sampling based robust ENMPC controller. 98

5.3 CPU-GPU queueing system. 100

5.4 Performance of the ENMPC controller on the CPU-GPU queueing system. 101

Chapter 1

Introduction

System and control theory deals with the problem of how to modify the behavior of dynamical systems through the use of feedback, especially in the presence of modeling uncertainty and disturbances. The corresponding concepts and techniques have been successfully applied to regulate various physical processes. Classical applications range from manufacturing, chemical systems and flight systems, to electrical circuits and power systems [35, 43]. In recent years, the scope of control theory has increasingly broadened to encompass different fields; examples include applying control techniques to the study of social networks and biological systems [54, 82] and control of systems at atomic and nano (space) scales [5, 18]. In turn, these applications and challenges from other disciplines are providing new theoretical challenges for the control community.

In this thesis, we study a set of estimation and control problems motivated by applications to next-generation Multi-Processor Systems on Chips (MPSoCs). Specifically, we consider problems related to state norm estimation, state estimation for positive systems, sensor selection and nonlinear output tracking. Although these problems are motivated by specific applications, the corresponding theory and techniques are developed for general dynamical systems. Below, we describe the specific challenges that we will be tackling.

1.1 State Estimation

Temperature monitoring is an important function of MPSoCs to guarantee system performance; overhigh processor temperature may bring reduction in system performance, timing delay variations or even permanent damages in the system [100]. Typically, the temperature estimation

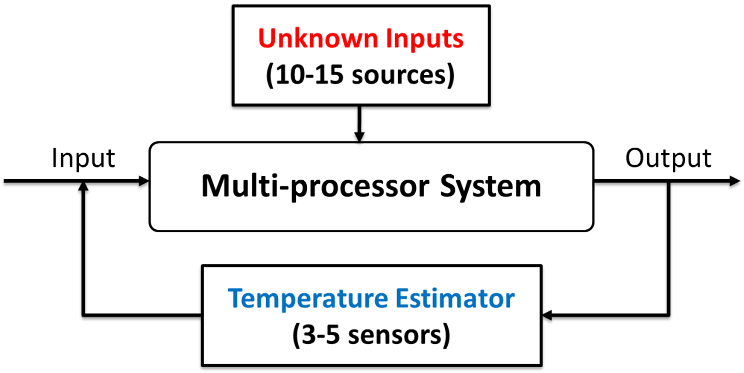


Figure 1.1: Temperature estimation for multi-processor systems.

techniques are based on a certain model of the thermal dynamics. A commonly studied thermal model in the state-space form is given as follows:

$$\begin{aligned} x[k+1] &= \underbrace{e^{-t_s F^{-1} G}}_{A} x[k] + \underbrace{G^{-1} F(I - e^{-t_s F^{-1} G}) F^{-1}}_{B} u[k] \\ y[k] &= C x[k], \end{aligned} \tag{1.1}$$

where x is the state vector which collects the temperature of specified thermal cells, u is the (unknown) input vector which represents the power consumption of different processors, y is the sensor measurement, and (A, B, C) are system matrices. Furthermore, I is the identity matrix, t_s is the sampling time, the matrix F captures the thermal capacitance of the cells, and the matrix G is the thermal conductance matrix. See [52, 122] for more details on this thermal model.

Motivated by the need to estimate the maximum temperature across a multi-processor system, in the first part of this thesis, we study state estimation for linear systems with unknown inputs. The problem of estimating the states of dynamical systems in the presence of unknown inputs has been studied extensively in the literature (e.g., see [49, 50, 141]). Such unknown inputs can be used to model uncertainty in the systems, including disturbances or faults [49], noise [50], and attacks [135]. For discrete-time linear systems with no constraints on the inputs, it has been shown that one can asymptotically reconstruct the states despite the unknown inputs if and only if the system is *strongly detectable* (or equivalently, has no unstable invariant zeros) [42, 68, 133]. However, in many applications, the property of strong detectability may not be satisfied. For example, a typical multi-processor system may have 10-15 unknown inputs while the system is only equipped with 3-5 sensors [123]; see Figure 1.1 for an illustration of the system. As we will see later in Section 2.2, in this case, the system cannot be strongly detectable since there are more unknown inputs than measurements.

When the system is not strongly detectable, one can either assume further information about the unknown inputs, or relax the estimation objective. A common approach, following the line of the Kalman filter, is to model the initial condition and inputs as random variables and stochastic processes, respectively, with certain statistics [2]. Another option is H_∞ or H_2 filtering, which attempts to minimize some norm of the operator that maps the unknown inputs to the estimation errors [36]. One can also take a deterministic approach where the initial condition and unknown inputs are assumed to be bounded in some set (e.g., ellipsoids or polytopes) and the goal is to find the set of possible states that are consistent with the observations; this leads to the so-called set-membership framework [10, 94]. Furthermore, partial state observers have also been considered which attempt to recover a particular function of the states [51, 134].

In contrast to the approaches discussed above, we investigate the influence of two alternative assumptions. First, we study the norm estimation problem. In certain applications (e.g., temperature monitoring in multi-processor systems), rather than aiming to reconstruct the states exactly, it suffices to provide an upper bound for the norm of the states (e.g., see [111]). In [67, 129], the authors proposed the notion of a *norm-estimator* for nonlinear systems which is driven by *known* inputs and outputs of the system and returns an estimate for the norm of the states. They showed that the system admits a norm-estimator if and only if it satisfies a property termed uniform IOSS (UIOSS).

We extend the concept of norm-estimation to the unknown input case by defining an *unknown input norm-observer*. In this setting, the norm of the initial condition and unknown inputs are bounded by some known constants and the objective is to estimate an upper bound for the norm of the states. In order to determine the conditions under which such an observer exists, we propose a notion of stability termed *bounded-input-bounded-output-bounded-state (BIBOBS) stability*, which is a fundamental property that is related to various existing system properties, including BIBO stability, BIBS stability, and IOSS (as we will discuss later in Section 2.5). We then provide a characterization of BIBOBS stability for discrete-time linear systems, which leads to a construction of an unknown input norm-observer.

Second, we consider state estimation for positive linear systems. The class of positive linear systems is a suitable model for applications where the states of the system are always positive (or at least nonnegative). For example, in the class of multi-processor systems, the positivity of the system comes from the physical nature of the thermal model. Besides the physical interpretation, the benefits of enforcing the property of positivity include simplifying the stability analysis [114] and bringing certain robustness properties [120]. We propose the notion of strong observability for positive systems, and provide a negative result by showing that positivity of the system is not helpful in designing (general) unknown input observers. We also consider the situation when we require the observer to return only positive estimates [30, 41].

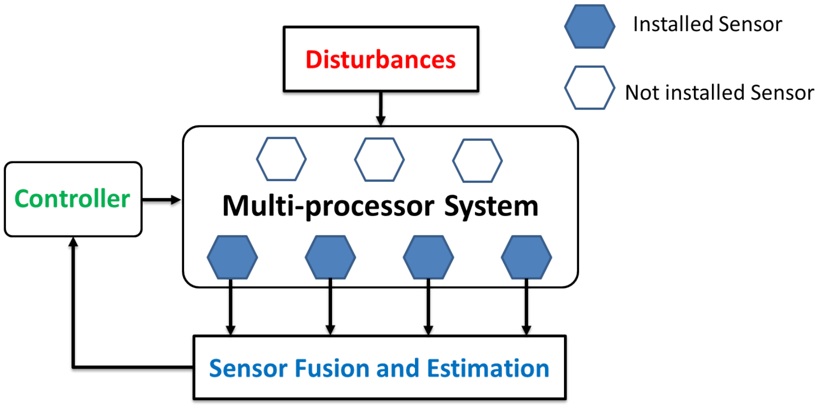


Figure 1.2: Sensor selection for multi-processor systems.

1.2 Sensor Selection

In the second part of this thesis, we consider the sensor selection problem for linear dynamical systems, which is motivated by the need to place thermal sensors in order to perform state estimation; see Figure 1.2 for an illustration of the system.

For the objective of estimating the state of a given linear Gauss-Markov system, there has been a growing literature in the past few years that studies how to dynamically select sensors at run-time to minimize certain metrics of the error covariance of the corresponding Kalman filter. This is known as the *sensor scheduling problem*, due to the fact that a different set of sensors can be chosen at each time-step (e.g., see [40, 55]).

However, in some applications (e.g., monitoring the state of a multi-processor system), we may not have the freedom to choose different sensors over run-time, and this leads to the design-time sensor selection problem (where the set of chosen sensors is not allowed to change over time). This problem has been studied in various forms, including cases where the objective is to guarantee a certain structural property of the system [108] or to optimize energy or information theoretic metrics [66, 130].[1] In [58], the authors studied the problem of estimating a *static* random variable and proposed various heuristics for sensor selection by using convex relaxation techniques. Sensor selection for parameter estimation was also studied in [146] where the uncertainty is due to deterministic and bounded disturbances. However, the results in [58, 146] do not directly translate to state estimation for dynamical systems and no performance guarantees were provided on the proposed algorithms.

[1] There have also been various recent studies of the dual design-time actuator placement problem (e.g., see [109, 140]).

In [26, 27], the authors studied the design-time actuator/sensor selection problem for continuous time linear dynamical systems using the sparsity-promoting framework from [78, 79, 109]. For the sensor selection problem, the objective is to design a Kalman gain matrix to minimize the resulting H_2 norm from the noise to the predicted estimation error. Sparsity is achieved by adding a penalty function that promotes column-sparsity of the gain matrix. In contrast to the formulation in [26, 27], in this thesis, we directly focus on minimizing functions of the steady state error covariances of discrete-time Kalman filters, and impose a hard constraint on the set of sensors to be chosen.

In [138, 139], the authors studied the design-time sensor selection problem for discrete-time linear time-varying systems over a finite horizon. They assumed that each sensor directly measures one component of the state vector, and the objective is either to minimize the estimation error with a cardinality constraint or to minimize the number of chosen sensors while guaranteeing a certain level of performance. Different from the formulation in [138, 139], we consider general measurement matrices and focus on minimizing the steady state estimation error of the Kalman filter.

In [147], the authors considered the same problem as the one we consider here, namely the design-time sensor selection problem for Kalman filtering in discrete-time linear dynamical systems with hard constraints. They showed that the sensor selection problem can be expressed as a semidefinite program (SDP). However, the results in [147] can only be applied to systems where the sensor noise terms are uncorrelated and no theoretical guarantees were provided on the performance of the proposed heuristics.

The objective of the sensor selection problem we study in this thesis is to choose a set of sensors (under certain constraints) to optimize either the *a priori* or the *a posteriori* error covariance of the corresponding Kalman filter; we will refer to these problems as the priori and posteriori *Kalman filtering sensor selection (KFSS)* problems, respectively. The priori KFSS problem is applicable for settings where a prediction of system states is needed and the posteriori KFSS problem is suitable for applications where the estimation can be conducted after receiving up-to-date measurements [2].

We explore the complexity of the priori and posteriori KFSS problems and investigate what factors of the system affect the performance of sensor selection algorithms by using the concept of the *sensor information matrix* [53]. We then study greedy algorithms and corresponding variants for the priori and posteriori KFSS problems.

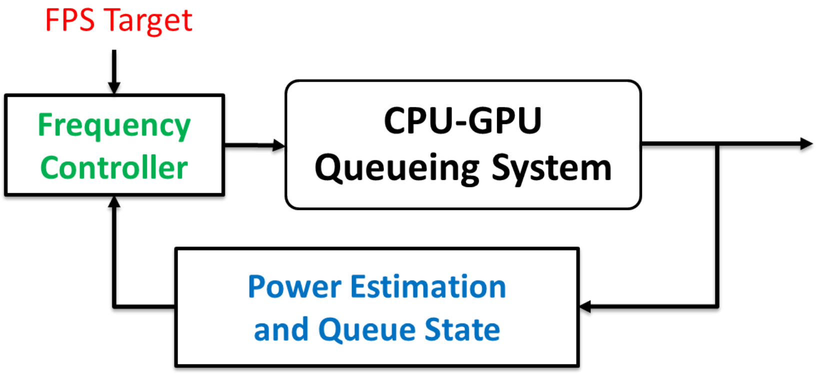


Figure 1.3: The CPU-GPU queueing system model proposed in [59].

1.3 Output Tracking

In the last part of this thesis, we study the output tracking problem for nonlinear systems with constraints, which is motivated by the need to optimize the energy consumption of the CPU-GPU subsystem in multi-processor systems while satisfying certain Quality of Service (QoS) requirements. Specifically, in [59], the authors proposed a queueing system model for the interaction between CPU and GPU; see Figure 1.3 for an illustration of the system. The objective is to drive the injection rate of the GPU queue to track a target number of Frames Per Second (FPS) while minimizing the power cost, and the controllable variables are the operating frequencies of the CPU and GPU. Furthermore, the queue occupancies need to be kept in a certain range. See Section 5.7.2 for more details on the system model.

There exist various approaches in the literature to address the tracking problem such as H_2 or H_∞ optimal control, see for example [73, 77]. Among these approaches, the model predictive control (MPC) based framework has been studied extensively due to its capability to handle constraints on the system [29, 31, 84, 86, 103].

A major disadvantage of the MPC is the need to solve an iterative optimization problem which may be challenging for online implementation, especially for systems with fast dynamics or controllers with limited online computational resources. Thus, in [7, 9], the authors proposed the explicit MPC (EMPC) approach for linear systems which has drawn much attention in the control community; see [1] and the references therein. Specifically, the authors derived an *explicit* form of the MPC control law in terms of the system state and used this explicit solution as a control look-up table online. However, it is not straightforward to extend the corresponding techniques to nonlinear systems [8, 38, 57].

One commonly studied scenario for the output tracking problem is the case where the reference signal is a constant. For more general classes of reference signals, it is often assumed that the desired trajectory is generated by an internal model [34,103]. However, in many applications, we may not have exact knowledge of the reference signal during the controller design stage, especially when the reference is generated by components running online. Furthermore, the lack of prior information on the reference signal brings further complexity to the implementation of EMPC.

In the literature, there are few works considering the tracking problem in the absence of known dynamics of the reference. In [31], the authors proposed a MPC controller for nonlinear systems with changing set-points by regarding the steady state and steady input as decision variables and using an offset cost function to penalize the tracking error. In [29], the author studied the nonlinear tracking problem with random references which are the outputs of a Markov process and proposed a MPC based approach which guarantees convergence to the desired reference when it remains constant. When the reference varies, the author characterized the set in which the tracking error lies. In [33], the authors addressed the problem of tracking target sets, i.e., the exact value of the desired output is not important as long as it stays in a certain set. The authors proposed a stable MPC formulation by imposing an additional cost term based on the concept of distance from a point to a set.

In this thesis, we study the output tracking problem for nonlinear systems with bounded reference signals. In order to reduce the online computation time and storage complexity while providing scalability to higher-dimensional state spaces, we extend the sampling-based explicit nonlinear model predictive controller (ENMPC) in [17] to the tracking problem by considering the reference as an extra dimension in the domain of the ENMPC. To alleviate the online search time of the traditional EMPC approaches, the basic idea of the sampling-based approach is to sample the state and reference signal space using deterministic sampling [17] and construct the ENMPC by using regression methods.

In [91], the authors proposed a tube-based robust control parametrization for the linear regulation problem by adding a feedback term to the nominal MPC, and a similar idea was generalized to nonlinear systems in [88, 89]. The basic idea of the tube-based robust nonlinear MPC is to generate a central path by using a nominal controller and to keep the states close to the central path by using an ancillary controller. By leveraging the tube-based robust MPC proposed in [88, 89], we also propose a robust variant of the sampling-based ENMPC for the case where there is an additive bounded disturbance.

1.4 Organization and Contributions

In Chapter 2 and Chapter 3, we study the problem of estimating the states of discrete-time linear systems with unknown inputs when the system is not strongly detectable. In Chapter 2, we consider the state norm estimation problem. Our contributions are as follows:

- We extend the concept of norm-estimation in [67, 129] to the unknown input case by defining an unknown input norm-observer.
- In order to characterize system properties that allow norm-estimation, we propose a notion of stability termed bounded-input-bounded-output-bounded-state (BIBOBS) stability. We show that the set of marginally stable eigenvalues (i.e., those with magnitude 1) plays a key role. Specifically, other than through unobservable strictly unstable eigenvalues (i.e., those with magnitude bigger than 1), there are only two ways to drive the states unbounded while keeping the output bounded: either by manipulating the controllable marginally stable eigenvalues with bounded inputs or by triggering the uncontrollable and unobservable marginally stable eigenvalues with carefully chosen initial conditions. As we show, care must be taken to identify the subset of marginally stable eigenvalues that cause such worst-case situations.
- We provide a comparison of BIBOBS stability with other classical stability properties, and illustrate the role of the concept of BIBOBS stability in the landscape of stability theory.

In Chapter 3, we consider state estimation for positive systems. Our contributions are as follows:

- We extend the notion of strong observability for positive systems, and show that the condition for a positive system to be strongly observable is the same as that for general systems. In other words, we show that positivity of the system is not helpful in designing (general) unknown input observers.
- For the case where positivity of the observer itself is desirable, we provide a linear programming based design procedure for delayed positive unknown input observers.

In Chapter 4, we study the design-time sensor selection problem for optimal filtering of discrete-time linear dynamical systems. Our contributions are as follows:

- We show that it is NP-hard to find the optimal solution of cost-constrained priori and posteriori KFSS problems, even under the assumption that the system is stable. It is often claimed in the literature that sensor selection problems are intractable [53, 58, 146]; however, except for certain problems with utility or energy based cost functions (e.g., see [11, 140]), to the best of our knowledge, there is still no explicit characterization of the complexity of the optimal filtering based sensor selection problems considered in this thesis.

- We provide insights into what factors of the system affect the performance of sensor selection algorithms. For the priori KFSS problem, we show that when the system is stable, the worst-case performance can be bounded by a parameter that depends only on the system dynamics matrix, and that the performance of a sensor selection algorithm cannot be arbitrarily bad if the system matrix is well conditioned, even under very large noise. For the posteriori KFSS problem, we show that for a given system, the worst-case performance of any selection of sensors can be upper-bounded in terms of the eigenvalues of the system noise covariance matrix and the corresponding sensor information matrix.

- We study greedy algorithms for the priori and posteriori KFSS problems and show that such algorithms are optimal (with respect to the corresponding KFSS problems) for two classes of systems. However, for general systems, as a negative result, we show that the cost functions of both the priori and posteriori KFSS problems do not necessarily have certain modularity properties, precluding the direct application of classical results from the theory of combinatorial optimization. Nevertheless, we show via simulations that greedy algorithms perform well in practice. Compared to the algorithms in [147], greedy algorithms provided in this thesis can be applied to a more general class of systems (where the sensor noises are correlated), are more efficient and (in simulations) provide comparable performance.

- We propose a variant of *a priori* covariance based and *a posteriori* covariance based greedy algorithms by optimizing an upper bound of the original cost functions (of the priori and posteriori KFSS problems) based on the Lyapunov equation. We show that the relaxed cost function is modular and that the running time of the corresponding greedy algorithm scales more slowly with the number of states in the system as compared to the original greedy algorithms, at the cost of a decrease in performance.

In Chapter 5, we study the output tracking problem for discrete-time nonlinear systems. Our contributions are as follows:

- We extend the sampling-based explicit nonlinear model predictive controller (ENMPC) in [17] to the tracking problem, and propose a low-complexity ENMPC to enable output tracking in the absence of complete reference signal information at design time. The proposed approach is suitable for applications with limited online computational resources, and the parameters can be easily customized to balance the trade off between performance and online computational complexity.
- We show that the proposed sampling based ENMPC guarantees stability and feasibility for all admissible references and is capable of steering the output of the system to any feasible set-point asymptotically.
- We extend the nominal ENMPC to enable robust output tracking for a class of additive disturbances in nonlinear systems, and show that the system output is restricted to an invariant neighborhood about the nominal trajectory, provided that the exogenous disturbance is sufficiently small.

1.5 Notation and Terminology

1.5.1 Sets

The set of integers, real numbers and complex numbers bigger than or equal to a are denoted as $\mathbb{Z}_{\geq a}$, $\mathbb{R}_{\geq a}$ and $\mathbb{C}_{\geq a}$, respectively.

For a closed set $\mathcal{X}$, $\mathcal{X}^c$ denotes its complement and int$(\mathcal{X})$ denotes an open set consisting of its interior points.

For two sets $\mathcal{X}, \mathcal{Y} \subset \mathbb{R}^n$ in a metric space (M, d), the Minkowski set sum is denoted by $\mathcal{X} \oplus \mathcal{Y} \triangleq \{x + y | x \in \mathcal{X}, y \in \mathcal{Y}\}$, the Pontryagin set difference is denoted by $\mathcal{X} \ominus \mathcal{Y} \triangleq \{x \in \mathcal{X} | x \oplus y \in \mathcal{X}, \forall y \in \mathcal{Y}\}$, and the distance between $\mathcal{X}$ and $\mathcal{Y}$ is denoted by $d(\mathcal{X}, \mathcal{Y}) \triangleq \inf_{x \in \mathcal{X}, y \in \mathcal{Y}} d(x, y)$.

For a set $\mathcal{X} \subset \mathbb{R}^n$ and a scalar α, we define $\alpha\mathcal{X} \triangleq \{\alpha x | x \in \mathcal{X}\}$.

For a set $\mathcal{X} \subset \mathbb{R}^n$, a matrix $M \in \mathbb{R}^{m \times n}$ and a scalar α, we define $M\mathcal{X} \triangleq \{Mx | x \in \mathcal{X}\}$ and $\alpha\mathcal{X} \triangleq \{\alpha x | x \in \mathcal{X}\}$.

For any set Z, we denote its cardinality and Lebesgue measure by $\mathrm{Card}(Z)$ and $\mathrm{Vol}(Z)$, respectively.

1.5.2 Matrices

For a square matrix $M \in \mathbb{C}^{n\times n}$, let M^T, M^H, $\mathrm{trace}(M)$, $\det(M)$, $\{\lambda_i(M)\}$ and $\{\sigma_i(M)\}$ be its transpose, conjugate transpose, trace, determinant, set of eigenvalues and set of singular values, respectively. The set of eigenvalues $\{\lambda_i(M)\}$ of M are ordered with nondecreasing magnitude (i.e., $|\lambda_1(M)| \geq \cdots \geq |\lambda_n(M)|$); the same order applies to the set of singular values $\{\sigma_i(M)\}$.

Let the null space and range space of a matrix M be $\mathcal{N}(M)$ and $\mathcal{R}(M)$, respectively.

The Euclidean norm of a vector and the corresponding induced matrix norm are both denoted by $\|\cdot\|$. For a vector x and a positive semidefinite matrix M, define $\|x\|_M^2 \triangleq x^T M x$.

A matrix is said to be *nonnegative* if all of its entries are nonnegative, *positive* if it is nonnegative and nonzero, and *strictly positive* if all of its entries are positive. A nonnegative, or positive, or strictly positive matrix A is denoted by $A \geqq 0$, $A \geq 0$, and $A > 0$, respectively [85]. The notation for vectors is similar.

A positive semi-definite matrix M is denoted by $M \succeq 0$ and $M \succeq N$ if $M - N \succeq 0$; the set of n by n positive semi-definite (resp. positive definite) matrices is denoted by $\mathbb{S}_+^n$ (resp. $\mathbb{S}_{++}^n$).

The identity matrix with dimension n is denoted by $I_{n\times n}$.

For a vector v, let $\mathrm{diag}(v)$ be the diagonal matrix with diagonal entries being the elements of v; for a set of matrices $\{M_i\}_{i=1}^q$, let $\mathrm{diag}(M_1, \cdots, M_q)$ be the block diagonal matrix with the i-th diagonal block being M_i.

1.5.3 Functions and Signals

For a random variable w, denote $\mathbb{E}[w]$ as its expectation.

For a signal z, we will denote its supremum norm over time interval $[0, k]$ by $\|z\|_{[0,k]} = \max_{0\leq j\leq k} \|z[j]\|$ [46].

A function $\alpha : \mathbb{R}_{\geq 0} \to \mathbb{R}_{\geq 0}$ is said to be of class $\mathcal{K}$ if it is continuous, strictly increasing and $\alpha(0) = 0$. If a $\mathcal{K}$-function is also unbounded, then it is said to be of class $\mathcal{K}_\infty$. A function $\beta : \mathbb{R}_{\geq 0} \times \mathbb{R}_{\geq 0} \to \mathbb{R}_{\geq 0}$ is said to be of class $\mathcal{KL}$ if $\beta(\cdot, k)$ is of class $\mathcal{K}$ and $\beta(r, k) \to 0$ as $k \to \infty, \forall r \geq 0$ [67].

Chapter 2

State Norm Estimation for Linear Dynamical Systems

2.1 Introduction

Estimation and filtering are one of the most important topics of control theory and signal processing. In the presence of unknown inputs, one can attempt to construct an *unknown input observer* to decouple the influence of unknown inputs [141]. However, for discrete-time linear systems with no constraints on the inputs, if the system is not strongly detectable (i.e., has some unstable invariant zeros), there does not exist any asymptotic observer which can recover the states. For example, this is the case in the application of temperature estimation for multi-processor systems since typically there are more unknown inputs than measurements in such systems [122].

When the system is not strongly detectable, in order to perform state estimation, one may need further information either about the initial condition or about the unknown inputs. In this chapter, instead of aiming to reconstruct the states exactly, we study a setting where the norm of the initial condition and unknown inputs are bounded by some known constants. The objective is to estimate an upper bound for the norm of the states. We extend the concept of norm estimation proposed in [67, 129] to the unknown input case, and determine the conditions under which such an unknown input norm-observer exists.

To solve the norm estimation problem, we propose a notion of stability termed bounded-input-bounded-output-bounded-state (BIBOBS) stability. In addition to its implications for unknown input norm-observers, the concept of BIBOBS stability has applications to system monitoring problems, such as the *false data injection* problem studied in [96, 97]; see Figure 2.1 for

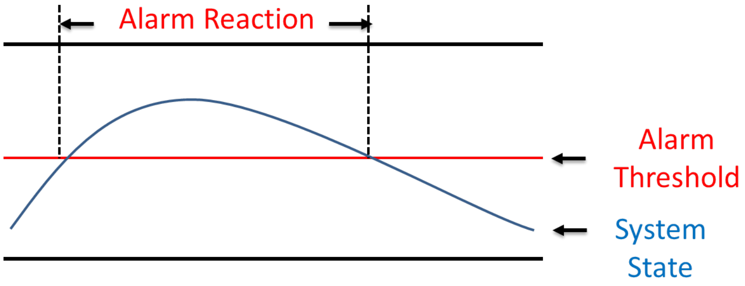


Figure 2.1: Illustration of the alarm system.

an illustration of the problem. In this setting, the system is a fault detection filter, the output is interpreted as the residue of the filter, the bound on the output represents the threshold above which an alarm is raised, and the states are interpreted as the estimation error; the goal of the attacker is to maximize the error (i.e., maximize the norm of the states) while remaining undetected through the output. If the attacker is constrained to apply bounded inputs (a scenario that is not considered in [96,97]), BIBOBS stability is required for preventing worst-case attacks (i.e., those causing arbitrarily large error without triggering the alarm [136]).

Although related stability notions for linear systems such as BIBO stability and BIBS stability have relatively simple characterizations [128], the proofs of the conditions for BIBOBS stability appear to be significantly more complicated. We show that the set of marginally stable eigenvalues (i.e., those with magnitude 1) plays a key role and thus we must carefully identify the subset of marginally stable eigenvalues that cause the worst-case situations. Moreover, we discuss the relationships between BIBOBS stability and other existing system properties.

The rest of this chapter is organized as follows. In Section 2.2, we provide some background on state estimation techniques. In Section 2.3, we define the concept of an unknown input norm-observer for linear systems with unknown inputs. In Section 2.4, we propose the notion of BIBOBS stability and provide necessary and sufficient conditions for linear systems to be BIBOBS stable. In Section 2.5, we discuss BIBOBS stability in the context of other existing stability notions. In Section 2.6, we illustrate the results via examples and simulations. Some concluding remarks are given in Section 2.7.

2.2 Background: State Estimation

One of the most important issues in state estimation consists in defining appropriate models of uncertainty. In the literature, three types of disturbances are typically studied. First is the stochastic setting, where the initial condition and disturbances are modeled as random variables. If certain statistics of the initial condition and disturbances are known, one can adopt efficient filtering strategies (e.g., Kalman filter) to reconstruct the states [15]. Second is to assume bounded disturbances, which is a deterministic approach to model uncertainties and leads to set-membership filtering [10, 21, 118]. This framework is applicable when the disturbances are not stochastic or have statistics that are difficult to identify. Third, one can assume that the disturbances are totally unknown, and in this case, the disturbances are often referred to as unknown inputs; this model places no constraints on the inputs compared to the previous models but requires stricter system properties [133].

In this section, we review some background on state estimation techniques. First, we introduce the concepts of strong observability and strong detectability from linear system theory, which are useful in the study of (totally) unknown inputs. Then we review the framework of set-membership filtering, which is similar to the norm estimation problem studied in this chapter. We will review the theory of Kalman filtering in Chapter 4, which serves as a basis for the sensor selection problem studied therein. Note that there also exist other important estimation frameworks in the literature, e.g., the H_∞ filtering approach; we omit a discussion on these techniques since they are not directly related to the approach studied in this thesis. See [127] for a more comprehensive review for the estimation and filtering techniques.

2.2.1 Linear System Estimation

Consider the discrete-time linear system

$$\begin{aligned} x[k+1] &= Ax[k] + Bu[k] \\ y[k] &= Cx[k] + Du[k] \end{aligned} \tag{2.1}$$

with state vector $x \in \mathbb{R}^n$, output (measurement) $y \in \mathbb{R}^p$, unknown input $u \in \mathbb{R}^m$, and system matrices (A, B, C, D) of appropriate dimensions. The initial condition of the system is $x[0]$. The unknown inputs u may represent disturbances, faults, attacks, or other uncontrolled uncertainties. As we have mentioned in Section 1.1, in the context of multi-processor systems, the state x represents the temperature of different thermal cells and the unknown input u captures the power consumption of different processors.

Let the *observability matrix* and *invertibility matrix* of system (2.1) (with delay L) be denoted by

$$\mathcal{O}_L = \begin{bmatrix} C \\ CA \\ \vdots \\ CA^L \end{bmatrix},$$

and

$$\mathcal{J}_L = \begin{bmatrix} D & 0 & 0 & \dots & 0 \\ CB & D & 0 & \dots & 0 \\ CAB & CB & D & \dots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ CA^{L-1}B & CA^{L-2}B & CA^{L-3}B & \dots & D \end{bmatrix},$$

respectively. Note that

$$y[k:k+L] = \mathcal{O}_L x[k] + \mathcal{J}_L u[k:k+L],$$

where $y[k:k+L] \in \mathbb{R}^{(L+1)p}$ and $u[k:k+L] \in \mathbb{R}^{(L+1)m}$ are the outputs and inputs over $L+1$ time steps, respectively.

Definition 1 (Strong Observability)**.** *The system (2.1) is said to be* ***strongly observable*** *if for any initial condition $x[0]$ and any sequence of unknown inputs $\{u[k]\}$, there exists some positive integer L such that $x[0]$ can be recovered from $y[0:L]$.*

In words, if one wants to reconstruct the states in finite time, the system must be *strongly observable*; otherwise, there exists some nonzero initial condition and an input sequence such that the system output is always zero, which is indistinguishable from the case of zero initial conditions and inputs.

Theorem 1 ([65])**.** *The system (2.1) is strongly observable if and only if*

$$\text{rank}(\begin{bmatrix} \mathcal{O}_L & \mathcal{J}_L \end{bmatrix}) = n + \text{rank}(\mathcal{J}_L)$$

for some $L \leq n$.

If the above equality holds for some L, then we can recover $x[k]$ by using $y[k:k+L]$ (without any information about $u[k:k+L]$). Besides the above algebraic characterization, there also exists a matrix pencil based characterization of strong observability.

Definition 2 (Matrix Pencil)**.** *For the linear system (2.1), the matrix*

$$P(z) = \begin{bmatrix} A - zI_n & B \\ C & D \end{bmatrix}$$

*is called the **matrix pencil** of the set (A, B, C, D), where $z \in \mathbb{C}$ is the variable.*

Theorem 2 ([126])**.** *The system (2.1) is strongly observable if and only if*

$$\operatorname{rank}(P(z)) = n + m$$

for all $z \leq \mathbb{C}$. In other words, the system is strongly observable if and only if it has no invariant zeros.

A relaxed notion of strong observability is the concept of strong detectability, which is defined as follows.

Definition 3 (Strong Detectability)**.** *The system (2.1) is said to be **strongly detectable** if for any initial condition $x[0]$ and any sequence of unknown inputs $\{u[k]\}$, $y[k] = 0, \forall k$, implies that $x[k] \to 0$ as $k \to \infty$.*

If the system is strongly detectable, then we can asymptotically reconstruct the states despite the disturbances. We also have the following matrix pencil based characterization for strong detectability.

Theorem 3 ([126])**.** *The system (2.1) is strongly detectable if and only if*

$$\operatorname{rank}(P(z)) = n + m$$

for all $|z| \geq 1$. In other words, the system is strongly detectable if and only if it has no unstable invariant zeros.

2.2.2 Set-membership Filtering

In this subsection, we review the set-membership filtering approach, which is applicable to the case where we only know that the disturbances are bounded, without any knowledge of other statistics. In this setting, the disturbances are assumed to be contained in bounded sets such as ellipsoids, intervals, or polytopes. Here we focus on the ellipsoid-based uncertainty model; see [94] for a discussion on other types of uncertainty models.

Consider the following version of system (2.1):

$$\begin{aligned} x[k+1] &= Ax[k] + w[k] \\ y[k] &= Cx[k] + v[k], \end{aligned} \tag{2.2}$$

where w and v are the process and measurement disturbances, respectively.

Assume that the uncertain quantities (the initial condition, input disturbance and measurement disturbance) are subject to the following constraints:

$$\begin{gathered} [x[0] - x_0]^T P^{-1} [x[0] - x_0] \leq 1 \\ w^T Q^{-1} w \leq 1 \\ v^T R^{-1} v \leq 1, \end{gathered}$$

where x_0 is some known vector, and P^{-1}, Q^{-1} and R^{-1} are positive definite weighting matrices. Let $X[k]$ be the set of states that is consistent with the constraints and available measurements at time-step k. In [10], the authors provided an outer ellipsoidal approximation $X^*[k]$ of $X[k]$ (i.e., $X^*[k]$ guarantees that $X[k] \subset X^*[k], \forall k$). Specifically, we have $X^*[k] = \{v | (v - x^*[k])^T \Sigma[k] (v - x^*[k]) \leq 1\}$, where $x^*[k]$ and $\Sigma[k]$ capture the center and volume of the ellipsoid, respectively. The weighting matrix $\Sigma[k]$ can be computed by using a certain recursive technique which is similar to the Riccati recursion; the corresponding algorithm can be regarded as a deterministic interpretation of the Kalman filter. However, the possible set of states $X[k]$ is, in general, not an ellipsoid, and it is difficult to get an exact description of $X[k]$. A practical technique to is to pursue both outer and inner approximations of $X[k]$; the problem of choosing these approximations to optimize certain metrics has also been studied (e.g., see [21]).

2.3 Unknown Input Norm-observers for Linear Systems with Unknown Inputs

As mentioned in the Section 2.2, one can reconstruct the states asymptotically despite the unknown inputs if and only if the system is strongly detectable. However, in many applications, strong observability or strong detectability of the system may not hold. For example, from Theorem 2 and Theorem 3, we can see that if the matrix B has full column rank and there are more unknown inputs than outputs (i.e., $m > p$ in system (2.1)), the system cannot be strongly observable or strongly detectable. In this case, we need to relax the objective of exactly reconstructing the states.

In practice, the inputs (either known or unknown) are often bounded in magnitude; for example, in the context of multi-processor systems, the unknown inputs are always bounded due to physical constraints and energy limitations [123]. Furthermore, one may have (or assume) prior information about certain properties of the initial condition of the system. Thus, in this section, we consider a setting similar to the set-membership approach where we assume that the norm of the initial condition $\|x[0]\|$ and the inputs are upper bounded by some known constants $x_{\text{max}} > 0$ and $u_{\text{max}} > 0$, respectively, i.e., the constants x_{max} and u_{max} satisfy that

$$\|x[0]\| \leq x_{\text{max}}, \forall k \in \mathbb{Z}_{\geq 0},$$

$$\|u[k]\| \leq u_{\text{max}}, \forall k \in \mathbb{Z}_{\geq 0}.$$

Instead of attempting to reconstruct the possible set of states consistent with the outputs as in the set-membership approach, our objective is to build an unknown input norm-observer for the states of the system by utilizing the information on the bounds of the initial condition and unknown inputs, defined as follows.

