Foundations and Trends® in Systems and Control

Sparse Actuator Control of Discrete-Time Linear Dynamical Systems

Suggested Citation: Geethu Joseph (2024), "Sparse Actuator Control of Discrete-Time Linear Dynamical Systems", Foundations and Trends® in Systems and Control: Vol. 11, No. 3, pp 186–284. DOI: 10.1561/2600000033.

Geethu Joseph
Delft University of Technology
g.joseph@tudelft.nl

now
the essence of knowledge
Boston — Delft

Contents

Sparse Actuator Control of Discrete-Time Linear Dynamical Systems

Geethu Joseph

Delft University of Technology, Netherlands; g.joseph@tudelft.nl

ABSTRACT

This monograph presents some exciting and new results on the analysis and design of control of discrete-time linear dynamical systems using sparse actuator control. Sparsity constraints arise naturally in the inputs of several linear systems due to limited resources or the underlying physics. The monograph deals with two types of sparsity constraints: time-varying and time-invariant supported sparse control inputs. It first provides a detailed theoretical discussion on controllability under sparsity constraints, including algebraic necessary and sufficient conditions for ensuring controllability. Several related formulations, covering stabilizability, output controllability, and nonnegative controllability under sparsity constraints, are also presented. Further, for sparsely controllable systems, the monograph describes two efficient, systematic, and rigorous approaches to designing sparse control inputs: compressed sensing algorithms and spare actuator scheduling algorithms. Overall, the concepts covered in the monograph provide various sparsity models, algorithms, and analysis tools that are readily accessible to

Geethu Joseph (2024), "Sparse Actuator Control of Discrete-Time Linear Dynamical Systems", Foundations and Trends® in Systems and Control: Vol. 11, No. 3, pp 186–284. DOI: 10.1561/2600000033.

systems and control, signal processing, and applied mathematics readers.

1

Introduction

This section introduces the notion of sparse actuator control, convinces the reader of its importance via a few applications, and provides an overview of the content of the monograph.

1.1 What is Sparse Actuator Control?

Sparse actuator control refers to a control signal that is sparse in actuator use, i.e., we use a small subset of actuators among the available ones. In this monograph, we focus on the control of discrete-time linear dynamical systems using a few actuators.

Linear dynamical systems are well-studied and widely accepted mathematical models for describing and analyzing various control systems that evolve over time. The model serves as the core engine in diverse areas such as control systems, signal processing, communications, etc. We represent a discrete linear dynamical system using the state space model,

$$\boldsymbol{x}_{k+1} = \boldsymbol{A}\boldsymbol{x}_k + \boldsymbol{B}\boldsymbol{u}_k, \tag{1.1}$$

for discrete-time indices $k = 1, 2, \ldots$. Here, $\boldsymbol{x}_k$ denotes the state vector at time k. The temporal evolution of the system state is determined by the state (transition) matrix $\boldsymbol{A}$ and the input matrix $\boldsymbol{B}$. The state $\boldsymbol{x}_{k+1}$ at a given time index $k+1$ depends linearly on the previous state $\boldsymbol{x}_k$ and is influenced by the input $\boldsymbol{u}_k$ applied at time k.

Numerous practical control problems deal with the task of designing control inputs $\boldsymbol{u}_k$'s to drive the system to a desired state. Typically, the design of inputs is constrained by the energy or steady-state error requirements and the level of control stability. These problems are generally posed as (convex) optimization problems and solved using techniques like least squares. This conventional control input design utilizes all the actuators or input variables (entries of control inputs $\boldsymbol{u}_k$'s). However, several resource-aware control systems demand simpler designs where only a subset of input variables are used to control the system. Simplicity is hard to achieve and makes the design problem more challenging. Mathematically, the simplicity can be encoded using the notion of sparsity. The research area associated with this phenomenon is known as *sparse actuator control.*

A vector is said to be sparse if it contains a lot of zeros entries compared to its dimension (length). Sparse actuator control of a discrete linear dynamical system deals with control inputs having very few nonzeroes entries (or active actuators) compared to their dimension. The index set of nonzero entries of a vector is defined as its support. In this monograph, we focus on control inputs whose support set cardinality is small compared to its dimension.

The sparsity-promoting strategies considered in the literature are divided into two categories. The first strategy, called the *time-varying support case*, allows the use of different subsets of input variables at different time indices to steer the system state. In the second strategy called *time- invariant support case*, the controller identifies a subset of input variables and uses the same subset at all times to control the system state. Clearly, the second strategy is more restricted and a special case of the first strategy. Further, in both cases, it may not be able to drive the system to a desired state because of the restrictions on the control inputs. We underscore that the main difficulty here is the identification of the small subsets of inputs that can drive the system to the desired state. If the subset is known, one can ignore the columns of the input matrix corresponding to the zero entries and reduce the control design problem to the standard control design problem. The subset identification is a combinatorial problem, and the sparsity constraint is non-convex. Consequently, the analysis and design of sparse actuator control are by far not trivial. This monograph is devoted to the fundamental limits, mathematical tools, and algorithms for the sparse actuator control of linear dynamical systems.

1.2 Motivation

Sparsity constraints naturally arise in several control systems. In this section, we point to a selection of control problems that can be modeled using linear dynamical systems where the input is constrained to be sparse due to cost and energy depletion issues. The section establishes the significance of the research topic of sparse actuator control and motivates the need to study it.

1.2.1 Communication-aware Control

A networked control system refers to a large system where the controlled object(s) and the controller communicate through a communication network. For example, consider a drone controlled by a centralized ground controller. The drone sends its sensor data to the ground controller, and based on this information, the controller sends out new control commands to the drone to adjust its position, velocity, and acceleration via the network. Such communication networks are often bandwidth-limited, which motivates the use of sparse control inputs (Heemels *et al.*, 2010; Tatikonda and Mitter, 2004; Liu *et al.*, 2020; Nagahara and Quevedo, 2011). The reason for promoting sparsity is that sparse signals admit compact representations (Foucart *et al.*, 2013), leading to a lower communication burden.

1.2.2 Network Opinion Manipulation

Consider a social network where people interact with each other and influence each other's opinions on a product or idea. For example, a social network can be people living in a specific area or social media communities, where a group of people share common interests and experiences. Also, network opinion can refer to a movie rating, inclination towards a political party, or customer rating of a product. The opinion evolution over time can be represented using a linear dynamical system whose state is the network opinion (opinion of all the individuals in the network), and the state matrix models the influence of each individual's opinion on others' opinions. The network information is manipulated by external agents such as paid bloggers, social media influencers, marketing agents, etc. Their influence can be modeled as an input applied to the dynamical system that models the opinion evolution. Further, the agents are often constrained by budget (financial or physical), which can be represented using sparse inputs, where sparsity denotes the budget constraints of the agent. As an example, consider a company that sends a salesperson to market their products by offering free samples. The number of free samples is limited, and not all the samples reach the target people at the same time. The influence of such manipulators can be modeled using sparse inputs whose support denotes the individuals who receive the free sample at a given time (Joseph *et al.*, 2021).

1.2.3 Malicious Data Injection Attacks

In an electric power network, malfunctioning or compromised devices, such as power system stabilizers, generator controllers and exciters, and cyclic loads, can inject forced oscillations (0.1–15 Hz) into the network. The sources triggering these oscillations are fewer compared to the potential sources (Anguluri *et al.*, 2023; Anguluri *et al.*, 2022; Siami *et al.*, 2020). Hence, the effect of anomalous sources can be modeled as sparse control inputs to the system. Further, sparse inputs can represent data injection attacks that target a limited number of sensors in the smart grid (Cárdenas *et al.*, 2008; Hao *et al.*, 2015; Sun and Li, 2022; Chen *et al.*, 2019). Similarly, sparse inputs can represent malicious attacks on cyber-physical systems (Ma and Shi, 2022; Tsang *et al.*, 2020).

1.2.4 Efficient Control in Biological Networks

Human metabolism is generally represented by directed networks. Some example biological networks are motivated by the application of control-theoretic ideas in the analysis of biological circuits (Marucci *et al.*, 2009), biochemical reaction networks (Liu *et al.*, 2013), and systems biology (Rajapakse *et al.*, 2012). Consider a directed network where the nodes model reactions and/or metabolites. The network can be externally affected by drugs that act only on a few nodes in the network. Here, sparse inputs target to reduce the adverse side effects due to drug administration and it motivates the need to control the system using sparse inputs.

Overall, the need for sparse actuator control stems from the system's cost constraints or simply from the physics of how the system operates. Therefore, using a limited number of actuators is desirable without significantly compromising the control performance, for example, in terms of the time or energy required to reach a certain desired state.

1.3 Overview of the Monograph

This monograph is organized into five sections (excluding this introductory section) that provide a detailed study of sparse actuator control. The problem formulation and associated analysis are readily accessible to signal processing, control/systems theory, and applied mathematics communities.

Section 2 formally introduces the notion of sparse actuator control with time-varying support and defines the notion of controllability under the sparsity constraint. The central questions of the section are as follows:

> *What are necessary and sufficient conditions for ensuring controllability under sparse inputs with possibly different supports? Can we devise a computationally simple test for sparse controllability? If a system is controllable using sparse inputs, how many control input vectors are needed to drive the system from a given initial state to an arbitrary final state?*

We show that, for any linear dynamical system controllable under the sparsity constraint, a sparse actuator schedule independent of the system state exists, which can drive the system to any desired state. Further, we derive simple algebraic conditions, which are both necessary and sufficient for the sparse controllability of the system. We show that the system is sparse controllable if and only if it is controllable *and* the sparsity level exceeds the nullity of the state matrix. Unlike the more traditional Kalman-type rank tests, the derived conditions can be verified in polynomial time complexity. Finally, we characterize the time-to-control or the minimum number of input vectors required to ensure sparse controllability and show that it is bounded by the state dimension. These results form a theoretical basis for designing sparse control inputs, which we discuss in the next section.