Definition 4 (Unknown Input Norm-observer)**.** *For system (2.1), we say that there exists an* ***unknown input norm-observer*** $\hat{x}$ *of the norm of the states* $\|x\|$ *if there exist functions* $\gamma_1, \gamma_2, \gamma_3 \in \mathcal{K}_\infty$ *such that*

$$\|x[k]\| \leq \hat{x}[k] \triangleq \gamma_1(x_{max}) + \gamma_2(\|y\|_{[0,k]}) + \gamma_3(u_{max}), \forall k, x[0], u.$$

The above definition of an unknown input norm-observer is similar to the concept of norm-estimation studied in [67]; the difference is that we do not use information on the unknown inputs (except for an upper bound on its norm) and we do not require the influence of the initial condition to asymptotically decay to zero. As we will see later in the next section, it turns out that these differences make the characterization of the unknown input norm-observer different from the concept of UIOSS proposed in [67].

2.4 BIBOBS Stability

In order to characterize system properties that allow norm-estimation, we introduce the concept of bounded-input-bounded-output-bounded-state (BIBOBS) stability. We start by formally defining the property of BIBOBS stability as follows.

Definition 5 (BIBOBS Stability)**.** *The system (2.1) is said to be* ***BIBOBS stable*** *if the state is bounded whenever the input and output of the system are bounded. Mathematically, a BIBOBS system satisfies*

$$\forall M_1, M_2, M_3 \in \mathbb{R}_{\geq 0} \text{ such that } \|x[0]\| \leq M_1, \|u[k]\| \leq M_2, \|y[k]\| \leq M_3, \forall k,$$
$$\Rightarrow \exists M \in \mathbb{R}_{\geq 0} \text{ such that } \|x[k]\| \leq M, \forall k.$$

In words, BIBOBS stability characterizes the ability of bounded disturbances to drive the state unbounded while remaining undetected (by keeping the output bounded). In the context of temperature estimation in multi-processor systems, the property of BIBOBS stability is necessary to avoid overhigh temperature to damage the system without being detected.

The following result indicates that the system being BIBOBS stable has a stronger implication: the norm of the state can be upper bounded by specific functions of the norms of the initial condition, input and output.

Proposition 1. *The system (2.1) is BIBOBS stable if and only if there exist functions $\alpha_1, \alpha_2, \alpha_3 \in \mathcal{K}_\infty$ such that*

$$\|x[k]\| \leq \alpha_1(\|x[0]\|) + \alpha_2(\|u\|_{[0,k]}) + \alpha_3(\|y\|_{[0,k]}), \forall k, x[0], u.$$

Proof. If the functions $\alpha_1, \alpha_2, \alpha_3 \in \mathcal{K}_\infty$ satisfying the condition in Proposition 1 exist, we can choose $M = \alpha_1(M_1) + \alpha_2(M_2) + \alpha_3(M_3)$ in Definition 5 and thus the system is BIBOBS stable.

For the other direction, by our main result in this section (i.e., Theorem 5), we will see that when the system is BIBOBS stable, such functions $\alpha_1, \alpha_2, \alpha_3 \in \mathcal{K}_\infty$ always exist (see Remark 2 for a specific construction). □

The condition in Proposition 1 thus serves as an alternative definition for BIBOBS stability. Based on Proposition 1, we can relate BIBOBS stability to the norm-estimation objective described in the previous section, as illustrated by the following result.

Theorem 4. *The system (2.1) admits an unknown input norm-observer if and only if it is BIBOBS stable.*

Proof. If the system is BIBOBS stable, by Proposition 1, we can just replace $\|x[0]\|$ and $\|u\|_{[0,k]}$, $\forall k$, by $x_{\max}$ and $u_{\max}$, respectively, and we get an unknown input norm-observer.

When the system is not BIBOBS stable, as we will prove later in this section (i.e., Lemma 1), for any $x_{\max}, u_{\max} > 0$, there exist some initial condition and input sequence such that the outputs are bounded but the states become unbounded, and thus there does not exist an unknown input norm-observer. □

In the rest of this section, we derive the conditions under which the system (2.1) is BIBOBS stable. As we will see, the constraint on the norm of the inputs limits their ability to drive the state of the system to be unbounded while remaining undetected via the outputs. A construction for the unknown input norm-observer will follow as a byproduct of the proof. We start with the following definition.

Definition 6 (Strictly Unstable and Marginally Stable Eigenvalue)**.** *For an eigenvalue λ of the matrix A, we say that λ is* ***strictly unstable*** *if it has magnitude bigger than* 1*, and* ***marginally stable*** *if it has magnitude* 1.

To give a characterization for BIBOBS stability, we consider a decomposition of the system (2.1) by first transforming the system into its Kalman canonical form and then applying a further transformation to convert the uncontrollable components into their Jordan forms. Let the transformation matrix be H. The transformed system is given by

$$\begin{aligned} x_H[k+1] &= A_H x_H[k] + B_H u[k] \\ y[k] &= C_H x_H[k] + Du[k], \end{aligned} \tag{2.3}$$

where $x_H = Hx$, $A_H = HAH^{-1}$, $B_H = HB$, and $C_H = CH^{-1}$ have the form

$$x_H = \begin{bmatrix} x_{co} \\ x_{c\bar{o}} \\ x_{\bar{c}o} \\ x_{\bar{c}\bar{o}} \end{bmatrix}, A_H = \begin{bmatrix} A_{co} & 0 & A_{13} & 0 \\ A_{21} & A_{c\bar{o}} & A_{23} & A_{24} \\ 0 & 0 & J_{\bar{c}o} & 0 \\ 0 & 0 & A_{43} & J_{\bar{c}\bar{o}} \end{bmatrix}, B_H = \begin{bmatrix} B_{co} \\ B_{c\bar{o}} \\ 0 \\ 0 \end{bmatrix},$$
$$C_H = \begin{bmatrix} C_{co} & 0 & C_{\bar{c}o} & 0 \end{bmatrix}.$$

In the above form, the state x_{co} is both controllable and observable, the state $x_{c\bar{o}}$ is controllable but not observable, the state $x_{\bar{c}o}$ is observable but not controllable, and the state $x_{\bar{c}\bar{o}}$ is neither controllable nor observable. The matrices $J_{\bar{c}o}$ and $J_{\bar{c}\bar{o}}$ are in Jordan forms. If some of these components do not exist, we consider their corresponding matrices to have size 0. Note that $\|H\|^{-1}\|x_H\| \leq \|x\| \leq \|H^{-1}\|\|x_H\|$.

The following theorem is our main result in this chapter and characterizes BIBOBS stability for linear systems.

Definition 7. *For an eigenvalue λ of $J_{\bar{c}\bar{o}}$, denote $A_{43}(\lambda)$ to be the matrix consisting of the rows of A_{43} corresponding to the Jordan blocks of λ.*

Theorem 5. *The system (2.1) is BIBOBS stable if and only if in the form (2.3), all of the following conditions are satisfied:*

(i) *The matrix $A_{c\bar{o}}$ is stable;*

(ii) *The matrix $J_{\bar{c}\bar{o}}$ does not contain strictly unstable eigenvalues, or marginally stable Jordan blocks with size bigger than 1;*

(iii) *For any shared marginally stable eigenvalue λ_m of $J_{\bar{c}o}$ and $J_{\bar{c}\bar{o}}$, each eigenvector $\bar{v}_m$ of $J_{\bar{c}o}$ corresponding to λ_m satisfies $\bar{v}_m \in \mathcal{N}(A_{43}(\lambda_m))$.*

Remark 1. *From Theorem 5, it is easy to see that BIBOBS stability is strictly weaker than detectability (which requires all unobservable eigenvalues to be stable); BIBOBS stability allows the block $J_{\bar{c}\bar{o}}$ in the form (2.3) to have marginally stable eigenvalues, as long as conditions (ii) and (iii) in Theorem 5 are satisfied.*

We split the proof of the above theorem into the following two lemmas. The first lemma proves the necessity of the conditions in Theorem 5. The basic idea behind the proof is that if the conditions in Theorem 5 fail, then for any constant upper bounds $x_{\max}, u_{\max}, y_{\max} > 0$, there is an initial state and a carefully constructed sequence of inputs $\{u[k]\}$ such that $\|y[k]\| \leq y_{\max}$ for all k while $\|x[k]\| \to \infty$ as $k \to \infty$.

Lemma 1. *The system (2.1) is BIBOBS stable only if in the form (2.3), all of the conditions in Theorem 5 are satisfied.*

Proof. Note that the conditions (i) and (ii) in Theorem 5 are equivalent to the following three conditions:

- The matrices $A_{c\bar{o}}$ and $J_{\bar{c}\bar{o}}$ do not contain strictly unstable eigenvalues;
- The matrix $J_{\bar{c}\bar{o}}$ does not contain marginally stable Jordan blocks with size bigger than 1;
- The matrix $A_{c\bar{o}}$ does not contain marginally stable eigenvalues.

We will first show that if any of the above three conditions is not satisfied, the system is not BIBOBS stable, and then show that if condition (iii) in Theorem 5 fails, the system is not BIBOBS stable.

The analysis for the case where $A_{c\bar{o}}$ or $J_{\bar{c}\bar{o}}$ contains strictly unstable eigenvalues is covered by the standard argument for undetectable systems. Specifically, if the matrix $A_{c\bar{o}}$ (resp. $J_{\bar{c}\bar{o}}$) contains some strictly unstable eigenvalue, we can choose the initial condition $x_{c\bar{o}}[0]$ (resp. $x_{\bar{c}\bar{o}}[0]$) to be the corresponding eigenvector, and set the initial condition for the other subsystems and the inputs to be zero. This results in the output being zero for all time, while $\|x[k]\| \to \infty$ as $k \to \infty$.

Now we show that if the matrix $J_{\bar{c}\bar{o}}$ contains some marginally stable eigenvalue λ_2 and at least one of the associated Jordan blocks has size bigger than 1, then there exists some choice of initial condition which causes the states to be unbounded while keeping the output zero. Choose the initial condition $\begin{bmatrix} x_{co}[0]^T & x_{c\bar{o}}[0]^T & x_{\bar{c}o}[0]^T \end{bmatrix}^T$ to be zero and let the inputs be identically zero. Since $x_{\bar{c}\bar{o}}$ is unobservable and $x_{\bar{c}\bar{o}}$ only influence the other unobservable state $x_{c\bar{o}}$, the output is always zero (under the current choice of initial conditions and inputs) and we just need to show that there exists some choice of initial condition $x_{\bar{c}\bar{o}}[0]$ which causes $\|x_{\bar{c}\bar{o}}[k]\|$ to be unbounded. Without loss of generality, suppose that $J_{\bar{c}\bar{o}} = \lambda_2 I + S$ where

$$S \triangleq \begin{bmatrix} 0 & 1 & & \\ & 0 & \ddots & \\ & & \ddots & 1 \\ & & & 0 \end{bmatrix}.$$

Note that if there exist other Jordan blocks in $J_{\bar{c}\bar{o}}$, we can choose the initial condition of the corresponding states to be zero. Choose the initial condition $x_{\bar{c}\bar{o}}[0]$ to be $[0 \; w \; 0 \; \cdots \; 0]^T$ where $w \neq 0$ is some constant. Then we have

$$\begin{aligned} \|x_{\bar{c}\bar{o}}[k]\| &= \|J_{\bar{c}\bar{o}}^k x_{\bar{c}\bar{o}}[0]\| \\ &= \|\sum_{i=0}^{k} \lambda_2^i S^{k-i} x_{\bar{c}\bar{o}}[0]\| \\ &= \| \begin{bmatrix} k\lambda_2^{k-1} w \; \lambda_2^k w \; 0 \; \cdots \; 0 \end{bmatrix}^T \| \\ &\geq k\|\lambda_2^{k-1} w\|. \end{aligned}$$

Thus, $\|x_{\bar{c}\bar{o}}[k]\|$ becomes unbounded as $k \to \infty$. Note that since we can always scale the constant w and the other components of the initial condition and the inputs are both zero, the analysis holds for any upper bounds $x_{\max}$ and $u_{\max}$.

Next we show that if the matrix $A_{c\bar{o}}$ contains a marginally stable eigenvalue λ_1, there exist some choice of initial condition and inputs that cause the states to be unbounded while keeping the output bounded. Note that in the form (2.3), the uncontrollable components are not influenced by the controllable components or the inputs. Thus, by choosing the initial condition of the uncontrollable components $\begin{bmatrix} x_{\bar{c}o}[0]^T & x_{\bar{c}\bar{o}}[0]^T \end{bmatrix}^T$ to be zero, we can focus on the following controllable subsystem:

$$\begin{aligned} \begin{bmatrix} x_{co}[k+1] \\ x_{c\bar{o}}[k+1] \end{bmatrix} &= \begin{bmatrix} A_{co} & 0 \\ A_{21} & A_{c\bar{o}} \end{bmatrix} \begin{bmatrix} x_{co}[k] \\ x_{c\bar{o}}[k] \end{bmatrix} + \begin{bmatrix} B_{co} \\ B_{c\bar{o}} \end{bmatrix} u[k] \\ y[k] &= \begin{bmatrix} C_{co} & 0 \end{bmatrix} \begin{bmatrix} x_{co}[k] \\ x_{c\bar{o}}[k] \end{bmatrix} + Du[k]. \end{aligned} \tag{2.4}$$

Denote the dimension of $\begin{bmatrix} x_{co}^T & x_{c\bar{o}}^T \end{bmatrix}^T$ by n_c, and denote the controllability matrix of subsystem (2.4) by $\mathcal{C}_{n_c-1}$. Let v_1 be any eigenvector of $A_{c\bar{o}}$ associated with λ_1. Choose the initial condition of subsystem (2.4) to be $\begin{bmatrix} 0 \\ v_1 \end{bmatrix}$. For $i = 0, 1, 2, \ldots$, choose the input sequence over time interval $[in_c, (i+1)n_c - 1]$ to be such that

$$\mathcal{C}_{n_c-1} u[in_c : (i+1)n_c - 1] = \begin{bmatrix} 0 \\ \lambda_1^{(i+1)n_c} v_1 \end{bmatrix}.$$

Note that since the subsystem (2.4) is controllable, $\mathcal{C}_{n_c-1}$ has full rank and such an input sequence always exists.[1] Further note that since $|\lambda_1| = 1$, the supremum norm of the input sequence is bounded. One can check that under this choice of initial condition and inputs, for any integer $i \geq 0$,

$$\begin{bmatrix} x_{co}[in_c] \\ x_{c\bar{o}}[in_c] \end{bmatrix} = \begin{bmatrix} 0 \\ (i+1)\lambda_1^{in_c} v_1 \end{bmatrix}$$

and

$$y[in_c : (i+1)n_c - 1] = \mathcal{J}_{n_c-1} u[in_c : (i+1)n_c - 1]$$

where $\mathcal{J}_{n_c-1}$ is the invertibility matrix of subsystem (2.4) [117]. We can see that in this case, the states become unbounded while the inputs and outputs are bounded and thus, the system is not BIBOBS stable. Note that since we can always scale the initial condition and inputs, the above analysis holds for any upper bounds $x_{\max}$ and $u_{\max}$.

Finally, we show that if condition (iii) in Theorem 5 is not satisfied, the system is not BIBOBS stable. In this case, $J_{\bar{c}\bar{o}}$ and $J_{\bar{c}o}$ share the same marginally stable eigenvalue λ_m, with an

[1] Note that the resulting input sequence may be complex valued; in this case, either the real part or the imaginary part (or both parts) of the sequence will be a real valued input sequence which drives the state to be unbounded while keeping the output bounded.

eigenvector $\bar{v}_m$ of $J_{\bar{c}o}$ such that $\bar{v}_m \notin \mathcal{N}(A_{43}(\lambda_m))$. Since condition (ii) in Theorem 5 must be satisfied (as we have argued above), without loss of generality, we can assume that $J_{\bar{c}\bar{o}}$ is of the form

$$J_{\bar{c}\bar{o}} = \begin{bmatrix} \lambda_m I & 0 & 0 \\ 0 & A_d & 0 \\ 0 & 0 & A_s \end{bmatrix},$$

where A_d is a diagonal matrix containing the marginally stable eigenvalues of $J_{\bar{c}\bar{o}}$ except λ_m and A_s contains the stable eigenvalues of $J_{\bar{c}\bar{o}}$.

Choose the initial condition $\begin{bmatrix} x_{co}[0]^T & x_{c\bar{o}}[0]^T & x_{\bar{c}\bar{o}}[0]^T \end{bmatrix}^T$ to be zero and set $x_{\bar{c}o}[0] = \bar{v}_m$ (so that $x_{\bar{c}o}[k] = \lambda_m^k \bar{v}_m, \forall k$). Then from (2.3) we know that the state $x_{\bar{c}\bar{o}}$ evolves as follows:

$$\begin{aligned} x_{\bar{c}\bar{o}}[k] &= J_{\bar{c}\bar{o}}^k x_{\bar{c}\bar{o}}[0] + \sum_{t=0}^{k-1} J_{\bar{c}\bar{o}}^{k-1-t} A_{43} x_{\bar{c}o}[t] \\ &= \sum_{t=0}^{k-1} J_{\bar{c}\bar{o}}^{k-1-t} A_{43} x_{\bar{c}o}[t] \\ &= \sum_{t=0}^{k-1} \begin{bmatrix} \lambda_m I & 0 & 0 \\ 0 & A_d & 0 \\ 0 & 0 & A_s \end{bmatrix}^{k-1-t} \begin{bmatrix} A_{43}(\lambda_m) \\ A_{43}(A_d) \\ A_{43}(A_s) \end{bmatrix} x_{\bar{c}o}[t] \\ &= \sum_{t=0}^{k-1} \begin{bmatrix} \lambda_m^{k-1-t} I & 0 & 0 \\ 0 & A_d^{k-1-t} & 0 \\ 0 & 0 & A_s^{k-1-t} \end{bmatrix} \begin{bmatrix} A_{43}(\lambda_m) \\ A_{43}(A_d) \\ A_{43}(A_s) \end{bmatrix} \lambda_m^t \bar{v}_m \\ &= \begin{bmatrix} k\lambda_m^{k-1} A_{43}(\lambda_m)\bar{v}_m \\ \sum_{t=0}^{k-1} A_d^{k-1-t} A_{43}(A_d)\lambda_m^t \bar{v}_m \\ \sum_{t=0}^{k-1} A_s^{k-1-t} A_{43}(A_s)\lambda_m^t \bar{v}_m \end{bmatrix}, \end{aligned}$$

where $A_{43}(A_d)$ and $A_{43}(A_s)$ denote the matrices consisting of the rows of A_{43} corresponding to A_d and A_s, respectively. Since $\bar{v}_m \notin \mathcal{N}(A_{43}(\lambda_m))$, $|\lambda_m| = 1$, we know that for any $k \in \mathbb{Z}_{\geq 0}$,

$$\begin{aligned} \|x_{\bar{c}\bar{o}}[k]\| &\geq k\|\lambda_m^{k-1} A_{43}(\lambda_m)\bar{v}_m\| \\ &= k\|A_{43}(\lambda_m)\bar{v}_m\|, \end{aligned}$$

and thus $\|x_{\bar{c}\bar{o}}\|$ is unbounded. Now we just need to show that under the current choice of initial condition, there exists some sequence of inputs which keeps the output bounded.

Note that since $|\lambda_m| = 1$ and $\|x_{\bar{c}o}[k]\| = \|\lambda_m^k \bar{v}_m\| = \|\bar{v}_m\|, \forall k$, $\|x_{\bar{c}o}\|$ is always bounded. Thus, we can focus on the other observable subsystem associated with state x_{co} which (from the form (2.3)) has the following dynamics:

$$x_{co}[k+1] = A_{co}x_{co}[k] + A_{13}x_{\bar{c}\bar{o}}[k] + B_{co}u[k].$$

Denote the dimension of x_{co} by n_{co}, and denote the controllability matrices of (A_{co}, B_{co}) and (A_{co}, A_{13}) by $\mathcal{C}_{n_{co}-1}$ and $\mathcal{C}'_{n_{co}-1}$, respectively. Choose the input sequence over time interval $[in_{co}, (i+1)n_{co}-1]$ to satisfy

$$\mathcal{C}_{n_{co}-1}u[in_{co} : (i+1)n_{co}-1] + \mathcal{C}'_{n_{co}-1}x_{\bar{c}\bar{o}}[in_{co} : (i+1)n_{co}-1] = 0.$$

Note that since (A_{co}, B_{co}) is controllable, $\mathcal{C}_{n_{co}-1}$ has full rank and such an input sequence always exists. Then one can check that $x_{co}[in_{co}] = 0, \forall i \in \mathbb{Z}_{\geq 0}$, and thus $\|y[k]\|$ is always bounded. Note that since we can always scale the initial condition $x_{\bar{c}\bar{o}}[0] = \bar{v}_m$, the analysis holds for any upper bounds $x_{\max}$ and $u_{\max}$.

Combining the above analysis, we see that if in the form (2.3), any of the conditions in Theorem 5 is not satisfied, the system is not BIBOBS stable, completing the proof. □

The next lemma proves the sufficiency part of Theorem 5. The basic idea of the proof is to show that if all of the conditions in Theorem 5 are satisfied, the undetectable subsystem (i.e., the subsystem associated with state $x_{\bar{c}\bar{o}}$) cannot be triggered by the other components of the system and thus we can estimate an upper bound for the norm of the states (in the sense of Proposition 1).

Lemma 2. *The system (2.1) is BIBOBS stable if in the form (2.3), all of the conditions in Theorem 5 are satisfied.*

Proof. We first group all of the states except $x_{\bar{c}\bar{o}}$ in the form (2.3) into a vector $\tilde{x}$, i.e., $x_H = \begin{bmatrix}\tilde{x}^T & x_{\bar{c}\bar{o}}^T\end{bmatrix}^T$. Denote the components of A_H, B_H and C_H associated with $\tilde{x}$ by $\tilde{A}$, $\tilde{B}$ and $\tilde{C}$, respectively, i.e., $A_H = \begin{bmatrix}\tilde{A} & \tilde{A}_{12} \\ \tilde{A}_{21} & J_{\bar{c}\bar{o}}\end{bmatrix}$ where $\tilde{A}_{12} = \begin{bmatrix}0 & A_{24}^T & 0\end{bmatrix}^T$ and $\tilde{A}_{21} = \begin{bmatrix}0 & 0 & A_{43}\end{bmatrix}$, $B_H = \begin{bmatrix}\tilde{B}^T & 0\end{bmatrix}^T$, and $C_H = \begin{bmatrix}\tilde{C} & 0\end{bmatrix}$. Note that $y[k] = \tilde{C}\tilde{x}[k], \forall k$.

If condition (i) in Theorem 5 is satisfied (i.e., the matrix $A_{c\bar{o}}$ is stable), then $(\tilde{A}, \tilde{C})$ is detectable and there exists some matrix L such that the matrix $\tilde{A} + L\tilde{C}$ is stable. For the subsystem associated with the state $\tilde{x}$, using the same trick as in [67] for characterizing IOSS, we can construct the following observer

$$\hat{x}[k+1] = \tilde{A}\hat{x}[k] + \tilde{B}u[k] + \tilde{A}_{12}x_{\bar{c}\bar{o}}[k] + L(\tilde{C}\hat{x}[k] - y[k])$$

with the property that if $\hat{x}[0] = \tilde{x}[0]$, then $\hat{x}[k] = \tilde{x}[k], \forall k, u$. Thus, we know that

$$\tilde{x}[k] = (\tilde{A} + L\tilde{C})^k\tilde{x}[0] + \sum_{i=0}^{k-1}(\tilde{A} + L\tilde{C})^{k-1-i}\left(\tilde{B}u[i] + \tilde{A}_{12}x_{\bar{c}\bar{o}}[i] - Ly[i]\right).$$

Since $\tilde{A} + L\tilde{C}$ is stable, there exists some constant $K_1 > 0$ and $\tilde{\lambda} \in (0, 1)$ such that for any $k \in \mathbb{Z}_{\geq 0}$, $\|(\tilde{A} + L\tilde{C})^k\| \leq K_1 \tilde{\lambda}^k$ [48]. For example, we can choose $\tilde{\lambda}$ to be the spectral radius of the matrix $\tilde{A} + L\tilde{C}$ and $K_1 = \|V^{-1}\|\|V\|$ where V is the matrix consisting of the eigenvectors of $\tilde{A} + L\tilde{C}$. Then for any $k \in \mathbb{Z}_{\geq 0}$, we have

$$\|\tilde{x}[k]\| \leq K_1 \tilde{\lambda}^k \|\tilde{x}[0]\| + \frac{K_1}{1 - \tilde{\lambda}} \left(\|\tilde{B}\|\|u\|_{[0,k]} + \|\tilde{A}_{12}\|\|x_{\bar{c}\bar{o}}\|_{[0,k]} + \|L\|\|y\|_{[0,k]} \right). \tag{2.5}$$

Based on inequality (2.5), Definition 5 (the definition of BIBOBS stability) and the fact that $\|x[k]\| \leq \|H^{-1}\|\|x_H[k]\|, \forall k$, if we can show that there exists some constant $K_2 > 0$ such that $\|x_{\bar{c}\bar{o}}\|_{[0,k]} \leq K_2 \|x_H[0]\|, \forall k$, then the system is BIBOBS stable. Specifically, for any $k \in \mathbb{Z}_{\geq 0}$, we would then have

$$\begin{aligned}
\|x[k]\| &\leq \|H^{-1}\|\|x_H[k]\| \\
&\leq \|H^{-1}\| \left(\|x_{\bar{c}\bar{o}}[k]\| + \|\tilde{x}[k]\| \right) \\
&\leq \|H^{-1}\|\|x_{\bar{c}\bar{o}}[k]\| + K_1 \tilde{\lambda}^k \|H^{-1}\|\|\tilde{x}[0]\| \\
&\qquad + \frac{K_1 \|H^{-1}\|}{1 - \tilde{\lambda}} \left(\|\tilde{B}\|\|u\|_{[0,k]} + \|\tilde{A}_{12}\|\|x_{\bar{c}\bar{o}}\|_{[0,k]} + \|L\|\|y\|_{[0,k]} \right) \\
&\leq (K_2 + K_1 \tilde{\lambda}^k) \|H^{-1}\|\|x_H[0]\| \\
&\qquad + \frac{K_1 \|H^{-1}\|}{1 - \tilde{\lambda}} \left(\|\tilde{B}\|\|u\|_{[0,k]} + K_2 \|\tilde{A}_{12}\|\|x_H[0]\| + \|L\|\|y\|_{[0,k]} \right) \\
&\leq \left(K_2 + K_1 \tilde{\lambda}^k + \frac{K_1 K_2}{1 - \tilde{\lambda}} \|\tilde{A}_{12}\| \right) \|H^{-1}\|\|H\|\|x[0]\| \\
&\qquad + \frac{K_1 \|H^{-1}\|}{1 - \tilde{\lambda}} \left(\|\tilde{B}\|\|u\|_{[0,k]} + \|L\|\|y\|_{[0,k]} \right).
\end{aligned} \tag{2.6}$$

In the rest of this proof, we will show that if conditions (ii) and (iii) in Theorem 5 are satisfied, then there exists a constant K_2 such that $\|x_{\bar{c}\bar{o}}\|_{[0,k]} \leq K_2 \|x_H[0]\|, \forall k$; in other words, if those conditions are satisfied, the undetectable subsystem associated with state $x_{\bar{c}\bar{o}}$ cannot be triggered by the detectable subsystem. Since in the form (2.3) the state $x_{\bar{c}\bar{o}}$ is only influenced by $x_{\bar{c}o}$ (through A_{43}), we can focus on the following uncontrollable subsystem:

$$\begin{bmatrix} x_{\bar{c}o}[k+1] \\ x_{\bar{c}\bar{o}}[k+1] \end{bmatrix} = \begin{bmatrix} J_{\bar{c}o} & 0 \\ A_{43} & J_{\bar{c}\bar{o}} \end{bmatrix} \begin{bmatrix} x_{\bar{c}o}[k] \\ x_{\bar{c}\bar{o}}[k] \end{bmatrix}. \tag{2.7}$$

Note that since condition (ii) in Theorem 5 is satisfied, we know that $J_{\bar{c}\bar{o}}$ does not contain strictly unstable eigenvalues or marginally stable Jordan blocks with size bigger than 1. In order to characterize the influence of $x_{\bar{c}o}$ on $x_{\bar{c}\bar{o}}$, we further decompose the matrices $J_{\bar{c}o}$ and $J_{\bar{c}\bar{o}}$ as follows:

$$J_{\bar{c}o} = \begin{bmatrix} \bar{A}_m & & \\ & \bar{A}_s & \\ & & \bar{A}_u \end{bmatrix}, J_{\bar{c}\bar{o}} = \begin{bmatrix} A_d & \\ & A_s \end{bmatrix},$$

where $\bar{A}_m$ (resp. the diagonal matrix A_d) contains the marginally stable eigenvalues of $J_{\bar{c}o}$ (resp. $J_{\bar{c}\bar{o}}$), $\bar{A}_s$ (resp. A_s) contains the stable eigenvalues of $J_{\bar{c}o}$ (resp. $J_{\bar{c}\bar{o}}$), and $\bar{A}_u$ contains the strictly unstable eigenvalues of $J_{\bar{c}o}$.

Since in the form (2.3), $x_{\bar{c}o}$ is not influenced by the other components of the system, we can investigate the dynamics of $x_{\bar{c}o}$ separately. According to the decomposition of $J_{\bar{c}o}$, we decompose the state $x_{\bar{c}o}$ as

$$x_{\bar{c}o} = \begin{bmatrix} x^m_{\bar{c}o} \\ x^s_{\bar{c}o} \\ x^u_{\bar{c}o} \end{bmatrix}.$$

Note that since the state $x_{\bar{c}o}$ is observable and the input is bounded, $\|x_{\bar{c}o}[k]\|$ must be bounded whenever $\|y[k]\|$ is bounded.

Denote the set of (marginally stable) eigenvalues of $\bar{A}_m$ and the set of corresponding eigenvectors by $\{\bar{\lambda}_{m,i}\}$ and $\{\tilde{v}_{m,i,j}\}$, respectively; the set of eigenvectors corresponding to a certain eigenvalue $\bar{\lambda}_{m,l}$ is $\{\tilde{v}_{m,l,j}\}$. Since the matrix $\bar{A}_m$ is in Jordan form, without loss of generality, we assume that the set of eigenvectors $\{\tilde{v}_{m,i,j}\}$ is a set of indicator vectors.

In order for $\|x_{\bar{c}o}[k]\|$ to be bounded, $x^u_{\bar{c}o}[0]$ must be zero. Moreover, $x^m_{\bar{c}o}[0]$ must be a linear combination of the eigenvectors of $\bar{A}_m$ (i.e., $x^m_{\bar{c}o}[0] = \sum_{i,j} \alpha_{i,j}\tilde{v}_{m,i,j}$ where $\{\alpha_{i,j}\}$ is the set of weights); otherwise the matrix $\bar{A}_m$ must contain some Jordan block that has size bigger than 1 and the corresponding subvector of $x^m_{\bar{c}o}[0]$ is not an eigenvector of that block, and based on a similar analysis as in Lemma 1 for the case where $J_{\bar{c}\bar{o}}$ contains marginally stable blocks with size bigger than 1, we know that $\|x_{\bar{c}o}[k]\|$ will be unbounded.

Thus, to guarantee that $\|y[k]\|$ is bounded, the initial condition $x_{\bar{c}o}[0]$ must have the form

$$x_{\bar{c}o}[0] = \begin{bmatrix} x^m_{\bar{c}o}[0] \\ x^s_{\bar{c}o}[0] \\ x^u_{\bar{c}o}[0] \end{bmatrix} = \begin{bmatrix} \sum_{i,j} \alpha_{i,j}\tilde{v}_{m,i,j} \\ x^s_{\bar{c}o}[0] \\ 0 \end{bmatrix}.$$

Combining the above analysis, we know that

$$
\begin{aligned}
x_{\bar{c}o}[k] &= J_{\bar{c}o}^{k} x_{\bar{c}o}[0] \\
&= \begin{bmatrix} \bar{A}_m^k & & \\ & \bar{A}_s^k & \\ & & \bar{A}_u^k \end{bmatrix} \begin{bmatrix} x_{\bar{c}o}^{m}[0] \\ x_{\bar{c}o}^{s}[0] \\ x_{\bar{c}o}^{u}[0] \end{bmatrix} \\
&= \underbrace{\sum_{i,j} \alpha_{i,j} \bar{\lambda}_{m,i}^{k} \underbrace{\begin{bmatrix} \tilde{v}_{m,i,j} \\ 0 \\ 0 \end{bmatrix}}_{\bar{v}_{m,i,j}}}_{\hat{x}_{\bar{c}o}^{m}[k]} + \underbrace{\begin{bmatrix} 0 \\ \bar{A}_s^k x_{\bar{c}o}^{s}[0] \\ 0 \end{bmatrix}}_{\hat{x}_{\bar{c}o}^{s}[k]} . \qquad (2.8)
\end{aligned}
$$

Note that $\bar{v}_{m,i,j}$ is the j-th eigenvector of $J_{\bar{c}o}$ corresponding to the eigenvalue $\bar{\lambda}_{m,i}$. Since $\{\bar{v}_{m,i,j}\}$ is a set of indicator vectors and the set of eigenvalues $\{\bar{\lambda}_{m,i}\}$ are marginally stable, we have

$$
\|\hat{x}_{\bar{c}o}^{m}[k]\| = \|\hat{x}_{\bar{c}o}^{m}[0]\| \leq \|x_{\bar{c}o}[0]\| \leq \|x_H[0]\|, \forall k.
$$

Finally, we are ready to consider the dynamics of $x_{\bar{c}\bar{o}}$. By equation (2.7), equation (2.8) and the decomposition of $J_{\bar{c}\bar{o}}$, we know that the state $x_{\bar{c}\bar{o}}$ evolves as follows:

$$
\begin{aligned}
x_{\bar{c}\bar{o}}[k] &= J_{\bar{c}\bar{o}}^{k} x_{\bar{c}\bar{o}}[0] + \sum_{t=0}^{k-1} J_{\bar{c}\bar{o}}^{k-1-t} A_{43} x_{\bar{c}o}[t] \\
&= J_{\bar{c}\bar{o}}^{k} x_{\bar{c}\bar{o}}[0] + \underbrace{\sum_{t=0}^{k-1} J_{\bar{c}\bar{o}}^{k-1-t} A_{43} \hat{x}_{\bar{c}o}^{s}[t]}_{P_m[k]} + \begin{bmatrix} P_d[k] \\ P_s[k] \end{bmatrix},
\end{aligned}
$$

where

$$
P_d[k] = \sum_{t=0}^{k-1} A_d^{k-1-t} A_{43}(A_d) \hat{x}_{\bar{c}o}^{m}[t]
$$

and

$$
P_s[k] = \sum_{t=0}^{k-1} A_s^{k-1-t} A_{43}(A_s) \hat{x}_{\bar{c}o}^{m}[t].
$$

Note that $A_{43}(A_d)$ and $A_{43}(A_s)$ denote the matrices consisting of the rows of A_{43} corresponding to A_d and A_s, respectively. Due to the structure of $J_{\bar{c}\bar{o}}$ and the stability of $\bar{A}_s$, there exist constants $K_2^1, K_2^2 > 0$ such that

$$
\|J_{\bar{c}\bar{o}}^{k} x_{\bar{c}\bar{o}}[0]\| \leq K_2^1 \|x_H[0]\|, \forall k,
$$

and

$$\|P_m[k]\| \leq K_2^2 \|x_H[0]\|, \forall k.$$

Moreover, we can regard P_s as the state of the system $(A_s, A_{43}(A_s))$ with zero initial condition and $\hat{x}^m_{\bar{c}o}$ being the input. Due to the stability of A_s and the fact that $\|\hat{x}^m_{\bar{c}o}[k]\| \leq \|x_H[0]\|, \forall k$, the system $(A_s, A_{43}(A_s))$ is input-to-state stable and there exists some constant $K_2^3 > 0$ such that

$$\|P_s[k]\| \leq K_2^3 \|x_H[0]\|, \forall k.$$

Now we just need to show that $\|P_d[k]\|$ is always bounded. Recall that A_d is a diagonal matrix with marginally stable eigenvalues. Denote the subvector of $P_d[k]$ corresponding to the l-th eigenvalue λ_l of A_d by $P_{d,l}[k]$. Then we have

$$\begin{aligned} P_{d,l}[k] &= \sum_{t=0}^{k-1} \lambda_l^{k-1-t} A_{43}(\lambda_l) \sum_{i,j} \alpha_{i,j} \bar{\lambda}_{m,i}^t \bar{v}_{m,i,j} \\ &= \lambda_l^{k-1} A_{43}(\lambda_l) \sum_{i,j} \alpha_{i,j} \left(\sum_{t=0}^{k-1} (\bar{\lambda}_{m,i} \lambda_l^{-1})^t \right) \bar{v}_{m,i,j}. \end{aligned}$$

Denote $\bar{\lambda}_{m,i}\lambda_l^{-1} = e^{\mathbf{i}\theta_{l,i}}$, where $\mathbf{i}$ is the imaginary unit, and let Θ_l be the set of indices of the eigenvalues of $\bar{A}_m$ which are equal to λ_l, i.e., $\Theta_l = \{i | \theta_{l,i} = 0\}$. Since condition (iii) in Theorem 5 is satisfied, if $\lambda_l = \bar{\lambda}_{m,i}$ (i.e., $\theta_{l,i} = 0$), then $\bar{v}_{m,i,j} \in \mathcal{N}(A_{43}(\lambda_l)), \forall j$, and thus $P_{d,l}[k]$ does not depend on the elements in Θ_l. Then we have

$$P_{d,l}[k] = \lambda_l^{k-1} A_{43}(\lambda_l) \sum_{i \notin \Theta_l, j} \alpha_{i,j} \frac{1 - e^{\mathbf{i}\theta_{l,i} k}}{1 - e^{\mathbf{i}\theta_{l,i}}} \bar{v}_{m,i,j}.$$

Since $\{\bar{v}_{m,i,j}\}$ is a set of indicator vectors, for any l, there exists some constant $K_{d,l} > 0$ such that

$$\|P_{d,l}[k]\| \leq K_{d,l} \|\hat{x}^m_{\bar{c}o}[0]\| \leq K_{d,l} \|x_H[0]\|, \forall k.$$

Thus, due to the fact that $\|P_d[k]\| \leq \sum_l \|P_{d,l}[k]\|, \forall k$, the constant $K_2^4 = \sum_l K_{d,l}$ guarantees that $\|P_d[k]\| \leq K_2^4 \|x_H[0]\|, \forall k$.

Combining all of the above analysis and the fact that

$$\|x_{\bar{c}\bar{o}}[k]\| \leq \|J^k_{\bar{c}\bar{o}} x_{\bar{c}\bar{o}}[0]\| + \|P_m[k]\| + \|P_s[k]\| + \|P_d[k]\|, \forall k,$$

the constant $K_2 = \sum_{i=1}^4 K_2^i$ guarantees that $\|x_{\bar{c}\bar{o}}\|_{[0,k]} \leq K_2 \|x_H[0]\|, \forall k$. Thus, the system is BIBOBS stable if all of the conditions in Theorem 5 are satisfied. □

Remark 2. *If the system satisfies the conditions for BIBOBS stability, a construction of the unknown input norm-observer follows by inequality (2.6). Specifically, in Proposition 1, we can choose the functions $\gamma_1, \gamma_2, \gamma_3 \in \mathcal{K}_\infty$ as follows*

$$\gamma_1(x_{max}) = \left(K_2 + K_1 \tilde{\lambda}^k + \frac{K_1 K_2}{1-\tilde{\lambda}} \|\tilde{A}_{12}\| \right) \|H^{-1}\| \|H\| x_{max}, \tag{2.9}$$

$$\gamma_2(u_{max}) = \frac{K_1}{1-\tilde{\lambda}} \|H^{-1}\| \|\tilde{B}\| u_{max}, \tag{2.10}$$

$$\gamma_3(\|y\|_{[0,k]}) = \frac{K_1}{1-\tilde{\lambda}} \|H^{-1}\| \|L\| \|y\|_{[0,k]}. \tag{2.11}$$

2.5 A Discussion of BIBOBS Stability

In this section, we discuss the relationships between BIBOBS stability and other classical system properties.

2.5.1 Related Linear System Stability Notions

We first consider BIBO stability, defined as follows.

Definition 8 (BIBO Stability)**.** *The system (2.1) with $x[0] = 0$ is said to be **BIBO stable** if every bounded input sequence excites a bounded output sequence, i.e., $\forall M_1 \in \mathbb{R}_{\geq 0}$ such that $\|u[k]\| \leq M_1, \forall k$, $\Rightarrow \exists M_2 \in \mathbb{R}_{\geq 0}$ such that $\|y[k]\| \leq M_2, \forall k$. Equivalently, the system is BIBO stable if there exists a function $\alpha \in \mathcal{K}_\infty$ such that if $x[0] = 0$, $\|y[k]\| \leq \alpha(\|u\|_{[0,k]}), \forall k, u$.*

Since the state does not play a role in the definition of BIBO stability, BIBO stability does not imply BIBOBS stability. In the converse direction, a BIBOBS stable system does not have to be BIBO stable since the output of a BIBOBS stable system can be unbounded with bounded input; a system is still BIBOBS stable as long as we can observe that the states become unbounded.

To see the difference between BIBO stability and BIBOBS stability, consider the following two scalar systems:

$$\begin{aligned} x[k+1] &= x[k] + u[k] \\ y[k] &= 0, \end{aligned} \tag{2.12}$$

$$\begin{aligned} x[k+1] &= x[k] + u[k] \\ y[k] &= x[k]. \end{aligned} \tag{2.13}$$

System (2.12) is BIBO stable (since the output is always zero) but not BIBOBS stable (since we can let the input be any constant at each time step and the state will be unbounded while the output is zero). System (2.13) is BIBOBS stable (since the output is the state) but not BIBO stable (since a bounded input can produce an unbounded output). Thus, we have the following result.

Proposition 2. *For system (2.1), BIBO stability does not imply BIBOBS stability, and vice versa.*

Next we consider the notion of BIBS stability, defined as follows.

Definition 9 (BIBS Stability)**.** *The system (2.1) is said to be* ***BIBS stable*** *if every bounded input sequence excites a bounded state sequence, i.e.,* $\forall M_1, M_2 \in \mathbb{R}_{\geq 0}$ *such that* $\|x[0]\| \leq M_1$ *and* $\|u[k]\| \leq M_2, \forall k,$ $\Rightarrow$ $\exists M_3 \in \mathbb{R}_{\geq 0}$ *such that* $\|x[k]\| \leq M_3, \forall k$*. Equivalently, the system is BIBS stable if there exist functions* $\alpha_1, \alpha_2 \in \mathcal{K}_\infty$ *such that* $\|x[k]\| \leq \alpha_1(\|x[0]\|) + \alpha_2(\|u\|_{[0,k]}), \forall k, x[0], u$*.*

The relationship between BIBS stability and BIBOBS stability is as follows.

Proposition 3. *For system (2.1), BIBS stability implies BIBOBS stability, but not vice versa.*

Proof. One direction is easy to show: since bounded input always results in bounded state for a BIBS stable system and the state-output mapping is linear, the system must be BIBOBS stable. For the other direction, consider again the system (2.13); the system is BIBOBS stable but not BIBS stable. □

To summarize these relationships, we have the following result, which shows that the intersection of BIBO stable and BIBOBS stable linear systems is exactly the set of BIBS stable linear systems. See Figure 2.2 for the relationships between BIBO stability, BIBS stability and BIBOBS stability.

Proposition 4. *The system (2.1) is BIBS stable if and only if it is both BIBO stable and BIBOBS stable.*

Proof. Note that for linear systems, BIBS stability also implies BIBO stability. Thus, if a linear system is BIBS stable, then it must be both BIBO stable and BIBOBS stable.

For the other direction, if a system is both BIBO stable and BIBOBS stable, then every bounded input results in a bounded output (due to BIBO stability) and thus results in a bounded state (due to BIBOBS stability). This implies that the system is BIBS stable. □

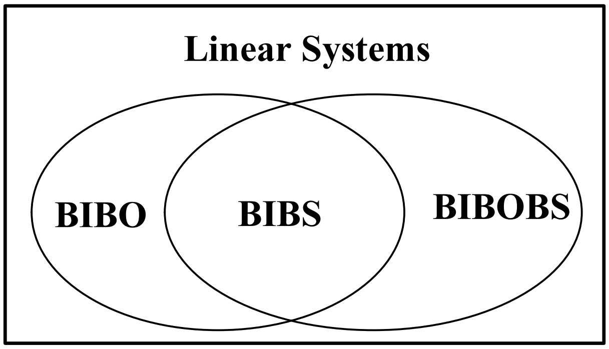


Figure 2.2: Venn diagram for the classes of linear systems that are BIBO stable, BIBS stable and BIBOBS stable.

Table 2.1: Comparison of different notions of stability.

BIBS Stability: $\exists \alpha_1, \alpha_2 \in \mathcal{K}_\infty$ such that $\|x[k]\| \leq \alpha_1(\|x[0]\|) + \alpha_2(\|u\|_{[0,k]}), \forall k, x[0], u$	ISS: $\exists \alpha \in \mathcal{K}_\infty, \beta \in \mathcal{KL}$ such that $\|x[k]\| \leq \beta(\|x[0]\|, k) + \alpha(\|u\|_{[0,k]}), \forall k, x[0], u$
BIBO Stability: $\exists \alpha \in \mathcal{K}_\infty$ such that if $x[0] = 0$, $\|y[k]\| \leq \alpha(\|u\|_{[0,k]}), \forall k, u$	IOS: $\exists \alpha \in \mathcal{K}_\infty, \beta \in \mathcal{KL}$ such that $\|y[k]\| \leq \beta(\|x[0]\|, k) + \alpha(\|u\|_{[0,k]}), \forall k, x[0], u$
BIBOBS Stability: $\exists \alpha_1, \alpha_2, \alpha_3 \in \mathcal{K}_\infty$ such that $\|x[k]\| \leq \alpha_1(\|x[0]\|) + \alpha_2(\|u\|_{[0,k]}) + \alpha_3(\|y\|_{[0,k]}), \forall k, x[0], u$	IOSS: $\exists \alpha_1, \alpha_2 \in \mathcal{K}_\infty, \beta \in \mathcal{KL}$ such that $\|x[k]\| \leq \beta(\|x[0]\|, k) + \alpha_1(\|u\|_{[0,k]}) + \alpha_2(\|y\|_{[0,k]}), \forall k, x[0], u$

2.5.2 Interpretation in Nonlinear Setting

Although we focus on linear systems in this chapter, the concept of BIBOBS stability can also be applied to nonlinear systems, and further insights can be obtained by comparing it with a set of properties that are normally defined for such systems.

In Table 2.1, we summarize the stability notions of interest; for a comprehensive discussion of these properties, we refer to [128]. Note that in Table 2.1, *ISS* represents *input-to-state stability* and *IOS* represents *input-output stability*. Note that in order to provide a uniform comparison, we use the condition given in Proposition 1 for BIBOBS stability.

From Table 2.1, we can see that the concept of BIBOBS stability fits naturally in the landscape of stability theory. Specifically, we can categorize the notions in Table 2.1 by two criteria. The first criterion is whether the output is taken into account: the notions in the first two rows do not consider output. The second criterion is whether the definition requires the influence of the initial condition to decay asymptotically: the notions in the second column all have this requirement.

Among these notions, BIBOBS stability is very similar to IOSS with the only difference being in the term related to $x[0]$. It is easy to see that IOSS is stronger than BIBOBS stability: for a system to be BIBOBS stable, the impact of initial conditions does not have to asymptotically decay over time.

Along this line, it is interesting to compare another property which imposes a further constraint on the term related to $x[0]$. In [46], the authors proposed a notion of $\mathcal{KL}$ norm-observability for nonlinear systems; roughly speaking, a system is $\mathcal{KL}$ norm-observable if it is IOSS and in the definition of IOSS, the function β can be chosen to decay arbitrarily fast in the second argument (see [46] for a formal definition). Since $\mathcal{KL}$ norm-observability and IOSS are equivalent to observability and detectability for linear systems, respectively, both of them are stronger than BIBOBS stability.

2.6 Illustrative Examples

In this section, we use several examples to illustrate the results in Lemma 1 and Lemma 2. Note that since the impact of condition (ii) in Theorem 5 is clear in the proof of Lemma 1, we omit the corresponding example.

2.6.1 Illustration of Condition (i) in Theorem 5

We first illustrate the case where $A_{c\bar{o}}$ contains marginally stable eigenvalues, i.e., when condition (i) in Theorem 5 is not satisfied.

Consider the system

$$\begin{aligned} A &= \begin{bmatrix} 2 & 0 \\ 1 & 1 \end{bmatrix}, \quad B = \begin{bmatrix} 1 & 1 \\ 1 & 0 \end{bmatrix}, \\ C &= \begin{bmatrix} 1 & 0 \end{bmatrix}, \quad D = 0. \end{aligned}$$

Note that the controllable but unobservable subsystem $(1, \begin{bmatrix} 1 & 0 \end{bmatrix}, 0)$ is marginally stable. Thus, by Lemma 1, we know that the system is not BIBOBS stable. Specifically, choose $x[0] = \begin{bmatrix} 0 \\ 1 \end{bmatrix}$, and

$$u[k] = \begin{cases} \begin{bmatrix} \frac{6}{11} \\ -\frac{4}{11} \end{bmatrix}, & \text{when } k \text{ is even} \\ \\ \begin{bmatrix} \frac{3}{11} \\ -\frac{7}{11} \end{bmatrix}, & \text{when } k \text{ is odd.} \end{cases}$$

Then we have

$$x[k] = \begin{cases} \begin{bmatrix} 0 \\ 1 + \frac{k}{2} \end{bmatrix}, & \text{when } k \text{ is even} \\ \\ \begin{bmatrix} \frac{2}{11} \\ \frac{17}{11} + \frac{k-1}{2} \end{bmatrix}, & \text{when } k \text{ is odd} \end{cases}$$

and

$$y[k] = \begin{cases} 0, & \text{when } k \text{ is even} \\ \frac{2}{11}, & \text{when } k \text{ is odd.} \end{cases}$$

Thus, the states become unbounded while the outputs are always bounded and the system is not BIBOBS stable.

2.6.2 Illustration of Condition (iii) in Theorem 5

Next we illustrate the case where $x_{\bar{c}\bar{o}}$ can be triggered by $x_{\bar{c}o}$, i.e., when condition (iii) in Theorem 5 is not satisfied.

Consider the system

$$A = \left[\begin{array}{cc:c} 1 & 1 & 0 \\ 0 & 1 & 0 \\ \hdashline 1 & 0 & 1 \end{array}\right] = \left[\begin{array}{c:c} J_{\bar{c}o} & 0 \\ \hdashline A_{43} & J_{\bar{c}\bar{o}} \end{array}\right], \quad B = \left[\begin{array}{c} 0 \\ \hdashline 0 \end{array}\right],$$

$$C = \left[\begin{array}{cc:c} 1 & 0 & 0 \end{array}\right] = \left[\begin{array}{c:c} C_{\bar{c}o} & 0 \end{array}\right], \quad D = 0.$$

Note that $J_{\bar{c}o}$ and $J_{\bar{c}\bar{o}}$ have the same marginally stable eigenvalue 1. Moreover, the eigenvector $\bar{v}_m = \begin{bmatrix} 1 \\ 0 \end{bmatrix}$ of $J_{\bar{c}o}$ corresponding to eigenvalue 1 is not in the null space of A_{43}. Thus, by Lemma 1, we know that the system is not BIBOBS stable. Specifically, by choosing $x_{\bar{c}o}[0] = \bar{v}_m$ and $x_{\bar{c}\bar{o}}[0] = 0$, we know that $\|x[k]\| = \left\| \begin{bmatrix} 1 & 0 & k \end{bmatrix}^T \right\|$ becomes unbounded while $\|y[k]\| = \left\| \begin{bmatrix} 1 & 0 & 0 \end{bmatrix}^T \right\|$ is always bounded. Note that if we replace A_{43} with $\begin{bmatrix} 0 & 1 \end{bmatrix}$, then $\bar{v}_m \in \mathcal{N}(A_{43})$ and thus by Lemma 2, the system is BIBOBS stable.

2.6.3 Construction of An Unknown Input Norm Observer

Finally, we consider a system that is BIBOBS stable and provide an unknown input norm observer for this system.

Consider the system

$$A = \left[\begin{array}{cc:c} 1.1 & & \\ & 1 & \\ \hdashline & & 1 \end{array}\right] = \left[\begin{array}{c:c} \tilde{A} & \tilde{A}_{12} \\ \hdashline \tilde{A}_{21} & J_{\bar{c}\bar{o}} \end{array}\right], \quad B = \left[\begin{array}{cc} 1 & 1 \\ 1 & 0 \\ \hdashline 0 & 0 \end{array}\right] = \left[\begin{array}{c} \tilde{B} \\ \hdashline 0 \end{array}\right],$$

$$C = \left[\begin{array}{cc:c} 1 & 1 & 0 \\ 0 & 1 & 0 \end{array}\right] = \left[\begin{array}{c:c} \tilde{C} & 0 \end{array}\right], \quad D = 0.$$

Since $(\tilde{A}, \tilde{C})$ is observable and $\tilde{A}_{21} = 0$, by Lemma 2, we know that the system is BIBOBS stable. Then we can choose a matrix L such that the matrix $\tilde{A} + L\tilde{C}$ is stable. For example, we choose

$$L = \begin{bmatrix} -1 & 1 \\ 0 & -1.1 \end{bmatrix}$$

such that $\mathrm{eig}(\tilde{A} + L\tilde{C}) = \{0.1, -0.1\}$. Then we have the bound

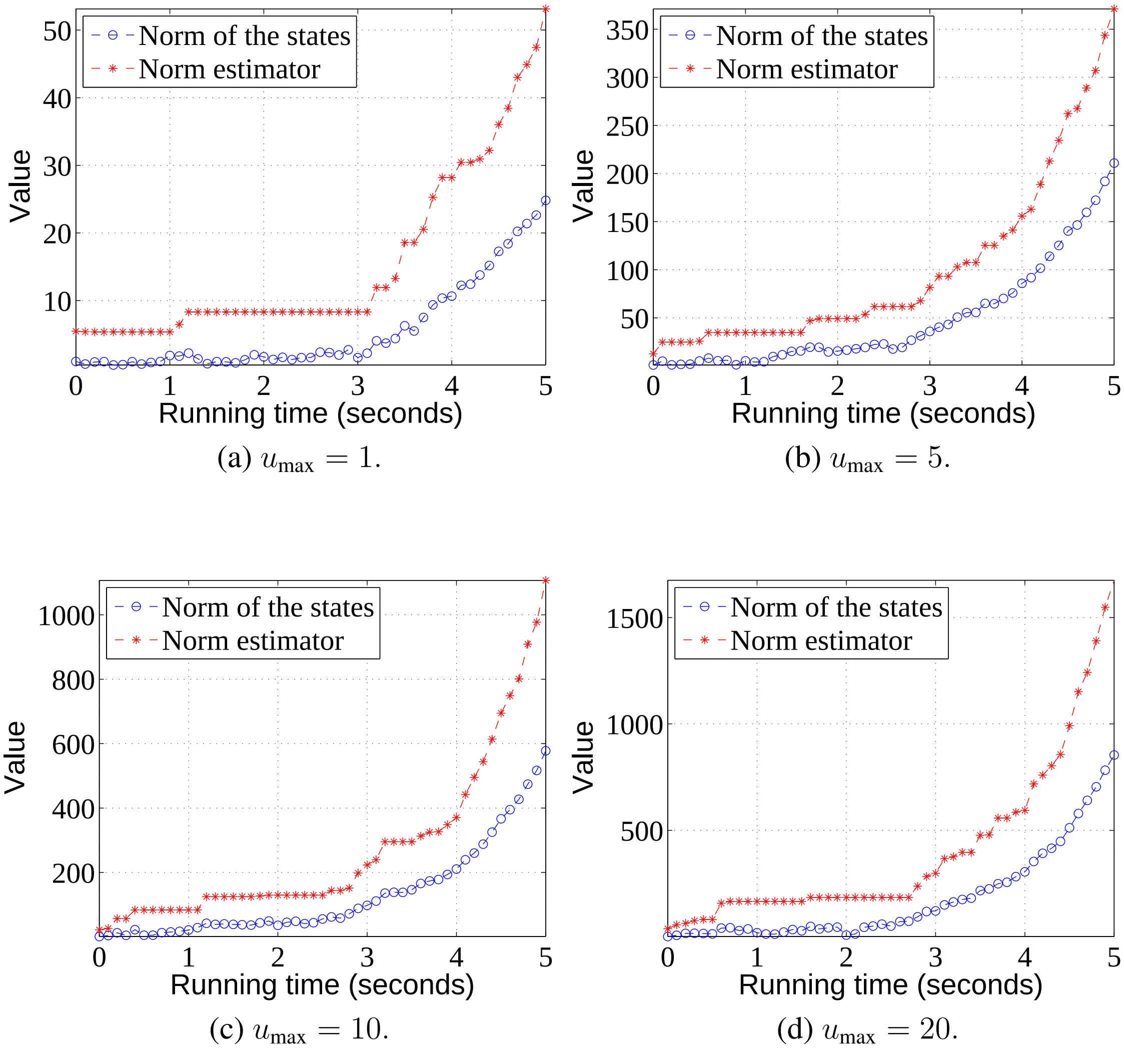
$$\|(\tilde{A} + L\tilde{C})^k\| \leq 0.1^k,$$

i.e., $K_1 = 1$ and $\tilde{\lambda} = 0.1$. Furthermore, we can choose $K_2 = 1$ since $\|x_{\bar{c}\bar{o}}\| \leq K_2\|x[0]\| = \|x[0]\|$. Thus, we have $\frac{K_1\|L\|}{1-\tilde{\lambda}} < 1.9$, $\frac{K_1\|\tilde{B}\|}{1-\tilde{\lambda}} < 1.8$, $H = I$, and $\|\tilde{A}_{12}\| = 0$. By using the functions in Remark 2, we can get an unknown input norm-observer $\hat{x}$ satisfying $\|x[k]\| \leq \hat{x}[k], \forall k, x[0], u$, as follows:

$$\hat{x}[k] = (1 + 0.1^k)x_{\max} + 1.8u_{\max} + 1.9\|y\|_{[0,k]}. \tag{2.14}$$

Now we provide simulation results to verify the performance of the above norm-observer. In the simulation, the upper bound for the norm of the initial condition is fixed to be 1 (i.e., $x_{\max} = 1$), and we compare the performance of the norm-observer with different upper bounds of the unknown input. Specifically, $u_{\max}$ is taken to be $1, 5, 10$ and 20, respectively. Note that the initial condition and unknown input are generated randomly within their upper bounds. The results are in Figure 2.3.

From Figure 2.3, we can see that the norm-observer (2.14) provides an upper bound on the norm of the states in all cases. Moreover, it is not surprising to observe that the scale of the state norm increases with the upper bound $u_{\max}$ of the unknown input.



(a) $u_{\max} = 1$.

(b) $u_{\max} = 5$.

(c) $u_{\max} = 10$.

(d) $u_{\max} = 20$.

Figure 2.3: Performance of the unknown input norm observer (2.14) with $x_{\max} = 1$ and different values of $u_{\max}$.

2.7 Summary

In this chapter, we studied the norm estimation problem for linear dynamical systems with unknown inputs. In order to characterize the conditions under which an unknown input norm-observer exists, we proposed the concept of BIBOBS stability and provide necessary and sufficient conditions for linear systems to be BIBOBS stable. When the system is not BIBOBS stable, we showed that the attacker is able to use bounded inputs to drive the states unbounded while keeping the output bounded. Specifically, other than through unobservable strictly unstable eigenvalues, such worst-case attacks can be achieved by manipulating the set of controllable but unobservable marginally stable eigenvalues or triggering the set of uncontrollable and unobservable marginally stable eigenvalues with carefully chosen initial conditions. On the other hand, when the system is BIBOBS stable, we provided a construction method for the unknown input norm-observer. Besides the application to norm estimation, we see that the concept of BIBOBS stability naturally supplements the other classical stability notions.

Chapter 3

State Estimation for Positive Linear Dynamical Systems

3.1 Introduction

The theory of positive systems has received considerable attention in different communities (e.g., economic modeling, behavioral science, and control), and builds upon concepts from nonnegative matrix theory [30, 85]. A positive system is a system in which the states are always positive (or at least nonnegative) and the positivity property naturally arises from the nature of the phenomenon under consideration [102, 124]. For example, in the application of temperature estimation for multi-processor systems, the positivity of the system comes from the physical nature of the thermal dynamics [123]. Other examples include transportation networks, industrial processes involving chemical reactors, compartmental systems (frequently used in biology and mathematics), stochastic models, and many other models for economic and social systems [30, 39, 47].

Due to the prevalence of positive systems, it is of interest to explore the role of positivity in state estimation. As we argued in the last chapter, when the system is not strongly observable, there exist some initial condition and a sequence of inputs such that the output is zero for all time, but the state is not. In this case, (finite-time) state estimation despite the unknown inputs is not possible for such initial condition and inputs. Suppose, however, that the system is known to be positive, and that the inputs are constrained to be nonnegative – does this additional knowledge assist in state estimation? The answer is no, as we will show in this chapter.

We also consider the situation when we require the observer to return only positive estimates [41]. For positive systems where state estimation with unknown inputs is possible, we extend the work in [113] to give a linear programming-based design method for positive unknown input observers.

The rest of this chapter is organized as follows. In Section 3.2, we provide some preliminary on positive systems theory. In Section 3.3, we extend the concept of strong observability to positive systems and show that the property of positivity is not helpful in state estimation. In Section 3.4, we consider the problem of constructing positive observers for positive systems. Some concluding remarks are given in Section 3.5.

3.2 Background: Positive Systems

In this section, we briefly review the concept of positive systems and illustrate the usefulness of the property of positivity. Consider the following discrete-time linear system

$$\begin{aligned} x[k+1] &= Ax[k] + Bu[k] \\ y[k] &= Cx[k] + Du[k], \end{aligned} \tag{3.1}$$

where $x \in \mathbb{R}^n$ is the state, $y \in \mathbb{R}^p$ is the output, and $u \in \mathbb{R}^m$ is the unknown input. We start with a definition of positive systems.

Definition 10 (Positive Systems)**.** *Given any nonnegative initial condition $x[0]$ and any sequence of nonnegative inputs $\{u[k]\}$ (i.e., $u[k] \geqq 0, \forall k$), the linear system (3.1) is said to be **positive** if the corresponding states and outputs are always nonnegative (i.e., $x[k], y[k] \geqq 0, \forall k$).*

In words, the system (3.1) is positive if nonnegative initial condition and inputs always result in nonnegative states and outputs. The following result specifies whether a discrete-time state-space model represents a positive system.

Theorem 6 ([30])**.** *The discrete-time linear system (3.1) is positive if and only if the system matrices (A, B, C, D) are all nonnegative.*

The study of positive systems builds upon the elegant theory of nonnegative matrices. One of the fundamental results in this theory is the well known Perron-Frobenius theorem, stated as follows [85].

Theorem 7 (Perron-Frobenius Theorem)**.** *For any positive matrix $A > 0$, the following statements hold:*

(1) there exists a positive real number λ_0 such that λ_0 is an eigenvalue of A and any other eigenvalue λ of A is strictly smaller than λ_0 (i.e., $|\lambda| < \lambda_0$);

(2) λ_0 has geometric and algebraic multiplicity 1;

(3) there exists an eigenvector v_0 of A with eigenvalue λ_0 such that all components of v_0 are positive (i.e., $v_0 > 0$).

In terms of system theory, λ_0 is the dominant eigenvalue (also called the Perron root or the Perron-Frobenius eigenvalue) and v_0 provides information of the long-term behavior of the homogeneous positive system $x[k+1] = Ax[k]$: the state vector aligns itself with the dominant eigenvector v_0. These results can be applied to various applications, e.g., stability analysis of distributed consensus algorithms [98].

Another remarkable result due to the positivity of the system is the equivalence between the stability of the system and the existence of its equilibrium. Specifically, consider the following nonhomogeneous system:

$$x[k+1] = Ax[k] + b$$

with $A \geq 0$ and $b > 0$. Then the system is stable if and only if there exists an $\bar{x} \geq 0$ such that $\bar{x} = A\bar{x} + b$, i.e., $\bar{x}$ is an equilibrium of the system [85].

A nonlinear counterpart to positive systems are so-called *monotone systems*, where the trajectories preserve a partial ordering on the states. The class of monotone systems has drawn much attention recently due to its applications to mathematical biology [3, 114].

3.3 Strong Observability of Positive Systems

In this section, we study the role of positivity in state estimation with unknown inputs. First we consider the case where the inputs are known. Analogous to the theory of general linear systems, in [41], the authors define observability for positive systems as follows.

Definition 11 (Observability of Positive Systems)**.** *A positive linear system (3.1) is said to be **observable** if for any nonnegative initial condition $x[0]$ and any **known** sequence of nonnegative inputs $\{u[k]\}$, there exists some $L \in \mathbb{Z}_{>0}$ such that $x[0]$ can be recovered from $\{y[k]\}_{k \leq L}$.*

Theorem 8 ([41])**.** *The positive linear system (3.1) is observable if and only if $\operatorname{rank}(\mathcal{O}_n) = n$, where $\mathcal{O}_n$ is the observability matrix of system (3.1)*

The above result shows that observability is not made easier by assuming positivity; if a given system is not observable, restricting the states and inputs to be positive does not provide more information about the unobserved states.

Then a natural question to ask is whether the property of positivity is helpful in conducting state estimation in the presence of unknown inputs. We extend the concept of strong observability for positive systems.

Definition 12 (Strong Observability of Positive Systems)**.** *A positive linear system (3.1) is said to be **strongly observable** if for any nonnegative initial state $x[0]$ and any **unknown** sequence of nonnegative inputs $\{u[k]\}$, there exists some $L \in \mathbb{Z}_{>0}$ such that $x[0]$ can be recovered from $\{y[k]\}_{k \leq L}$.*

We will use the following lemma in our characterization of strong observability of positive systems. We omit the proof as it depends only on the rank of the system matrices, and thus is identical to the proof for general systems [126]. Recall that $\mathcal{O}_L$ and $\mathcal{J}_L$ are the observability matrix and invertibility matrix of system (3.1) with delay L; see Section 2.2.1 for the detailed forms of these matrices.

Lemma 3. *Given the positive linear system (3.1), let $G(L) = \operatorname{rank}(\begin{bmatrix}\mathcal{O}_L & \mathcal{J}_L\end{bmatrix}) - \operatorname{rank}(\mathcal{J}_L)$ be a function of $L \in \mathbb{Z}_{\geq 0}$. Then $G(L)$ has the following properties:*

- *$G(L)$ is a nondecreasing function and $G(L) \leq n$;*
- *If $G(L') = G(L'+1)$, then $G(L) = G(L'), \forall L \geq L'$;*
- *$G(L) = G(n), \forall L \geq n$.*

The following result tells us that the condition for strong observability of positive systems is the same as the one for general systems. Thus, knowing the system to be positive does not help in state estimation with arbitrary positive inputs.

Proposition 5. *The positive linear system (3.1) is strongly observable if and only if*

$$\operatorname{rank}(\begin{bmatrix}\mathcal{O}_n & \mathcal{J}_n\end{bmatrix}) = n + \operatorname{rank}(\mathcal{J}_n).$$

Proof. (*Sufficiency*) Recall from the discussion in Section 2.2.1 that if the condition in the proposition holds, the system is strongly observable even for arbitrary (potentially negative) states and inputs. Thus, the condition in the proposition is also sufficient to recover the initial states under positivity constraints.

(*Necessity*)[1] Assume that $\mathrm{rank}(\begin{bmatrix}\mathcal{O}_n & \mathcal{J}_n\end{bmatrix}) < n + \mathrm{rank}(\mathcal{J}_n)$. Then either the system is not observable (i.e., $\mathrm{rank}(\mathcal{O}_n) < n$) or some column of $\mathcal{O}_n$ can be expressed as a linear combination of other columns of $\mathcal{J}_n$ and $\mathcal{O}_n$. In the former case, there does not exist a observer even if the inputs are known (by Theorem 8).

In the latter case, there exist nonzero vectors v_1 and v_2 such that $\mathcal{O}_n v_1 + \mathcal{J}_n v_2 = 0$. Let $x_1 = \max\{v_1, 0\}$, $x_2 = \max\{-v_1, 0\}$, $u_1 = \max\{v_2, 0\}$ and $u_2 = \max\{-v_2, 0\}$. Note that for a vector v, $\max\{v, 0\} = v$ if $v \geq 0$ and $\max\{v, 0\} = 0$ otherwise. Then we have

$$\begin{aligned}
&\mathcal{O}_n(x_1 - x_2) + \mathcal{J}_n(u_1 - u_2) = 0,\\
&x_1 \neq x_2, u_1 \neq u_2,\\
&x_1, x_2, u_1, u_2 \geq 0.
\end{aligned}$$

In other words, we know that $\mathcal{O}_n x_1 + \mathcal{J}_n u_1 = \mathcal{O}_n x_2 + \mathcal{J}_n u_2$ and the nonnegative initial state x_1 with input sequence u_1 is not distinguishable from the nonnegative initial state x_2 with input sequence u_2 over $n+1$ time-steps. By Lemma 3, we know that the above analysis holds for any number of time-steps larger than n and thus the system is not strongly observable.

Combining the above analysis, we know that the system is not strongly observable if the rank condition in the proposition is not satisfied. □

Remark 3. *Note that even if we require all the system matrices to be strictly positive (as opposed to just nonnegative), the above results still hold with an identical proof.*

3.4 Positive Observers with Unknown Inputs

Theorem 8 and Proposition 5 tell us that the positivity of the original system is not helpful in relaxing the algebraic condition for state estimation with positive inputs. If the system is strongly observable, however, one can construct unknown input observers to recover the states. Note that since the states of the system are always positive, negative estimates may not be useful, and in some applications (e.g., observers for compartmental systems [41]), the positivity of the observers themselves is desirable. Thus, we now consider the design of positive observers.

[1] This proof generalizes the proof of Theorem 4.7 in [41] (stated as Theorem 8 in this section) to the case where there are unknown inputs.

Definition 13 (Positive Observers)**.** *Given positive system (3.1) and any nonnegative initial estimate $\hat{x}[0]$, an observer is said to be **positive** if the corresponding estimates are always nonnegative (i.e., $\hat{x}[k] \geqq 0, \forall k$).*

The positive observer problem has been studied extensively in the literature [4, 41, 92, 113, 125, 143]. Necessary and sufficient conditions for the existence of a (Luenberger-type) positive observer (for systems with known inputs) have been given in [113]. Note that the existence condition for positive observers is different from that for general ones, i.e., when the (positive) system is observable, there may still not exist positive observers. For example, consider the example from [41] where the system matrices are $A = \begin{bmatrix} 1 & 1 \\ 2 & 3 \end{bmatrix}, B = 0, C = \begin{bmatrix} 1 & 0 \end{bmatrix}, D = 0$. One can check (by using Theorem 8) that the system is observable but there does not exist a (Luenberger-type) positive observer for the system; see [41] for details.

While these existing works study the case where there are no inputs (or assume that the inputs are known), there are few works considering the positive observer problem with unknown inputs. In [121], the authors designed positive unknown input observers for continuous-time positive linear systems using current system output (i.e., no delay in the observer); the observer parameters are obtained by solving a certain linear matrix inequality (LMI). To the best of our knowledge, the *delayed* positive unknown input observer problem has not been studied.

Here we look at an unknown input observer of the form

$$\hat{x}[k+1] = E\hat{x}[k] + Fy[k : k+L], \tag{3.2}$$

with state estimate $\hat{x} \in \mathbb{R}^n$, delay $L \in \mathbb{Z}_{\geq 0}$ and system matrices (E, F) of appropriate dimensions. Recall that $y[k : k+L]$ is the output of the system up to time-step $k+L$, and thus (3.2) represents a *delayed* observer.

It is known that for systems with unknown inputs, a delayed observer of the above form is typically necessary (and sufficient if the system is strongly detectable) [133]. We show this to be the case for positive systems as well.

Lemma 4. *The observer (3.2) is positive if and only if the matrices E, $F\mathcal{O}_L$ and $F\mathcal{J}_L$ are all nonnegative.*

Proof. (*Sufficiency*) Note that

$$\begin{aligned} \hat{x}[k+1] &= E\hat{x}[k] + Fy[k : k+L] \\ &= E\hat{x}[k] + F\mathcal{O}_L x[k] + F\mathcal{J}_L u[k : k+L]. \end{aligned}$$

Since $\hat{x}[0]$ is nonnegative and $x[k], u[k] \geqq 0, \forall k$, it is easy to see that $\hat{x}[k] \geqq 0, \forall k$, if E, $F\mathcal{O}_L$ and $F\mathcal{J}_L$ are all nonnegative.

(*Necessity*) Note that the positivity of the observer needs to hold for any (nonnegative) choices of the initial estimates $\hat{x}[0]$, initial condition $x[0]$ and input sequences $\{u[k]\}$. If any of the matrices E, $F\mathcal{O}_L$ and $F\mathcal{J}_L$ is not nonnegative, then there always exists a combination of $\hat{x}[0]$, $x[0]$ and $u[0 : L]$ such that $\hat{x}[1]$ is negative, which contradicts the positivity of the observer. For example, assume some element E_{ij} of E is negative. Then a natural choice is $x[0] = 0$, $u[0 : L] = 0$, and $\hat{x}[0] = e_j$, where $e_j \in \{0, 1\}^n$ with only the j-th element equal to 1, and it is easy to see that $\hat{x}_i[1]$ will be negative. □

Remark 4. *Note that similar to the analysis in [113], we do not require F to be nonnegative; this is because $y[k : k + L]$ is composed of two nonnegative ingredients $x[k]$ and $u[k : k + L]$ which thus constrain $y[k : k + L]$.*

In order to choose the matrices E and F to achieve asymptotic estimation, we look at the dynamics of the estimation error:

$$\begin{aligned} e[k+1] &= \hat{x}[k+1] - x[k+1] \\ &= E\hat{x}[k] + Fy[k : k+L] - Ax[k] - Bx[k] \\ &= Ee[k] + Fy[k : k+L] + (E - A)x[k] - Bu[k] \\ &= Ee[k] + (E - A + F\mathcal{O}_L)x[k] + F\mathcal{J}_L u[k : k+L] - Bu[k]. \end{aligned}$$

Given the positive system (3.1), in order to achieve asymptotic estimation and guarantee the positivity of the observer, the matrices E and F should satisfy the following properties:

1. $E = A - F\mathcal{O}_L$ is stable and nonnegative;
2. $F\mathcal{O}_L$ is nonnegative;
3. $F\mathcal{J}_L = B'$, where $B' = \begin{bmatrix} B & 0 & \cdots & 0 \end{bmatrix}$.