Section 3 addresses the design of sparse control with time-varying support for a given linear dynamical system. This section seeks an answer to the following question:

> *Given a controllable linear dynamical system, how do we design sparse inputs that take the system from a given initial state to a desired final state?*

We formulate the sparse control input design in two ways. In the first approach, we formulate it as a sparse recovery problem and use the compressed sensing algorithms to solve the problem. This approach does not necessarily assume that the system is controllable, but the corresponding sparse actuator schedule depends on the initial and final states. In the second approach, we assume that the system is controllable. We design a global sparse actuator schedule that applies to any pair of initial and final states and then derive the control inputs based on the designed schedule.

Section 4 extends the idea of sparse control to other related control theory notions. The focal question addressed in the section is as follows:

> *How are the stabilizability, output controllability, and nonnegative controllability of a linear dynamical system affected by the sparsity constraints on the input?*

We show three key results in this section. The first result, perhaps surprisingly, shows that sparsity constraints do not have any effect on the stabilizability. All stabilizable systems are sparse stabilizable. The second result is on the algebraic characterization of sparse output controllability. We derive bounds on minimal sparsity levels that ensure output controllability. Finally, we show that any sparse controllable and nonnegative controllable systems are sparse nonnegative controllable. We also briefly discuss three sets of design algorithms: the first estimates sparse control inputs for system stabilization, the second estimates sparse control inputs to achieve a desired output, and the third estimates nonnegative sparse control inputs to reach a desired state.

Section 5 looks at a more stringent sparsity constraint, where all the sparse inputs have nonzero entries at the same indices. So, the section deals with the question:

> *If a linear dynamical system can be controlled using only a few actuators among the available ones, how do we choose the actuators and design the corresponding control inputs?*

We prove that the problem of finding the minimum sparsity level to make the system controllable under time-invariant support is an NP hard problem. Nonetheless, there are several approximate design algorithms that can choose a small number of actuators to control the system. There are two design approaches: one is compressed sensing algorithms that are initial and final state-dependent, and the second approach is state-independent actuator scheduling-based.

Section 6 summarizes the open problems or questions on the new area of sparse actuator control, points to the weaknesses that still have to be strengthened, and offers some concluding remarks.

Each section ends with a subsection titled "Notes" where we provide supplementary facts, additional comments, and some open questions.

Notation is usually introduced when it is used for the first time. The collection of symbols used in the text can be found on page 273.

1.4 Out of the Scope Topics

This monograph is by no means exhaustive but presents some notable recent research connecting linear dynamical systems and sparse inputs. Other important problems related to sparsity and control have been studied in the literature but are outside the scope of this monograph. We briefly discuss a few of them below.

1.4.1 Sparsity in Time or Maximum Hands-Off

An important and widely studied control paradigm related to sparsity is known as maximum hands-off control (Nagahara *et al.*, 2015). This approach is characterized by applying zero control for most of the time, resulting in minimal active periods or the shortest active duration. Since actuators remain inactive for extended periods, this method significantly reduces fuel consumption, power usage, and communication burden. It is worth noting that this strategy leverages sparsity over time, while this monograph focuses on sparsity across actuator use.

The concept of time sparsity is modeled using the ℓ_0-norm, which serves as a penalty function to measure the duration of the control signal's active support. However, the ℓ_0-norm is challenging to optimize directly and is typically approximated using its convex relaxation, the ℓ_1-norm. Their equivalence holds under an assumption called normality. Most research in this area centers on continuous-time systems, though these ideas are also extended to discrete-time models (Mai and Yin, 2023).

Sparsity methods from compressed sensing and their applicability to systems and control, covering standard sparsity methods in finite-dimensional vector spaces and optimal control methods in infinite-dimensional function spaces, has been extensively covered in Nagahara (2020) and Nagahara (2023). The idea of maximum hands-off control has been extended to general linear systems (Ikeda and Kashima, 2018; Chatterjee *et al.*, 2016; Nagahara *et al.*, 2016; Ito *et al.*, 2021), and time-varying systems (Mai and Yin, 2024). Additionally, when the normality assumption does not hold, non-convex penalty functions for promoting sparsity have also been explored in the literature (Ikeda, 2024).

1.4.2 Sparse Input Estimation From Observations

The design of sparse control inputs is closely related to the problem of sparse input estimation in linear dynamical systems. In many cases, the initial state of the system, whether sparse or non-sparse, is also unknown. By leveraging the sparsity in the system, this problem can be framed as finding the sparse solution to a linear system of equations. Various compressed sensing approaches, such as basis pursuit, sparse Bayesian learning, reweighed-ℓ_1, and reweighed ℓ_2, have been used to estimate the sparse control inputs. The algorithms and guarantees for sparse input estimation from observations, as well as other variants of the problem, are discussed in Sefati *et al.* (2015), Kafashan *et al.* (2016), Fosson *et al.* (2019), and Chakraborty *et al.* (2024) and their references.

Despite similarities and a shared compressed sensing-inspired approach to the solution, sparse control design and sparse input estimation differ significantly. In the sparse input estimation problem, the output trajectory of the system is already known, and the goal is to estimate the sparsest inputs that can drive the system along a given trajectory. In contrast, the sparse control problem does not have a predefined trajectory. Instead, the objective is to find sparse inputs while also considering system expenditures such as the energy budget, making the problems distinct.

1.4.3 Sparsity in Feedback

Another area of interest is enforcing sparsity in the controller's feedback gain matrix. Some variants of this approach focus on maximizing the number of nonzero rows in the feedback gain matrix to make the state feedback vector sparse (Polyak *et al.*, 2014; Arastoo *et al.*, 2016). This approach helps reduce the bandwidth requirement when feedback is communicated to the plant via a wireless link. Another variant aims to minimize the number of nonzero entries in the feedback matrix (Lin *et al.*, 2013; Fardad and Jovanović, 2014; Babazadeh and Nobakhti, 2016). This strategy reduces the number of communication links between the many components of large-scale and networked control systems.

1.4.4 Sparse Sensing and Sparse Observability

Since sensing problems are dual counterparts of control problems, sparse sensing or sensor scheduling represents a related area of study. In such contexts, the literature often distinguishes between two approaches: myopic strategies, which prioritize the immediate effects of selected actuators, and non-myopic strategies, which take a forward-looking perspective alongside immediate impacts (Hashemi *et al.*, 2020; Ballotta *et al.*, 2020; Vafaee and Siami, 2024).

Sensor scheduling differs significantly, as its performance metrics typically include estimation error, computational constraints, and transmission delay (especially in networked systems). In contrast, control design primarily focuses on minimizing energy usage and adhering to time-to-control constraints. Some studies have also explored joint actuator-sensor selection (Ye *et al.*, 2022), which is also not the focus here.

Another related topic includes the observability of linear systems when the initial state is sparse (Dai and Yüksel, 2013; Sanandaji *et al.*, 2014; Joseph and Murthy, 2018; Joseph and Murthy, 2019). However, sparse actuator controllability assumes a general initial state and sparse control inputs, demanding distinct analyses.

1.4.5 Minimal Input Set for Structured Systems

Certain studies have explored minimal input selection for structured systems. The primary objective is to identify a minimal cardinality set of inputs that ensures a structurally controllable network (Chapman and Mesbahi, 2013; Trefois and Delvenne, 2015). Additionally, other research has investigated methods to adjust the system configuration in order to reduce the size of the minimal input set (Abbas *et al.*, 2023; Joseph *et al.*, 2023).

While these studies address sparsity, they require graph theory-based analysis and the concept of zero-forcing sets specific to structural systems, which are not directly related to the topic of this monograph.

Interested readers are referred to the above works and references therein for extensive treatment of the topics.

1.5 Notes

Linear dynamical systems are extensively applied across numerous fields, including control systems (Zhou *et al.*, 1996), signal processing (Anderson and Moore, 2005), communications (Prasad *et al.*, 2014), economics (Brockwell *et al.*, 2002), mechanical and civil engineering (Pope III *et al.*, 2002; Shao *et al.*, 2006), and healthcare (Neumann *et al.*, 2009; Hvistendahl *et al.*, 2013). The study of linear dynamical systems with sparsity constraints dates back to 1972 (Athans, 1972). Recent research has focused on the problem of identifying sequences of sparse control inputs, for both fixed and time-varying sets of control nodes, as well as other related challenges, all of which are explored in detail in this monograph.

Several sparse control design algorithms covered in this monograph are inspired by the field of compressed sensing, particularly in Sections 3 and 5. Compressed sensing, also known as compressive sensing, compressive sampling, sparse sampling, or sparse signal recovery, is a signal processing technique that allows for efficient acquisition and reconstruction of signals by solving an underdetermined linear system. This method makes use of the principle of sparsity, enabling the recovery of signals from significantly fewer samples than traditionally required by the Nyquist-Shannon sampling theorem through optimization techniques.

Since the advent of compressed sensing theory in the early 2000s, the field has introduced a plethora of computationally efficient algorithms and sophisticated analytical tools to handle sparsity. The initial steps in this field were marked by seminal papers that combined ℓ_1-norm minimization with randomness in the measurement matrices (Candès *et al.*, 2006; Donoho, 2006). These foundational works paved the way for a robust framework that has significantly influenced various applications in signal processing and beyond. For thorough overviews of compressed sensing, refer to the works of Baraniuk (2007), Wakin *et al.* (2008), Kutyniok (2013), Fornasier and Rauhut (2015), and Foucart *et al.* (2013).

Moreover, as discussed in Section 1.2.1, sparse control offers compact representations. The advances in compressed sensing have made it possible to represent sparse vectors with fewer samples compared to non-sparse vectors, thereby reducing the communication burden. This advantage is particularly significant in systems where communication efficiency and bandwidth are critical. By minimizing the amount of data required for accurate signal representation and transmission, compressed sensing facilitates more efficient and effective control strategies in various cyber-physical applications.

2

Controllability of Sparse Inputs with Time-Varying Support

In this section, we mathematically introduce the notion of sparse control with time-varying support and discuss how the sparsity constraint affects the controllability of linear dynamical systems. Specifically, we characterize the controllability and time-to-control of the system under sparsity constraints.

2.1 Sparse Control With Time-Varying Support

We first define the notion of sparsity and support set.

Definition 2.1. The support of a vector $\boldsymbol{u} \in \mathbb{R}^m$, denoted as $\text{supp}(\boldsymbol{u})$ is the index set of its nonzero values,

$$\text{supp}(\boldsymbol{u}) = \{j : \boldsymbol{u}_j \neq 0\}. \tag{2.1}$$

A vector is said to be s-sparse if it has at most s nonzero entries,

$$\|\boldsymbol{u}\|_0 = |\text{supp}(\boldsymbol{u})| \leq s, \tag{2.2}$$

where $\|\cdot\|_0$ is the ℓ_0 norm that counts the nonzero entries of a vector.

We next define sparse control with time-varying support. For this, we recall the state space model of the linear dynamical system $(\boldsymbol{A} \in \mathbb{R}^{n\times n}, \boldsymbol{B} \in \mathbb{R}^{n\times m})$, with state $\boldsymbol{x}_k \in \mathbb{R}^n$ and input $\boldsymbol{u}_k \in \mathbb{R}^m$,

$$\boldsymbol{x}_{k+1} = \boldsymbol{A}\boldsymbol{x}_k + \boldsymbol{B}\boldsymbol{u}_k, \tag{2.3}$$

for discrete-time indices $k = 1, 2, \ldots$. Here, we impose the constraint that the inputs are s-sparse for a given sparsity level, $\|\boldsymbol{u}_k\|_0 \leq s$, for $k = 1, 2, \ldots$.

This section is dedicated to analyzing the controllability and time-to-control of the system under sparsity constraints. We start with the classical notion of controllability and associated results.

2.2 Classical Controllability Theory

An important aspect of the design of control systems is the ability to drive the system from any initial state to any desired state, called controllability. Formally, the linear dynamical system $(\boldsymbol{A}, \boldsymbol{B})$ in (2.3) is said to be controllable if, for any initial state $\boldsymbol{x}_1$ and any final state $\boldsymbol{x}_{\mathrm{f}}$, there exist a time index $K < \infty$ and inputs $\{\boldsymbol{u}_k,\ k = 1, 2, \ldots, K\}$ that steer the system from the state $\boldsymbol{x}_1$ to $\boldsymbol{x}_{K+1} = \boldsymbol{x}_{\mathrm{f}}$. Controllability is critical in the analysis and control of systems as it ensures the feasibility of designing control inputs that can achieve the desired system performance. Further, from the state space model in (2.3), we have

$$\boldsymbol{x}_{K+1} = \boldsymbol{A}^K\boldsymbol{x}_1 + \boldsymbol{\Phi}^{(K)}\boldsymbol{u}^{(K)}, \tag{2.4}$$

where $\boldsymbol{\Phi}^{(K)} \in \mathbb{R}^{n\times Km}$ is defined as

$$\boldsymbol{\Phi}^{(K)} = \begin{bmatrix}\boldsymbol{A}^{K-1}\boldsymbol{B} & \boldsymbol{A}^{K-2}\boldsymbol{B} & \ldots & \boldsymbol{B}\end{bmatrix}, \tag{2.5}$$

and $\boldsymbol{u}^{(K)} \in \mathbb{R}^{Km}$ is formed by concatenating the K input vectors,

$$\boldsymbol{u}^{(K)} = \begin{bmatrix}\boldsymbol{u}_1 \\ \boldsymbol{u}_2 \\ \vdots \\ \boldsymbol{u}_K\end{bmatrix}. \tag{2.6}$$

The controllability of a linear dynamical system can be tested using the classical Kalman rank test or, equivalently, using the Popov-Belevitch-Hautus (PBH) test.

Theorem 2.1 (Chen, 1984). The linear dynamical system ($\boldsymbol{A} \in \mathbb{R}^{n\times n}, \boldsymbol{B} \in \mathbb{R}^{n\times m}$) in (2.3) is controllable if and only if it satisfies the Kalman rank condition given by

$$\operatorname{rank}\left\{\begin{bmatrix}\boldsymbol{A}^{n-1}\boldsymbol{B} & \boldsymbol{A}^{n-2}\boldsymbol{B} & \ldots & \boldsymbol{B}\end{bmatrix}\right\} = n. \tag{2.7}$$

Equivalently, the linear dynamical system $(\boldsymbol{A}, \boldsymbol{B})$ in (2.3) is controllable if and only if it satisfies the PBH rank condition given by

$$\operatorname{rank}\left\{\begin{bmatrix}\boldsymbol{A} - \lambda \boldsymbol{I} & \boldsymbol{B}\end{bmatrix}\right\} = n, \quad \forall\, \lambda \in \mathbb{C}. \tag{2.8}$$

For controllable linear dynamical systems, we define the controllability index as follows. From the Kalman rank condition in (2.7), we infer that for a controllable system $(\boldsymbol{A}, \boldsymbol{B})$, the controllability matrix $\boldsymbol{\Phi}^{(n)} \in \mathbb{R}^{n\times nm}$ in (2.5) in has n linearly independent columns and the set of independent columns may not be unique. The controllability index refers to the smallest integer K^* such that $\boldsymbol{\Phi}^{(K^*)} \in \mathbb{R}^{n\times K^* m}$ has full row rank, i.e.,

$$K^* = \min_{k=1,2,\ldots} k \text{ s. t. } \operatorname{rank}\left\{\boldsymbol{\Phi}^{(k)}\right\} = n. \tag{2.9}$$

Therefore, K^* denotes the least number of control inputs that can steer the system from any initial state to any desired final state at $k = K^*$. In other words, the controllability index, also called *time-to-control*, characterizes the minimum time required to steer the dynamic system towards a desired state. The controllability index is bounded as follows:

Theorem 2.2 (Chen, 1984). The controllability index K^* defined in (2.9) of a controllable linear dynamical system ($\boldsymbol{A} \in \mathbb{R}^{n\times n}, \boldsymbol{B} \in \mathbb{R}^{n\times m}$) in (2.3) is bounded as

$$\frac{n}{\operatorname{rank}\{\boldsymbol{B}\}} \leq K^* \leq n - \operatorname{rank}\{\boldsymbol{B}\} + 1. \tag{2.10}$$

We next discuss how the above results from the classical control theory change when we introduce the sparsity constraints described in Section 2.1.

2.3 Sparse Controllability

We define the modified version of controllability under sparsity constraints as follows.

Definition 2.2. The linear dynamical system ($\boldsymbol{A} \in \mathbb{R}^{n\times n}, \boldsymbol{B} \in \mathbb{R}^{n\times m}$) in (2.3) is said to be s-sparse controllable for a given $s \geq 1$ if, for any initial state $\boldsymbol{x}_1 \in \mathbb{R}^n$ and any final state $\boldsymbol{x}_{\mathrm{f}} \in \mathbb{R}^n$, there exist a time index $K < \infty$ and inputs $\{\boldsymbol{u}_k \in \mathbb{R}^m : \|\boldsymbol{u}_k\|_0 \leq s\ k = 1, 2, \ldots, K\}$ that steer the system from the state $\boldsymbol{x}_1$ to final state $\boldsymbol{x}_{K+1} = \boldsymbol{x}_{\mathrm{f}}$.

We first derive a controllability test similar to the Kalman rank test for sparse controllability.

Theorem 2.3 (Joseph and Murthy, 2021)**.** The linear dynamical system ($\boldsymbol{A} \in \mathbb{R}^{n\times n}, \boldsymbol{B} \in \mathbb{R}^{n\times m}$) in (2.3) is s-sparse controllable if and only if there exist an integer $K < \infty$ and sets $\{\mathcal{S}_k \subseteq \{1, 2, \ldots, m\},\ |\mathcal{S}_k| \leq s\}_{k=1}^K$ such that

$$\operatorname{rank}\left\{\begin{bmatrix}\boldsymbol{A}^{K-1}\boldsymbol{B}_{\mathcal{S}_1} & \boldsymbol{A}^{K-2}\boldsymbol{B}_{\mathcal{S}_2} & \ldots & \boldsymbol{B}_{\mathcal{S}_K}\end{bmatrix}\right\} = n, \tag{2.11}$$

where $\boldsymbol{B}_{\mathcal{S}_k} \in \mathbb{R}^{n\times|\mathcal{S}_k|}$ is the submatrix formed by the columns of $\boldsymbol{B}$ indexed by $\mathcal{S}_k$, for any $k = 1, 2, \ldots, K$.