Remark 5. *Note that there exists a positive observer of the form (3.2) achieving asymptotic estimation if and only if there exist matrices E and F satisfying the above properties. A necessary condition for the existence of a matrix F satisfying the third condition is that the system (3.1) is invertible (i.e., $\operatorname{rank}(\mathcal{J}_L) - \operatorname{rank}(\mathcal{J}_{L-1}) = m$ for some L) [133]. The smallest nonnegative integer L for which this rank condition holds is called the delay of integration, and can be larger than 1. Thus, a delayed observer of the form (3.2) is typically necessary for state estimation, even for positive systems.*

Next we will relate the existence of the positive observer to the feasibility of a linear programming problem, which is similar to the approach adopted in [113]. We will use the following result.

Lemma 5 ([120]). *Given a positive matrix $A \in \mathbb{R}^{n\times n}_{\geq 0}$, A is stable if and only if there exists $w \in \mathbb{R}^n$ such that $w > 0$ and $(I - A)w > 0$.*

Let

$$\mathcal{O}_L = \begin{bmatrix} \mathcal{O}_L^1 & \cdots & \mathcal{O}_L^n \end{bmatrix},$$

and

$$\mathcal{J}_L = \begin{bmatrix} \mathcal{J}_L^1 & \cdots & \mathcal{J}_L^{(L+1)m} \end{bmatrix},$$

where $\mathcal{O}_L^1, \ldots, \mathcal{O}_L^n, \mathcal{J}_L^1, \ldots, \mathcal{J}_L^{(L+1)m} \in \mathbb{R}^{(L+1)p}$ are the columns of $\mathcal{O}_L$ and $\mathcal{J}_L$, respectively.

Theorem 9. *Given the positive system (3.1), the following statements are equivalent:*

- *There exists a positive observer (3.2) with gain matrices E and F achieving asymptotic estimation.*

- *The following linear programming problem in the variables $w = \begin{bmatrix} w_1 & \cdots & w_n \end{bmatrix}^T \in \mathbb{R}^n$, $z_1, \ldots, z_n \in \mathbb{R}^{(L+1)p}$ is feasible:*

$$A^T w - \mathcal{O}_L^T \sum_{i=1}^{n} z_i > 0 \tag{3.3}$$

$$A_{ij} w_i - z_i^T \mathcal{O}_L^j \geq 0, \ \forall\, 1 \leq i, j \leq n \tag{3.4}$$

$$z_i^T \mathcal{O}_L^j \geq 0, \ \forall\, 1 \leq i, j \leq n \tag{3.5}$$

$$z_i^T \mathcal{J}_L^j = B_{ij} w_i, \ \forall\, 1 \leq i \leq n, \forall\, 1 \leq j \leq m \tag{3.6}$$

$$z_i^T \mathcal{J}_L^j = 0, \ \forall\, 1 \leq i \leq n, \forall\, m < j \leq (L+1)m \tag{3.7}$$

$$w > 0. \tag{3.8}$$

Furthermore, the gain matrix $F = \begin{bmatrix} \frac{z_1}{w_1} & \cdots & \frac{z_n}{w_n} \end{bmatrix}^T$ and $E = A - F\mathcal{O}_L$.

Proof. Note that a matrix is nonnegative and stable if and only if its transpose is nonnegative and stable; thus, by Lemma 5, we know that inequalities (3.3), (3.4) and (3.8) are equivalent to the condition that $A - F\mathcal{O}_L$ is nonnegative and stable. It is easy to check that inequality (3.5) is equivalent to the condition that $F\mathcal{O}_L \geqq 0$ (provided w is positive), and equality (3.6) and inequality (3.7) are equivalent to the condition that $F\mathcal{J}_L = B'$. Thus, the result follows by the previous analysis of error dynamics. □

Remark 6. *Note that an alternative approach to obtain positive observers is to first construct a general observer and then project the estimate into the nonnegative quadrant. In contrast, the approach we considered in this section is to take positivity of the observer into account at the design phase. As we have mentioned in Section 1.1, the benefits of enforcing the property of positivity include simplifying the stability analysis (e.g., allowing the linear-programming based design algorithm) and bringing certain robustness properties (e.g., stability under uncertainties or time-varying perturbations in the system matrices) [114, 120].*

3.5 Summary

In this chapter, we explored the influence of the positivity of the system on state estimation with unknown inputs; such systems arise often in applications where the quantities are positive. We extended the concept of strong observability to positive systems and showed that the property of positivity is not helpful in performing state estimation. For cases where it is desirable to require the observer to output nonnegative estimates for positive systems, we also studied the construction of delayed positive unknown input observers and proposed a linear programming based design procedure.

Chapter 4

Sensor Selection for Linear Dynamical Systems

4.1 Introduction

One of the key problems in control system design is to select an appropriate set of actuators or sensors (either at design-time or at run-time) in order to achieve certain performance objectives [142]. In the context of sensor networks and robotics coordination, the problem of sensor scheduling has received much attention, where the objective is to dynamically select sensors at run-time to minimize a certain cost function (e.g., energy consumption or estimation error) [40, 56, 71, 95]. However, the assumption that the sensors can be chosen at run-time may not be feasible in many other applications, e.g., temperature monitoring in multi-processor systems [148].

The design-time sensor selection problem (where the set of chosen sensors does not evolve over time) has also been studied in various forms. In [22, 108], the problem of sensor selection for structured systems [28] is considered; the goal is to choose a subset of sensors such that the system is guaranteed to satisfy a certain structural property (e.g., fault diagnosability [22] or structural observability [108]). Alternative formulations include selecting a feasible set of sensors to optimize an energy related metric [131] or an information theoretic metric [66].

In this chapter, we consider the design-time sensor selection problem for optimal filtering of discrete-time linear dynamical systems. Specifically, we study the problem of choosing a set of sensors (under certain constraints) to minimize either the *a priori* or the *a posteriori* error covariance of the corresponding Kalman filter. As mentioned in Section 1.2, we will refer to these problems as the priori and posteriori *Kalman filtering sensor selection (KFSS)* problems, respectively.

We will first study the complexity of the priori and posteriori KFSS problems and show that it is NP-hard to find the optimal solution of these problems, even when the system is stable. Then we will provide insights into what factors of the system affect the performance of sensor selection algorithms by using the concept of the *sensor information matrix* [53].

Since it is intractable to find the optimal selection of sensors in general, a reasonable tradeoff is to design appropriate approximation algorithms. For the (run-time) sensor scheduling problem, it has been shown that certain greedy algorithms can be applied to obtain guaranteed performance due to the fact that certain cost functions have the nice property of being submodular [55], and this inspires us to study greedy algorithms for the priori and posteriori KFSS problems. While the cost functions of both the priori and posteriori KFSS problems do not necessarily have certain modularity properties, we show via simulations that greedy algorithms perform well in practice.

We also consider the problem of optimizing an upper bound of the original cost functions (of the priori and posteriori KFSS problems) based on the Lyapunov equation and propose a variant of *a priori* covariance based and *a posteriori* covariance based greedy algorithms. We show that the relaxed cost function has a strong property of being modular and the running time of the corresponding greedy algorithm scales more slowly with the number of states in the system.

The rest of the chapter is organized as follows. In Section 4.2, we provide some background on sensor scheduling. In Section 4.3, we formulate the (design-time) sensor selection problems. In Section 4.4, we analyze the complexity of the priori and posteriori KFSS problems. In Section 4.5, we provide worst-case guarantees on the performance of sensor selection algorithms. In Section 4.6, we propose and study three greedy algorithms for sensor selection, and illustrate their performance and complexity in Section 4.7. We conclude in Section 4.8.

4.2 Background: Sensor Scheduling

In this section, we provide a general framework for the (discrete-time) sensor scheduling problem and briefly review the corresponding literature.

Consider the discrete-time linear system

$$x[k+1] = Ax[k] + w[k], \tag{4.1}$$

where $x[k] \in \mathbb{R}^n$ is the system state, $w[k] \in \mathbb{R}^n$ is a zero-mean white Gaussian noise process with $\mathbb{E}\left[w[k](w[k])^T\right] = W$ for all $k \in \mathbb{N}$, and $A \in \mathbb{R}^{n \times n}$ is the system dynamics matrix. We assume throughout this chapter that the pair $(A, W^{\frac{1}{2}})$ is stabilizable.

The set of sensors to be chosen must come from a given set $\mathcal{Q}$ consisting of q sensors. Each sensor $i \in \mathcal{Q}$ provides a measurement of the form

$$y_i[k] = C_i x[k] + v_i[k], \tag{4.2}$$

where $C_i \in \mathbb{R}^{s_i \times n}$ is the state measurement matrix for that sensor, and $v_i[k] \in \mathbb{R}^{s_i}$ is a zero-mean white Gaussian noise process. For convenience, we define

$$y[k] \triangleq \begin{bmatrix} y_1[k] \\ \vdots \\ y_q[k] \end{bmatrix}, C \triangleq \begin{bmatrix} C_1 \\ \vdots \\ C_q \end{bmatrix}, v[k] \triangleq \begin{bmatrix} v_1[k] \\ \vdots \\ v_q[k] \end{bmatrix}.$$

Then the measurement equation corresponding to the output of all sensors is

$$y[k] = Cx[k] + v[k]. \tag{4.3}$$

We denote $\mathbb{E}\left[v[k](v[k])^T\right] = V$ and take $\mathbb{E}\left[v[k](w[j])^T\right] = 0$ for all $j, k \in \mathbb{N}$.

Let $z_i[k]$ be the indicator variable of sensor i at time-step k, i.e., $z_i[k] \in \{0, 1\}$ and $z_i[k] = 1$ if and only if the measurement of sensor i is chosen at time-step k. Based on the requirement of the specific application, one can specify certain constraints on the selection of sensors (e.g., impose an upper bound on the number of sensors that can be chosen at each time-step). Let $z[k] \in \{0, 1\}^q$ be the indicator vector of the set of chosen sensors at time-step k, and denote the set of feasible indicator vectors (which satisfy the specified constraints) at time-step k to be $\mathcal{Z}[k]$.

Given an estimation strategy for the state x, the sensor scheduling problem is to design a scheduling policy $\mathcal{P} = \{z[k] | z[k] \in \mathcal{Z}[k], \forall k\}$ such that the average error covariance of the corresponding estimator $\hat{x}_{\mathcal{P}}$ is minimized. Specifically, for the finite-horizon version of the problem over time interval $[0, T]$, the objective is to solve the following optimization problem:

$$\min_{\mathcal{P}, \hat{x}_{\mathcal{P}}} J_T \triangleq \frac{1}{T} \sum_{k=1}^{T} \mathbb{E}[(x[k] - \hat{x}_{\mathcal{P}}[k])^T Q_k (x[k] - \hat{x}_{\mathcal{P}}[k])],$$

where $\{Q_k\}$ is a sequence of semidefinite weighting matrices.

For the infinite-horizon version, the goal is to solve the following problem:

$$\min_{\mathcal{P}, \hat{x}_{\mathcal{P}}} \limsup_{T \to \infty} J_T.$$

Note that the specific form of the estimator $\hat{x}_{\mathcal{P}}$ depends on the estimation strategy adopted and its performance depends on the scheduling policy $\mathcal{P}$. A common choice of the estimation strategy is Kalman filtering and the corresponding sensor scheduling problem has been studied extensively, especially the finite-horizon version of the problem.

In [40], the Kalman filter is used for state estimation after data fusion and the authors proposed a stochastic sensor scheduling algorithm to minimize an upper bound on the expected steady state estimation error covariance. In [58], the authors gave a convex relaxation based approach for parameter estimation, which provided a general framework for various cost functions (e.g., performance criteria or energy and topology constraints); however, [58] assumes that the sensor measurements are uncorrelated, which may not be true in practice. Thus, in [95], another framework is proposed to handle correlated measurements. Some other interesting works include [144], where optimal and suboptimal sensor scheduling algorithms based on tree pruning techniques are given, and [74], where the optimization problem is decomposed into coupled small convex optimization problems which can be solved in a distributed fashion. However, so far, the solutions proposed for the finite-horizon problem either are computationally inefficient or consist of heuristics with no guaranteed performance (except for the greedy policies relying on submodularity of the corresponding objective functions) [55].

Recently, the infinite-horizon sensor scheduling problem has received increasing attentions, e.g., see [56,71,74,99,150]. In [71], the authors considered the continuous-time sensor scheduling problem; leveraged the results of the classical *Restless Bandit Problem*, they provided analytical expressions for a simplified scalar version of the problem and proposed a family of periodically switching policies for the multi-dimensional systems. In [74], the authors studied a discrete-time version of the problem. While the original problem is deterministic, they proposed a stochastic strategy with performance bounds; moreover, they prove the monotonicity and trace-convexity properties of the underlying discrete-time modified ARE (MARE) and provided a closed-form solution for a special class of MARE. In [56], the authors considered the discrete-time sensor scheduling problem and characterized the conditions under which there exists a schedule with uniformly bounded estimation error covariance. When such conditions are satisfied, they proposed a scheduling algorithm that guarantees bounded error covariance. In [150], some interesting properties of the solutions of the discrete-time infinite-horizon scheduling problem are demonstrated; specifically, the authors showed that both the optimal infinite-horizon average-per-stage cost and the corresponding optimal schedules are independent of the covariance matrix of the initial state, and the optimal estimation cost can be approximated arbitrarily closely by some periodic schedule.

Finally, we note that the actuator scheduling problem (i.e., the problem of scheduling actuators to minimize control efforts) can be regarded as a dual problem of the sensor scheduling problem; see [26] for more details.

4.3 Problem Formulation

In this section, we formulate the design-time sensor selection problem formally. The system model is the same as the sensor scheduling problem (i.e., equations (4.1) and (4.3) in Section 4.2). The difference is that the set of chosen senors do not change over time (i.e., the indicator variables do not evolve over time). This is due to the fact that in the configuration of multi-processor systems, the sensors can only be installed on the system at design-time. Let $z \in \{0,1\}^q$ be the indicator vector of the installed sensors, i.e., $z_i = 1$ if and only if sensor $i \in \mathcal{Q}$ is installed. Define the selection matrix $Z \triangleq \text{diag}(z_1 I_{s_1 \times s_1}, \cdots, z_q I_{s_q \times s_q})$ and denote $\tilde{C} \triangleq ZC$ and $\tilde{V} \triangleq ZVZ^T$.

Each sensor $i \in \mathcal{Q}$ has an associated *cost* $r_i \in \mathbb{R}_{\geq 0}$, representing, for example, monetary costs of purchasing and installing that sensor or the energy consumption of that sensor. Define the cost vector $r \triangleq [r_1 \cdots r_q]^T$. We also assume there is a *sensor budget* $\beta \in \mathbb{R}_{\geq 0}$, representing the total cost that can be spent on sensors from $\mathcal{Q}$.

For any given subset of sensors that is installed, the Kalman filter provides the optimal estimate of the state using the measurements from those sensors (in the sense of minimizing mean square estimation error, under the stated assumptions on the noise processes).

The algorithm for the Kalman filter consists of two steps: the measurement update and time update. Let $\Sigma_{k|k-1}(z)$ and $\Sigma_{k|k}(z)$ be the *a priori* error covariance matrix and the *a posteriori* error covariance matrix (at time-step k) of the Kalman filter when the set of sensors indicated by the vector z are installed, respectively. Then for a given selection z, we have

$$\Sigma_{k|k-1}(z) = A\Sigma_{k-1|k-1}(z)A^T + W,$$

and

$$\Sigma_{k|k}(z) = \Sigma_{k|k-1}(z) - \Sigma_{k|k-1}(z)\tilde{C}^T(\tilde{C}\Sigma_{k|k-1}(z)\tilde{C}^T + \tilde{V})^{-1}\tilde{C}\Sigma_{k|k-1}(z).$$

If the pair $(A, \tilde{C})$ is detectable (and given the stabilizability of $(A, W^{\frac{1}{2}})$), both $\Sigma_{k|k-1}(z)$ and $\Sigma_{k|k}(z)$ will converge to unique limits [2]; denote the limits of $\Sigma_{k|k-1}(z)$ and $\Sigma_{k|k}(z)$ by $\Sigma(z)$ and $\Sigma^*(z)$, respectively. We will also use $\Sigma(\mathcal{S})$ and $\Sigma^*(\mathcal{S})$ to denote these quantities for a specific set of sensors $\mathcal{S} \subseteq \mathcal{Q}$.

The limit $\Sigma(z)$ of the *a priori* error covariance satisfies the following *discrete algebraic Riccati equation (DARE)* [2]:

$$\Sigma(z) = A\Sigma(z)A^T + W - A\Sigma(z)\tilde{C}^T(\tilde{C}\Sigma(z)\tilde{C}^T + \tilde{V})^{-1}\tilde{C}\Sigma(z)A^T. \tag{4.4}$$

The DARE is an important research topic in control and filtering, and has many applications (e.g., the LQR problem, canonical factorization, and H_∞ control problem) [70]. However, there is still no known closed form solution for DARE [116].

Using the matrix inversion lemma [48], the DARE (4.4) can also be written as

$$\Sigma(z) = W + A(\Sigma^{-1}(z) + \underbrace{\tilde{C}^T\tilde{V}^{-1}\tilde{C}}_{R(z)})^{-1}A^T, \tag{4.5}$$

where the matrix $R(z)$ is the so-called sensor information matrix corresponding to the indicator vector z.[1] Note that $R(z)$ is a function of the indicator vector z and subsumes the information contribution of the chosen sensors specified by z.

The limit $\Sigma^*(z)$ of the *a posteriori* error covariance satisfies the following equation [15]:

$$\Sigma^*(z) = \big((A\Sigma^*(z)A^T + W)^{-1} + R(z)\big)^{-1}. \tag{4.6}$$

Note that $\Sigma(z)$ and $\Sigma^*(z)$ are coupled as follows [15]:

$$\Sigma^*(z) = \Sigma(z) - \Sigma(z)\tilde{C}^T(\tilde{C}\Sigma(z)\tilde{C}^T + \tilde{V})^{-1}\tilde{C}\Sigma(z), \tag{4.7}$$

or equivalently,

$$\Sigma^*(z) = (\Sigma^{-1}(z) + R(z))^{-1}. \tag{4.8}$$

Further note that the inverses in the equations (4.4)-(4.8) are interpreted as pseudo-inverses if the arguments are not invertible.[2]

Definition 14 (Feasible Sensor Selection)**.** *The sensor selection $z \in \{0,1\}^q$ is said to be **feasible** if both $\Sigma_{k|k-1}(z)$ and $\Sigma_{k|k}(z)$ converge to finite limits (denoted by $\Sigma(z)$ and $\Sigma^*(z)$, respectively) as $k \to \infty$, and the limits do not depend on $\Sigma_{0|0}(z)$. When z is not feasible, define $\mathrm{trace}(\Sigma(z)) = \infty$ and $\mathrm{trace}(\Sigma^*(z)) = \infty$.*

We now propose the following priori and posteriori Kalman filtering sensor selection (KFSS) problems. Denote $s = \sum_{i=1}^q s_i$.

Problem 1 (Priori KFSS Problem)**.** *Given a system dynamics matrix $A \in \mathbb{R}^{n\times n}$, a measurement matrix $C \in \mathbb{R}^{s\times n}$, a system noise covariance matrix $W \in \mathbb{S}_+^n$, a sensor noise covariance matrix $V \in \mathbb{S}_+^s$, a cost vector $r \in \mathbb{R}_{\geq 0}^q$, and a budget $\beta \in \mathbb{R}_{\geq 0}$, the priori KFSS problem is to solve the optimization problem:*

$$\begin{array}{cl} \min\limits_{z} & \mathrm{trace}(\Sigma(z)) \\ \textit{s.t.} & r^T z \leq \beta \\ & z \in \{0,1\}^q \end{array}$$

where $\Sigma(z)$ is given by equation (4.4).

[1] Note that the sensor information matrix is different from the *Fisher information matrix*, which is the inverse of the error covariance matrix [2].

[2] For the special case of $V = 0$, the matrix inversion lemma does not hold under pseudo-inverses (unless $z = 0$) and thus we can compute $\Sigma(z)$ and $\Sigma^*(z)$ by equations (4.4) and (4.7).

Problem 2 (Posteriori KFSS Problem)**.** *Given a system dynamics matrix $A \in \mathbb{R}^{n\times n}$, a measurement matrix $C \in \mathbb{R}^{s\times n}$, a system noise covariance matrix $W \in \mathbb{S}^n_+$, a sensor noise covariance matrix $V \in \mathbb{S}^s_+$, a cost vector $r \in \mathbb{R}^q_{\geq 0}$, and a budget $\beta \in \mathbb{R}_{\geq 0}$, the posteriori KFSS problem is to solve the optimization problem:*

$$\begin{aligned} \min_{z} \quad & \operatorname{trace}(\Sigma^*(z)) \\ \textit{s.t.} \quad & r^T z \leq \beta \\ & z \in \{0,1\}^q \end{aligned}$$

where $\Sigma^*(z)$ is given by equation (4.6).

Note that the only difference between Problem 1 and Problem 2 is the cost function (the former is to minimize $\operatorname{trace}(\Sigma(z))$ and the latter is to minimize $\operatorname{trace}(\Sigma^*(z))$). In the following sections, we will discuss the complexity of the two KFSS problems and investigate approaches to address these problems.

4.4 Complexity of the Priori and Posteriori KFSS Problems

In this section, we show that the priori and posteriori KFSS problems are both NP-hard. We will use the following well-known result on Kalman filtering [2].

Lemma 6. *When the pair $(A, W^{\frac{1}{2}})$ is stabilizable, the indicator vector z is feasible if and only if the pair $(A, \tilde{C})$ is detectable.*

To show the complexity of the two KFSS problems, we will relate them to the problems described below.

Problem 3. *Given a matrix $A \in \mathbb{R}^{n\times n}$, the problem of finding a diagonal matrix $M \in \mathbb{R}^{n\times n}$ with the fewest nonzero elements such that the pair (A, M) is controllable (resp. stabilizable, detectable) is referred to as the minimum controllability (resp. minimum stabilizability, minimum detectability) problem.*

Theorem 10. *The priori KFSS problem and the posteriori KFSS problem are NP-hard.*

Proof. We first give a reduction from the minimum detectability problem to the priori KFSS problem (resp. posteriori KFSS problem). Given $A \in \mathbb{R}^{n\times n}$ for the minimum detectability problem and some $p \in \{1, \ldots, n\}$, the instance for the corresponding priori KFSS problem (resp. posteriori KFSS problem) with parameter p is the system matrix A, the set $\mathcal{Q}$ of n sensors with the measurement matrix $C = I_{n\times n}$, the system noise covariance matrix $W = I_{n\times n}$, the sensor noise covariance matrix $V = I_{n\times n}$, the cost vector $r = [1 \cdots 1]^T$ and the budget $\beta = p$.[3]

Suppose there is an algorithm $\mathcal{A}$ that determines the minimum value of $\mathrm{trace}(\Sigma(z))$ (resp. $\mathrm{trace}(\Sigma^*(z))$) over all z satisfying $b^T z \leq B$, or outputs a flag if there is no feasible sensor selection; recall that $\mathrm{trace}(\Sigma(z)) = \infty$ (resp. $\mathrm{trace}(\Sigma^*(z)) = \infty$) for any selection z that is not feasible. If the output of algorithm $\mathcal{A}$ is finite, by Lemma 6, we know that the solution to the minimum detectability problem (i.e., the minimum number of nonzero entries of the diagonal matrix $M \in \mathbb{R}^{n\times n}$ such that (A, M) is detectable) is at most p. In order to solve the minimum detectability problem, we need to call algorithm $\mathcal{A}$ at most n times (i.e., increase p from 1 to n). Thus, if the minimum detectability problem is NP-hard, then the priori KFSS problem (resp. posteriori KFSS problem) is also NP-hard.

The NP-hardness of the minimum detectability problem follows from the proof of NP-hardness of the minimum controllability problem in [105]. Specifically, given $n_1, n_2 \in \mathbb{Z}_{\geq 1}$ and a collection $\mathcal{C}$ of n_1 nonempty subsets of $\{1, \cdots, n_2\}$, let $A(\mathcal{C}) = U^{-1}\,\mathrm{diag}(1, \cdots, n_1+n_2+1)U$, where U is some invertible matrix related to $\mathcal{C}$.[4] In [105], the author proved that $\mathcal{C}$ has a hitting set with cardinality s if and only if there exists a diagonal matrix B with no more than s nonzero entries such that $(A(\mathcal{C}), B)$ is controllable. Since the hitting set problem is NP-hard, the minimum controllability problem is also NP-hard.

Note that the set of eigenvalues of $A(\mathcal{C})$ is $\{1, \cdots, n_1 + n_2 + 1\}$, which are all unstable. Thus, to find a matrix B such that $(A(\mathcal{C}), B)$ is stabilizable is equivalent to finding a matrix B such that $(A(\mathcal{C}), B)$ is controllable, which implies that the minimum stabilizability problem is NP-hard. By the duality of stabilizability and detectability, the minimum detectability problem is also NP-hard, completing the proof. □

Note that the above result shows that it is NP-hard to find a feasible solution for the priori and posterior KFSS problems, even when all sensors have identical costs. In the following, we show that the priori and posterior KFSS problems are still NP-hard if the system dynamics matrix A is stable (so that *all* sensor selections are feasible), but when the sensor costs can be arbitrary. The reductions are both from the optimization form of the 0-1 knapsack problem [60].

Definition 15 (0-1 Knapsack Problem)**.** *Given a set of n items, each with a value α_i and a weight*

[3] Note that here $s = n$ (i.e., $s_i = 1, \forall i$).

[4] Note that U is constructed based on the incidence matrix of $\mathcal{C}$; we omit the construction details and refer to the proof of Theorem 1.1 in [105].

β_i, and a weight budget B, the 0-1 *knapsack problem is to solve the optimization problem:*

$$\begin{aligned}\max_{z} \quad & \alpha_i z_i \\ \textit{s.t.} \quad & \sum_{i=1}^{n} \beta_i z_i \leq B \\ & z \in \{0,1\}^n\end{aligned}$$

where z is the indicator vector.

Theorem 11. *The priori KFSS problem is NP-hard even under the additional assumption that the system dynamics matrix A is stable.*

Proof. Consider the case where the measurement of each sensor is a scalar (i.e., $s_i = 1, \forall i$). In this case, when $A = aI_{n\times n}$ with $0 < a < 1$ being some constant, $C = I_{n\times n}$, $V = 0$, and $W = \operatorname{diag}([w_1 \cdots w_n])$, we know that

$$\Sigma = \begin{bmatrix} \Sigma_{11} & & \\ & \ddots & \\ & & \Sigma_{nn} \end{bmatrix},$$

where for $i = 1, \dots, n$,

$$\Sigma_{ii} = \begin{cases} w_i, & z_i = 1 \\ \frac{w_i}{1-a^2}, & z_i = 0 \end{cases}.$$

Thus, the reduction of the *a priori* estimation error by adding sensor i is

$$\Sigma_{ii}(z_i = 0) - \Sigma_{ii}(z_i = 1) = \frac{a^2}{1-a^2} w_i, \forall i.$$

Given the number of items n, the set of values $\{\alpha_i\}$, the set of weights $\{\beta_i\}$ and the weight budget B for the 0-1 knapsack problem, the corresponding instance for the priori KFSS problem is the *stable* system matrix $A = \frac{1}{2} I_{n\times n}$ (i.e., take the constant $a = \frac{1}{2}$), the set $\mathcal{Q}$ of n sensors with the measurement matrix $C = I_{n\times n}$, the system noise covariance matrix $W = \operatorname{diag}([w_1 \cdots w_n])$ with $w_i = \frac{1-a^2}{a^2}\alpha_i = 3\alpha_i$, the sensor noise covariance matrix $V = 0$, the cost vector $r = [\beta_1 \cdots \beta_n]^T$ and the budget $\beta = B$.

Then we can see that an indicator vector z for the 0-1 knapsack problem is optimal if and only if it is optimal for the corresponding priori KFSS problem. Since the optimization form of the 0-1 knapsack problem is NP-hard, the priori KFSS problem is NP-hard even under the additional assumption that the matrix A is stable, completing the proof. □

Theorem 12. *The posteriori KFSS problem is NP-hard even under the additional assumption that the system dynamics matrix A is stable.*

Proof. Consider the case where the measurement of each sensor is a scalar (i.e., $s_i = 1, \forall i$). In this case, when $A = 0$, $C = I_{n\times n}$, $V = 0$, and $W = \text{diag}([w_1 \cdots w_n])$, we have

$$\Sigma^* = \begin{bmatrix} \Sigma_{11}^* & & \\ & \ddots & \\ & & \Sigma_{nn}^* \end{bmatrix},$$

where for $i = 1, \dots, n$,

$$\Sigma_{ii}^* = \begin{cases} 0, & z_i = 1 \\ w_i, & z_i = 0 \end{cases}.$$

Thus, the reduction of the *a posteriori* estimation error by adding sensor i is

$$\Sigma_{ii}^*(z_i = 0) - \Sigma_{ii}^*(z_i = 1) = w_i, \forall i.$$

Given the number of items n, the set of values $\{\alpha_i\}$, the set of weights $\{\beta_i\}$ and the weight budget B for the 0-1 knapsack problem, the corresponding instance for the posteriori KFSS problem is the (stable) system matrix $A = 0$, the set $\mathcal{Q}$ of n sensors with the measurement matrix $C = I_{n\times n}$, the system noise covariance matrix $W = \text{diag}([w_1 \cdots w_n])$ with $w_i = \alpha_i$, the sensor noise covariance matrix $V = 0$, the cost vector $r = [\beta_1 \cdots \beta_n]^T$ and the budget $\beta = B$.

Following the same argument as in the proof of Theorem 11 and the fact that the optimization form of the 0-1 knapsack problem is NP-hard, the posteriori KFSS problem is NP-hard even under the additional assumption that the matrix A is stable. □

Remark 7. *There are few works in the literature that explicitly characterize the complexities of sensor selection problems. Exceptions include [11] where a utility-based sensor selection problem is shown to be NP-hard, and [140] where the NP-hardness of an energy metric based sensor selection problem is established.*

In the rest of this chapter, we focus on the case where the pair (A, C_i) is detectable, $\forall i \in \{1, \cdots, q\}$. Note that in this case, any choice of sensors (except $z = \mathbf{0}$ if A is unstable) is feasible.

4.5 Upper Bounds on the Performance of Sensor Selection Algorithms

In this section, we study worst-case bounds on the performance of sensor selection algorithms for the priori and posteriori KFSS problems. Specifically, we consider the following ratio $r(\Sigma)$:

$$r(\Sigma) \triangleq \frac{\text{trace}(\Sigma_{\text{worst}})}{\text{trace}(\Sigma_{\text{opt}})},$$

where Σ_{opt} and Σ_{worst} are the solutions of the DARE corresponding to the optimal selection of sensors and the worst-case feasible selection, respectively, and also the ratio

$$r(\Sigma^*) \triangleq \frac{\text{trace}(\Sigma^*_{\text{worst}})}{\text{trace}(\Sigma^*_{\text{opt}})},$$

which is defined similarly.

Remark 8. *Note that for any a priori covariance (resp. a posteriori covariance) based sensor selection algorithm, the performance of that algorithm is within $r(\Sigma)$ (resp. $r(\Sigma^*)$) times the optimal performance. In other words, the quantities $r(\Sigma)$ and $r(\Sigma^*)$ characterize the 'spectrum' of the performance of all feasible selections.*

Note that since it is in general difficult to obtain the analytical solution of the DARE (4.4) (and also the steady state *a posteriori* error covariance from equation (4.8)), the problem of providing bounds for the DARE solution has been studied extensively in the literature; see [69] and the references therein.

However, the existing upper bounds on the DARE solution typically assume that the system is stable [72] or that the corresponding sensor information matrix is nonsingular [63, 64]; the latter assumption is restrictive in the context of sensor selection. Thus, in this section, we focus on the case where the system is stable to obtain more insights into the factors that affect the performance of sensor selection algorithms.

4.5.1 Upper Bound for $r(\Sigma)$

We first derive an upper bound on the ratio $r(\Sigma)$ for the priori KFSS problem when the system is stable. We will be using the following results.

Lemma 7 ([64]). *For $\Sigma \succeq 0$ satisfying the DARE (4.5) with $W \succ 0$, we have*

$$\Sigma(z) \succeq A(W^{-1} + R(z))^{-1}A^T + W. \tag{4.9}$$

Lemma 8 ([149]). *For Hermitian matrices $M, N \in \mathbb{C}^{n\times n}$, we have*

$$\lambda_n(M)\,\mathrm{trace}(N) \leq \mathrm{trace}(MN) \leq \lambda_1(M)\,\mathrm{trace}(N).$$

Lemma 9 ([48]). *For Hermitian matrices $M, N \in \mathbb{C}^{n\times n}$, we have the following Weyl's inequalities:*

$$\lambda_n(M+N) \geq \lambda_n(M) + \lambda_n(N),$$
$$\lambda_1(M+N) \leq \lambda_1(M) + \lambda_1(N).$$

Lemma 10 ([81]). *A square matrix $A \in \mathbb{R}^{n\times n}$ is Schur stable if and only if there exists a nonsingular matrix P such that $\sigma_1(PAP^{-1}) < 1$.*

Remark 9. *Note that in the above lemma, the matrix P can be constructed by using the eigenvalues and (generalized) eigenvectors of A [81]. Thus, for any stable square matrix A, there exists some positive constant $\alpha_A \triangleq \frac{\sigma_1^2(P)}{\sigma_n^2(P)(1-\sigma_1^2(PAP^{-1}))}$ which only depends on A, where the matrix P is nonsingular and satisfies $\sigma_1(PAP^{-1}) < 1$. This constant α_A will be used to establish the performance upper bounds in the rest of this section.*

To incorporate the nature of the sensor set $\mathcal{Q}$, our results will use the sensor information matrix $R(z)$ from (4.5) which encapsulates both the measurement matrix $\tilde{C}$ and the sensor noise covariance matrix $\tilde{V}$ corresponding to the indicator vector z.

Theorem 13. *For a given cost vector r and budget β, let $\mathcal{R} = \{R(z)\}$ be the set of all sensor information matrices such that the constraint $r^Tz \leq \beta$ is satisfied. Denote $\lambda_1^{max} \triangleq \max\{\lambda_1(R)|R \in \mathcal{R}\}$. Then for the system (5.5) with stable A and $W \succ 0$,*

$$r(\Sigma) \leq \frac{\alpha_A(1+\lambda_1^{max}\lambda_n(W))\,\mathrm{trace}(W)}{n\sigma_n^2(A)\lambda_n(W) + (1+\lambda_1^{max}\lambda_n(W))\,\mathrm{trace}(W)}, \tag{4.10}$$

where α_A is some positive constant that only depends on A, as defined in Remark 9.

Proof. We first provide an upper bound for $\operatorname{trace}(\Sigma_{\text{worst}})$. Consider the case where $z = \mathbf{0}$ (i.e., no sensors are chosen). Note that since A is stable, $z = \mathbf{0}$ is feasible (in the sense of Definition 14). In this case, the DARE (4.4) becomes the Lyapunov equation

$$\Sigma(\mathbf{0}) = A\Sigma(\mathbf{0})A^T + W.$$

Define

$$\bar{\Sigma} = P\Sigma(\mathbf{0})P^T$$

and

$$\bar{W} = PWP^T,$$

where P is nonsingular and satisfies $\sigma_1(PAP^{-1}) < 1$. Note that since the matrix A is stable, by Lemma 10, such a matrix P always exists.

Let $D = PAP^{-1}$. Then we get

$$\bar{\Sigma} = D\bar{\Sigma}D^T + \bar{W}.$$

By Lemma 8, we know that

$$\begin{aligned} \operatorname{trace}(D\bar{\Sigma}D^T) &= \operatorname{trace}(D^T D\bar{\Sigma}) \\ &\leq \sigma_1^2(D)\operatorname{trace}(\bar{\Sigma}) \end{aligned}$$

and thus

$$\operatorname{trace}(\bar{\Sigma}) \leq \frac{\operatorname{trace}(\bar{W})}{1 - \sigma_1^2(D)}.$$

Since $W, \Sigma \succeq 0$ and the matrix $P^T P$ is symmetric, by Lemma 8, we know that

$$\begin{aligned} \operatorname{trace}(\bar{\Sigma}) &= \operatorname{trace}(P^T P\Sigma(\mathbf{0})) \\ &\geq \sigma_n^2(P)\operatorname{trace}(\Sigma(\mathbf{0})), \end{aligned}$$

and

$$\begin{aligned} \operatorname{trace}(\bar{W}) &= \operatorname{trace}(P^T PW) \\ &\leq \sigma_1^2(P)\operatorname{trace}(W). \end{aligned}$$

Combining the above analysis, we obtain

$$\begin{aligned} \operatorname{trace}(\Sigma_{\text{worst}}) &\leq \operatorname{trace}(\Sigma(\mathbf{0})) \\ &\leq \frac{\sigma_1^2(P)}{\sigma_n^2(P)} \frac{\operatorname{trace}(W)}{1 - \sigma_1^2(D)} \\ &= \alpha_A \operatorname{trace}(W), \end{aligned}$$

where the last equality is due to Lemma 10 and Remark 9.