Proof. From Definition 2.2, we see that the system is s-sparse controllable if and only if for any vectors $\boldsymbol{x}_1$ and $\boldsymbol{x}_{\mathrm{f}}$, there exist an integer $K < \infty$ and $\{\boldsymbol{u}_k \in \mathbb{R}^m : \|\boldsymbol{u}_k\|_0 \leq s\}_{k=1}^K$ such that

$$\boldsymbol{x}_{K+1} - \boldsymbol{A}^K\boldsymbol{x}_1 = \begin{bmatrix}\boldsymbol{A}^{K-1}\boldsymbol{B} & \boldsymbol{A}^{K-2}\boldsymbol{B} & \ldots & \boldsymbol{B}\end{bmatrix}\begin{bmatrix}\boldsymbol{u}_1\\ \boldsymbol{u}_2\\ \vdots\\ \boldsymbol{u}_K\end{bmatrix}. \tag{2.12}$$

So, the system is s-sparse controllable if and only if for any vectors $\boldsymbol{x}_1$ and $\boldsymbol{x}_{\mathrm{f}}$, there exist an integer $K < \infty$ and support sets $\{\mathcal{S}_k \subseteq \{1, 2, \ldots, m\},\ |\mathcal{S}_k| \leq s\}_{k=1}^K$, and $\tilde{\boldsymbol{u}} \in \mathbb{R}^{\sum_{k=1}^K |\mathcal{S}_k|}$ such that

$$\boldsymbol{x}_{K+1} - \boldsymbol{A}^K\boldsymbol{x}_1 = \begin{bmatrix}\boldsymbol{A}^{K-1}\boldsymbol{B}_{\mathcal{S}_1} & \boldsymbol{A}^{K-2}\boldsymbol{B}_{\mathcal{S}_2} & \ldots & \boldsymbol{B}_{\mathcal{S}_K}\end{bmatrix}\tilde{\boldsymbol{u}}. \tag{2.13}$$

Hence, the system is s-sparse controllable if and only if

$$\mathbb{R}^n = \bigcup_{\substack{\{\mathcal{S}_k \subseteq \{1,2,\ldots,m\},\\ |\mathcal{S}_k| \leq s\}_{k=1}^K}} \operatorname{ColSpace}\left\{\begin{bmatrix}\boldsymbol{A}^{K-1}\boldsymbol{B}_{\mathcal{S}_1} & \boldsymbol{A}^{K-2}\boldsymbol{B}_{\mathcal{S}_2} & \ldots & \boldsymbol{B}_{\mathcal{S}_K}\end{bmatrix}\right\}, \tag{2.14}$$

where ColSpace$\{\cdot\}$ denotes the column space of a matrix. Further, the above relation implies that the union of a finite number of vector spaces is another vector space, which happens if and only if one of the vector spaces contains every other vector space. As a result, Statement (a) holds if and only if there exist an integer $K < \infty$ and support sets $\{\mathcal{S}_k \subseteq \{1, 2, \ldots, m\},\ |\mathcal{S}_k| \leq s\}_{k=1}^{K}$, such that

$$\mathbb{R}^n = \text{ColSpace}\left\{\left[\boldsymbol{A}^{K-1}\boldsymbol{B}_{\mathcal{S}_1} \quad \boldsymbol{A}^{K-2}\boldsymbol{B}_{\mathcal{S}_2} \quad \ldots \quad \boldsymbol{B}_{\mathcal{S}_K}\right]\right\}, \tag{2.15}$$

which is equivalent to the desired sparse controllability test. Thus, the proof is complete. □

The above result gives an interesting insight into sparse control with time-varying support. By definition, the support set at any time index can vary depending on the initial and final states. However, Theorem 2.3 shows that if the system is sparse controllable, there exists an actuator schedule independent of the initial and final states that can steer the system using fewer actuators. Further, Theorem 2.3 implies the invariance of sparse controllability under equivalent transformation.

Corollary 2.4. The s-sparse controllability property of a linear dynamical system ($\boldsymbol{A} \in \mathbb{R}^{n\times n}, \boldsymbol{B} \in \mathbb{R}^{n\times m}$) is invariant under reordering of $\boldsymbol{B}$ and equivalent transformation.

Also, Theorem 2.3 provides insights into sparse controllability for the special cases when $\boldsymbol{B}$ is not full rank.

Corollary 2.5. A controllable linear dynamical system ($\boldsymbol{A} \in \mathbb{R}^{n\times n}, \boldsymbol{B} \in \mathbb{R}^{n\times m}$) is s-sparse controllable if $s > \text{rank}\{\boldsymbol{B}\}$. In particular, when $\text{rank}\{\boldsymbol{B}\} = 1$, sparse controllability is equivalent to controllability.

Proof. If $s > \text{rank}\{\boldsymbol{B}\}$, then we can find s columns of $\boldsymbol{B}$ indexed by $\mathcal{S}$ that span its column space. Then, using Theorem 2.1, we have

$$n = \text{rank}\left\{\left[\boldsymbol{A}^{K-1}\boldsymbol{B} \quad \boldsymbol{A}^{K-2}\boldsymbol{B} \quad \ldots \quad \boldsymbol{B}\right]\right\} \tag{2.16}$$

$$= \text{rank}\left\{\left[\boldsymbol{A}^{K-1}\boldsymbol{B}_{\mathcal{S}} \quad \boldsymbol{A}^{K-2}\boldsymbol{B}_{\mathcal{S}} \quad \ldots \quad \boldsymbol{B}_{\mathcal{S}}\right]\right\}. \tag{2.17}$$

Then, by Theorem 2.3, the system is s-sparse controllable. □

Although Theorem 2.3 gives a sparse controllability test, this condition can not be verified easily because the number of possible support sets is exponential in n. Moreover, similar to the Kalman rank test, the above test also requires us to compute different powers of $\boldsymbol{A}$, which can lead to numerical instability. So, we next present a simpler condition to test sparse controllability.

Theorem 2.6 (Joseph and Murthy, 2021)**.** For the linear dynamical system ($\boldsymbol{A} \in \mathbb{R}^{n\times n}, \boldsymbol{B} \in \mathbb{R}^{n\times m}$) in (2.3), the following statements are equivalent:

(a) The system $(\boldsymbol{A}, \boldsymbol{B})$ is s-sparse controllable (see Definition 2.2).

(b) There exist an integer $K < \infty$ and sets $\{\mathcal{S}_k \subseteq \{1, 2, \ldots, m\}, |\mathcal{S}_k| \leq s\}_{k=1}^{K}$ such that

$$\operatorname{rank}\left\{\begin{bmatrix}\boldsymbol{A}^{K-1}\boldsymbol{B}_{\mathcal{S}_1} & \boldsymbol{A}^{K-2}\boldsymbol{B}_{\mathcal{S}_2} & \ldots & \boldsymbol{B}_{\mathcal{S}_K}\end{bmatrix}\right\} = n. \tag{2.18}$$

(c) For any nonzero $\lambda \in \mathbb{C}$, we have $\operatorname{rank}\left\{\begin{bmatrix}\boldsymbol{A} - \lambda\boldsymbol{I} & \boldsymbol{B}\end{bmatrix}\right\} = n$ and there exists a set $\mathcal{S} \subseteq \{1, 2, \ldots, m\}$ such that $1 \leq |\mathcal{S}| \leq s$ and $\operatorname{rank}\left\{\begin{bmatrix}\boldsymbol{A} & \boldsymbol{B}_{\mathcal{S}}\end{bmatrix}\right\} = n$.

Proof. The equivalence of Statements (a) and (b) is proved via Theorem 2.3. Therefore, proving the equivalence between Statements (b) and (c) is enough. We first show that Statement (b) implies Statement (c). Suppose that Statement (b) holds. We derive that

$$\begin{aligned}
\mathbb{R}^n &= \operatorname{ColSpace}\left\{\begin{bmatrix}\boldsymbol{A}^{K-1}\boldsymbol{B}_{\mathcal{S}_1} & \boldsymbol{A}^{K-2}\boldsymbol{B}_{\mathcal{S}_2} & \ldots & \boldsymbol{B}_{\mathcal{S}_K}\end{bmatrix}\right\} && (2.19)\\
&\subseteq \operatorname{ColSpace}\left\{\begin{bmatrix}\boldsymbol{A}^{K-1}\boldsymbol{B} & \boldsymbol{A}^{K-2}\boldsymbol{B} & \ldots & \boldsymbol{B}\end{bmatrix}\right\} && (2.20)\\
&\subseteq \mathbb{R}^n. && (2.21)
\end{aligned}$$

Hence, we have

$$\operatorname{rank}\left\{\begin{bmatrix}\boldsymbol{A}^{K-1}\boldsymbol{B} & \boldsymbol{A}^{K-2}\boldsymbol{B} & \ldots & \boldsymbol{B}\end{bmatrix}\right\} = n. \tag{2.22}$$

Now, using the equivalence of the Kalman and PHB rank conditions by Theorem 2.1, we conclude that for all $\lambda \in \mathbb{C}$,

$$\operatorname{rank}\left\{\begin{bmatrix}\boldsymbol{A} - \lambda\boldsymbol{I} & \boldsymbol{B}\end{bmatrix}\right\} = n, \tag{2.23}$$

which establishes the first condition of Statement (c). Further, we note that if Statement (b) holds,

$$\mathbb{R}^n = \text{ColSpace}\left\{\left[\boldsymbol{A}^{K-1}\boldsymbol{B}_{\mathcal{S}_1} \quad \boldsymbol{A}^{K-2}\boldsymbol{B}_{\mathcal{S}_2} \quad \ldots \quad \boldsymbol{B}_{\mathcal{S}_K}\right]\right\} \tag{2.24}$$

$$\subseteq \text{ColSpace}\left\{\left[\boldsymbol{A} \quad \boldsymbol{B}_{\mathcal{S}_K}\right]\right\} \subseteq \mathbb{R}^n, \tag{2.25}$$

which follows from the fact that the first $\sum_{k=1}^{K-1} |\mathcal{S}_k|$ columns of the matrix belong to the column space of $\boldsymbol{A}$. Hence, Statement (b) implies Statement (c).