Next we derive a lower bound for $\mathrm{trace}(\Sigma_{\text{opt}})$. Specifically, for any given z, we have

$$\begin{aligned}
\mathrm{trace}(\Sigma(z)) &\geq \mathrm{trace}(A(W^{-1}+R(z))^{-1}A^T+W) && (4.11)\\
&\geq \lambda_n(A^TA)\,\mathrm{trace}((W^{-1}+R(z))^{-1})+\mathrm{trace}(W) && (4.12)\\
&= \sigma_n^2(A)\sum_{i=1}^{n}\frac{1}{\lambda_i(W^{-1}+R(z))}+\mathrm{trace}(W)\\
&\geq \frac{n\sigma_n^2(A)}{\lambda_1(W^{-1}+R(z))}+\mathrm{trace}(W)\\
&\geq \frac{n\sigma_n^2(A)}{\lambda_1(W^{-1})+\lambda_1(R(z))}+\mathrm{trace}(W) && (4.13)\\
&\geq \frac{n\sigma_n^2(A)}{\frac{1}{\lambda_n(W)}+\lambda_1^{\text{max}}}+\mathrm{trace}(W).
\end{aligned}$$

Note that inequality (4.11) is due to Lemma 7, inequalities (4.12) is due to Lemma 8 and inequality (4.13) is due to Lemma 9. Further note that the derived lower bound for $\mathrm{trace}(\Sigma(z))$ holds for any sensor selection z and thus holds for $\mathrm{trace}(\Sigma_{\text{opt}})$.

The result follows by combining the upper bound for $\mathrm{trace}(\Sigma_{\text{worst}})$ and the lower bound for $\mathrm{trace}(\Sigma_{\text{opt}})$. □

The above result also yields a simpler upper bound for $r(\Sigma)$ which highlights the role of the system dynamics matrix A.

Corollary 1. *If the given system (5.5) is stable, there exists a constant α_A which only depends on the matrix A such that $r(\Sigma) \leq \alpha_A$. Furthermore, if the matrix A is stable and normal (i.e., $A^TA = AA^T$), then*

$$r(\Sigma) \leq \frac{1}{1-\lambda_1^2(A)}.$$

Proof. The proof of the first part (i.e., $r(\Sigma) \leq \alpha_A$) immediately follows by noting that the denominator in (4.10) is lower bounded by $(1+\lambda_1^{\text{max}}\lambda_n(W))\,\mathrm{trace}(W)$.

When A is normal, the set of singular values of A coincides with its eigenvalues [48] (i.e., $\sigma_i(A) = |\lambda_i(A)|, \forall i$). Since A is stable, we know that $\sigma_1(A) = |\lambda_1(A)| < 1$. Thus, in Lemma 10, we can choose the transformation matrix P to be the identity matrix (i.e., $P = I$). Then by Remark 9, we have $r(\Sigma) \leq \alpha_A = \frac{1}{1-\lambda_1^2(A)}$, completing the proof. □

Remark 10. *Note that for fixed A and W, the upper bound of $r(\Sigma)$ in (4.10) approaches α_A as λ_1^{max} gets bigger. In other words, when the measurement (from the past time-step) is more accurate, the worst-case difference among all feasible sensor selection algorithms is mainly determined by the system dynamics. Since there exists an upper bound for $r(\Sigma)$ which only depends on the system dynamics matrix A, no sensor selection algorithm will provide arbitrarily bad performance as long as A is well conditioned, regardless of the statistics of the noise processes and the nature of the sensor set $\mathcal{Q}$. In particular, if $A = 0$, then the state $x[k+1]$ in (5.5) is uncorrelated with $x[k]$, and thus measurements of the current state are not useful in predicting the next state. This is corroborated by the fact that $r(\Sigma) = 1$ in this case.*

4.5.2 Upper Bound for $r(\Sigma^*)$

Next we provide an upper bound on the ratio $r(\Sigma^*)$ for the posteriori KFSS problem when the system is stable. We will use the following result.

Lemma 11 ([48]). *For matrices $M, N \in \mathbb{S}_+^n$, if $M \succeq N$, we have $M^{-1} \preceq N^{-1}$.*

Theorem 14. *For given cost vector r and budget β, let $\mathcal{R} = \{R(z)\}$ be the set of all sensor information matrices such that the constraint $r^T z \leq \beta$ is satisfied. Denote $\lambda_1^{max} \triangleq \max\{\lambda_1(R) | R \in \mathcal{R}\}$. Then for the system (5.5) with stable A and $W \succ 0$,*

$$r(\Sigma^*) \leq \alpha_A \left(\frac{\lambda_1(W)}{\lambda_n(W)} + \lambda_1^{max} \lambda_1(W) \right), \tag{4.14}$$

where α_A is some positive constant that only depends on A, as defined in Remark 9.

Proof. We first give an upper bound for $\text{trace}(\Sigma^*_{\text{worst}})$. Since $R(z) \succeq 0, \forall z$, by Lemma 11 and equation (4.8), we know that $\Sigma^*(z) \preceq \Sigma(z), \forall z$. Thus, a simple upper bound for $\text{trace}(\Sigma^*_{\text{worst}})$ is

$$\begin{aligned} \text{trace}(\Sigma^*_{\text{worst}}) &\leq \text{trace}(\Sigma_{\text{worst}}) \\ &\leq \alpha_A \, \text{trace}(W) \\ &\leq n \alpha_A \lambda_1(W), \end{aligned}$$

where α_A is defined in Remark 9.

Next we give a lower bound for $\text{trace}(\Sigma^*_{\text{opt}})$. For convenience, define the following notation:

$$\begin{aligned} X_1(z) &\triangleq (A(W^{-1} + R(z))^{-1} A^T + W)^{-1} + R(z), \\ X_2(z) &\triangleq A(W^{-1} + R(z))^{-1} A^T + W. \end{aligned}$$

Note that $X_2(z)$ is the matrix lower bound for $\Sigma(z)$ given in Lemma 7 and $X_1(z) = X_2^{-1}(z) + R(z)$. Thus, by Lemma 11 and equation (4.8), we have

$$\begin{aligned}\Sigma^*(z) &= (\Sigma^{-1}(z) + R(z))^{-1} \\ &\succeq (X_2^{-1}(z) + R(z))^{-1} \\ &= X_1^{-1}(z).\end{aligned}$$

Moreover, it is easy to see that $X_2(z) \succeq W$ and thus $\lambda_n(X_2(z)) \geq \lambda_n(W)$.

Then for any given z, we have

$$\begin{aligned}\operatorname{trace}(\Sigma^*(z)) &\geq \operatorname{trace}(X_1^{-1}(z)) \\ &= \sum_{i=1}^{n} \frac{1}{\lambda_i(X_1(z))} \qquad (4.15) \\ &\geq \frac{n}{\lambda_1(X_1(z))} \\ &\geq \frac{n}{\lambda_1(X_2^{-1}(z)) + \lambda_1(R(z))} \qquad (4.16) \\ &\geq \frac{n}{\frac{1}{\lambda_n(X_2(z))} + \lambda_1^{\max}} \qquad (4.17) \\ &\geq \frac{n}{\frac{1}{\lambda_n(W)} + \lambda_1^{\max}}.\end{aligned}$$

Note that inequality (4.16) is due to Lemma 9. Since the above lower bound holds for any sensor selection z, it also holds for $\operatorname{trace}(\Sigma^*_{\text{opt}})$.

The result follows by combining the upper bound for $\operatorname{trace}(\Sigma^*_{\text{worst}})$ and the lower bound for $\operatorname{trace}(\Sigma^*_{\text{opt}})$. □

Remark 11. *As argued in Remark 10, when the system is stable, $r(\Sigma)$ can be upper bounded by a constant which only depends on the system matrix A. However, the above result suggests that $r(\Sigma^*)$ depends on both the system noise covariance matrix W and the achievable 'quality' of measurements (which is characterized by λ_1^{max}). In particular, when $C = I_{n\times n}, V = \bar{v}I_{n\times n}$ and $W = \bar{w}I_{n\times n}$, where $\bar{v}, \bar{w} > 0$ are some constants, we have $\lambda_1(W) = \lambda_n(W) = \bar{w}$, $\lambda_1^{max} = \frac{1}{\bar{v}}$ and*

$$r(\Sigma^*) \leq \alpha_A(1 + \frac{\bar{w}}{\bar{v}}).$$

Thus, for a fixed matrix A, the worst-case difference among all feasible sensor selection algorithms becomes smaller if the system noise gets smaller (i.e., $\bar{w}$ gets smaller) or the measurements become more inaccurate (i.e., $\bar{v}$ gets bigger).

4.6 Greedy Algorithms

In this section, we explore simple greedy algorithms to solve the priori and posteriori KFSS problems, given as Algorithm 1 and Algorithm 2, respectively. We focus on the case where $r = [1 \cdots 1]^T$ and $\beta = p$ for some $p \in \{1, \cdots, q\}$ (i.e., our goal is to choose p sensors out of the total q sensors to optimize the performance of the Kalman filter). In other words, an indictor vector z is valid if $z \in \mathcal{Z}_p$ where $\mathcal{Z}_p$ is defined to be the set of indicator vectors with no more than p nonzero elements. The basic idea of greedy algorithms is to iteratively pick sensors that provide the largest incremental decrease in the steady state (*a priori* or *a posteriori*) error covariance.

Algorithm 1 *A Priori* Covariance based Greedy Algorithm

```
Input: System dynamics matrix $A$, set of all sensors $\mathcal{Q}$, noise covariances $W$ and $V$, and
    constant $p$
Output: A set $\mathcal{S}$ of chosen sensors
$k \leftarrow 0, \mathcal{S} \leftarrow \emptyset$
for $k \leq p$ do
    for $i \in \mathcal{Q} \cap \bar{\mathcal{S}}$ do
        Calculate $\mathrm{trace}(\Sigma_{i,\mathcal{S}}) \triangleq \mathrm{trace}(\Sigma(\mathcal{S} \cup \{i\}))$
    end for
    Choose $j$ with $\mathrm{trace}(\Sigma_{j,\mathcal{S}}) = \min_i \mathrm{trace}(\Sigma_{i,\mathcal{S}})$
    $\mathcal{S} \leftarrow \mathcal{S} \cup \{j\}$, $\mathcal{Q} \leftarrow \mathcal{Q} \setminus \{j\}$, $k \leftarrow k + 1$
end for
```

Algorithm 2 *A Posteriori* Covariance based Greedy Algorithm

```
Input: System dynamics matrix $A$, set of all sensors $\mathcal{Q}$, noise covariances $W$ and $V$, and
    constant $p$
Output: A set $\mathcal{S}$ of chosen sensors
$k \leftarrow 0, \mathcal{S} \leftarrow \emptyset$
for $k \leq p$ do
    for $i \in \mathcal{Q} \cap \bar{\mathcal{S}}$ do
        Calculate $\mathrm{trace}(\Sigma^*_{i,\mathcal{S}}) \triangleq \mathrm{trace}(\Sigma^*(\mathcal{S} \cup \{i\}))$
    end for
    Choose $j$ with $\mathrm{trace}(\Sigma^*_{j,\mathcal{S}}) = \min_i \mathrm{trace}(\Sigma^*_{i,\mathcal{S}})$
    $\mathcal{S} \leftarrow \mathcal{S} \cup \{j\}$, $\mathcal{Q} \leftarrow \mathcal{Q} \setminus \{j\}$, $k \leftarrow k + 1$
end for
```

In the rest of this section, we will show that greedy algorithms are optimal (with respect to the corresponding KFSS problem) for two different classes of systems. However, for general systems, we provide a negative result showing that the trace of the steady state *a priori* error covariance and *a posteriori* error covariance (and other related metrics) do not satisfy certain modularity properties in general, which precludes the direct application of classical results on submodular function optimization. Finally, we will propose a variant of *a priori* covariance based and *a posteriori* covariance based greedy algorithms by optimizing a relaxed objective function.

4.6.1 Optimality of Greedy Algorithms for Two Classes of Systems

First note that when the sensor noises are uncorrelated, i.e., $\mathbb{E}\left[v_i[k_1](v_j[k_2])^T\right] = 0, \forall i \neq j, k_1, k_2$, then the sensor noise covariance matrix V is block diagonal; in this case, let $V = \text{diag}(V_1, \cdots, V_q)$ where $V_i = \mathbb{E}\left[v_i[k](v_i[k])^T\right]$. Then the sensor information matrix $R(z)$ can be written as

$$R(z) = \sum_{i=1}^{q} z_i R_i$$

where $R_i \triangleq C_i^T V_i^{-1} C_i$ is the sensor information matrix associated with sensor i. The following result characterizes the relationship between the partial orders on information matrices to the partial orders on the corresponding *a priori* and *a posteriori* error covariances.

Lemma 12 ([53, 147]). *For two selections of sensors z and z', if $R(z) \succeq R(z')$, then we have $\Sigma(z) \preceq \Sigma(z')$ and $\Sigma^*(z) \preceq \Sigma^*(z')$.*

In other words, when $R(z) \succeq R(z')$, the sensor selection associated with z is better than the one associated with z' (in the sense of having smaller *a priori* and *a posteriori* error covariances).[5]

The following result shows that when the sensor noises are uncorrelated and the set of information matrices $\{R_i\}$ is totally ordered, then Algorithm 1 and Algorithm 2 are optimal (with respect to the corresponding KFSS problems).

Proposition 6. *If the sensor noises are uncorrelated and the set of information matrices $\{R_i\}$ is totally ordered with respect to the order relation of positive semidefiniteness, then the optimal solution of the priori and posteriori KFSS problems with $r = [1 \cdots 1]^T$ and $\beta = p$ is the set of sensors $\mathcal{P} \subseteq \mathcal{Q}$ such that $|\mathcal{P}| = p$ and $R_i \succeq R_j, \forall i \in \mathcal{P}, j \in \mathcal{Q} \setminus \mathcal{P}$. Furthermore, both Algorithm 1 and Algorithm 2 output this optimal set of sensors.*

[5] Note that the partial order based on positive semidefiniteness between the matrices directly leads to an order on their traces.

Proof. We first show that the optimal solution of the priori and posteriori KFSS problems is the specified set of sensors $\mathcal{P}$. Denote $z_{\mathcal{P}}$ as the indicator vector associated with set $\mathcal{P}$. Since the set of information matrices $\{R_i\}$ is totally ordered, we have $R(z_{\mathcal{P}}) \succeq R(z), \forall z \in \mathcal{Z}_p$. Thus, by Lemma 12, we know that $\text{trace}(\Sigma(z_{\mathcal{P}})) \leq \text{trace}(\Sigma(z))$ and $\text{trace}(\Sigma^*(z_{\mathcal{P}})) \leq \text{trace}(\Sigma^*(z)), \forall z \in \mathcal{Z}_p$, which implies that the set of sensors $\mathcal{P}$ is the optimal solution of the priori and posteriori KFSS problems.

Next we show by induction that the output of Algorithm 1 and Algorithm 2 is the set of sensors $\mathcal{P}$. Without loss of generality, let $R_1 \succeq \cdots \succeq R_q$. Then $\mathcal{P} = \{1, \cdots, p\}$. By Lemma 12, we know that $\Sigma(\{1\}) \preceq \Sigma(\{i\}), \forall i$ (resp. $\Sigma^*(\{1\}) \preceq \Sigma^*(\{i\}), \forall i$); thus, after the first loop, the output of Algorithm 1 (resp. Algorithm 2) is to choose the first sensor.

Assume that Algorithm 1 (resp. Algorithm 2) outputs the first k sensors after the k-th loop. By Lemma 12, we know that $\Sigma(\{1, \cdots, k, k+1\}) \preceq \Sigma(\{1, \cdots, k, i\}), \forall i > k$ (resp. $\Sigma^*(\{1, \cdots, k, k+1\}) \preceq \Sigma^*(\{1, \cdots, k, i\}), \forall i > k$); thus, the output of Algorithm 1 (resp. Algorithm 2) is $\{1, \cdots, k, k+1\}$ after the $(k+1)$-th loop. Thus, after the p-th loop, the final output of both algorithms is the set of sensors $\mathcal{P}$, completing the proof. □

Remark 12. *Note that in [147], the authors showed that for a given p, under the same conditions as in Proposition 6, the optimal solution of the posteriori KFSS problem is the set of p sensors with 'largest' information matrices. However, their algorithm is a special case of Algorithm 2 and they do not consider the priori KFSS problem.*

Next we consider another class of systems where the system dynamics matrix A is stable and block-diagonal and the matrices C, W and V are all block-diagonal. Specifically, consider the following block-diagonal system matrices:

$$\begin{aligned} A &= \text{diag}(A_1, \cdots, A_q), \\ W &= \text{diag}(W_1, \cdots, W_q), \\ C &= \text{diag}(C_1^d, \cdots, C_q^d), \\ V &= \text{diag}(V_1, \cdots, V_q), \end{aligned} \tag{4.18}$$

where $A_i, C_i^d, W_i, V_i \in \mathbb{R}^{s_i \times s_i}, \forall i$. Note that $C_i \in \mathbb{R}^{s_i \times n}$ and C_i^d contains s_i columns of C_i. The following result shows that Algorithm 1 and Algorithm 2 are optimal in this case.

Proposition 7. *For the priori and posteriori KFSS problems with $r = [1 \cdots 1]^T$, $\beta = p$, stable A and the system matrices A, C, W and V all being block-diagonal as specified in (4.18), Algorithm 1 is optimal for the priori KFSS problem and Algorithm 2 is optimal for the posteriori KFSS problem.*

Proof. Note that since A is stable, A_i is stable, $\forall i$, and thus any selection of sensors is feasible (in the sense of Definition 14). Further note that for this class of systems, the *a priori* and *a posteriori* error covariances Σ and Σ^* are both block-diagonal. Specifically, we have

$$\Sigma = \text{diag}(\Sigma_1, \cdots, \Sigma_q)$$

where

$$\Sigma_i = W_i + A_i \left(\Sigma_i^{-1} + z_i (C_i^d)^T V_i^{-1} C_i^d\right)^{-1} A_i^T, \forall i,$$

and

$$\Sigma^* = \text{diag}(\Sigma_1^*, \cdots, \Sigma_q^*)$$

where

$$\Sigma_i^* = \left((A_i \Sigma_i^* A_i^T + W_i)^{-1} + z_i (C_i^d)^T V_i^{-1} C_i^d\right)^{-1}, \forall i.$$

Thus, for any indicator vector z, we have

$$\text{trace}(\Sigma(z)) = \sum_i \text{trace}(\Sigma_i(z_i))$$

and

$$\text{trace}(\Sigma^*(z)) = \sum_i \text{trace}(\Sigma_i^*(z_i)).$$

We will show that Algorithm 1 is optimal in this case and the optimality of Algorithm 2 follows the same reasoning.

Define

$$\Delta_i \triangleq \text{trace}(\Sigma_i(z_i = 0)) - \text{trace}(\Sigma_i(z_i = 1))$$

which is the reduction of the *a priori* estimation error by adding sensor i. Without loss of generality, assume that $\Delta_1 \geq \cdots \geq \Delta_q$. Since $\text{trace}(\Sigma(\{1, \cdots, p\})) \leq \text{trace}(\Sigma(z)), \forall z \in \mathcal{Z}_p$, the optimal solution for the priori KFSS problem is the first p sensors. By following a similar argument as in the proof of Proposition 6, we know that after the k-th loop, the output of Algorithm 1 is the first k sensors and thus the final output of Algorithm 1 is the first p sensors which is the optimal solution, completing the proof. □

Remark 13. *Note that since both A and C are block-diagonal, the stability of A is necessary to guarantee that any selection of sensors is feasible. Further note that when the sensor noise covariance matrix V is block-diagonal (as specified in (4.18)), the sensor noises are uncorrelated; however, there does not exist a total order for the set of information matrices in general. For example, consider a system with system matrices as specified in (4.18). Then the sensor information matrices associated with the first and second sensors are* $R_1 = \text{diag}(R_1^d, 0, 0, \cdots, 0)$

and $R_2 = \mathrm{diag}(0, R_2^d, 0, \cdots, 0)$, *respectively, where* $R_i^d \triangleq (C_i^d)^T V_i^{-1} C_i^d$, *and* $R_1 - R_2 = \mathrm{diag}(R_1^d, -R_2^d, 0, \cdots, 0)$ *is indefinite unless* $R_1^d = 0$ *or* $R_2^d = 0$. *Thus, the set of instances in Proposition 6 and Proposition 7 are disjoint. Roughly speaking, the class of systems in Proposition 6 has the property that one can rank the contribution of each sensor while the class of systems in Proposition 7 possesses the property that the influence of each sensor is separable.*

4.6.2 Lack of Submodularity of the Cost Functions

Outside of the special cases discussed in Proposition 6 and Proposition 7, there are few tools available to give performance guarantees on greedy algorithms. One such tool is the concept of submodularity, which has been used in the analysis of greedy algorithms for the sensor scheduling problem, as mentioned in the beginning of this section. Now we briefly review this concept (see [83] for a comprehensive discussion) and show that the trace of the steady state *a priori* and *a posteriori* error covariances (and other related metrics) do not satisfy this property in general.

Definition 16 (Normalized and Monotone)**.** *Let E be a finite set and define the set function $f : 2^E \to \mathbb{R}$. The set function f is normalized if $f(\mathbf{0}) = 0$, and is monotone if for every $X \subseteq Y \subseteq E$, $f(X) \leq f(Y)$.*

Definition 17 (Submodularity)**.** *Consider a set E and a set function $f : 2^E \to \mathbb{R}$. The set function f is submodular if for every $X, Y \subseteq E$ with $X \subseteq Y$ and every $x \in E \setminus Y$, $f(X \cup \{x\}) - f(X) \geq f(Y \cup \{x\}) - f(Y)$, and is supermodular if $-f$ is submodular.*

An alternative way to define submodularity is through the property of *diminishing marginal return*, i.e., a set function f is submodular if for every $X, Y \subseteq E$ with $X \subseteq Y$ and every $x \in E \setminus Y$, $f(X \cup \{x\}) - f(X) \geq f(Y \cup \{x\}) - f(Y)$. The concept of submodularity is very useful in the study of combinatorial optimization problems, and the role of submodularity in discrete optimization is similar to the role of convexity in continuous optimization.

A common approach to maximize a submodular function is to use a greedy algorithm, which repeatedly chooses elements to maximize the marginal return. The performance of such a greedy algorithm is characterized as follows.

Theorem 15 ([83])**.** *If the cost function f to be maximized is normalized, monotone and submodular, then the performance of greedy algorithm is within a factor of $1 - \frac{1}{e}$ of the optimal.*

For the priori or posteriori covariance matrices induced by the set of indicator vectors in $\mathcal{Z}_p$, we will consider the problem of *maximizing* the following three performance metrics:

- $F_1(\cdot) = -\operatorname{trace}(\cdot)$
- $F_2(\cdot) = -\log\det(\cdot)$
- $F_3(\cdot) = -\lambda_1(\cdot)$

where the metric F_1 captures the mean squared error, the metric F_2 captures the volume of the confidence ellipsoid (which is the ellipsoid that contains the estimation error with a certain probability), and the metric F_3 captures the worst-case error covariance. Note that maximizing F_1 is equivalent to minimizing $-F_1$ as in the priori and posteriori KFSS problems.

In [55], the authors showed that the metric F_2 is submodular for the single-step sensor scheduling problem while F_1 and F_3 are neither submodular nor supermodular. One question of interest is whether any of these metrics is submodular or supermodular for the priori and posteriori KFSS problems. However, the following counterexamples show that these metrics are neither supermodular nor submodular in general.

Definition 18. *For metric $F_i(\Sigma)$ (resp. $F_i(\Sigma^*)$) and two sets of sensors X and Y, let the change of utility by adding Y to X be $\Delta_{F_i}(Y|X)$ (resp. $\Delta^*_{F_i}(Y|X)$), i.e., $\Delta_{F_i}(Y|X) = F_i\left(\Sigma(X \cup Y)\right) - F_i\left(\Sigma(X)\right)$ and $\Delta^*_{F_i}(Y|X) = F_i\left(\Sigma^*(X \cup Y)\right) - F_i\left(\Sigma^*(X)\right)$.*

Example 1 (Lack of submodularity of $F_i(\Sigma)$)**.** *Consider an instance of the priori KFSS problem with*

$$\begin{aligned} A &= \begin{bmatrix} 0.3 & 0.4 \\ 0.2 & 0.6 \end{bmatrix}, \quad & C &= \begin{bmatrix} 1 & 0.5 & 0.7 & 0 & 0.3 \\ 0 & 0.5 & 0.3 & 0.7 & 0.7 \end{bmatrix}^T, \\ W &= I_{2\times 2}, & V &= I_{5\times 5}, \end{aligned}$$

$s_i = 1, \forall i$, $r = [1 \cdots 1]^T$ and $\beta = 4$. Note that A is stable and thus all selections of sensors are feasible. One can check that

$$\Delta_{F_i}(\{1\}|\{2,3\}) < \Delta_{F_i}(\{1\}|\{2,3,4\}), i \in \{1,2,3\},$$

which contradicts the submodularity of the corresponding metrics, and

$$\Delta_{F_i}(\{1\}|\{2\}) > \Delta_{F_i}(\{1\}|\{2,3\}), i \in \{1,2,3\},$$

which contradicts the supermodularity of the corresponding metrics.

Example 2 (Lack of submodularity of $F_i(\Sigma^*)$)**.** *Consider the same instance in Example 1 for the posteriori KFSS problem. One can check that*

$$\Delta^*_{F_i}(\{2\}|\{1,3\}) > \Delta^*_{F_i}(\{2\}|\{1,3,4\}), i \in \{1,2,3\},$$

which contradicts the supermodularity of the corresponding metrics, and

$$\Delta^*_{F_i}(\{5\}|\{2,3\}) < \Delta^*_{F_i}(\{5\}|\{1,2,3\}), i \in \{1,3\},$$

which contradicts the submodularity of the corresponding metrics. Finally, consider another instance as follows:

$$A = \begin{bmatrix} 0.4 & 0.9 & 0.8 & 0.9 & 1.1 \\ 0.2 & 0.1 & -0.4 & 0.4 & -0.5 \\ -0.1 & 2.1 & 0.7 & 1.7 & 1.3 \\ -0.9 & -0.8 & -0.8 & -1.3 & -1 \\ 0.4 & -1.7 & 0.2 & -1.3 & -0.3 \end{bmatrix}, W = \begin{bmatrix} 6.3 & 1.6 & 1.6 & 0.9 & 0.6 \\ 1.6 & 9.2 & 1.1 & 1.6 & 1.2 \\ 1.6 & 1.1 & 6.6 & 1.5 & 1.8 \\ 0.9 & 1.6 & 1.5 & 5.5 & 1 \\ 0.6 & 1.2 & 1.8 & 1 & 5.9 \end{bmatrix},$$

$$C = \begin{bmatrix} -3 & 10 & 7 & -3 & 2 \\ -1 & -7 & 3 & -6 & 0 \\ -8 & 3 & 3 & 3 & 5 \\ -2 & 5 & -3 & 2 & -10 \\ 0 & 4 & -3 & 5 & 1 \end{bmatrix}, \qquad V = \begin{bmatrix} 2.3 & 0.2 & 0 & 0 & 0 \\ 0.2 & 0.7 & 0 & 0 & 0 \\ 0 & 0 & 2.3 & 0 & 0 \\ 0 & 0 & 0 & 2.5 & 0 \\ 0 & 0 & 0 & 0 & 2.3 \end{bmatrix},$$

$s_i = 1, \forall i$, $r = [1 \cdots 1]^T$ *and* $\beta = 4$*. Note that* A *is stable. One can check that*

$$\Delta^*_{F_2}(\{1\}|\{2\}) < \Delta^*_{F_2}(\{1\}|\{2,3,4\}),$$

which contradicts the submodularity of $F_2(\Sigma^*)$*.*

The above negative results imply that one may not be able to use classical results from combinatorial optimization to analyze Algorithm 1 and Algorithm 2; despite this, our simulations in Section 4.7 show that these greedy algorithms perform well in practice.

Remark 14. *It has been observed that greedy algorithms often perform well in practice for different types of "subset selection problem" (i.e., the problem of selecting a subset of items from the total set to optimize a certain utility function) [23, 137]; however, there are few theoretical explanations to such observation in the absence of submodularity. In [24], the authors proposed a concept termed* ***submodularity ratio*** *as a measure of "approximate submodularity" and this concept may serve as a predictor for the performance of greedy algorithms. We leave an exploration of this venue for future work.*

4.6.3 Approximated KFSS Problem

Note that due to the nonlinear nature of the DARE and equation (4.6), it is in general difficult to obtain the corresponding analytical solutions. Thus, in this subsection, we will consider an approximation for the cost functions in the priori and posteriori KFSS problems and explore its structural properties. Specifically, the solution of the DARE can be approximated by the solutions of two Lyapunov equations as follows.

Lemma 13 ([14]). *Let P_w and $P_v(z)$ satisfy the Lyapunov equations*

$$P_w = AP_wA^T + W$$

and

$$P_v(z) = AP_v(z)A^T + R(z),$$

respectively, where $R(z)$ is the sensor information matrix from (4.5). For $\Sigma(z) \succeq 0$ satisfying the DARE (4.4), we have

$$(P_v(z) + P_w^{-1})^{-1} \preceq \Sigma(z) \preceq P_v(z)^{-1} + P_w.$$

Note that since $\Sigma^*(z) = (\Sigma^{-1}(z) + R(z))^{-1} \preceq \Sigma(z)$, $P_v(z)^{-1} + P_w$ is also a matrix upper bound for $\Sigma^*(z)$. As a heuristic, we consider the problem of minimizing the upper bound $\mathrm{trace}(P_v(z)^{-1}+P_w)$ on $\mathrm{trace}(\Sigma(z))$ and $\mathrm{trace}(\Sigma^*(z))$ (rather than $\mathrm{trace}(\Sigma(z))$ and $\mathrm{trace}(\Sigma^*(z))$ themselves). Since the term P_w does not depend on the specific selection of sensors, we further approximate the problem by seeking to maximize $\mathrm{trace}(P_v(z))$.

Problem 4 (Approximated KFSS Problem)**.** *Given a system matrix $A \in \mathbb{R}^{n\times n}$, a measurement matrix $C \in \mathbb{R}^{s\times n}$, a sensor noise covariance matrix $V \in \mathbb{S}^s_+$ and the number of sensors to be chosen p, the approximated KFSS problem is to solve the following optimization problem:*

$$\begin{aligned} \max_{z} \quad & \mathrm{trace}(P_v(z)) \\ \textit{s.t.} \quad & 1^T z \leq p \\ & z \in \{0,1\}^q \end{aligned} \tag{4.19}$$

where $P_v(z)$ is defined in Lemma 13.

Here we propose the following Lyapunov equation based greedy algorithm, given as Algorithm 3. Note that in order to guarantee that the Lyapunov equation has a feasible solution, we assume that the system matrix A is stable. We show that the cost function of the approximated KFSS problem is modular when the measurement noises of the sensors are uncorrelated, which indicates that the contribution of adding each sensor is separable, and thus Algorithm 3 provides optimal performance for the approximated KFSS problem in this case. Note that the proof of Theorem 16 is a relatively straightforward extension of the results in [131] on the continuous-time Lyapunov equation to the discrete-time case.

Algorithm 3 Lyapunov Equation based Greedy Algorithm

```
Input: System dynamics matrix $A$, set of all sensors $\mathcal{Q}$, noise covariance $V$, and constant $p$
Output: A set $\mathcal{S}$ of chosen sensors
$k \leftarrow 0, \mathcal{S} \leftarrow \emptyset$
for $k \leq p$ do
    for $i \in \mathcal{Q} \cap \bar{\mathcal{S}}$ do
        Solve $\mathrm{trace}(P_v^{i,\mathcal{S}}) = \mathrm{trace}(P_v(\mathcal{S} \cup \{i\}))$
    end for
    Choose $j$ with $\mathrm{trace}(P_v^{j,\mathcal{S}}) = \max_i \mathrm{trace}(P_v^{i,\mathcal{S}})$
    $\mathcal{S} \leftarrow \mathcal{S} \cup \{j\}$, $\mathcal{Q} \leftarrow \mathcal{Q} \setminus \{j\}$, $k \leftarrow k + 1$
end for
```

Lemma 14 ([83]). *A set function $f : 2^{\mathcal{Q}} \to \mathbb{R}$ is modular if $\forall \mathcal{S} \subset \mathcal{Q}$, $f(\mathcal{S}) = f(\mathbf{0}) + \sum_{s \in \mathcal{S}} f(s)$.*

Theorem 16. *Let $R_{\mathcal{S}} = C_{\mathcal{S}}^T V_{\mathcal{S}}^{-1} C_{\mathcal{S}}$ be the sensor information matrix associated with the set $\mathcal{S}$ of chosen sensors and let $P_v(\mathcal{S})$ be the solution of $P_v = AP_vA^T + R_{\mathcal{S}}$. Define $\mathrm{trace}(P_v(\mathbf{0})) = 0$. If the matrix A is stable and the measurement noises of the sensors are uncorrelated and $V \succ 0$, then the cost function $\mathrm{trace}(P_v(\mathcal{S}))$ is normalized, monotone and modular.*

Proof. For any $\mathcal{S} \subset \mathcal{Q}$, since the covariance matrix V is block diagonal and $V \succ 0$, we know that $R_{\mathcal{S}}$ can be decomposed as $R_{\mathcal{S}} = \sum_{s \in \mathcal{S}} C_s^T V_s^{-1} C_s$ where C_s and V_s are the rows of C and measurement noise covariance matrix corresponding to sensor s, respectively. Thus, we get

$$\begin{aligned}P_v(\mathcal{S}) &= \sum_{k=0}^{\infty} A^k C_{\mathcal{S}}^T V_{\mathcal{S}}^{-1} C_{\mathcal{S}} (A^T)^k \\ &= \sum_{k=0}^{\infty} A^k \left(\sum_{s\in\mathcal{S}} C_s^T V_s^{-1} C_s \right) (A^T)^k \\ &= \sum_{s\in\mathcal{S}} \sum_{k=0}^{\infty} A^k C_s^T V_s^{-1} C_s (A^T)^k \\ &= \sum_{s\in\mathcal{S}} P_v(s).\end{aligned}$$

Note that the first and last equalities are due to the analytic solution of the discrete-time Lyapunov equation [45]. Since the trace operator is a linear function and $\text{trace}(P_v(\mathbf{0})) = 0$, by using Lemma 14, we know that the function $\text{trace}(P_v(\mathcal{S}))$ is normalized, monotone and modular. □

Corollary 2. *Let the set of sensors chosen by Algorithm 3 be $\mathcal{S}$ and the optimal solution of the approximated KFSS problem be $\text{trace}(P_v^{opt})$. If the matrix A is stable and the measurement noises of the sensors are uncorrelated and $V \succ 0$, then $\text{trace}(P_v(\mathcal{S})) = \text{trace}(P_v^{opt})$.*

4.7 Simulation

In this section, we provide simulation results for the performance of the DARE based greedy algorithm (Algorithm 1), the posteriori covariance based Greedy Algorithm (Algorithm 2), and the Lyapunov equation based greedy algorithm (Algorithm 3) and discuss their complexity.

4.7.1 Performance Evaluation

In order to illustrate the performance of greedy algorithms considered in this chapter, we will compare them with the following sensor selection strategies:

- *Sparse optimization (abbr.: SparseOpt)* approach for the priori KFSS problem from [26, 109]. Sparsity is achieved by adding a penalty function on the columns of the gain matrix. Since there is in general no systematic method to choose the weight of the penalty function, we fix the weight in the simulations and select the sensors corresponding to the columns of the gain matrix with largest l_1 norm.

- *Priori convex relaxation (abbr.: PriConRe)* approach for the priori KFSS problem from [147]. Note that we modify the algorithm in [147] for packet-dropping channels to handle the case of reliable channels (corresponding to the priori KFSS problem considered in this chapter).

- *Posteriori convex relaxation (abbr.: PostConRe)* approach for the posteriori KFSS problem from [147].

- *A random selection (abbr.: Random)* of sensors for both the priori and posteriori KFSS problems. We use this as a benchmark.

We consider two cases: the system is stable and the system is unstable but detectable. For the first case, we randomly generate 300 stable systems all having dimension 5 (i.e., $n = 5$). For each system, the goal is to choose 5 sensors out of a total of 20 (i.e., $p = 5$, $q = 20$, $r = [1 \cdots 1]^T$ and $\beta = p$) and the measurement of each sensor is a scalar (i.e., $s_i = 1, \forall i$). The results are summarized in Table 4.1.

For the second case, we also randomly generate 300 systems. For each system, the system matrix A is unstable and the pair (A, C_i) is detectable, $\forall i \in \{1, \cdots, q\}$. The other parameters are the same (i.e., $n = 5$, $p = 5$, $q = 20$, $r = [1 \cdots 1]^T$, $\beta = p$ and $s_i = 1, \forall i$). The results are summarized in Table 4.2.