Next, to prove the equivalence between Statements (b) and (c), we show that if Statement (b) does not hold, Statement (c) is also violated. Suppose that for any $K < \infty$,

$$\text{rank}\left\{\left[\boldsymbol{A}^{K-1}\boldsymbol{B}_{\mathcal{S}_1} \quad \boldsymbol{A}^{K-2}\boldsymbol{B}_{\mathcal{S}_2} \quad \ldots \quad \boldsymbol{B}_{\mathcal{S}_K}\right]\right\} < n, \tag{2.26}$$

for any sets $\{\mathcal{S}_k \subseteq \{1, 2, \ldots, m\},\ |\mathcal{S}_k| \leq s\}_{k=1}^{K}$. Let $\mathcal{S} \subseteq \{1, 2, \ldots, m\}$ such that $1 \leq |\mathcal{S}| \leq s$. We can construct sets $\mathcal{T}_k \subseteq \{1, 2, \ldots, m\}$ for $k = 1, 2, \ldots, m-1$ such that $|\mathcal{T}_k| \leq s$ and

$$\bigcup_{k=1}^{m-1} \mathcal{T}_k \bigcup \mathcal{S} = \{1, 2, \ldots, m\}. \tag{2.27}$$

Then, using the assumption in (2.26), with $K = nm$, we get

$$\begin{aligned}\text{rank}\{[\boldsymbol{A}^{nm-1}\boldsymbol{B}_{\mathcal{T}_1} \quad \boldsymbol{A}^{nm-2}\boldsymbol{B}_{\mathcal{T}_1} \quad \ldots \quad \boldsymbol{A}^{n(m-1)}\boldsymbol{B}_{\mathcal{T}_1} \\ \boldsymbol{A}^{n(m-1)-1}\boldsymbol{B}_{\mathcal{T}_2} \quad \boldsymbol{A}^{n(m-1)-2}\boldsymbol{B}_{\mathcal{T}_2} \quad \ldots \quad \boldsymbol{A}^{n(m-2)}\boldsymbol{B}_{\mathcal{T}_2} \\ \ldots \quad \boldsymbol{A}^{2n-1}\boldsymbol{B}_{\mathcal{T}_{m-1}} \quad \boldsymbol{A}^{2n-2}\boldsymbol{B}_{\mathcal{T}_{m-1}} \quad \ldots \quad \boldsymbol{A}^{n}\boldsymbol{B}_{\mathcal{T}_{m-1}} \\ \ldots \quad \boldsymbol{A}^{n-1}\boldsymbol{B}_{\mathcal{S}} \quad \boldsymbol{A}^{n-2}\boldsymbol{B}_{\mathcal{S}} \quad \ldots \quad \boldsymbol{B}_{\mathcal{S}}]\} < n.\end{aligned} \tag{2.28}$$

We rearrange the columns of the above matrix to get

$$\begin{aligned}\text{rank}\{[\boldsymbol{A}^{nm-1}\boldsymbol{B}_{\mathcal{T}_1} \quad \boldsymbol{A}^{n(m-1)-1}\boldsymbol{B}_{\mathcal{T}_2} \quad \ldots \quad \boldsymbol{A}^{2n-1}\boldsymbol{B}_{\mathcal{T}_{m-1}} \quad \boldsymbol{A}^{n-1}\boldsymbol{B}_{\mathcal{S}} \\ \boldsymbol{A}^{nm-2}\boldsymbol{B}_{\mathcal{T}_1} \quad \boldsymbol{A}^{n(m-1)-2}\boldsymbol{B}_{\mathcal{T}_2} \quad \ldots \quad \boldsymbol{A}^{2n-2}\boldsymbol{B}_{\mathcal{T}_{m-1}} \quad \boldsymbol{A}^{n-2}\boldsymbol{B}_{\mathcal{S}} \\ \ldots \boldsymbol{A}^{n(m-1)}\boldsymbol{B}_{\mathcal{T}_1} \quad \boldsymbol{A}^{n(m-2)}\boldsymbol{B}_{\mathcal{T}_2} \ldots \quad \boldsymbol{A}^{n}\boldsymbol{B}_{\mathcal{T}_{m-1}} \quad \boldsymbol{B}_{\mathcal{S}}]\} < n.\end{aligned} \tag{2.29}$$

Now we define the following matrix,

$$\hat{\boldsymbol{B}} = \left[\boldsymbol{A}^{n(m-1)}\boldsymbol{B}_{\mathcal{T}_1} \quad \boldsymbol{A}^{n(m-2)}\boldsymbol{B}_{\mathcal{T}_2} \ldots \quad \boldsymbol{A}^{n}\boldsymbol{B}_{\mathcal{T}_{m-1}} \quad \boldsymbol{B}_{\mathcal{S}}\right], \tag{2.30}$$

to rewrite (2.29) as

$$\text{rank}\left\{\begin{bmatrix} \boldsymbol{A}^{n-1}\hat{\boldsymbol{B}} & \boldsymbol{A}^{n-2}\hat{\boldsymbol{B}} & \dots & \hat{\boldsymbol{B}} \end{bmatrix}\right\} < n. \tag{2.31}$$

Next, from the Kalman rank condition in Theorem 2.1, we conclude that the system $(\boldsymbol{A}, \hat{\boldsymbol{B}})$ is not controllable and it violates the PBH rank condition. So there exists $\lambda \in \mathbb{C}$ such that

$$\text{rank}\left\{\begin{bmatrix} \boldsymbol{A} - \lambda \boldsymbol{I} & \hat{\boldsymbol{B}} \end{bmatrix}\right\} < n. \tag{2.32}$$

Consequently, there exists $\boldsymbol{z} \in \mathbb{C}^n$ such that $\boldsymbol{z} \neq \boldsymbol{0}$ and

$$\boldsymbol{z}^\mathsf{T}\begin{bmatrix} \boldsymbol{A} - \lambda \boldsymbol{I} & \hat{\boldsymbol{B}} \end{bmatrix} = \boldsymbol{0}. \tag{2.33}$$

This relation leads us to

$$\boldsymbol{z}^\mathsf{T}\boldsymbol{A} = \lambda \boldsymbol{z}^\mathsf{T} \tag{2.34}$$

$$\boldsymbol{z}^\mathsf{T}\hat{\boldsymbol{B}} = \boldsymbol{0}. \tag{2.35}$$

Substituting for $\hat{\boldsymbol{B}}$ from (2.29) and using (2.34), we arrive at

$$\boldsymbol{0} = \boldsymbol{z}^\mathsf{T}\hat{\boldsymbol{B}} = \boldsymbol{z}^\mathsf{T}\begin{bmatrix} \boldsymbol{A}^{n(m-1)}\boldsymbol{B}_{\mathcal{T}_1} & \boldsymbol{A}^{n(m-2)}\boldsymbol{B}_{\mathcal{T}_2} \dots & \boldsymbol{A}^{n}\boldsymbol{B}_{\mathcal{T}_{m-1}} & \boldsymbol{B}_{\mathcal{S}} \end{bmatrix} \tag{2.36}$$

$$= \boldsymbol{z}^\mathsf{T}\begin{bmatrix} \lambda^{n(m-1)}\boldsymbol{B}_{\mathcal{T}_1} & \lambda^{n(m-2)}\boldsymbol{B}_{\mathcal{T}_2} \dots & \lambda^{n}\boldsymbol{B}_{\mathcal{T}_{m-1}} & \boldsymbol{B}_{\mathcal{S}} \end{bmatrix}. \tag{2.37}$$

We now consider two cases: $\lambda = 0$ and $\lambda \neq 0$ separately as follows.

Case 1: If $\lambda = 0$, from (2.37), we derive

$$\boldsymbol{0} = \boldsymbol{z}^\mathsf{T}\boldsymbol{B}_{\mathcal{S}}. \tag{2.38}$$

This condition combined with (2.34) implies

$$\boldsymbol{0} = \boldsymbol{z}^\mathsf{T}\begin{bmatrix} \boldsymbol{A} - \lambda \boldsymbol{I} & \boldsymbol{B}_{\mathcal{S}} \end{bmatrix} = \boldsymbol{z}^\mathsf{T}\begin{bmatrix} \boldsymbol{A} & \boldsymbol{B}_{\mathcal{S}} \end{bmatrix}. \tag{2.39}$$

Hence, for any set $\mathcal{S} \subseteq \{1, 2, \dots, m\}$, we derive $\text{rank}\left\{\begin{bmatrix} \boldsymbol{A} & \boldsymbol{B}_{\mathcal{S}} \end{bmatrix}\right\} < n$, violating Statement (c).

Case 2: If $\lambda \neq 0$, from (2.37), we derive

$$\boldsymbol{0} = \boldsymbol{z}^\mathsf{T}\begin{bmatrix} \boldsymbol{B}_{\mathcal{T}_1} & \boldsymbol{B}_{\mathcal{T}_2} \dots & \boldsymbol{B}_{\mathcal{T}_{m-1}} & \boldsymbol{B}_{\mathcal{S}} \end{bmatrix} = \boldsymbol{z}^\mathsf{T}\boldsymbol{B}, \tag{2.40}$$

using (2.27). This condition combined with (2.34) implies

$$\boldsymbol{0} = \boldsymbol{z}^\mathsf{T}\begin{bmatrix} \boldsymbol{A} - \lambda \boldsymbol{I} & \boldsymbol{B} \end{bmatrix}. \tag{2.41}$$

Hence, there exists $\lambda \neq 0$ such that $\text{rank}\left\{\begin{bmatrix} \boldsymbol{A} - \lambda \boldsymbol{I} & \boldsymbol{B} \end{bmatrix}\right\} < n$, violating Statement (c).

To summarize, Statement (c) is violated if Statement (b) does not hold. So Statement (c) implies Statement (b), and the proof is complete. □

The proof technique of Theorem 2.6 is quite general and can be extended to other sparsity structures.