From Table 4.1a and Table 4.2a, we see that for the priori KFSS problem, the priori convex relaxation approach from [147] provides a set of sensors with smaller trace than the other algorithms in a plurality of cases. However, this algorithm also exhibits larger variance than the other algorithms (with high worst case deviation from optimality). On average, the sparse optimization approach from [26, 109] outperforms all the other algorithms, and this approach has the smallest variance. As illustrated in Table 4.2a and Table 4.2a, Algorithm 1 exhibits comparable average performance to the other algorithms.

From Table 4.1b and Table 4.2b, we see that Algorithm 2 and the posteriori convex relaxation approach from [147] each outperforms the other in a comparable number of cases. However, once again, Algorithm 2 provides more consistent results with better average performance.

From Table 4.1a and Table 4.1b, we see that in general Algorithm 3 performs worse than the other algorithms. Although the performance of Algorithm 3 is not appealing, as we will show in the next subsection, the algorithm is more efficient than the other algorithms.

To summarize, the *a priori* and *a posteriori* error covariance based greedy algorithms have comparable performance with the other sensor selection algorithms in general. Moreover, as we have argued in Section 1.4, compared to the priori and posteriori convex relaxation approaches in [147], greedy algorithms can be applied to a more general class of systems where the sensor noise covariance matrix V is not necessarily block-diagonal.

Table 4.1: Performance comparison of different algorithms over 300 randomly generated **stable** systems with scalar measurements and diagonal V. For Algorithm $\mathcal{A}$, the table presents the **average**, **standard deviation** and **worst-case values** of $\frac{\text{trace}(\Sigma_{\mathcal{A}})}{\text{trace}(\Sigma_{\text{opt}})}$ (in 4.1a) and $\frac{\text{trace}(\Sigma^*_{\mathcal{A}})}{\text{trace}(\Sigma^*_{\text{opt}})}$ (in 4.1b) over the 300 runs. The last column presents the percentage of the 300 systems for which the corresponding algorithm **outperforms all the other algorithms**.

(a) Priori KFSS Problem

	Average	Standard Deviation	Worst	Outperforms Other Algorithms
Algorithm 1	1.2	0.8	7.4	12%
Algorithm 3	2.4	6.3	59.5	9.5%
PriConRe	1.3	2.2	31.6	46%
SparseOpt	1.1	0.3	4.1	32.5%
Random	16.5	81.6	1020.8	0%

(b) Posteriori KFSS Problem

	Average	Standard Deviation	Worst	Outperforms Other Algorithms
Algorithm 2	3.6	2.5	21.0	36.5%
Algorithm 3	31.5	49.0	305.4	12%
PostConRe	12.1	25.0	185.8	49%
Random	51.0	60.6	334.0	2.5%

Table 4.2: Performance comparison of different algorithms over 300 randomly generated **unstable** systems with scalar measurements and diagonal V. For Algorithm $\mathcal{A}$, the table presents the **average**, **standard deviation** and **worst-case values** of $\frac{\mathrm{trace}(\Sigma_{\mathcal{A}})}{\mathrm{trace}(\Sigma_{\mathrm{opt}})}$ (in 4.2a) and $\frac{\mathrm{trace}(\Sigma^*_{\mathcal{A}})}{\mathrm{trace}(\Sigma^*_{\mathrm{opt}})}$ (in 4.2b) over the 300 runs. The last column presents the percentage of the 300 systems for which the corresponding algorithm **outperforms all the other algorithms**.

(a) Priori KFSS Problem

	Average	Standard Deviation	Worst	Outperforms Other Algorithms
Algorithm 1	1.4	1.3	13.5	11.3%
PriConRe	1.5	2.7	41.3	47.0%
SparseOpt	1.1	0.3	4.7	41.7%
Random	11.3	26.0	307.3	0%

(b) Posteriori KFSS Problem

	Average	Standard Deviation	Worst	Outperforms Other Algorithms
Algorithm 2	4.6	3.9	37.2	49.7%
PostConRe	17.8	52.4	600.7	50.3%
Random	55.0	46.5	456.7	0%

4.7.2 Complexity Analysis

To compare the complexity of the previous algorithms, note that the complexity of solving the DARE and the Lyapunov equation are $O(n^3)$ and $O(n^2)$, respectively, where n is the number of states [25]. If we aim to choose p sensors from a set of q sensors, then the complexity of Algorithm 1 and Algorithm 3 are $O(pqn^3)$ and $O(pqn^2)$, respectively. Since we can obtain Σ^* from Σ by equation (4.8), the complexity of Algorithm 2 is also $O(pqn^3)$.

As argued in [26], when the weight of the sparsity penalty function is *fixed*, the complexity of the sparse optimization approach is $O((n+s)^6)$ by using the interior point method[6] (recall that $s = \sum_{i=1}^{q} s_i$ is the dimension of the combined output y); however, the process of choosing an appropriate weight for the sparsity penalty function (in order to obtain the desired level of sparsity) requires additional computation. Moreover, the complexities of the priori and the posteriori convex relaxation approaches from [147] are both $O((n+s)^6)$ by using the interior point method [13].

Thus, the complexity of greedy algorithms is lower than those of the other SDP based approaches. Figure 4.1 shows simulations that support this conclusion. Note that the simulation is conducted on a typical 2.4-GHz personal computer, the goal is to choose 5 sensors out of 20 (i.e., $p = 5$, $q = 20$, $r = [1 \cdots 1]^T$ and $\beta = p$) and we take the measurement of each sensor to be a scalar (i.e., $s_i = 1, \forall i$). Further note that we found our solver ran out of memory when the number of states n exceeded 50 for the SDP based approaches.

4.8 Summary

In this chapter, we studied the Kalman filtering based design-time sensor selection problem. We showed that it is NP-hard to find the optimal solutions for both the priori and posteriori KFSS problems. Then by studying the ratios between the worst-case and optimal selections, we provided insights into the factors that affect the performance of sensor selection algorithms. Finally, we investigated greedy algorithms and corresponding variant for the priori and posteriori KFSS problems. Although the cost functions in the priori and posteriori KFSS problems do not possess certain modularity properties in general, the simulations indicated that greedy algorithms perform well in practice. Moreover, compared to other sensor selection strategies, greedy algorithms are more efficient.

[6] In [26], the authors present a customized algorithm to reduce the complexity to $O(n^6)$.

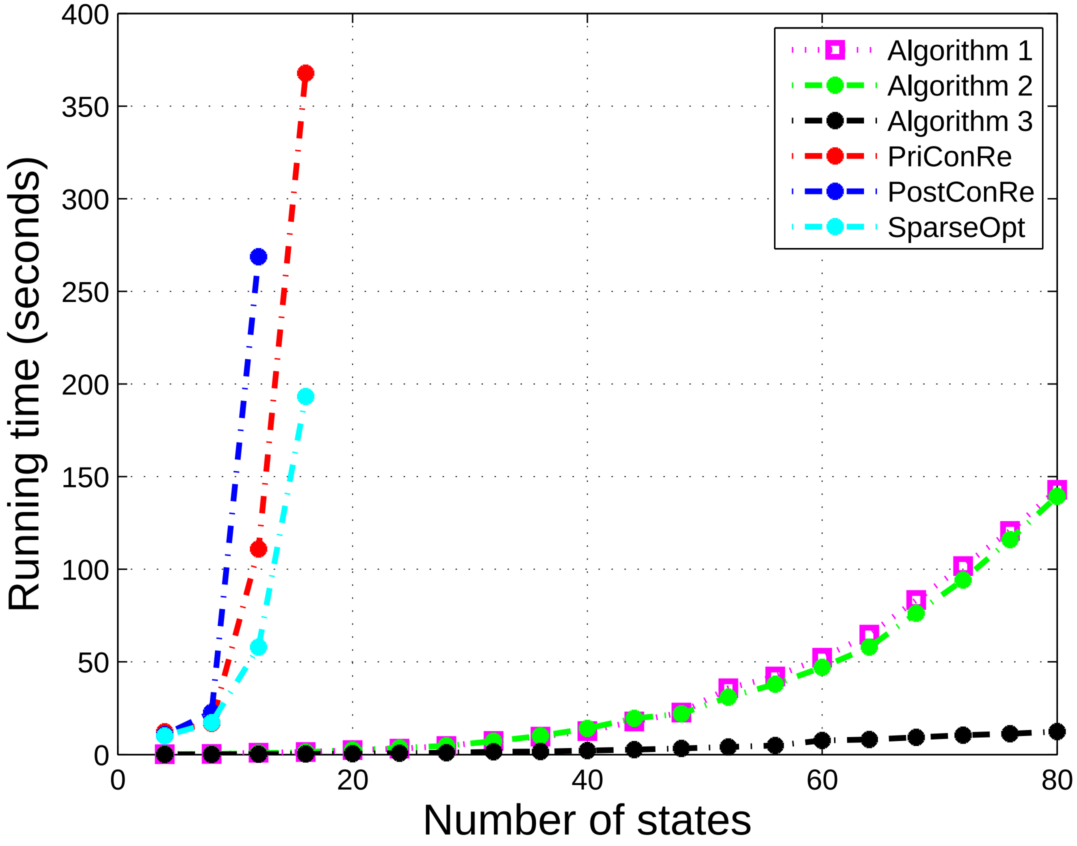


Figure 4.1: Complexity comparison of different algorithms. The x-axis is the number of states n and the y-axis is the running time of the algorithm.

Chapter 5

Output Tracking for Nonlinear Dynamical Systems

5.1 Introduction

A fundamental design problem in system and control engineering is to steer the state of the system to track a specified time-varying reference signal. For example, in the control of CPU-GPU subsystem in mobile platforms, the main objective is to drive the system output to track some specified number of Frames Per Second (FPS) [59, 110]. Other applications include trajectory tracking for mobile robots [87], altitude modification in flight control [80, 112] and behavior tracking in biomedical systems [16, 19].

Among various approaches for the tracking problem, the model predictive control (MPC) based framework has drawn much attention due to its capability to handle hard constraints on the system (i.e., constraints on the system states and inputs) and satisfactory performance in practice [12, 32, 37, 76, 115]. A major issue in the implementation of MPC is the need to solve an optimization problem online iteratively, which requires a large amount of online computation resources.

In order to address this issue, the so called explicit MPC (EMPC) approach has been proposed, and the idea is to move most of the online computations offline. However, most of the works on EMPC focus on linear systems and the study on nonlinear EMPC is far from mature, where the prospects of EMPC are even higher [38, 115]. Existing studies on nonlinear EMPC normally involve partitioning the feasible regions into boxes or hypercubes (e.g., the approaches based on convex multi-parametric nonlinear programming [8, 57]) and thus the complexity may grow exponentially as the problem size increases in the worst-case, which prohibits its application to systems with very limited online computational and storage resources (e.g., the multiprocessor subsystem in mobile platforms).

In this chapter, we study the nonlinear tracking problem in the absence of exact knowledge on the reference signal at design time. Instead, we assume that the reference signals are contained in some bounded set. In order to reduce the online computational and storage complexity and provide scalability to higher dimensional state spaces, we extend the sampling-based EMPC regulator in [17] to the tracking problem by regarding the reference as an extra parameter. Instead of using the common method of partitioning the feasible region into critical subregions, the basic idea of the sampling based approach is to sample the state and reference signal space using deterministic sampling [17] and construct the EMPC by using sparse-sample based regression of orthogonal polynomial basis functions.

Based on the results for the nominal system when there is no uncertainty, we also provide an extension of the sampling based EMPC for the case where there is an additive bounded disturbance. Utilizing the same idea as in [88, 89] on tube-based robust control parametrization, we propose a sampling based robust EMPC which consists of a nominal controller and an ancillary controller, and the output of the system is guaranteed to stay in a certain neighborhood of the nominal trajectory provided that the disturbance is small enough.

The rest of this chapter is organized as follows. In Section 5.2, we review the model predictive control techniques. In Section 5.3, we present the class of systems that we are considering and formally discuss the output tracking problem. In Section 5.4, we present the sampling based ENMPC approach for output tracking, and provide stability and feasibility guarantees for the proposed methodology in Section 5.5. In Section 5.6, we extend the nominal ENMPC to a robust variant in order to attain robustness against a class of additive disturbances. In Section 5.7, we illustrate the performance and efficiency of the proposed approaches with simulations. Finally, some concluding remarks are given in Section 5.8.

5.2 Background: Model Predictive Control

In this section, we provide some background on model predictive control for the tracking problem. Since the theory on nonlinear MPC is not as well developed as linear MPC, we use linear MPC to illustrate the corresponding techniques.

Consider the following discrete-time linear time-invariant system:

$$\begin{aligned} x[k+1] &= Ax[k] + Bu[k] \\ y[k] &= Cx[k], \end{aligned} \tag{5.1}$$

where $x \in \mathbb{R}^n$ is the state, $y \in \mathbb{R}^p$ is the output and $u \in \mathbb{R}^m$ is the input. Suppose that the system is subject to state and input constraints such that $x[k] \in \mathbb{X}, u[k] \in \mathbb{U}, \forall k$, where $\mathbb{X}$ and $\mathbb{U}$ are the state and input constraint sets, respectively.

In order for the output of system (5.1) to track a constant reference y_r without any steady state offset (i.e., $\lim_{k\to\infty} \|y[k] - y_r\| = 0$), there must exist a pair of steady state and associated constant input $(x_r, u_r) \in \mathbb{X} \times \mathbb{U}$ such that

$$\begin{aligned} x_r &= Ax_r + Bu_r \\ y_r &= Cx_r. \end{aligned}$$

The following result characterizes the condition under which the system is capable to track any constant reference in the absence of state and input constraints.

Lemma 15 ([107]). *For system (5.1) without state and input constraints, there exists a pair of offset-free steady state and associated constant input (x_r, u_r) for any set-point if and only if*

$$\operatorname{rank}\left(\begin{bmatrix} I - A & B \\ C & 0 \end{bmatrix}\right) = n + p. \tag{5.2}$$

Remark 15. *Note that the condition (5.2) implies that the number of controlled variables cannot exceed either the number of states or the number of control inputs (i.e., $p \leq \min\{n, m\}$).*

The basic idea of MPC is to use a model of the system to predict its future evolution and apply the control law in a receding horizon fashion. At each time-step, a certain constrained optimization problem is solved (for a specified prediction horizon) and the obtained input sequence will only be applied for the immediately following time-step; then at the next time-step, a new optimal control problem based on up-to-date measurements of the state is solved over a shifted horizon. Specifically, for a given reference y_r and the associated constant state and input (x_r, u_r), at each time-step k, we solve the following problem:

$$\begin{aligned} \min_{u} \quad & \|x_{k+N_p} - x_r\|_P^2 + \sum_{t=k}^{k+N_p-1} \left(\|x_t - x_r\|_Q^2 + \|u_t - u_r\|_R^2\right) \\ \text{s.t.} \quad & x_{t+1} = Ax_t + Bu_t, \\ & x_k = x[k], \\ & x_t \in \mathbb{X}, \forall t \in [k, k+N_p], \\ & u_t \in \mathbb{U}, \forall t \in [k, k+N_p], \\ & x_{k+N_p} \in \mathbb{X}_f, \end{aligned} \tag{5.3}$$

where $P \succ 0$, $Q \succ 0$ and $R \succ 0$ are symmetric weighting matrices and $\mathbb{X}_f$ is the terminal set. Note that here we denote the actual values by $(x[k], u[k])$ and the predicted values by (x_k, u_k).

Typically the terminal weighting matrix P is chosen to be the solution of the following problem:

$$\begin{aligned} \max_{P,Q,R} \quad & \operatorname{trace}(P) \\ \text{s.t.} \quad & P = Q + A^T(P - PB(B^TPB + R)^{-1}B^TP)A. \end{aligned} \tag{5.4}$$

Note that here we allow the state cost Q and input cost R to be design parameters. If the costs Q and R are fixed, the terminal cost P is just given by the solution of the corresponding discrete algebraic Riccati equation (DARE) (i.e., the constraint in Problem (5.4)).

Denote K to be the corresponding Kalman gain matrix (i.e., $K = -(R + B^TPB)^{-1}B^TPA$) and define Ω to be the maximal positively invariant set corresponding to the input $u = Kx$ [61]. For a given set-point $y_r \in \mathcal{Y}_r$, the terminal set $\mathbb{X}_f$ can be chosen to be $\mathbb{X}_f = \{x_r\} \oplus \Omega$ where x_r is the corresponding steady state.

Let the optimal solution of Problem (5.3) be $U_k^* = \begin{bmatrix}(u_k^*)^T \cdots & (u_{k+N_p-1}^*)^T\end{bmatrix}^T$. Then we apply the first sample of the obtained optimal control sequence to the system (i.e., apply $u[k] = u_k^*$) and repeat the procedure for the next time-step $k+1$ based on the new measurement of the state $x[k+1]$.

One disadvantage of the MPC approach is that we need to solve an optimization problem online and this may not be practical especially when we have fast dynamics or limited online computation resources. For example, in the multi-processor systems, the online computation and storage resources are very limited.

To reduce the computational complexity, in [7, 9], the authors proposed the explicit MPC (EMPC) approach. The basic idea of EMPC is to provide an *explicit* form of the control inputs in terms of the state (and reference signal in the tracking problem) and use this explicit solution as a control look-up table online. Define the sets of all possible steady states and inputs to be $\mathcal{X}$ and $\mathcal{U}$, respectively. For a given time-step k, the optimal control law $u_k^*(x[k], x_r, u_r)$ is a piecewise affine function of $x[k]$, x_r and u_r, defined over a polyhedral partition of the feasible parameters. In other words, the optimal control law has the following form:

$$u_k^*(x[k], x_r, u_r) = \begin{cases} H_1\tilde{x}_k + h_1, & \text{if } \tilde{x}_k \in \mathcal{P}_1 \triangleq \{\tilde{x} | L_1\tilde{x} \le l_1\} \\ \quad\vdots & \qquad\vdots \\ H_{N_r}\tilde{x}_k + h_{N_r}, & \text{if } \tilde{x}_k \in \mathcal{P}_{N_r} \triangleq \{\tilde{x} | L_{N_r}\tilde{x} \le l_{N_r}\} \end{cases} .$$

where $\tilde{x}_k = \begin{bmatrix} x[k] \\ x_r \\ u_r \end{bmatrix}$ and $\{\mathcal{P}_i\}_{i=1}^{N_r}$ form a polyhedral partition of $\mathbb{X} \times \mathcal{X} \times \mathcal{U}$ [12].

Although the explicit MPC approach can provide the optimal control law with lightweight online computation, the storage complexity (i.e., the number of regions N_r over which the control law is defined) may grow exponentially in the worst-case, which limits its application to relatively small problems (e.g., one/two inputs, up to five-ten states and three/four free control moves) [145].

5.3 Problem Formulation

In this section, we formally formulate the nonlinear output tracking problem studied in this chapter. We consider a class of discrete-time nonlinear systems, modeled by

$$\begin{aligned} x[k+1] &= f(x[k], u[k]) \\ y[k] &= h(x[k], u[k]) \end{aligned} \tag{5.5}$$

where $x[k] \in \mathbb{R}^n$ is the state, $y[k] \in \mathbb{R}^p$ is the output and $u[k] \in \mathbb{R}^m$ is the input. We assume that the functions f and h are continuously differentiable and $(0, 0)$ is an equilibrium of the system (i.e., $f(0, 0) = 0$).

The system (5.5) is subject to the following constraints:

$$\begin{aligned} x[k] &\in \mathbb{X}, \\ u[k] &\in \mathbb{U}, \end{aligned}$$

for any $k \geq 0$, where the state constraint set $\mathbb{X} \subseteq \mathbb{R}^n$ and the control constraint set $\mathbb{U} \subseteq \mathbb{R}^m$ are convex, compact and contain the origin in their interiors.

We assume that the reference signal y_{ref} takes values within a compact set $\mathbb{Y}_{\text{ref}} \subseteq \mathbb{R}^p$. We also assume that the designer has information regarding the bounds on the values y_{ref} can take, but not the reference signal itself.

For each set-point $y_r \in \mathbb{Y}_{\text{ref}}$, similar to the linear tracking problem, it is desirable to find a pair of steady state and steady input $(x_r, u_r) \in \mathbb{X} \times \mathbb{U}$ such that

$$\begin{aligned} x_r &= f(x_r, u_r) \\ y_r &= h(x_r, u_r). \end{aligned} \tag{5.6}$$

However, such steady state and input (x_r, u_r) may not exist or may not be reachable under the state and input constraints. In such scenarios, we obtain a pair of (x_r, u_r) for the reference value y_r by solving the following problem:

$$\begin{aligned}
(x_r^\star, u_r^\star) = \arg\min_{x_r, u_r} & \; \|y_r - h(x_r, u_r)\|_{Q_r}^2 \\
\text{s.t.} \quad & x_r = f(x_r, u_r), \\
& x_r \in \mathbb{X}, \\
& u_r \in \mathbb{U},
\end{aligned} \tag{5.7}$$

where $Q_r \succ 0$ is a symmetric weighting matrix. In other words, for a given set-point, we steer the output of the system to the closest admissible steady output $y_r^* = h(x_r^\star, u_r^\star)$ while fulfilling the state and input constraints.

Remark 16. *Note that since $(0, 0)$ is an equilibrium of the system, and $\mathbb{X}$ and $\mathbb{U}$ are convex compact sets containing the origin in their interiors, Problem (5.7) is always feasible. When the solution of Problem (5.7) is not unique, we can choose the one with smallest $\|u_r^\star\|$ which corresponds to the steady state input with smallest energy.*

For the system (5.5), our objective is to design an ENMPC controller which guarantees recursive feasibility and stability with respect to the desired reference.

5.4 Sampling based Nominal ENMPC

In this section, we present the sampling based ENMPC approach for the nonlinear tracking problem. The basic idea of the sampling based ENMPC is to sample the augmented space $\mathbb{X} \times \mathbb{Y}_{\text{ref}}$ (i.e., regard both the state and the reference as parameters). At each sampling point, we solve a constrained optimization problem to obtain the corresponding optimal nonlinear MPC control input. Then we construct the ENMPC control surface using linear regression with a pre-defined set of tensored polynomial basis functions.

5.4.1 Sampling Scheme

To guarantee that the samples are distributed sufficiently evenly, we use the following notion of a low-discrepancy sequence [17, 62].

Definition 19. *The* ***discrepancy*** *of a sequence* $\{z(i)\}_{i=1}^N \subset \mathcal{Z} \subset \mathbb{R}^{n_z}$ *is defined as*

$$D_N \triangleq \sup_{Z \in \mathcal{I}} \left| \frac{\#Z_N}{N} - \frac{\mathrm{Vol}(Z)}{\mathrm{Vol}(\mathcal{Z})} \right|,$$

where $\mathcal{I} \subset \mathcal{Z}$ *is the set of* n_z*-dimensional intervals of the form*

$$\prod_{j=1}^{n_z} [a_j, b_j) = \{z \in \mathcal{Z} | a_j \leq z_j < b_j\}$$

and $\#Z_N$ *is defined as* $\#Z_N \triangleq \mathrm{Card}\{i \in \{1, \cdots, N\} | z(i) \in Z\}$*. A sequence* $\{z(i)\}_{i=1}^N$ *is said to be* ***low-discrepancy*** *if* $\lim_{N\to\infty} D_N = 0$.

In order to obtain low-discrepancy sequences, we adopt the quasi-random sampling scheme where the set of N samples $\{z(i)\}_{i=1}^N$ are drawn from the Halton sequence [17,62]. Note that one can also use other sampling schemes such as Sobol sequences or multi-level sparse grids [17].

5.4.2 ENMPC Design

At each sampling point $z(i) = (x(i), y_r(i)) \in \mathbb{X} \times \mathbb{Y}_{\text{ref}}$, we determine a pair of steady augmented state pair $(x_r(i), u_r(i))$ for the set-point $y_r(i)$ by solving Problem (5.7). Then we solve the following finite-horizon output tracking problem:

$$\begin{aligned} \min_{u} \quad & V_{N_p}(x(i), x_r(i), u_r(i)) \\ \text{s.t.} \quad & x_{k+1} = f(x_k, u_k), \\ & x_0 = x(i), \\ & x_k \in \mathbb{X}, \quad \forall k \in [0, N_p], \\ & u_k \in \mathbb{U}, \quad \forall k \in [0, N_p], \\ & x_{N_p} \in \mathbb{X}_f, \end{aligned} \tag{5.8}$$

where

$$V_{N_p}(x(i), x_r(i), u_r(i)) \triangleq \|x_{N_p} - x_r(i)\|_P^2 + \sum_{k=0}^{N_p-1} (\|x_k - x_r(i)\|_Q^2 + \|u_k - u_r(i))\|_R^2),$$

$P \succ 0$, $Q \succ 0$ and $R \succ 0$ are symmetric weighting matrices, N_p is the prediction horizon and $\mathbb{X}_f$ is the terminal set. In Section 5.5, we will discuss the properties that the cost matrices and terminal set should satisfy to guarantee stability and feasibility.

Remark 17. *Note that since we do not have exact knowledge of the reference signal (i.e., how the reference varies during the predictive horizon), we solve the problem* (5.8) *by fixing the reference* y_r *over the entire predictive horizon.*

Let the first control action of the optimal control sequence obtained by solving Problem (5.8) be $u^*(z)$. Our aim is to approximate the optimal control law u^* by using linear regression via tensored polynomial basis functions. Let such an approximation be $\tilde{u}(z)$. Then, we obtain a feasible controller $\hat{u}(z)$ by projecting $\tilde{u}(z)$ onto the constraint set $\mathbb{U}$.

Specifically, similar to the regression approach in [17], for a set of M orthogonal polynomial basis functions (e.g., Chebyshev polynomials or Legendre polynomials [17]) $\{\mathbb{B}_i(z)\}_{i=1}^M \subset \mathbb{R}^{n+p}$, we express $\tilde{u}(z)$ as a linear combination of the basis functions as follows:

$$\tilde{u}(z,\alpha) = \sum_{i=1}^{M} \alpha_i \mathbb{B}_i(z), \tag{5.9}$$

where $\alpha_i \in \mathbb{R}^{m\times(n+p)}$ is the coefficient matrix. We will use the set of optimal control inputs $\{u^*(z(i))\}_{i=1}^N$ at the sampling points to get the coefficients $\{\alpha_i\}$.

Note that in order to obtain the orthogonal polynomial basis functions, we can use the tensor product based construction which consists of the set of mutual products of the one-dimensional polynomials whose total degree is less than or equal to some desired order. For example, the set of 3-degree Legendre polynomials $\mathcal{L}_3$ is

$$\mathcal{L}_3 = \left\{ p_{j_1}(z_1) \times p_{j_2}(z_2) \times \cdots \times p_{j_{n+p}}(z_{n+p}) | j_1 + j_2 + \cdots + j_{n+p} \le 3 \right\},$$

where z_s is the s-th element of the vector z, and p_j is the j-th one dimensional polynomial such that

$$\begin{aligned} p_0(z_s) &= 1, \\ p_1(z_s) &= z_s, \\ p_2(z_s) &= \frac{1}{2}(3z_s^2 - 1), \\ p_3(z_s) &= \frac{1}{2}(5z_s^3 - 3z_s). \end{aligned}$$

Define the matrices

$$
\begin{aligned}
G &\triangleq \begin{bmatrix} \mathbb{B}_1(z(1)) & \cdots & \mathbb{B}_1(z(N)) \\ \vdots & \ddots & \vdots \\ \mathbb{B}_M(z(1)) & \cdots & \mathbb{B}_M(z(N)) \end{bmatrix}, \\
g &\triangleq \begin{bmatrix} u^*(z(1)) & \cdots & u^*(z(N)) \end{bmatrix}, \\
\alpha &\triangleq \begin{bmatrix} \alpha_1 & \cdots & \alpha_M \end{bmatrix}.
\end{aligned}
$$

We can obtain the optimal coefficient α^* by solving the following problem with some prespecified constant $\bar{\alpha} > 0$:

$$
\alpha^* = \arg\min_{\alpha} \{\|\alpha G - g\|_\infty : \|\alpha\|_\infty \leq \bar{\alpha}\}. \tag{5.10}
$$

Note that the parameter $\bar{\alpha}$ is used to reduce the sensitivity of the regression surface to approximation errors and to limit the number of bits required to store the coefficients. Then the corresponding control law is $\tilde{u}(z, \alpha^*)$, with $\tilde{u}$ defined in (5.9).

Note that in general we may have $\tilde{u}(z) \notin \mathbb{U}$; in this case, we can choose $\hat{u}(z)$ to be the closest point to $\tilde{u}(z)$ which satisfies the input constraint. Thus, the sampling based ENMPC control law $\hat{u}(z)$ is constructed as follows:

$$
\hat{u}(z) = \arg\min_{u \in \mathbb{U}} \left\| \sum_{i=1}^{M} \alpha_i^* \mathbb{B}_i(z) - u \right\|. \tag{5.11}
$$

Remark 18. *Note that if the input constraint set $\mathbb{U}$ is in the form of a product of closed intervals, we can apply the above projection componentwise.*

The design of the sampling based nominal ENMPC is summarized in Algorithm 4.

5.5 Feasibility and Stability Analysis

In this section, we analyze the stability and feasibility properties of the proposed sampling based ENMPC. Note that for each set-point y_r, the corresponding steady state and input (x_r, u_r) only appear in the cost function of Problem (5.8) (which is the problem of solving for the optimal control inputs) and the constraints are independent of y_r. Thus, the feasibility region of Problem (5.8) is independent of y_r, as argued in [31].

Suppose that there exists a subregion $\mathbb{X}_{\text{feas}} \subseteq \mathbb{X}$ such that for all $x \in \mathbb{X}_{\text{feas}}$ and $y_r \in \mathbb{Y}_{\text{ref}}$, Problem (5.8) is feasible. Methods for obtaining an estimate of $\mathbb{X}_{\text{feas}}$ using samples can be found in [17, 106], and the references therein. We make the following assumptions.

Algorithm 4 [Sampling based Nominal ENMPC]

```
Offline Computations
Input: System functions $f$ and $h$, constraints $\mathbb{X}$ and $\mathbb{U}$, reference constraint $\mathbb{Y}_{\text{ref}}$, number of samples $N$, number of basis functions $M$ and upper bound on coefficients $\bar{\alpha}$
Output: Coefficient $\alpha^*$
Sample $\mathbb{X} \times \mathbb{Y}_{\text{ref}}$ using a low-discrepancy sequence
For each sampling point $z(i) = (x(i), y_r(i))$, solve Problem (5.8)
Solve Problem (5.10) to get the coefficients $\{\alpha_i^*\}$

Online Computations
Input: Coefficient $\alpha^*$ and input constraint $\mathbb{U}$
Output: Control input $u$
At each time-step $k \geq 0$:
Obtain the state $x[k]$ and reference $y_{\text{ref}}[k]$
Compute values of the basis functions $\{\mathbb{B}_i(z)\}_{i=1}^M$ where $z = (x[k], y_{\text{ref}}[k])$
$u[k] = \arg\min_{u \in \mathbb{U}} \| \sum_{i=1}^M \alpha_i^* \mathbb{B}_i(z) - u \|$
```

Assumption 1. *Recall that $u^*(x, y_r)$ is the optimal control law obtained by solving (5.8).*

(i) *The feasible region $\mathbb{X}_{feas}$ and the terminal set $\mathbb{X}_f$ are open and there exists a compact set $\mathbb{X}_{attr}$ such that $\mathbb{X}_f \subset \mathbb{X}_{attr} \subset \mathbb{X}_{feas}$.*

(ii) *For any $y_r \in \mathbb{Y}_{ref}$, there exist a pair of steady state and input $(x_r, u_r) \in \mathbb{X}_f \times \mathbb{U}$ such that (5.6) is satisfied, and the linearization of the function $f(\cdot, \cdot)$ at (x_r, u_r) is stabilizable. .*

(iii) *For every $y_r \in \mathbb{Y}_{ref}$, $u^*(x, y_r)$ is continuously differentiable on $\mathbb{X}_{feas}$.*

(iv) *For every $y_r \in \mathbb{Y}_{ref}$, the control constraints are satisfied inside the terminal set (i.e., $u^*(x, y_r) \in \mathbb{U}, \forall x \in \mathbb{X}_f$) and $\mathbb{X}_f$ is positively invariant under $u^*(x, y_r)$ (i.e., $f(x, u^*(x, y_r)) \in \mathbb{X}_f, \forall x \in \mathbb{X}_f$).*

(v) *For every $y_r \in \mathbb{Y}_{ref}$ and $x \in \mathbb{X}_f$, the function $V_f(\cdot) \triangleq \| \cdot \|_P^2$, where P is the terminal cost matrix in (5.8), is a local control Lyapunov function, i.e., $V_f(f(x, u^*) - x_r) - V_f(x - x_r) \leq -\|x - x_r\|_Q^2 - \|u^* - u_r\|_R^2$.*

In [17], the authors showed that for the continuous-time nonlinear regulation problem, by using the deterministic sampling scheme with sufficiently large M and N, one can obtain an arbitrarily close approximation of the nonlinear MPC control surface and achieve arbitrarily small distance between the trajectories generated by applying u^* and $\hat{u}$ within the prediction horizon N_p. This result can be easily extended to the tracking problem by replacing the state space $\mathbb{X}$ with the augmented space $\mathbb{X} \times \mathbb{Y}_{\text{ref}}$. To this end, we denote $\Phi(k\,; x_0, u)$ as the state of system (5.5) at time-step k when the initial state at time $k = 0$ is x_0 and the input signal is u.

Lemma 16 ([17])**.** *Let $\mathcal{W}$ be a compact set satisfying $\mathcal{W} \subset \mathbb{X}_{feas} \times \mathbb{Y}_{ref}$. For any constant $\epsilon > 0$, if Assumption 1 is satisfied, then by using low-discrepancy samples in $\mathbb{X}_{feas} \times \mathbb{Y}_{ref}$, there exist some $N(\epsilon), M(\epsilon) \in \mathbb{Z}_{>0}$ and $\bar{\alpha} > 0$ such that*

$$\|\hat{u}(z) - u^*(z)\| < \epsilon, \forall N > N(\epsilon), M > M(\epsilon), z \in \mathcal{W},$$

and

$$\|\Phi(k; x, \hat{u}) - \Phi(k; x, u^*)\| < \epsilon,$$

for all $N > N(\epsilon)$, $M > M(\epsilon)$ $z \in \mathcal{W}$, and $k \in [0, N_p]$.

Remark 19. *Note that Lemma 16 is a discrete-time counterpart of the result in [17]. The basic idea of the proof is to utilize the continuity of the model and optimal solution u^* and the fact that the projection in (5.11) to the convex set $\mathbb{U}$ is non-expansive on Hilbert spaces [6, 17].*

Remark 20. *Note that the condition (iii) in Assumption 1 for continuity of the optimal solution $u^*(x, y_r)$ is required for Lemma 16 to hold. However, in general, $u^*(x, y_r)$ may not be continuous on $\mathbb{X}_{feas}$ [93]. In this case, we need to divide $\mathbb{X}_{feas}$ into subregions such that $u^*(x, y_r)$ is continuously differentiable on each subregion and construct ENMPC separately for each subregion.*

We are now ready to show that if Assumption 1 is satisfied, the proposed sampling based ENMPC guarantees stability and feasibility and achieves asymptotic tracking.

Theorem 17. *Suppose Assumption 1 is satisfied. Then for every set-point $y_r \in \mathbb{Y}_{ref}$, there exist positive integers M, N sufficiently large such that the trajectory of the system (5.5) controlled by the ENMPC controller $\hat{u}(x, y_r)$ defined in (5.11) is feasible and stable and the system output achieves asymptotic tracking (i.e., $\lim_{k\to\infty} \|y[k] - y_r\| \to 0$) with a region of attraction $\mathbb{X}_{attr}$ satisfying condition (i) in Assumption 1.*

Proof. We first prove the feasibility of the controller by using similar arguments as for the regulation problem in [17]. Note that the input constraint is satisfied by the projection operation described in (5.11). Therefore, we need to demonstrate that the state trajectory satisfies the state constraints for any $k \geq 0$.

Since $\mathbb{X}_{\text{feas}}$ is open and $\mathbb{X}_{\text{attr}} \subset \mathbb{X}_{\text{feas}}$, there exists some compact set $\mathcal{X}_1(x) \subset \mathbb{X}_{\text{feas}}$ such that

$$\Phi(k; x, u^*) \in \mathcal{X}_1(x), \forall x \in \mathbb{X}_{\text{attr}}, k \in [0, N_p],$$

and there exists some positive distance δ_1 such that $d(\mathcal{X}_1(x), \mathbb{X}_{\text{feas}}^c) > \delta_1$. By Lemma 16, we know that for any $x \in \mathbb{X}_{\text{attr}}$ and $y_r \in \mathbb{Y}_{\text{ref}}$, there exist some $N(\delta_1)$ and $M(\delta_1)$ such that

$$\|\Phi(k; x, \hat{u}) - \Phi(k; x, u^*)\| \leq \delta_1, \forall N > N(\delta_1), M > M(\delta_1), k \in [0, N_p].$$

Thus, we know that when N and M are sufficiently large,

$$\Phi(k; x, \hat{u}) \in \mathbb{X}_{\text{feas}} \subseteq \mathbb{X}, \forall x \in \mathbb{X}_{\text{attr}}, k \in [0, N_p],$$

and the state constraint is satisfied over the period $[0, N_p]$.

Now we consider the case for $k > N_p$. By conditions (i) and (iv) in Assumption 1 and the feasibility of Problem (5.8), we know that there exists some compact set $\mathcal{X}_2$ such that $\Phi(k; x, u^*)$ enters and stays in $\mathcal{X}_2$ for $k > N_p$, i.e.,

$$\Phi(k; x, u^*) \in \mathcal{X}_2, \forall x \in \mathbb{X}_{\text{attr}}, k > N_p.$$

Similar to the analysis for the case where $k \in [0, N_p]$, since $\mathbb{X}_f$ is open, there exists some positive distance δ_2 such that $d(\mathcal{X}_2, \mathbb{X}_f^c) > \delta_2$, and by Lemma 16, for any $x \in \mathbb{X}_{\text{attr}}$ and $y_r \in \mathbb{Y}_{\text{ref}}$, there exist some $N(\delta_2)$ and $M(\delta_2)$ such that

$$\|\Phi(k; x, \hat{u}) - \Phi(k; x, u^*)\| \leq \delta_2, \forall N > N(\delta_2), M > M(\delta_2), k > N_p.$$

Thus, by choosing $N > \max(N(\delta_1), N(\delta_2))$ and $M > \max(M(\delta_1), M(\delta_2))$, we have

$$\Phi(k; x, \hat{u}) \in \mathbb{X}_{\text{feas}} \subseteq \mathbb{X}, \forall k, x \in \mathbb{X}_{\text{attr}},$$

completing the proof of feasibility.

Next we prove the asymptotic tracking property. By the positive definiteness of the stage cost matrices Q and R, we know that there exists some constant $\beta > 0$ such that for any $y_r \in \mathbb{Y}_{\text{ref}}$,

$$\|x - x_r\|_Q^2 + \|u - u_r\|_R^2 \geq \beta \|x - x_r\|_2^2, \forall x \in \mathbb{X}, u \in \mathbb{U}.$$

Note that by the analysis for feasibility, $\hat{u}(x, y_r)$ and the resulting state trajectory are both feasible for any set-point y_r and initial condition $x \in \mathbb{X}_f$. Denote the value of the cost function in Problem (5.8) under control law u and initial state x as $V_{N_p}(x, y_r; u)$. Then by condition (v) in

Assumption 1 and standard techniques for stability of MPC (c.f. [90, 119]), we know that for any $y_r \in \mathbb{Y}_{\text{ref}}$ and $x \in \mathbb{X}_f$,

$$V_{N_p}(f(x, \hat{u}), y_r; \hat{u}) - V_{N_p}(x, y_r; \hat{u}) \leq -\beta \|x - x_r\|_2^2. \tag{5.12}$$

Since $V_{N_p}(\Phi(k; x, \hat{u}), y_r; \hat{u})$ is a non-increasing sequence bounded below by zero over the set $\mathbb{X}_f$, we have

$$V_{N_p}(\Phi(k+1; x, \hat{u}), y_r; \hat{u}) - V_{N_p}(\Phi(k; x, \hat{u}), y_r; \hat{u}) \to 0$$

as $k \to \infty$ for any $x \in \mathbb{X}_f$, and thus $\Phi(k; x, \hat{u}) \to x_r$ and $\hat{u} \to u_r$ as $k \to \infty$. By condition (ii) in Assumption 1, we know that $y[k] \to y_r$ as $k \to \infty$ and the sampling based ENMPC controller achieves asymptotic tracking.

Finally, we prove stability of the proposed controller. By the construction procedure of sampling based ENMPC, we know that for fixed y_r and M, N, $\hat{u}(x, y_r)$ is only a function of the (current) state x. Thus, the function $V_{N_p}(x, y_r; \hat{u})$ serves as a Lyapunov function for system $x[k+1] = f(x[k], \hat{u}(x[k], y_r))$ over $\mathbb{X}_f$ and stability of the system follows by using standard Lyapunov arguments [115], thereby completing the proof. □

5.6 Extension To Robust ENMPC

In this section, we will provide an extension of the sampling based nominal ENMPC for nonlinear systems with additive disturbances. Specifically, we consider the following system model:

$$\begin{aligned} \bar{x}[k+1] &= f(\bar{x}[k], \bar{u}[k]) + w[k] \\ \bar{y}[k] &= h(\bar{x}[k], \bar{u}[k]) \end{aligned} \tag{5.13}$$

where $w \in \mathbb{W} \subset \mathbb{R}^{n_w}$ is the disturbance and $\mathbb{W}$ is a convex and compact set containing the origin. Note that system (5.5) can be regarded as the nominal model of the above system by setting $w = 0$. Denote $\bar{\Phi}(k; \bar{x}_0, \bar{u}, w)$ to be the state of system (5.13) at time-step k when the initial state at time 0 is $\bar{x}_0$, the input signal is $\bar{u}$ and the disturbance input is w.

By adopting the same parametrization as in [88, 89], the sampling based robust ENMPC controller consists of two parts: the nominal controller and the ancillary controller. The role of the nominal controller is to generate a desired trajectory for the nominal system to follow and the ancillary controller is designed to restrict the output $\bar{y}$ of the uncertain system (5.13) to a neighborhood of the nominal system output y by minimizing the cost of their deviations. See Figure 5.1 for an illustration of the robust controller.

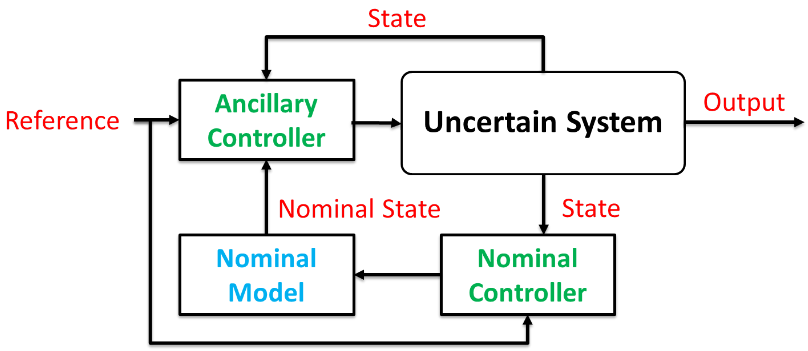


Figure 5.1: Illustration of the robust ENMPC controller.

For the nominal controller, at each sampling point, we will solve Problem (5.8) with tightened constraints. Specifically, at each sampling point $z(i) = (x(i), y_r(i)) \in \mathbb{X} \times \mathbb{Y}_{\text{ref}}$, we will solve the following modified MPC problem:

$$\begin{aligned} \min_{u} \quad & V_{N_p}(x(i), y_r(i)) = \|x_{N_p} - x_r(i)\|_P^2 \\ & + \sum_{k=0}^{N_p-1} \left(\|x_k - x_r(i)\|_Q^2 + \|u_k - u_r(i)\|_R^2\right) \\ \text{s.t.} \quad & x_{k+1} = f(x_k, u_k), \\ & x_0 = x(i), \\ & x_k \in \mathbb{X}_0, \forall k \in [0, N_p], \\ & u_k \in \mathbb{U}_0, \forall k \in [0, N_p], \\ & x_{N_p} \in \mathbb{X}_f, \end{aligned} \tag{5.14}$$

where $\mathbb{X}_0 \subset \mathbb{X}$ and $\mathbb{U}_0 \subset \mathbb{U}$ are convex and compact constraint sets containing the origin in their interior. We will refer to this problem as the **tightened nominal control problem**.

Denoting the first move of the optimal solution of Problem (5.14) to be $u^*_{\text{nomi}}(z)$, we can obtain an approximated control law $\hat{u}_{\text{nomi}}(z)$ of $u^*_{\text{nomi}}(z)$ by using the same methods as in Section 5.4.2. Note that in the projection (5.11), the input constraint $\mathbb{U}$ is replaced by $\mathbb{U}_0$. The chosen number of samples and number of basis functions are denoted by N_{nomi} and M_{nomi}, respectively. Furthermore, the coefficients obtained by solving Problem (5.10) with $u^*(z)$ replaced by $u^*_{\text{nomi}}(z)$ is denoted by α^*_{nomi}.

Remark 21. *The role of the tightened constraint sets $\mathbb{X}_0$ and $\mathbb{U}_0$ is to provide feasibility guarantees for the state and input of the uncertain system in spite of disturbance inputs; we will illustrate this later in Remark 24 after presenting the main results in this section.*

For the ancillary controller, at each sampling point $v(i) = (\bar{x}(i), x(i), y_r(i)) \in (\mathbb{X} \oplus \mathbb{W}) \times \mathbb{X} \times \mathbb{Y}_{\text{ref}}$, we solve the following problem:

$$\begin{aligned} \min_{\bar{u}} \quad & \bar{V}_{\bar{N}_p}(\bar{x}(i), x(i), y_r(i)) \\ \text{s.t.} \quad & \bar{x}_{k+1} = f(\bar{x}_k, \bar{u}_k) \\ & \bar{x}_0 = \bar{x}(i), \\ & \bar{u}_k \in \mathbb{U}, \forall k \in [0, \bar{N}_p], \\ & \bar{x}_{\bar{N}_p} \in \bar{\mathbb{X}}_f, \end{aligned} \tag{5.15}$$

where

$$\begin{aligned} \bar{V}_{\bar{N}_p}(\bar{x}(i), x(i), y_r(i)) \triangleq \|\bar{x}_{\bar{N}_p} - \Phi(\bar{N}_p; x(i), u^*_{\text{nomi}})\|^2_{\bar{P}} \\ + \sum_{k=0}^{\bar{N}_p - 1} \|\bar{x}_k - \Phi(k; x(i), u^*_{\text{nomi}})\|^2_{\bar{Q}} + \|\bar{u}_k - u^*_{\text{nomi}}\|^2_{\bar{R}}, \end{aligned}$$

$\bar{P} \succ 0$, $\bar{Q} \succeq 0$ and $\bar{R} \succ 0$ are weighting matrices, $\bar{N}_p$ is the prediction horizon, and $\bar{\mathbb{X}}_f$ is the terminal set. Note that the terminal cost $\bar{P}$ and terminal set $\bar{\mathbb{X}}_f$ are chosen to satisfy similar conditions as in Assumption 1 to guarantee stability.

Denote the first move of the optimal solution of Problem (5.15) to be $\bar{u}^*_{\text{anci}}(v)$; as before, we can obtain an approximated control law $\hat{u}_{\text{anci}}(v)$ using samples drawn from $\bar{u}^*_{\text{anci}}(v)$. The chosen number of samples and number of basis functions are denoted by N_{anci} and M_{anci}, respectively. Furthermore, the coefficients obtained by solving Problem (5.10) with $u^*(z)$ replaced by $u^*_{\text{anci}}(z)$ is denoted by α^*_{anci}.

Remark 22. *Note that in order to reduce the offline computational complexity, we may utilize the data of the nominal controller to design the ancillary controller. For example, we can choose $N_{nomi} = N_{anci}$ and choose each sampling point v for the ancillary controller as $v = (\bar{x}, z)$ where z is the corresponding sampling point for the nominal controller. Then we can apply $\hat{u}_{nomi}(z)$ obtained by solving Problem (5.14) to the cost function of Problem (5.15).*

The design of sampling based robust ENMPC is summarized in Algorithm 5. In the rest of this section, we will extend the results in [88, 89] for tube-based robust nonlinear MPC to the sampling-based robust ENMPC and show that if certain assumptions on the ENMPC parameters are satisfied and the disturbance is small enough, then the ancillary controller is able to keep the state of the uncertain system in a neighborhood of the central path generated by the nominal controller.

Algorithm 5 [Sampling based Robust ENMPC]

```
Offline Computations
Input: System functions $f$ and $h$, constraints $\mathbb{X}_0, \mathbb{U}_0, \mathbb{X}$ and $\mathbb{U}$, reference constraint $\mathbb{Y}_{\text{ref}}$, numbers of samples $N_{\text{nomi}}, N_{\text{anci}}$, and numbers of basis functions $M_{\text{nomi}}, M_{\text{anci}}$
Output: Coefficients $\alpha^*_{\text{nomi}}$ and $\alpha^*_{\text{anci}}$
Nominal Controller
For each low-discrepancy sample $z(i) \in \mathbb{X} \times \mathbb{Y}_{\text{ref}}$, solve Problem (5.8) with tightened constraints $\mathbb{X}_0$ and $\mathbb{U}_0$
Obtain $\alpha^*_{\text{nomi}}$ by solving Problem (5.10) with $u^*(z)$ replaced by $u^*_{\text{nomi}}(z)$
Ancillary Controller
For each low-discrepancy sample $v(i) \in (\mathbb{X} \oplus \mathbb{W}) \times \mathbb{X} \times \mathbb{Y}_{\text{ref}}$, solve Problem (5.15)
Obtain $\alpha^*_{\text{anci}}$ by solving Problem (5.10) with $u^*(z)$ replaced by $\bar{u}^*_{\text{anci}}(v)$

Online Computations
Input: Coefficients $\alpha^*_{\text{nomi}}$, $\alpha^*_{\text{anci}}$ and input constraints $\mathbb{U}_0, \mathbb{U}$
Output: Control input $\bar{u}$
Initialize: Obtain the state $\bar{x}[0]$ and set $x[0] = \bar{x}[0]$
At each time-step $k \geq 0$:
Obtain the state $\bar{x}[k]$ and reference $y_{\text{ref}}[k]$
$\hat{u}_{\text{anci}}(v) = \arg\min_{u \in \mathbb{U}} \| \sum_{i=1}^{M} \alpha^*_{\text{anci},i} \mathbb{B}_i(v) - u \|$ where $v = (\bar{x}[k], x[k], y_{\text{ref}}[k])$
Apply $\bar{u}[k] = \hat{u}_{\text{anci}}(v)$ to the uncertain system (5.13)
$\hat{u}_{\text{nomi}}(z) = \arg\min_{u \in \mathbb{U}_0} \| \sum_{i=1}^{M} \alpha^*_{\text{nomi},i} \mathbb{B}_i(z) - u \|$ where $z = (x[k], y_{\text{ref}}[k])$
Apply $u[k] = \hat{u}_{\text{nomi}}(z)$ to the nominal system (5.5)
```

Denote the feasibility region of the tightened nominal control problem and Problem (5.15) as $\mathbb{X}^0_{\text{feas}}$ and $\bar{\mathbb{X}}_{\text{feas}}(x)$, respectively. Note that $\bar{\mathbb{X}}_{\text{feas}}(x)$ is a function of the nominal initial state x. Define $\mathcal{X}_{\text{feas}} \triangleq \{(\bar{x}, x) | x \in \mathbb{X}^0_{\text{feas}}, \bar{x} \in \bar{\mathbb{X}}_{\text{feas}}(x)\}$. Further define the function $\theta(\bar{x}, x, u) \triangleq \bar{\Phi}(N; \bar{x}, u, 0) - \Phi(N; x, u^*_{\text{nomi}})$.

We make the following assumptions on the robust ENMPC parameters.

Assumption 2.

(i) The set $\mathcal{X}_{feas}$ is open and there exists a compact set $\bar{\mathbb{X}}_{attr} \subset \mathcal{X}_{feas}$.

(ii) For any $y_r \in \mathbb{Y}_{ref}$, u^*_{nomi} is continuously differentiable on $\mathbb{X}^0_{feas}$ and for any $x \in \mathbb{X}^0_{feas}$, u^*_{anci} is

continuously differentiable on $\bar{\mathbb{X}}_{feas}(x)$.

(iii) *For any $y_r \in \mathbb{Y}_{ref}$, there exists some constant $c > 0$ such that $\bar{V}_{\bar{N}_p}(\bar{x}, x, y_r; \hat{u}_{anci}) \leq c\|\bar{x} - x\|^2, \forall(\bar{x}, x) \in (\bar{\mathbb{X}}_{feas}(x) \oplus \mathbb{W}) \times \mathbb{X}^0_{feas}$, where $\bar{V}_{\bar{N}_p}(\bar{x}, x, y_r; u)$ is the value of the cost function in Problem (5.15) when the input u is applied.*

(iv) *For any $y_r \in \mathbb{Y}_{ref}$ and all $(\bar{x}, x) \in \mathcal{X}_{feas}$, $(\partial/\partial u)\theta(\bar{x}, x, u^*_{anci})$ has full rank n where $(\partial/\partial u)\theta$ denotes the matrix whose (i, j)-th element is $\partial\theta_i/\partial u_j$.*

Remark 23. *Note that conditions (iii) and (iv) in Assumption 2 are assumptions that have been previously made in [88] in order to provide stability guarantees for the tube-based robust MPC.*

Let $\bar{V}^*_{\bar{N}_p}(\bar{x}, x, y_r)$ be the optimal value function of Problem (5.15). In [88], the authors proved the following property of $\bar{V}^*_{\bar{N}_p}(\bar{x}, x, y_r)$ when conditions (iii) and (iv) in Assumption 2 hold.

Lemma 17 ([88])**.** *If conditions (iii) and (iv) in Assumption 2 are satisfied, then for any $y_r \in \mathbb{Y}_{ref}$, there exist some constants $c_2 > c_1 > 0$ and $\gamma_1 \triangleq (1 - \frac{c_1}{c_2}) \in (0, 1)$ such that $\forall(\bar{x}, x) \in \mathcal{X}_{feas}$,*

$$c_1\|\bar{x} - x\| \leq \bar{V}^*_{\bar{N}_p}(\bar{x}, x, y_r) \leq c_2\|\bar{x} - x\|,$$

$$\bar{V}^*_{\bar{N}_p}(f(\bar{x}, u^*_{anci}), f(x, u^*_{nomi}), y_r) \leq \gamma_1\bar{V}^*_{\bar{N}_p}(\bar{x}, x, y_r).$$

Now we extend the above result to the suboptimal value function $\bar{V}_{\bar{N}_p}(\bar{x}, x, y_r; \hat{u}_{\text{anci}})$.

Lemma 18. *If Assumption 2 is satisfied and the parameters $N_{nomi}, M_{nomi}, N_{anci}, M_{anci}$ are sufficiently large, then for any $y_r \in \mathbb{Y}_{ref}$, there exists some constant $\gamma_2 \in (\gamma_1, 1)$ such that*

$$\begin{aligned}&\bar{V}_{\bar{N}_p}(f(\bar{x}, \hat{u}_{anci}), f(x, \hat{u}_{nomi}), y_r; \hat{u}_{anci})\\ &\qquad\qquad \leq \gamma_2\bar{V}_{\bar{N}_p}(\bar{x}, x, y_r; \hat{u}_{anci}), \forall(\bar{x}, x) \in \bar{\mathbb{X}}_{attr}.\end{aligned}$$

Proof. First note that since conditions (i) and (ii) are satisfied, by a similar analysis as in the proof of Theorem 17, we know that

$$(f(\bar{x}, \hat{u}_{\text{anci}}), f(x, \hat{u}_{\text{nomi}})) \in \mathcal{X}_{\text{feas}}, \forall(\bar{x}, x) \in \mathcal{X}_{\text{feas}},$$

and thus feasibility of the controller is guaranteed.

Let c_1 and c_2 be constants defined in Lemma 17. Since $\bar{V}_{\bar{N}_p}(\bar{x}, x, y_r; \hat{u}_{\text{anci}}) \geq \bar{V}^*_{\bar{N}_p}(\bar{x}, x, y_r)$ and by the same analysis as for equation (5.12) in the proof of Theorem 17, we know that there exist some constants $0 < c_1' < c_1$ and $c_2' > c_2$ such that

$$\bar{V}_{\bar{N}_p}(\bar{x}, x, y_r;\ \hat{u}_{\text{anci}}) \leq c_2' \|\bar{x} - x\|,$$

and

$$\bar{V}_{\bar{N}_p}(f(\bar{x}, \hat{u}_{\text{anci}}), f(x, \hat{u}_{\text{nomi}}), y_r; \hat{u}_{\text{anci}}) - \bar{V}_{\bar{N}_p}(\bar{x}, x, y_r; \hat{u}_{\text{anci}}) \leq -c_1' \|\bar{x} - x\|.$$

Thus, we have

$$\bar{V}_{\bar{N}_p}(f(\bar{x}, \hat{u}_{\text{anci}}), f(x, \hat{u}_{\text{nomi}}), y_r; \hat{u}_{\text{anci}}) \leq \gamma_2 \bar{V}_{\bar{N}_p}(\bar{x}, x, y_r; \hat{u}_{\text{anci}}),$$

where $\gamma_2 \triangleq (1 - \frac{c_1'}{c_2'}) \in (\gamma_1, 1)$, completing the proof. □

Given the nominal state trajectory $\Phi(k; x, \hat{u}_{\text{nomi}})$, the objective of tube-based robust MPC is to find a sequence of tubes centered at the nominal state such that the state of the uncertain system lies in these tubes, i.e., to find a constant set $\mathcal{S}$ such that $\bar{\Phi}(k; \bar{x}, \hat{u}_{\text{anci}}, w) \in \{\Phi(k; x, \hat{u}_{\text{nomi}})\} \oplus \mathcal{S}, \forall k, w$ [115]. In [88, 89], the authors showed that one can use the sublevel set of the optimal cost function to characterize the set of tubes. Based on this idea, for any $x \in \mathbb{X}^0_{\text{feas}}$, $y_r \in \mathbb{Y}_{\text{ref}}$, and a parameter $d \geq 0$, we define the following set-valued function:

$$\mathcal{S}_d(x, y_r) \triangleq \{\bar{x} | \bar{V}_{\bar{N}_p}(\bar{x}, x, y_r; \hat{u}_{\text{anci}}) \leq d\}. \tag{5.16}$$

Note that since $\mathcal{S}_0(x, y_r) = \{x\}$, the set $\mathcal{S}_d(x, y_r)$ characterizes a neighborhood of the nominal state x [115]. Define

$$w_{\max} \triangleq \max\{\|w\| : w \in \mathbb{W}\}$$

as a metric characterizing the size of the disturbance set.

In the following theorem, we will show that by using the sampling based robust ENMPC controller, the state of the uncertain system always stays in the state-tube $\{\mathcal{S}_d(\Phi(k; x, \hat{u}_{\text{nomi}}), y_r)\}_{k \geq 0}$ and converges to a set $\mathcal{S}_d(x_r, y_r)$ provided that the disturbance is small enough.

Theorem 18. *Suppose that Assumption 1 for the nominal system (5.5) and Assumption 2 for the uncertain system (5.13) are satisfied. For every set-point $y_r \in \mathbb{Y}_{ref}$ and all $(\bar{x}[0], x[0]) \in \bar{\mathbb{X}}_{attr}$, there exist positive integers $N_{nomi}, M_{nomi}, N_{anci}, M_{anci}$ sufficiently large such that*

$$\bar{\Phi}(k; \bar{x}[0], \hat{u}_{anci}, w) \in \mathcal{S}_d(\Phi(k; x[0], \hat{u}_{nomi}), y_r), \forall k,$$

and $\bar{\Phi}(k; \bar{x}[0], \hat{u}_{anci}, w)$ converges to the set $\mathcal{S}_d(x_r, y_r)$ for all disturbances satisfying $w_{max} \leq \frac{(1-\gamma_2)d}{c}$.

Proof. By Lemma 18 and condition (iii) in Assumption 2, we know that $\forall(\bar{x}, x) \in \bar{\mathbb{X}}_{\text{attr}}, w \in \mathbb{W}$,

$$\begin{aligned}\bar{V}_{\bar{N}_p}(f(\bar{x}, \hat{u}_{\text{anci}}) + w, &f(x, \hat{u}_{\text{nomi}}), y_r; \hat{u}_{\text{anci}}) \\ &\leq \gamma_2 \bar{V}_{\bar{N}_p}(\bar{x}, x, y_r; \hat{u}_{\text{anci}}) + c w_{\max}\end{aligned}$$

where the constant c satisfies condition (iii). By using the same techniques as in Proposition 4 in [89], we know that

$$\begin{aligned}\bar{V}_{\bar{N}_p}(f(\bar{x}, \hat{u}_{\text{anci}}) + w, &f(x, \hat{u}_{\text{nomi}}), y_r; \hat{u}_{\text{anci}}) \\ &\leq \bar{V}_{\bar{N}_p}(\bar{x}, x, y_r; \hat{u}_{\text{anci}}) \leq d\end{aligned}$$

provided that $c w_{\max} \leq (1 - \gamma_2)\bar{V}_{\bar{N}_p}(\bar{x}, x, y_r; \hat{u}_{\text{anci}}) \leq (1 - \gamma_2)d$. Note that since we choose $\bar{x}[0] = x[0]$, we have $\bar{x}[0] \in \mathcal{S}_d(x[0], y_r), \forall d \geq 0$, and the robust positive invariance of the state-tube $\{\mathcal{S}_d(\Phi(k; x, \hat{u}_{\text{nomi}}), y_r)\}_{k \geq 0}$ follows.

By Theorem 17, we know that for any $y_r \in \mathbb{Y}_{\text{ref}}$ and x in the attraction region, $\Phi(k; x, \hat{u}_{\text{nomi}}) \to x_r$ as $k \to \infty$. Since the state-tube is robust positively invariant, the limit point of any realization of $\bar{\Phi}(k; \bar{x}[0], \hat{u}_{\text{anci}}, w)$ lies in the set $\mathcal{S}_d(x_r, y_r)$, completing the proof. □

Remark 24. *Note that if the tightened constraint sets $\mathbb{X}_0$ and $\mathbb{U}_0$ in the tightened nominal control problem are chosen such that $\mathcal{S}_d(x_r, y_r) \subset \mathbb{X}$ for all $y_r \in \mathbb{Y}_{ref}$, then the constraints of the uncertain system (5.13) are satisfied. In [89], the authors proposed a simple method to determine $\mathbb{X}_0$ and $\mathbb{U}_0$. Specifically, one can choose $\mathbb{X}_0 = \alpha\mathbb{X}$ and $\mathbb{U}_0 = \beta\mathbb{U}$ where $\alpha, \beta \in (0, 1)$ are some scalar tuning parameters. Then the problem is reduced to appropriately choosing the two constants α and β.*

5.7 Simulation

In this section, we illustrate the performance of the proposed sampling based ENMPC controller with simulations. Offline solutions of the constrained optimization problems are computed by using the `GODLIKE` toolbox in MATLAB [104]. The sampling points are drawn from the Halton sequence [62].

5.7.1 2-D System

We first consider a simple two-dimensional nonlinear system which is similar to the examples studied in [17, 20] as follows:

$$\begin{aligned} \begin{bmatrix} x_1[k+1] \\ x_2[k+1] \end{bmatrix} &= \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix} \begin{bmatrix} x_1[k] \\ x_2[k] \end{bmatrix} + \begin{bmatrix} x_1[k] \\ x_2[k] \end{bmatrix} u[k] + \begin{bmatrix} w_1[k] \\ w_2[k] \end{bmatrix} \\ y[k] &= \begin{bmatrix} 1 & 1 \end{bmatrix} \begin{bmatrix} x_1[k] \\ x_2[k] \end{bmatrix}, \end{aligned} \tag{5.17}$$

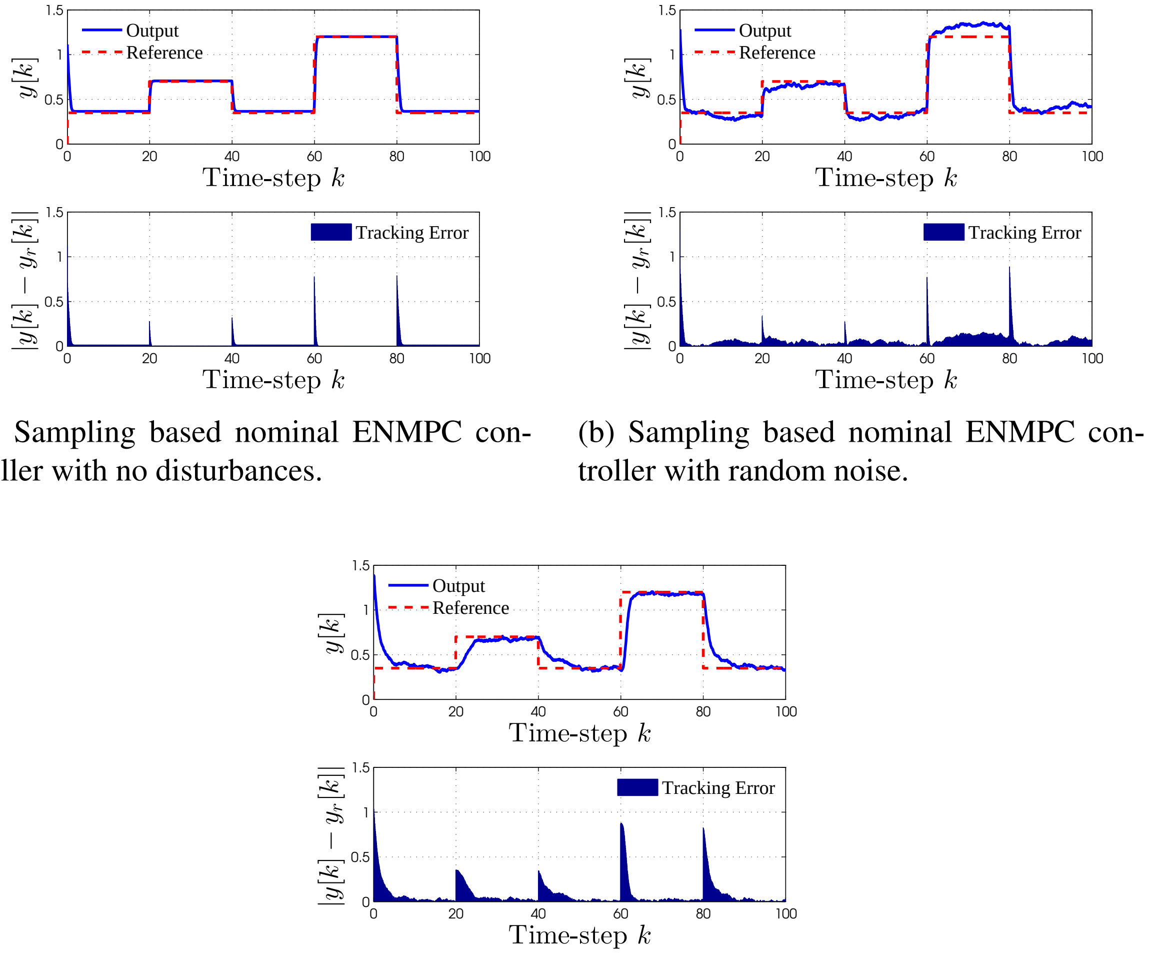
where $x_1, x_2, u, y, w_1, w_2 \in \mathbb{R}$. The state and input constraints are as follows:

$$\begin{aligned} \mathbb{X} &= \{x \in \mathbb{R}^2 : \|x\|_\infty \leq 2\}, \\ \mathbb{U} &= \{u \in \mathbb{R} : |u| \leq 2\}. \end{aligned}$$

Furthermore, the reference signal y_{ref} takes values in the interval $[0, 2]$ and the disturbance w is a random noise process satisfying $\|w\|_\infty \leq 0.2$. We will refer to the controllers generated by Algorithm 4 and Algorithm 5 as the sampling based nominal ENMPC controller and the sampling based robust ENMPC controller, respectively.

For the sampling based nominal ENMPC controller, we use $N = 2000$ sampling points (which required approximately 1 hour of off-line computation), and the basis functions are chosen to be the Legendre polynomials up to degree 3 with $M = 35$ coefficients. For the sampling based robust ENMPC controller, we use $N_{\text{nomi}} = 2000$ and $N_{\text{anci}} = 10000$ sampling points for the nominal and ancillary controllers, respectively (which required approximately 5 hours of off-line computation), and the basis functions are chosen to be the Legendre polynomials up to degree 3 for both controllers. The prediction horizons are all chosen to be 5. Note that there exist different methods to determine the ENMPC parameters such that the corresponding conditions in Assumption 1 are satisfied (e.g., see [20, 115]). Further note that the expansion of N_{anci} compared to N_{nomi} is due to the sampling space being extended to a higher dimension for the ancillary controller. The results are in Figure 5.2.

From Figure 5.2a, we can see that for the nominal system (5.5), the sampling based nominal ENMPC controller achieves offset free tracking of the reference signal, and a relatively large tracking error only appears when the set point changes. However, for the uncertain system (5.13) with random noise, we can observe from Figure 5.2b that the tracking performance is not satisfactory by simply ignoring the noise and applying the sampling based nominal ENMPC controller. By contrast, the result in Figure 5.2c shows that the tracking error is reduced at steady state by using the sampling based robust ENMPC controller and the output of the uncertain system is



(a) Sampling based nominal ENMPC controller with no disturbances.

(b) Sampling based nominal ENMPC controller with random noise.

(c) Sampling based robust ENMPC controller with random noise.

Figure 5.2: Performance of the sampling based nominal ENMPC controller and the sampling based robust ENMPC controller.

driven to stay close to the reference. Note that the plots in Figure 5.2b and Figure 5.2c are for the same realization of disturbances.

Next we compare the running time of the sampling based ENMPC controller with the online NMPC controller using a set of 100 randomly generated initial conditions. The simulation is conducted on a typical 2.4-GHz personal computer. The results are in Table 5.1.

Table 5.1: Complexity comparison of different NMPC approaches over 100 randomly generated initial conditions. The table presents the **average** and **standard deviation** of the time required to compute a control action (unit: millisecond).

	Average	Standard Deviation
Algorithm 4	3.2	0.1
Online NMPC	82.8	6
Algorithm 5	5.8	0.4
Online Tube-based Robust NMPC	213.5	26

From Table 5.1, we can see that both the sampling based nominal and robust ENMPC controllers are more efficient than the online NMPC controller. Note that the online computational time of the ENMPC controller can be further reduced by choosing a smaller number of basis functions, at the cost of losing tracking performance.

5.7.2 CPU-GPU Queueing System

In this subsection, we test the performance of the sampling based ENMPC approach on the CPU-GPU queueing system proposed in [59], which is the motivating application for the results in this chapter; see Figure 5.3 for the model of the system. The objective is to drive the injection rate of the GPU queue (i.e., λ_{GPU}) to track a target number of Frames Per Second (FPS) of the display, and the controllable variables are the operating frequencies of the CPU and GPU (i.e., f_{CPU} and f_{GPU}) which determine the injection rates of the corresponding queues (i.e., λ_{CPU} and λ_{GPU}).

The dynamics of the queueing system is given as follows:

$$\begin{aligned} q_{CPU}[k+1] &= q_{CPU}[k] + \lambda_{CPU}[k]T_s - \lambda_{GPU}[k]T_s \\ q_{GPU}[k+1] &= q_{GPU}[k] + \lambda_{GPU}[k]T_s - \mu_{GPU}T_s, \end{aligned}$$

where q_{CPU} (resp. q_{GPU}) and λ_{CPU} (resp. λ_{GPU}) are the state and injection rate of the CPU queue (resp. GPU queue), respectively, T_s is the sampling time, and μ_{GPU} is a constant target FPS.

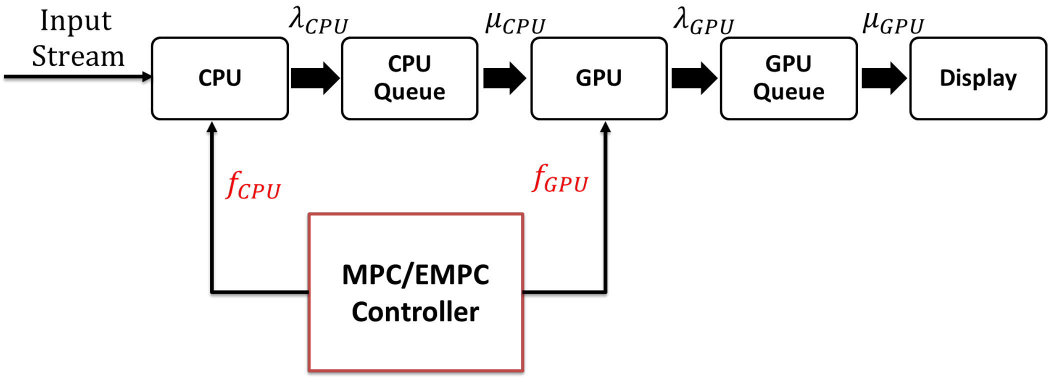


Figure 5.3: CPU-GPU queueing system.