Corollary 2.7. The linear dynamical system ($\boldsymbol{A} \in \mathbb{R}^{n\times n}, \boldsymbol{B} \in \mathbb{R}^{n\times m}$) in (2.3) with feasible inputs satisfying the constraint $\mathrm{Supp}(\boldsymbol{u}_k) \in \mathbb{S}$ where $\mathbb{S}$ is a subset of the powerset of $\{1, 2, \ldots, m\}$ and $\bigcup_{\mathcal{S}\in\mathbb{S}} = \{1, 2, \ldots, m\}$. Then, the system is controllable under the above constraint if and only if for any nonzero $\lambda \in \mathbb{C}$, we have $\mathrm{rank}\left\{\begin{bmatrix}\boldsymbol{A} - \lambda \boldsymbol{I} & \boldsymbol{B}\end{bmatrix}\right\} = n$ and there exists a set $\mathcal{S} \in \mathbb{S}$ such that $\mathrm{rank}\left\{\begin{bmatrix}\boldsymbol{A} & \boldsymbol{B}_{\mathcal{S}}\end{bmatrix}\right\} = n$.

Proof. The proof is identical to the proof of Theorem 2.6 except that instead of choosing support sets with cardinality at most s, we choose the support sets from $\mathbb{S}$. □

We note that Theorem 2.6 assumes that $\mathbb{S} = \{\mathcal{S} \subseteq \{1, 2, \ldots, m\} : |\mathcal{S}| \leq s\}$, but Theorem 2.7 generalizes it to any arbitrary set $\mathbb{S}$ satisfying $\bigcup_{\mathcal{S}\in\mathbb{S}} = \{1, 2, \ldots, m\}$. Further, the conditions of the new controllability test, Statement (c) of Theorem 2.6 is simpler than that given by Theorem 2.3. Nonetheless, the second condition needs to be verified for support sets of size s, which might appear to be a combinatorial problem. However, the condition can be further simplified as follows.

Theorem 2.8 (Joseph and Murthy, 2021)**.** The linear dynamical system ($\boldsymbol{A} \in \mathbb{R}^{n\times n}, \boldsymbol{B} \in \mathbb{R}^{n\times m}$) in (2.3) is s-sparse controllable if and only if the system is controllable and the sparsity level $s \geq \max\{1, n - \mathrm{rank}\{\boldsymbol{A}\}\}$.

Proof. From Theorem 2.1 and Theorem 2.6, it suffices to establish that the following statements are equivalent:

(a) For any nonzero $\lambda \in \mathbb{C}$, we have $\mathrm{rank}\left\{\begin{bmatrix}\boldsymbol{A} - \lambda \boldsymbol{I} & \boldsymbol{B}\end{bmatrix}\right\} = n$ and there exists a set $\mathcal{S}$ such that $1 \leq |\mathcal{S}| \leq s$ and $\mathrm{rank}\left\{\begin{bmatrix}\boldsymbol{A} & \boldsymbol{B}_{\mathcal{S}}\end{bmatrix}\right\} = n$.

(b) For any $\lambda \in \mathbb{C}$, we have $\mathrm{rank}\left\{\begin{bmatrix}\boldsymbol{A} - \lambda \boldsymbol{I} & \boldsymbol{B}\end{bmatrix}\right\} = n$ and $s \geq \max\{1, n - \mathrm{rank}\{\boldsymbol{A}\}\}$.

We first show that Statement (a) implies Statement (b). If Statement (a) holds

$$n = \operatorname{rank}\left\{\begin{bmatrix} \boldsymbol{A} & \boldsymbol{B}_{\mathcal{S}} \end{bmatrix}\right\} \leq \operatorname{rank}\left\{\begin{bmatrix} \boldsymbol{A} & \boldsymbol{B} \end{bmatrix}\right\} \leq n. \tag{2.42}$$

Therefore, we conclude that for any $\lambda \in \mathbb{C}$, we have $\operatorname{rank}\left\{\begin{bmatrix} \boldsymbol{A} - \lambda \boldsymbol{I} & \boldsymbol{B} \end{bmatrix}\right\} = n$, which is the first condition of Statement (b). Also, we derive

$$n = \operatorname{rank}\left\{\begin{bmatrix} \boldsymbol{A} & \boldsymbol{B}_{\mathcal{S}} \end{bmatrix}\right\} \leq \operatorname{rank}\{\boldsymbol{A}\} + \operatorname{rank}\{\boldsymbol{B}_{\mathcal{S}}\} \leq \operatorname{rank}\{\boldsymbol{A}\} + s. \tag{2.43}$$

As a consequence, we deduce that $s \geq n - \operatorname{rank}\{\boldsymbol{A}\}$. Hence, we establish the second condition of Statement (b) and prove that Statement (a) implies Statement (b).

Next, we show that Statement (b) implies Statement (a). Statement (b) immediately implies the first condition of Statement (a) and $\operatorname{rank}\left\{\begin{bmatrix} \boldsymbol{A} & \boldsymbol{B} \end{bmatrix}\right\} = n$. Then, there exists $n - \operatorname{rank}\{\boldsymbol{A}\}$ columns in $\boldsymbol{B}$ that are linearly independent of the columns of $\boldsymbol{A}$. So if $s \geq n - \operatorname{rank}\{\boldsymbol{A}\}$, we can always find a support set $\mathcal{S}$ with $|\mathcal{S}| = n - \operatorname{rank}\{\boldsymbol{A}\} \leq s$ such that $\operatorname{rank}\left\{\begin{bmatrix} \boldsymbol{A} & \boldsymbol{B}_{\mathcal{S}} \end{bmatrix}\right\} = n$. Hence, Statement (b) implies Statement (a), and the proof is complete. □

The first condition of Theorem 2.8 is the same as the PBH test, which is not surprising. If a system is not controllable (without any restrictions on the input), it is not sparse controllable. So, the PBH test is a necessary condition for sparse controllability. Theorem 2.8 establishes that any controllable system is sparse controllable if the lower limit on sparsity is met. Also, the first condition is independent of the sparsity level, and the lower limit in the second condition does not depend on the input matrix $\boldsymbol{B}$. Furthermore, the PBH test also provides the rank of $\boldsymbol{A}$ and therefore, the controllability test in Theorem 2.8 has the same complexity as the classical PBH test in Theorem 2.1. So, unlike the combinatorial test in Theorem 2.3, the new controllability test has polynomial complexity in n, and the complexity is independent of the sparsity level s.

Another important insight from Theorem 2.8 about reversible linear dynamical systems i.e., linear dynamical systems whose state matrix $\boldsymbol{A}$ is invertible, is as follows:

Corollary 2.9. A controllable linear dynamical system ($\boldsymbol{A} \in \mathbb{R}^{n\times n}, \boldsymbol{B} \in \mathbb{R}^{n\times m}$) with rank $\{\boldsymbol{A}\} \geq n-1$ is s-sparse controllable for any $s \geq 1$.

So far, we have discussed the conditions under which a linear dynamical system is sparse controllable. Next, we discuss the controllability index under sparsity constraints.

2.4 Sparse Controllability Index

The controllability index K_s^* under the constraint that the inputs are s-sparse can be defined similarly to the unconstrained case.

$$K_s^* = \min_{\substack{\{\mathcal{S}_k \subseteq \{1,2,\ldots,m\},\\ |\mathcal{S}_k| \leq s\}_{k=1}^K}} K$$
$$\text{s. t. } \operatorname{rank}\left\{\begin{bmatrix}\boldsymbol{A}^{K-1}\boldsymbol{B}_{\mathcal{S}_1} & \boldsymbol{A}^{K-2}\boldsymbol{B}_{\mathcal{S}_2} & \ldots & \boldsymbol{B}_{\mathcal{S}_K}\end{bmatrix}\right\} = n. \quad (2.44)$$

Therefore, K_s^* denotes the time-to-control or the least number of control inputs that can steer the system from any initial state to any desired final state at $k = K_s^*$, using s-sparse inputs.

We next bound the sparse controllability index under sparsity constraints, similar to Theorem 2.2. We start with the lower bound.

Lemma 2.10. The s-sparse controllability index K_s^* defined in (2.44) of a controllable linear dynamical system ($\boldsymbol{A} \in \mathbb{R}^{n\times n}, \boldsymbol{B} \in \mathbb{R}^{n\times m}$) in (2.3) is bounded as

$$K^* \geq \frac{n}{\min\{\operatorname{rank}\{\boldsymbol{B}\}, s\}}. \quad (2.45)$$

Proof. To prove the lower bound, let $\{\mathcal{S}_k \subseteq \{1,2,\ldots,m\},\ |\mathcal{S}_k| \leq s\}_{k=1}^{K_s^*}$ such that

$$n = \operatorname{rank}\left\{\begin{bmatrix}\boldsymbol{A}^{K_s^*-1}\boldsymbol{B}_{\mathcal{S}_1} & \boldsymbol{A}^{K-2}\boldsymbol{B}_{\mathcal{S}_2} & \ldots & \boldsymbol{B}_{\mathcal{S}_{K_s^*}}\end{bmatrix}\right\}. \quad (2.46)$$

Then, we deduce that

$$n \leq \sum_{k=1}^{K_s^*} \operatorname{rank}\left\{\boldsymbol{A}^{K_s^*-k}\boldsymbol{B}_{\mathcal{S}_k}\right\} \leq \sum_{k=1}^{K^*} \operatorname{rank}\left\{\boldsymbol{B}_{\mathcal{S}_k}\right\} = K^* \min\{\operatorname{rank}\{\boldsymbol{B}\}, s\}. \quad (2.47)$$

Rearranging the above relation, we arrive at the lower bound on K_s^*. □

We next derive the upper bound for the sparse controllability index using the following lemma.

Lemma 2.11. For a controllable linear dynamical system ($\boldsymbol{A} \in \mathbb{R}^{n\times n}, \boldsymbol{B} \in \mathbb{R}^{n\times m}$) in (2.3), we define $\boldsymbol{\Phi}^{(K,s)}$ as

$$\begin{aligned}\boldsymbol{\Phi}^{(K,s)} \in \underset{\boldsymbol{\Phi}\in\mathbb{R}^{n\times Ks}}{\arg\max}\ & \operatorname{rank}\{\boldsymbol{\Phi}\} \\ \text{s. t. } & \boldsymbol{\Phi} = \left[\boldsymbol{A}^{K-1}\boldsymbol{B}_{\mathcal{S}_1} \quad \boldsymbol{A}^{K-2}\boldsymbol{B}_{\mathcal{S}_2} \quad \dots \quad \boldsymbol{B}_{\mathcal{S}_K}\right], \\ & \text{and } \mathcal{S}_k \subseteq \{1,2,\dots,m\},\ |\mathcal{S}_k| \le s,\ k = 1,2,\dots,K. \end{aligned} \tag{2.48}$$

Then, for any $K \le K_s^*$, the rank of the matrix satisfies

$$\operatorname{rank}\left\{\boldsymbol{\Phi}^{(K,s)}\right\} > \operatorname{rank}\left\{\boldsymbol{\Phi}^{(K-1,s)}\right\}. \tag{2.49}$$

Proof. We notice that $K_s^* = \min\{K : \operatorname{rank}\left\{\boldsymbol{\Phi}^{(K,s)}\right\} = n\}$ and

$$\operatorname{rank}\left\{\boldsymbol{\Phi}^{(K_s^*,s)}\right\} = n > \operatorname{rank}\left\{\boldsymbol{\Phi}^{(K_s^*-1,s)}\right\}, \tag{2.50}$$

establishing the desired result for $K = K_s^*$.

Next, we prove the result for $K < K_s^*$. Suppose (2.49) is not true for some $K < K_s^*$, i.e., $\operatorname{rank}\left\{\boldsymbol{\Phi}^{(K,s)}\right\} \le \operatorname{rank}\left\{\boldsymbol{\Phi}^{(K-1,s)}\right\}$, and let $\mathcal{T}_k \subseteq \{1,2,\dots,m\}$ with $|\mathcal{T}_k| \le s$, for $k = 1,2,\dots,K-1$, be such that

$$\operatorname{rank}\left\{\boldsymbol{\Phi}^{(K-1,s)}\right\} = \operatorname{rank}\left\{\left[\boldsymbol{A}^{K-2}\boldsymbol{B}_{\mathcal{T}_1} \quad \boldsymbol{A}^{K-3}\boldsymbol{B}_{\mathcal{T}_2} \quad \dots \quad \boldsymbol{B}_{\mathcal{T}_{K-1}}\right]\right\}. \tag{2.51}$$

Then, for any $\mathcal{S} \subseteq \{1,2,\dots,m\}$ with $|\mathcal{S}| \le s$, we have

$$\operatorname{rank}\left\{\boldsymbol{\Phi}^{(K,s)}\right\} \ge \operatorname{rank}\left\{\left[\boldsymbol{A}^{K-1}\boldsymbol{B}_{\mathcal{S}} \quad \boldsymbol{A}^{K-2}\boldsymbol{B}_{\mathcal{T}_1} \quad \boldsymbol{A}^{K-3}\boldsymbol{B}_{\mathcal{T}_2} \quad \dots \quad \boldsymbol{B}_{\mathcal{T}_{K-1}}\right]\right\} \tag{2.52}$$

$$\ge \operatorname{rank}\left\{\left[\boldsymbol{A}^{K-2}\boldsymbol{B}_{\mathcal{T}_1} \quad \boldsymbol{A}^{K-3}\boldsymbol{B}_{\mathcal{T}_2} \quad \dots \quad \boldsymbol{B}_{\mathcal{T}_{K-1}}\right]\right\} \tag{2.53}$$

$$= \operatorname{rank}\left\{\boldsymbol{\Phi}^{(K-1,s)}\right\}. \tag{2.54}$$

However, if (2.49) is not true, we arrive at $\operatorname{rank}\left\{\boldsymbol{\Phi}^{(K,s)}\right\} = \operatorname{rank}\left\{\boldsymbol{\Phi}^{(K-1,s)}\right\}$. Also, the above relation holds for any $\mathcal{S}$, i.e.,

$$\operatorname{ColSpace}\left\{\boldsymbol{A}^{K-1}\boldsymbol{B}_{\mathcal{S}}\right\} \subset \operatorname{ColSpace}\left\{\left[\boldsymbol{A}^{K-2}\boldsymbol{B}_{\mathcal{T}_1} \quad \boldsymbol{A}^{K-3}\boldsymbol{B}_{\mathcal{T}_2} \quad \dots \quad \boldsymbol{B}_{\mathcal{T}_{K-1}}\right]\right\}. \tag{2.55}$$

Therefore, we derive

$$\text{ColSpace}\left\{\boldsymbol{A}^{K-1}\boldsymbol{B}\right\} \subset \text{ColSpace}\left\{\left[\boldsymbol{A}^{K-2}\boldsymbol{B}_{\mathcal{T}_1} \quad \boldsymbol{A}^{K-3}\boldsymbol{B}_{\mathcal{T}_2} \quad \ldots \quad \boldsymbol{B}_{\mathcal{T}_{K-1}}\right]\right\}. \tag{2.56}$$

Consequently, we deduce that

$$\begin{aligned}&\text{ColSpace}\left\{\boldsymbol{A}^{K_s^*-1}\boldsymbol{B}\right\} \\ &\quad \subset \text{ColSpace}\left\{\left[\boldsymbol{A}^{K_s^*-2}\boldsymbol{B}_{\mathcal{T}_1} \quad \boldsymbol{A}^{K_s^*-3}\boldsymbol{B}_{\mathcal{T}_2} \quad \ldots \quad \boldsymbol{A}^{K_s^*-K}\boldsymbol{B}_{\mathcal{T}_{K-1}}\right]\right\}.\end{aligned} \tag{2.57}$$

In contrast, (2.50) implies that for any $\mathcal{S}_k \subseteq \{1, 2, \ldots, m\}$ with $|\mathcal{S}_k| \leq s$, for $k = 1, 2, \ldots, K_s^* - 1$, we get

$$\text{ColSpace}\left\{\boldsymbol{A}^{K_s^*-1}\boldsymbol{B}\right\} \subsetneq \text{ColSpace}\left\{\left[\boldsymbol{A}^{K_s^*-2}\boldsymbol{B}_{\mathcal{S}_1} \quad \boldsymbol{A}^{K_s^*-3}\boldsymbol{B}_{\mathcal{S}_2} \quad \ldots \quad \boldsymbol{B}_{\mathcal{S}_{K_s^*-1}}\right]\right\}. \tag{2.58}$$

Hence, the negation of our relation (2.49) leads to falsehood, and the proof is complete. □

Finally, we bound the sparse controllability index from above and below using Theorems 2.10 and 2.11.

Theorem 2.12 (Joseph and Murthy, 2021). The s-sparse controllability index K_s^* defined in (2.44) of a controllable linear dynamical system $(\boldsymbol{A} \in \mathbb{R}^{n\times n}, \boldsymbol{B} \in \mathbb{R}^{n\times m})$ in (2.3) is bounded as

$$\frac{n}{\min\{\text{rank}\{\boldsymbol{B}\}, s\}} \leq K^* \leq n - \min\{\text{rank}\{\boldsymbol{B}\}, s\} + 1. \tag{2.59}$$

Proof. The lower bound directly follows from Theorem 2.10. Next, to prove the upper bound, using the definition in Theorem 2.11, we deduce that

$$n = \text{rank}\left\{\boldsymbol{\Phi}^{(K_s^*,s)}\right\} \geq \text{rank}\left\{\boldsymbol{\Phi}^{(K_s^*-1,s)}\right\} + 1 \geq \ldots \geq \text{rank}\left\{\boldsymbol{\Phi}^{(1,s)}\right\} + K_s^* - 1 \tag{2.60}$$

$$= \max_{\substack{\mathcal{S}\subseteq\{1,2,\ldots,m\}\\ |\mathcal{S}|\leq s}} \text{rank}\{\boldsymbol{B}_{\mathcal{S}}\} + K_s^* - 1 = \min\{\text{rank}\{\boldsymbol{B}\}, s\} + K_s^* - 1. \tag{2.61}$$

Rearranging the above relation, we arrive at the upper bound. □

As s increases from 1 to rank $\{\boldsymbol{B}\}$, the upper and lower bounds decrease, indicating that the system has more flexibility and thus requires fewer input vectors to ensure controllability. When $s = \text{rank}\{\boldsymbol{B}\}$, the above result reduces to the bounds for (unconstrained) controllability index given by Theorem 2.2. The upper and lower bounds meet when $\min\{\text{rank}\{\boldsymbol{B}\}, s\} = 1$.

2.5 Summary

- A linear dynamical system is s-sparse controllable if it is controllable using s-sparse inputs.
- If a linear dynamical system is s-sparse controllable, there exists an actuator schedule independent of the initial and final states, which can guarantee the controllability of the system.
- A linear dynamical system is s-sparse controllable if and only if it is controllable and the sparsity level exceeds the nullity of the state matrix.
- An s-sparse controllable linear dynamical system ($\boldsymbol{A} \in \mathbb{R}^{n\times n}, \boldsymbol{B} \in \mathbb{R}^{n\times m}$) needs at least $\frac{n}{\min\{\text{rank}\{\boldsymbol{B}\}, s\}}$ and at most $n - \min\{\text{rank}\{\boldsymbol{B}\}, s\} + 1$ control inputs to ensure controllability.