The injection rates λ_{CPU} and λ_{GPU} are determined by f_{CPU} and f_{GPU} as follows:

$$\lambda_{CPU}[k] = \Phi\left(T_{CPU}^{ref}[k], f_{CPU}[k]\right)$$
$$\lambda_{GPU}[k] = \Phi\left(T_{GPU}^{ref}[k], f_{GPU}[k]\right),$$

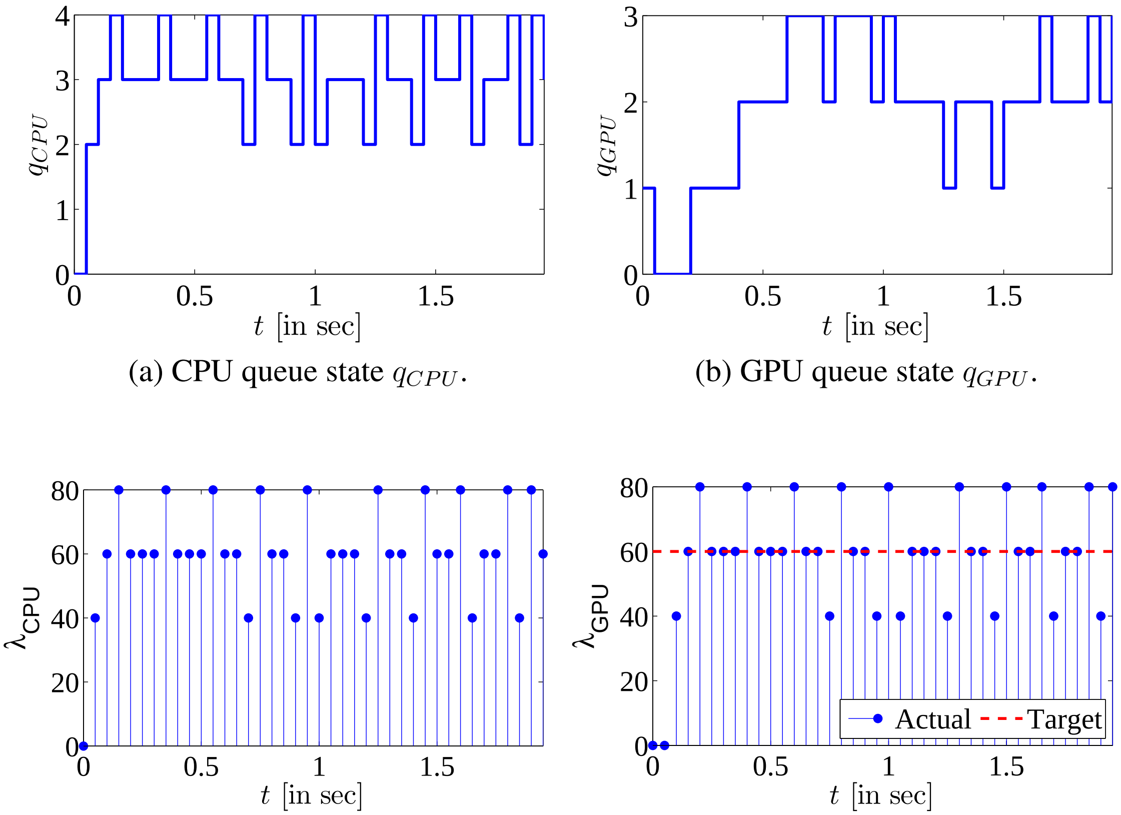
where the parameter $T_{CPU}^{ref}[k]$ (resp. $T_{GPU}^{ref}[k]$) is the average processing time for the tokens in the CPU queue (resp. GPU queue) at time-step k when the CPU (resp. GPU) is operating at some specified reference frequency. Note that T_{CPU}^{ref} and T_{GPU}^{ref} only depend on the characteristics of the tokens to be processed (i.e., the input stream to the system). For a given sequence of tokens (i.e., given T_{CPU}^{ref} or T_{GPU}^{ref}), the function Φ represents the mapping from the frequency adopted (i.e., f_{CPU} or f_{GPU}) to the corresponding injection rate; we omit the specific form of the function Φ and refer to [59] for more details.

In order to construct the ENMPC controller, we regard the variables $q_{CPU}, q_{GPU}, T_{CPU}^{ref}$ and T_{GPU}^{ref} as parameters of the optimization problem. We use $N = 5000$ sampling points over the sampling region which is $q_{CPU}, q_{GPU} \in [0, 5]$ and $T_{CPU}^{ref}, T_{GPU}^{ref} \in [10, 20]$.[1] We choose the basis functions to be the Legendre polynomials up to degree 2 with $M = 15$ coefficients and the prediction horizon is chosen to be 2. Furthermore, the sampling time $T_s = 50$ (unit: millisecond) and the target FPS $\mu_{GPU} = 60$.

In Figure 5.4, we illustrate the performance of the ENMPC controller using data collected from a mobile platform. From Figure 5.4d, we can see that the controller is able to drive the

[1] Note that the unit of T_{CPU}^{ref} and T_{GPU}^{ref} is millisecond.

injection rate of the GPU queue close to the desired target (i.e., μ_{GPU}). Moreover, from Figure 5.4a and Figure 5.4b, we can see that the queue occupancies are kept in a desired range (i.e., $q_{CPU}, q_{GPU} \in [0, 5]$).



(a) CPU queue state q_{CPU}.

(b) GPU queue state q_{GPU}.

(c) CPU queue injection rate λ_{CPU}.

(d) GPU queue injection rate λ_{GPU}.

Figure 5.4: Performance of the ENMPC controller on the CPU-GPU queueing system. The red dashed line in Figure 5.4d is the target FPS to track.

5.8 Summary

In this chapter, we studied the output tracking problem for nonlinear constrained systems and proposed a sampling based ENMPC approach to address the problem. The basic idea is to sample the augmented state and reference signal space using a low-discrepancy sequence and approximate the optimal control surface based on the information at the sampling points. We also extended the tube-based robust control in [88,89] to robust ENMPC by using the sampling based approach. As we showed, the proposed approaches achieve asymptotic tracking and guarantee stability and feasibility of the system, given that certain mild conditions are satisfied. Moreover, the sampling based (robust) ENMPC is easy to implement and is suitable for applications with limited online computation and storage resources, such as MPSoCs.

Chapter 6

Conclusions and Future Research

6.1 Conclusions

In this thesis, we studied a set of estimation and control problems, driven by applications in Multi-Processor Systems on Chips (MPSoCs). These applications stimulated new formulations for extensively studied problems (e.g., the state estimation problems in Chapter 2 and Chapter 3), motivated new objectives for existing problems in the literature (e.g., the design-time sensor selection problem in Chapter 4), and brought new challenges which required us to invent applicable techniques (e.g., the resource constrained output tracking problem in Chapter 5).

In Chapter 2 and Chapter 3, we studied the state estimation problem for linear dynamical systems with unknown inputs when the system is not strongly detectable. In other words, we studied the case where it is impossible to exactly reconstruct the system states. Under this situation, in Chapter 2, we considered the problem of constructing an unknown input norm-observer, which can be regarded as a relaxed estimation objective for cases where perfect estimation cannot be achieved, and proposed the notion of BIBOBS stability to solve the unknown input norm estimation problem. We showed that under certain conditions, the inputs and initial condition can be chosen so that the states corresponding to the eigenvalues with magnitude 1 are persistently excited or triggered while the states of the other systems are maintained in a bounded orbit; thus, care must be taken to avoid such situations.

In Chapter 3, we explored the influence of the other assumption on the system: the property of positivity. We showed that the additional information on positivity is not helpful in relaxing the conditions under which perfect estimation is achievable. We also considered the case where the positivity of the observers is needed and provided a construction method for positive observers.

In Chapter 4, we studied the priori and posteriori KFSS problems for linear dynamical systems. We showed that these problems are both NP-hard (even under the additional assumption that the system is stable). We also provided upper bounds for the performance of the worst-case selection of sensors and highlighted the factors that dominate the worst-case performance. Then we studied *a priori* covariance based and *a posteriori* covariance based greedy algorithms for sensor selection. We showed that these algorithms are optimal for two classes of systems. For general systems, we provided a negative result showing that the corresponding cost functions are neither supermodular nor submodular; however, simulations indicate that these algorithms perform well in practice. For the Lyapunov equation based greedy algorithm (which attempts to minimize an upper bound on the original objective functions), we showed that this algorithm achieves optimal performance with respect to its cost function. Although this algorithm performs less well than the original algorithm in terms of minimizing the steady state Kalman filtering error covariance, the run-time scales better with the system size.

Finally, in Chapter 5, we proposed an efficient sampling based explicit nonlinear MPC (ENMPC) for the nonlinear output tracking problem by augmenting the state space with the reference space. We provided feasibility and asymptotic tracking guarantees for the nominal controlled system. We designed an ancillary ENMPC to eliminate the influence of additive disturbances and provided ultimate bounds for the robust ENMPC controlled system states. The proposed approaches are efficient for online implementation and can be easily modified to balance the trade off between performance and online computational complexity.

To summarize, the theory and techniques developed in this thesis provide efficient algorithms for estimation, configuration and control of MPSoCs and yield insights into the underlying structure of the corresponding problems.

6.2 Future Research

Along the line of our results in this thesis, there are still many interesting directions for future research. Here we briefly state some of the potential directions.

- **Partial State Norm Estimation**

  In Chapter 2, we studied the state norm estimation problem where the objective is to provide an upper bound on the norm of the whole set of states. However, in some cases, we may only be interested in estimating a subset of the states. For example, in [134], the authors proposed a design procedure for partial state observers which recovers a particular function of the states. Thus, it is of interest to study the partial state norm estimation problem where the objective is to estimate an upper bound on the norm of a certain subset of states.

- **State Estimation with Mixed Disturbances**

  As we have mentioned in Section 2.2, it is important to define appropriate models of uncertainty to conduct state estimation. While there have been extensive works focusing on a specific type of disturbance, little attention has been paid to a combination of different disturbances. Exceptions include [50,132] where a combination of stochastic disturbances and unknown inputs are considered, and [44, 101] where a combination of stochastic and set-membership disturbances is studied. Thus, besides norm estimation and estimation for positive systems studied in this thesis, it will be interesting to study other combinations of disturbances.

- **Sensor Selection with Different Estimation Strategies**

  The design-time sensor selection problem studied in Chapter 4 is based on the choice of Kalman filter as the underlying estimator. While Kalman filtering is applicable for various applications, there exist scenarios under which the corresponding theory is not applicable (e.g., when it is difficult to obtain statistics of the noises or there exist requirements on the worst-case performance). Thus, it is of interest to study the sensor selection problem based on other estimation strategies. For example, the H_∞ filtering framework is a promising candidate, especially the algebraic Riccati equation (ARE) based approach [75] which shares the same foundation with the Kalman filtering framework.

- **Sampling based ENMPC with Model Uncertainties**

  In Section 5.6, we proposed a robust variant to deal with additive bounded disturbances while assuming that the system model is accurate. In many applications, it may be difficult to identify a reliable model of the system. Thus, for future work, it will be interesting to extend the sampling based ENMPC to handle other classes of modeling or parametric uncertainties.

References

[1] A. Alessio and A. Bemporad. A survey on explicit model predictive control. In *Nonlinear Model Predictive Control*, pages 345–369, 2009.

[2] B. D. O. Anderson and J. B. Moore. *Optimal Filtering*. Dover Publications, 2012.

[3] D. Angeli and E. D. Sontag. Monotone control systems. *IEEE Transactions on Automatic Control*, 48(10):1684–1698, 2003.

[4] J. Back and A. Astolfi. Design of positive linear observers for positive linear systems via coordinate transformations and positive realizations. *SIAM Journal on Control and Optimization*, 47(1):345–373, 2008.

[5] C. J. Bardeen, V. V. Yakovlev, K. R. Wilson, S. D. Carpenter, P. M. Weber, and W. S. Warren. Feedback quantum control of molecular electronic population transfer. *Chemical Physics Letters*, 280(1):151–158, 1997.

[6] H. H. Bauschke and P. L. Combettes. *Convex Analysis and Monotone Operator Theory in Hilbert spaces*. Springer Science & Business Media, 2011.

[7] A. Bemporad, F. Borrelli, and M. Morari. Model predictive control based on linear programming - the explicit solution. *IEEE Transactions on Automatic Control*, 47(12):1974–1985, 2002.

[8] A. Bemporad and C. Filippi. An algorithm for approximate multiparametric convex programming. *Computational Optimization and Applications*, 35(1):87–108, 2006.

[9] A. Bemporad, M. Morari, V. Dua, and E. N. Pistikopoulos. The explicit linear quadratic regulator for constrained systems. *Automatica*, 38(1):3–20, 2002.

[10] D. Bertsekas and I. Rhodes. Recursive state estimation for a set-membership description of uncertainty. *IEEE Transactions on Automatic Control*, 16(2):117–128, 1971.

[11] F. Bian, D. Kempe, and R. Govindan. Utility based sensor selection. In *Proc. of the 5th International Conference on Information Processing in Sensor Networks*, pages 11–18, 2006.

[12] F. Borrelli. *Constrained Optimal Control of Linear and Hybrid Systems*, volume 290. Springer, 2003.

[13] S. Boyd and L. Vandenberghe. *Convex Optimization*. Cambridge University Press, 2004.

[14] R. S. Bucy and P. D. Joseph. *Filtering for Stochastic Processes with Applications to Guidance*, volume 326. Americ. Math. Soc., 1987.

[15] D. E. Catlin. *Estimation, Control, and the Discrete Kalman Filter*, volume 71. Applied Mathmatical Sciences, 1989.

[16] A. Chakrabarty, G. T. Buzzard, M. J. Corless, S. H. Zak, and A. E. Rundell. Correcting hypothalamic-pituitary-adrenal axis dysfunction using observer-based explicit nonlinear model predictive control. In *Proc. of the 36th Annual International Conference of the Engineering in Medicine and Biology Society*, pages 3426–3429, 2014.

[17] A. Chakrabarty, V. Dinh, M. J. Corless, A. E. Rundell, S. H. Zak, and G. T. Buzzard. Support vector machine informed explicit nonlinear model predictive control. *IEEE Transactions on Automatic Control*, 2016. to appear.

[18] T. Chang and X. Sun. Analysis and control of monolithic piezoelectric nano-actuator. *IEEE Transactions on Control Systems Technology*, 9(1):69–75, 2001.

[19] B. Chen and C. Wu. Robust optimal reference-tracking design method for stochastic synthetic biology systems: T–S fuzzy approach. *IEEE Transactions on Fuzzy Systems*, 18(6):1144–1159, 2010.

[20] H. Chen and F. Allgower. A quasi-infinite horizon nonlinear model predictive control scheme with guaranteed stability. *Automatica*, 34(10):1205–1217, 1998.

[21] F. L. Chernousko. Ellipsoidal state estimation for dynamical systems. *Nonlinear Analysis: Theory, Methods & Applications*, 63(5):872–879, 2005.

[22] C. Commault and J-M. Dion. Optimal sensor location for fault detection and isolation in linear structured systems. In *Proc. of the European Control Conference*, volume 3, 2003.

[23] A. Das and D. Kempe. Algorithms for subset selection in linear regression. In *Proceedings of the 14th ACM Symposium on Theory of Computing*, pages 45–54, 2008.

[24] A. Das and D. Kempe. Submodular meets spectral: Greedy algorithms for subset selection, sparse approximation and dictionary selection. *arXiv:1102.3975*, 2011.

[25] B. N. Datta. *Numerical Methods for Linear Control Systems: Design and Analysis*. Academic Press, 2004.

[26] N. K. Dhingra, M. R. Jovanović, and Z. Q. Luo. An ADMM algorithm for optimal sensor and actuator selection. In *Proc. of 53rd IEEE Conference on Decision and Control*, pages 4039–4044, 2014.

[27] N. K. Dhingra, M. R. Jovanović, and Z. Q. Luo. Optimal sensor and actuator selection for large-scale dynamical systems. In *Proc. of the 2015 Asilomar Conference on Signals, Systems, and Computers*, 2015.

[28] J. Dion, C. Commault, and J. Van Der Woude. Generic properties and control of linear structured systems: a survey. *Automatica*, 39(7):1125–1144, 2003.

[29] P. Falugi. Model predictive control for tracking randomly varying references. *International Journal of Control*, 88(4):745–753, 2015.

[30] L. Farina and S. Rinaldi. *Positive Linear Systems: Theory and Applications*, volume 50. John Wiley & Sons, 2011.

[31] A. Ferramosca, D. Limón, I. Alvarado, T.o Alamo, and E. F. Camacho. MPC for tracking of constrained nonlinear systems. In *Proc. of the IEEE Conference on Decision and Control*, pages 7978–7983, 2009.

[32] A. Ferramosca, D. Limon, I. Alvarado, and E. F. Camacho. Cooperative distributed MPC for tracking. *Automatica*, 49(4):906–914, 2013.

[33] A. Ferramosca, D. Limon, A. H. González, D. Odloak, and E. F. Camacho. MPC for tracking zone regions. *Journal of Process Control*, 20(4):506–516, 2010.

[34] B. A. Francis and W. M. Wonham. The internal model principle of control theory. *Automatica*, 12(5):457–465, 1976.

[35] G. F. Franklin, J. D. Powell, and A. E. Naeini. *Feedback Control of Dynamic Systems*. Prentice Hall, 1994.

[36] J. C. Geromel, J. Bernussou, G. Garcia, and M. C. DE Oliveira. H_2 and H_∞ robust filtering for discrete-time linear systems. *SIAM Journal on Control Optimization*, 38(5):1353–1368, 2000.

[37] R. Gondhalekar and C. N. Jones. MPC of constrained discrete-time linear periodic systems - A framework for asynchronous control: Strong feasibility, stability and optimality via periodic invariance. *Automatica*, 47(2):326–333, 2011.

[38] A. Grancharova, T. A. Johansen, and P. Tøndel. Computational aspects of approximate explicit nonlinear model predictive control. In *Assessment and Future Directions of Nonlinear Model Predictive Control*, pages 181–192. Springer, 2007.

[39] C. Grussler. Model reduction of positive systems. Master Thesis, 2012.

[40] V. Gupta, T. H. Chung, B. Hassibi, and R. M. Murray. On a stochastic sensor selection algorithm with applications in sensor scheduling and sensor coverage. *Automatica*, 42(2):251–260, 2006.

[41] H. M. Härdin and J. H. van Schuppen. Observers for linear positive systems. *Linear Algebra and Its Applications*, 425(2):571–607, 2007.

[42] M. L. J. Hautus. Strong detectability and observers. *Linear Algebra and Its Applications*, 50:353–368, 1983.

[43] J. L. Hellerstein, Y. Diao, S. Parekh, and D. M. Tilbury. *Feedback Control of Computing Systems*. John Wiley & Sons, 2004.

[44] T. Henningsson. Recursive state estimation for linear systems with mixed stochastic and set-bounded disturbances. In *Proc. of the 47th IEEE Conference on Decision and Control*, pages 678–683, 2008.

[45] J. P. Hespanha. *Linear Systems Theory*. Princeton University Press, 2009.

[46] J. P. Hespanha, D. Liberzon, D. Angeli, and E. D. Sontag. Nonlinear norm-observability notions and stability of switched systems. *IEEE Transactions on Automatic Control*, 50(2):154–168, 2005.

[47] J. Van Den Hof. *System theory and system identification of compartmental systems*. PhD thesis, University of Groningen, 1996.

[48] R. A. Horn and C. R. Johnson. *Matrix Analysis*. Cambridge University Press, 2012.

[49] M. Hou and P. C. Müller. Fault detection and isolation observers. *International Journal of Control*, 60(5):827–846, 1994.

[50] M. Hou and R. J. Patton. Optimal filtering for systems with unknown inputs. *IEEE Transactions on Automatic Control*, 43(3):445–449, 1998.

[51] M. Hou, A. C. Pugh, and P. C. Müller. Disturbance decoupled functional observers. *IEEE Transactions on Automatic Control*, 44(2):382–386, 1999.

[52] W. Huang, S. Ghosh, S. Velusamy, K. Sankaranarayanan, K. Skadron, and M. R. Stan. Hotspot: A compact thermal modeling methodology for early-stage VLSI design. *IEEE Transactions on Very Large Scale Integration Systems*, 14(5):501–513, 2006.

[53] M. F. Huber. Optimal pruning for multi-step sensor scheduling. *IEEE Transaction on Automatic Control*, 57(5):1338–1343, 2012.

[54] M. O. Jackson. *Social and Economic Networks*, volume 3. Princeton University Press, 2008.

[55] S. T. Jawaid and S. L. Smith. Submodularity and greedy algorithms in sensor scheduling for linear dynamical systems. *Automatica*, 61:282–288, 2015.

[56] Syed Talha Jawaid and Stephen L Smith. A complete algorithm for the infinite horizon sensor scheduling problem. In *Proc. of American Control Conference*, 2014. to appear.

[57] T. A. Johansen. Approximate explicit receding horizon control of constrained nonlinear systems. *Automatica*, 40(2):293–300, 2004.

[58] S. Joshi and S. Boyd. Sensor selection via convex optimization. *IEEE Trans. on Signal Processing*, 57(2):451–462, 2009.

[59] D. Kadjo, R. Ayoub, M. Kishinevsky, and P. V. Gratz. A control-theoretic approach for energy efficient CPU-GPU subsystem in mobile platforms. In *Proc. of the 52nd Annual Design Automation Conference*, page 62, 2015.

[60] H. Kellerer, U. Pferschy, and D. Pisinger. *Knapsack Problems*. Springer Science & Business Media, 2004.

[61] E. C. Kerrigan. Robust constraint satisfaction: Invariant sets and predictive control. PhD Thesis, University of Cambridge, 2001.

[62] L. Kocis and W. J. Whiten. Computational investigations of low-discrepancy sequences. *ACM Transactions on Mathematical Software*, 23(2):266–294, 1997.

[63] N. Komaroff. Upper bounds for the solution of the discrete Riccati equation. *IEEE Transactions on Automatic Control*, 37(9):1370–1373, 1992.

[64] N. Komaroff. Iterative matrix bounds and computational solutions to the discrete algebraic Riccati equation. *IEEE Transactions on Automatic Control*, 39(8):1676–1678, 1994.

[65] W. Kratz. Characterization of strong observability and construction of an observer. *Linear Algebra and Its Applications*, 221:31–40, 1995.

[66] A. Krause, A. Singh, and C. Guestrin. Near-optimal sensor placements in Gaussian processes: Theory, efficient algorithms and empirical studies. *The Journal of Machine Learning Research*, 9:235–284, 2008.

[67] M. Krichman, E. D. Sontag, and Y. Wang. Input-output-to-state stability. *SIAM Journal on Control and Optimization*, 39(6):1874–1928, 2001.

[68] J. Kurek. The state vector reconstruction for linear systems with unknown inputs. *IEEE Transactions on Automatic Control*, 28(12):1120–1122, 1983.

[69] W. H. Kwon, Y. S. Moon, and S. C. Ahn. Bounds in algebraic Riccati and Lyapunov equations: a survey and some new results. *International Journal of Control*, 64(3):377–389, 1996.

[70] P. Lancaster and L. Rodman. *Algebraic Riccati Equations*. Oxford University Press, 1995.

[71] J. Le Ny, E. Feron, and Munther A. Dahleh. Scheduling continuous-time Kalman filters. *IEEE Transactions on Automatic Control*, 56(6):1381–1394, 2011.

[72] C. H. Lee. Upper matrix bound of the solution for the discrete Riccati equation. *IEEE Transactions on Automatic Control*, 42(6):840–842, 1997.

[73] F. L. Lewis and V. L. Syrmos. *Optimal Control*. John Wiley & Sons, 1995.

[74] C. Li and N. Elia. Stochastic sensor scheduling via distributed convex optimization. *Automatica*, 58:173–182, 2015.

[75] H. Li and M. Fu. A linear matrix inequality approach to robust H_∞ filtering. *IEEE Transactions on Signal Processing*, 45(9):2338–2350, 1997.

[76] D. Limon, I. Alvarado, T. Alamo, and E. F. Camacho. MPC for tracking piecewise constant references for constrained linear systems. *Automatica*, 44(9):2382–2387, 2008.

[77] C. Lin, Q. Wang, and T. Lee. H_∞ output tracking control for nonlinear systems via TS fuzzy model approach. *IEEE Transactions on Systems, Man and Cybernetics*, 36(2):450–457, 2006.

[78] F. Lin, M. Fardad, and M. R. Jovanović. Sparse feedback synthesis via the alternating direction method of multipliers. In *Proc. of the American Control Conference*, pages 4765–4770, 2012.

[79] F. Lin, M. Fardad, and M. R. Jovanovic. Design of optimal sparse feedback gains via the alternating direction method of multipliers. *IEEE Transactions on Automatic Control*, 58(9):2426–2431, 2013.

[80] C. Liu, W. Chen, and J. Andrews. Tracking control of small-scale helicopters using explicit nonlinear MPC augmented with disturbance observers. *Control Engineering Practice*, 20(3):258–268, 2012.

[81] J. Liu and J. Zhang. The open question of the relation between square matrix's eigenvalues and its similarity matrix's singular values in linear discrete system. *International Journal of Control, Automation and Systems*, 9(6):1235–1241, 2011.

[82] Y. Liu, J. Slotine, and A.-L. Barabási. Controllability of complex networks. *Nature*, 473(7346):167–173, 2011.

[83] L. Lovász. Submodular functions and convexity. In *Mathematical Programming The State of the Art*, pages 235–257. Springer, 1983.

[84] P. Lu. Constrained tracking control of nonlinear systems. *Systems & Control Letters*, 27(5):305–314, 1996.

[85] D. G. Luenberger. *Introduction to Dynamical Systems: Theory, Models, and Applications*. Wiley, 1979.

[86] U. Maeder, F. Borrelli, and M. Morari. Linear offset-free model predictive control. *Automatica*, 45(10):2214–2222, 2009.

[87] I. Maurovic, M. Baotic, and I. Petrovic. Explicit model predictive control for trajectory tracking with mobile robots. In *Proc. of the IEEE/ASME International Conference on Advanced Intelligent Mechatronics*, pages 712–717, 2011.

[88] D. Q. Mayne and E. C. Kerrigan. Tube-based robust nonlinear model predictive control. In *Proc. of the IFAC Symposium on Nonlinear Control Systems*, pages 36–41, 2007.

[89] D. Q. Mayne, E. C. Kerrigan, E. J. Van Wyk, and P. Falugi. Tube-based robust nonlinear model predictive control. *International Journal of Robust and Nonlinear Control*, 21(11):1341–1353, 2011.

[90] D. Q. Mayne, J. B. Rawlings, C. V. Rao, and P. O. M. Scokaert. Constrained model predictive control: Stability and optimality. *Automatica*, 36(6):789–814, 2000.

[91] D. Q. Mayne, M. M. Seron, and S. V. Raković. Robust model predictive control of constrained linear systems with bounded disturbances. *Automatica*, 41(2):219–224, 2005.

[92] P. J. McCarthy, C. Nielsen, and S. L. Smith. Cardinality constrained robust optimization applied to a class of interval observers. In *Proc. of the American Control Conference*, pages 5337–5342, 2014.

[93] E. S. Meadows, M. A. Henson, J. W. Eaton, and J. B. Rawlings. Receding horizon control and discontinuous state feedback stabilization. *International Journal of Control*, 62(5):1217–1229, 1995.

[94] M. Milanese and A. Vicino. Optimal estimation theory for dynamic systems with set-membership uncertainty: an overview. *Automatica*, 27(6):997–1009, 1991.

[95] Y. Mo, R. Ambrosino, and B. Sinopoli. Sensor selection strategies for state estimation in energy constrained wireless sensor networks. *Automatica*, 47(7):1330–1338, 2011.

[96] Y. Mo, E. Garone, A. Casavola, and B. Sinopoli. False data injection attacks against state estimation in wireless sensor networks. In *Proc. of the 49th IEEE Conference on Decision and Control*, pages 5967–5972, 2010.

[97] Y. Mo and B. Sinopoli. False data injection attacks in control systems. In *Proc. of the 1st Workshop on Secure Control Systems*, 2010.

[98] L. Moreau. Stability of continuous-time distributed consensus algorithms. In *Proc. of the IEEE Conference on Decision and Control*, volume 4, pages 3998–4003, 2004.

[99] A. I. Mourikis and S. I. Roumeliotis. Optimal sensor scheduling for resource-constrained localization of mobile robot formations. *IEEE Transactions on Robotics*, 22(5):917–931, 2006.

[100] S. Murali, A. Mutapcic, D. Atienza, R. Gupta, S. Boyd, and G. De Micheli. Temperature-aware processor frequency assignment for MPSoCs using convex optimization. In *Proc. of the 5th IEEE/ACM/IFIP International Conference on Hardware/Software Codesign and System Synthesis*, pages 111–116, 2007.

[101] B. Noack, F. Pfaff, and U. D. Hanebeck. Optimal kalman gains for combined stochastic and set-membership state estimation. In *Proc. of the 51st IEEE Conference on Decision and Control*, pages 4035–4040, 2012.

[102] Y. Ohta, H. Maeda, and S. Kodama. Reachability, observability, and realizability of continuous-time positive systems. *SIAM Journal on Control and Optimization*, 22(2):171–180, 1984.

[103] S. Olaru and D. Dumur. Compact explicit MPC with guarantee of feasibility for tracking. In *Proc. of IEEE Conference on Decision and Control and European Control Conference*, pages 969–974, 2005.

[104] R. Oldenhuis and J. Vandekerckhove. GODLIKE - a robust single-& multi-objective optimizer. *MATLAB Central, Mathworks*, 2009.

[105] A. Olshevsky. Minimal controllability problems. *IEEE Transactions on Control of Network Systems*, 1(3):249–258, 2014.

[106] C. J. Ong, D. Sui, and E. G. Gilbert. Enlarging the terminal region of nonlinear model predictive control using the support vector machine method. *Automatica*, 42(6):1011–1016, 2006.

[107] G. Pannocchia and J. B. Rawlings. Disturbance models for offset-free model-predictive control. *AIChE Journal*, 49(2):426–437, 2003.

[108] S. Pequito, S. Kar, and A. P. Aguiar. A framework for structural input/output and control configuration selection of large-scale systems. *IEEE Transactions on Automatic Control*, 61(2):303–318, 2016.

[109] B. Polyak, M. Khlebnikov, and P. Shcherbakov. An LMI approach to structured sparse feedback design in linear control systems. In *Proc. of European Control Conference*, pages 833–838, 2013.

[110] A. Prakash, H. Amrouch, M. Shafique, T. Mitra, and J. Henkel. Improving mobile gaming performance through cooperative CPU-GPU thermal management. In *Proc. of the 53rd Annual Design Automation Conference*, page 47, 2016.

[111] L. Praly and Y. Wang. Stabilization in spite of matched unmodeled dynamics and an equivalent definition of input-to-state stability. *Mathematics of Control, Signals and Systems*, 9(1):1–33, 1996.

[112] G. V. Raffo, M. G. Ortega, and F. R. Rubio. An integral predictive/nonlinear H_∞ control structure for a quadrotor helicopter. *Automatica*, 46(1):29–39, 2010.

[113] M. A. Rami, F. Tadeo, and U. Helmke. Positive observers for linear positive systems, and their implications. *International Journal of Control*, 84(4):716–725, 2011.

[114] A. Rantzer. Distributed control of positive systems. In *Proc. of Joint 50th IEEE Conference on Decision and Control and European Control Conference*, pages 6608–6611, 2011.

[115] J. B. Rawlings and D. Q. Mayne. *Model Predictive Control: Theory and Design*. Nob Hill Publishing, 2009.

[116] A. J. Rojas. On the discrete-time algebraic Riccati equation and its solution in closed-form. In *Proc. of the 18th IFAC World congress*, pages 162–167, 2011.

[117] M. K. Sain and J. L. Massey. Invertibility of linear time-invariant dynamical systems. *IEEE Transactions on Automatic Control*, 14(2):141–149, 1969.

[118] F. Schweppe. Recursive state estimation: unknown but bounded errors and system inputs. *IEEE Transactions on Automatic Control*, 13(1):22–28, 1968.

[119] P. O. M. Scokaert, D. Q. Mayne, and J. B. Rawlings. Suboptimal model predictive control (feasibility implies stability). *IEEE Transactions on Automatic Control*, 44(3):648–654, 1999.

[120] M. E. Sezer and D. D. Siljak. On stability of interval matrices. *IEEE Transactions on Automatic Control*, 39(2):368–371, 1994.

[121] B. Shafai, S. Nazari, and A. Oghbaee. Positive unknown input observer design for positive linear systems. In *Proc. of the 19th International Conference on System Theory, Control and Computing*, pages 360–365, 2015.

[122] S. Sharifi, R. Ayoub, and T. S. Rosing. Tempomp: Integrated prediction and management of temperature in heterogeneous MPSoCs. In *Proc. of the Conference on Design, Automation and Test in Europe*, pages 593–598, 2012.

[123] S. Sharifi, C. Liu, and T. S. Rosing. Accurate temperature estimation for efficient thermal management. In *Proc. of the 9th International Symposium on Quality Electronic Design*, pages 137–142, 2008.

[124] R. Shorten, F. Wirth, and D. Leith. A positive systems model of TCP-like congestion control: Asymptotic results. *IEEE/ACM Transactions on Networking*, 14(3):616–629, 2006.

[125] Z. Shu, J. Lam, H. Gao, B. Du, and L. Wu. Positive observers and dynamic output-feedback controllers for interval positive linear systems. *IEEE Transactions on Circuits and Systems I*, 55(10):3209–3222, 2008.

[126] L. M. Silverman. Discrete Riccati equations: Alternative algorithms, asymptotic properties and system theory interpretations. *Control and Dynamic Systems*, 12:313–386, 1976.

[127] D. Simon. *Optimal State Estimation: Kalman, H_∞, and Nonlinear Approaches*. John Wiley & Sons, 2006.

[128] E. D. Sontag. Input to state stability. *The Control Systems Handbook: Control System Advanced Methods*, pages 1034–1054, 2011.

[129] E. D. Sontag and Y. Wang. Output-to-state stability and detectability of nonlinear systems. *Systems & Control Letters*, 29(5):279–290, 1997.

[130] T. Summers, F. Cortesi, and J. Lygeros. On submodularity and controllability in complex dynamical networks. *IEEE Transactions on Control of Network Systems*, 3(1):91–101, 2015. to appear.

[131] T. H. Summers and J. Lygeros. Optimal sensor and actuator placement in complex dynamical networks. In *Proc. of IFAC World Congress*, pages 3784–3789, 2014.

[132] S. Sundaram and C. N. Hadjicostis. Optimal state estimators for linear systems with unknown inputs. In *Proc. of the 45th IEEE Conference on Decision and Control*, pages 4763–4768, 2006.

[133] S. Sundaram and C. N. Hadjicostis. Delayed observers for linear systems with unknown inputs. *IEEE Transactions on Automatic Control*, 52(2):334–339, 2007.

[134] S. Sundaram and C. N. Hadjicostis. Partial state observers for linear systems with unknown inputs. *Automatica*, 44(12):3126–3132, 2008.

[135] S. Sundaram and C. N Hadjicostis. Distributed function calculation via linear iterative strategies in the presence of malicious agents. *IEEE Transaction on Automatic Control*, 56(7):1495–1508, 2011.

[136] A. Teixeira, I. Shames, H. Sandberg, and K. H. Johansson. A secure control framework for resource-limited adversaries. *Automatica*, 51:135–148, 2015.

[137] J. A. Tropp. Greed is good: Algorithmic results for sparse approximation. *IEEE Transactions on Information Theory*, 50(10):2231–2242, 2004.

[138] V. Tzoumas, A. Jadbabaie, and G. J. Pappas. Sensor placement for optimal Kalman filtering: Fundamental limits, submodularity, and algorithms. In *Proc. of American Control Conference*, 2016. to appear.

[139] V. Tzoumas, A. Jadbabaie, and G. J. Pappas. Sensor placement for optimal Kalman filtering: Fundamental limits, submodularity, and algorithms. *arXiv: 1509.08146*, 2016.

[140] 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*, pages 67–78, 2015.

[141] M. E. Valcher. State observers for discrete-time linear systems with unknown inputs. *IEEE Transactions on Automatic Control*, 44(2):397–401, 1999.

[142] M. Van De Wal and B. De Jager. A review of methods for input/output selection. *Automatica*, 37(4):487–510, 2001.

[143] J. Van Den Hof. Positive linear observers for linear compartmental systems. *SIAM Journal on Control and Optimization*, 36(2):590–608, 1998.

[144] M. P. Vitus, W. Zhang, A. Abate, J. Hu, and C. J. Tomlin. On efficient sensor scheduling for linear dynamical systems. *Automatica*, 48(10):2482–2493, 2012.

[145] Y. Wang and S. Boyd. Fast model predictive control using online optimization. *IEEE Transactions on Control Systems Technology*, 18(2):267–278, 2010.

[146] Y. Wang, M. Sznaier, and F. Dabbene. A convex optimization approach to worst-case optimal sensor selection. In *Proc. of 52nd IEEE Conference on Decision and Control*, pages 6353–6358, 2013.

[147] C. Yang, J. Wu, X. Ren, W. Yang, H. Shi, and L. Shi. Deterministic sensor selection for centralized state estimation under limited communication resource. *IEEE Transactions on Signal Processing*, 63(9):2336–2348, 2015.

[148] F. Zanini, D. Atienza, C. N. Jones, and G. De Micheli. Temperature sensor placement in thermal management systems for MPSoCs. In *Proc. of the IEEE International Symposium on Circuits and Systems*, pages 1065–1068, 2010.

[149] F. Zhang and Q. Zhang. Eigenvalue inequalities for matrix product. *IEEE Transactions on Automatic Control*, 51(9):1506–1509, 2006.

[150] L. Zhao, W. Zhang, J. Hu, A. Abate, and C. J. Tomlin. On the optimal solutions of the infinite-horizon linear sensor scheduling problem. *IEEE Transactions on Automatic Control*, 2014. to appear.