2.6 Notes

The classical controllability theory and its results, along with their proofs and detailed discussions, are extensively covered in Chen (1984). Meanwhile, the sparse controllability-related findings discussed in this section are sourced from Joseph and Murthy (2021). It's worth noting that the proofs delineated in Joseph and Murthy (2021) differ from those provided in this work.

Additionally, Joseph and Murthy (2021) discusses other notable results. For instance, in classical control theory, the Kalman decomposition is a pivotal mathematical technique employed to convert a given discrete-time linear dynamical system into a canonical form, thereby distinguishing its controllable components. This transformation facilitates the analysis and design of system control. Expanding upon this technique within the realm of sparse controllability, Joseph and Murthy (2021) introduces a method for decomposing the state space. This process, relying on Theorem 2.4, recognizes that s-sparse controllability, akin to conventional controllability, remains unchanged under a change of basis. Consequently, the state space can be decomposed via a suitable linear transformation, dividing it into sparse-controllable, sparse-uncontrollable, and uncontrollable subspaces. For further insights, refer to Joseph and Murthy (2021, Section V).

Furthermore, in continuation of the sparse controllability findings, Joseph *et al.* (2021) investigates the application of sparse actuator control in social networks. This work delves into the potential for an external manipulative agent to influence the opinions of individuals within a social network. The agent aims to steer them towards a desired state while imposing limited additive influence on their inherent opinions. Focused on a social network modeled as an Erdős-Rényi random graph, where each individual adjusts their opinion based on a weighted average of their neighbors' opinions, the study uncovers intriguing results. It reveals that in the absence of constraints on the agent's influence, they can guide the network opinion to any desired value within a finite time, indicating that the system is almost surely controllable. However, when the control input is constrained by sparsity, the network's opinion becomes controllable with a certain probability. A lower bound for this probability elucidates how the controllability of network opinion hinges on factors such as network size, connectivity, and the sparsity constraints faced by the manipulative agent.

3

Design of Sparse Inputs with Time-Varying Support

This section introduces the problem of designing or estimating sparse inputs (with time-varying support) of a linear dynamical system and covers the sparsity-aware algorithms to solve the control design problem.

3.1 Sparse Input Design Problems

We recall the state space model of the linear dynamical system $(\boldsymbol{A}, \boldsymbol{B})$ in (2.3), with state $\boldsymbol{x}_k \in \mathbb{R}^n$ and input $\boldsymbol{u}_k \in \mathbb{R}^m$,

$$\boldsymbol{x}_{k+1} = \boldsymbol{A}\boldsymbol{x}_k + \boldsymbol{B}\boldsymbol{u}_k, \tag{3.1}$$

for discrete-time indices $k = 1, 2, \ldots$. Here, we impose the constraint that the inputs are sparse $\|\boldsymbol{u}_k\|_0 \leq s \ll m$, for $k = 1, 2 \ldots$. The goal here is to estimate the sparse inputs given the initial state $\boldsymbol{x}_1$ and the desired final state $\boldsymbol{x}_{\mathrm{f}}$. Specifically, the inputs $\{\boldsymbol{u}_k,\ k = 1, 2, \ldots, K\}$ should steer the system from the state $\boldsymbol{x}_1$ to $\boldsymbol{x}_{K+1} = \boldsymbol{x}_{\mathrm{f}}$, for some finite K. This problem is feasible if the linear dynamical system is controllable and the sparsity level s exceeds the nullity of $\boldsymbol{A}$, as given by Theorem 2.8. So, throughout this section, assume that the above two conditions hold.

From Theorem 2.12, we note that a system needs at most n sparse inputs to drive its state from any initial state to any final state. So, for a given initial state $\boldsymbol{x}_1 \in \mathbb{R}^n$ and final state $\boldsymbol{x}_{\mathrm{f}} \in \mathbb{R}^n$ the state-dependent actuator schedule-based design solves for n sparse inputs $\{\boldsymbol{u}_k\}_{k=1}^n$ such that

$$\boldsymbol{x}_{\mathrm{f}} - \boldsymbol{A}^n \boldsymbol{x}_1 = \sum_{k=1}^{n} \boldsymbol{A}^{n-k} \boldsymbol{B} \boldsymbol{u}_k. \tag{3.2}$$

which follows from (3.1).

Let the vector of sparse unknowns be

$$\boldsymbol{u} = \begin{bmatrix} \boldsymbol{u}_1 \\ \boldsymbol{u}_2 \\ \vdots \\ \boldsymbol{u}_n \end{bmatrix}. \tag{3.3}$$

Then, we rewrite (3.2) as

$$\bar{\boldsymbol{x}} = \boldsymbol{x}_{\mathrm{f}} - \boldsymbol{A}^n \boldsymbol{x}_1 = \sum_{k=1}^{n} \boldsymbol{A}^{n-k} \boldsymbol{B} \boldsymbol{u}_k = \boldsymbol{\Phi} \boldsymbol{u}, \tag{3.4}$$

where we define the matrix $\boldsymbol{\Phi}$ as

$$\boldsymbol{\Phi} = \begin{bmatrix} \boldsymbol{A}^{n-1}\boldsymbol{B} & \boldsymbol{A}^{n-2}\boldsymbol{B} & \dots & \boldsymbol{B} \end{bmatrix}. \tag{3.5}$$

Hence, solving for sparse inputs is the same as solving the system of equations

$$\bar{\boldsymbol{x}} = \boldsymbol{\Phi} \boldsymbol{u}, \tag{3.6}$$

for a sparse solution $\boldsymbol{u}$.

We divide the sparse input design algorithms for the time-varying support case into two categories: state-dependent actuator scheduling and state-independent actuator scheduling, as presented next.

3.2 State-dependent Actuator Scheduling

The goal of the state-dependent actuator scheduling-based approach is to design n sparse inputs $\{\boldsymbol{u}_k\}_{k=1}^n$ by directly solving (3.6). We note that this approach does not assume that the system is sparse controllable, but it assumes (3.6) has a solution. In other words, this approach assumes that $\bar{\boldsymbol{x}}$ belongs to ColSpace$\{\boldsymbol{\Phi}\}$, which need not be equal to $\mathbb{R}^n$.

There are two approaches to finding a sparse solution to (3.6). In the first approach, we minimize the average sparsity of the inputs given by

$$\frac{1}{n}\sum_{k=1}^{n}\|\boldsymbol{u}_k\|_0 = \frac{1}{n}\|\boldsymbol{u}\|_0 . \tag{3.7}$$

Therefore, by dropping the constant factor of $1/n$ from the objective function, the sparse input design problem is

$$\underset{\boldsymbol{u}\in\mathbb{R}^{nm}}{\arg\min} \quad \|\boldsymbol{u}\|_0 \quad \text{s. t.} \quad \bar{\boldsymbol{x}} = \boldsymbol{\Phi}\boldsymbol{u}. \tag{3.8}$$

The above problem is the same as the classical sparse signal recovery or compressed sensing problem. Hence, we can use the sparsity-aware algorithms from the compressed sensing literature to solve the problem. This approach is referred to as *sparse input design via sparse recovery algorithms*. We briefly describe some of the sparse recovery algorithms to solve (3.8) in Section 3.3.

The problem with the above approach is that it is possible to have different sparsity levels for inputs at different times. For example, a solution can contain an input $\boldsymbol{u}_k$ with all nonzero entries because there is no control over the sparsity of the individual inputs. To tackle this problem, in the second approach, the sparsity level on each input is limited by a constant $s < m$ as follows:

$$\underset{\boldsymbol{u}\in\mathbb{R}^{nm}}{\arg\min} \quad \|\boldsymbol{u}\|_0 \ \text{s. t.}\ \bar{\boldsymbol{x}} = \boldsymbol{\Phi}\boldsymbol{u},\ \boldsymbol{u} = \begin{bmatrix}\boldsymbol{u}_1\\ \boldsymbol{u}_2\\ \vdots\\ \boldsymbol{u}_n\end{bmatrix},\ \text{and}\ \|\boldsymbol{u}_k\|_0 \le s,\ \text{for}\ k = 1, 2, \ldots, n. \tag{3.9}$$

Due to the additional constraint, the above optimization problem seeks $\boldsymbol{u}$ obtained by concatenating n number of s-sparse vector, and the feasibility set of (3.9) is a subset of the feasibility set of (3.6). This special type of sparse vector with an additional structure is called the piecewise sparse vector, and this approach in (3.9) is referred to as *sparse input design via piecewise sparse recovery algorithms*.

3.3 Design via Sparse Recovery Algorithms

This section presents a selection of popular sparse recovery algorithms used in compressive sensing to solve (3.6). We present two types of algorithms: convex optimization methods and greedy methods.

3.3.1 Convex Optimization Methods

We note that the optimization problem in (3.6),

$$\underset{\boldsymbol{u}\in\mathbb{R}^{nm}}{\arg\min}\ \ \|\boldsymbol{u}\|_0 \quad \text{s. t.} \quad \bar{\boldsymbol{x}} = \boldsymbol{\Phi}\boldsymbol{u}, \tag{3.10}$$

is nonconvex due to the ℓ_0-norm in the objective function. Nonetheless, we note that the ℓ_q norm $\|\boldsymbol{u}\|_q = \left(\sum_i \boldsymbol{u}_i^q\right)^{1/q}$ approaches $\|\boldsymbol{u}\|_0$ when $q > 0$ is close to 0. Consequently, we can approximate (3.10) as

$$\underset{\boldsymbol{u}\in\mathbb{R}^{nm}}{\arg\min}\ \ \|\boldsymbol{u}\|_q \quad \text{s. t.} \quad \bar{\boldsymbol{x}} = \boldsymbol{\Phi}\boldsymbol{u}, \tag{3.11}$$

for some $q > 0$. For $q < 1$, the problem in (3.11) is nonconvex and hard to solve. However, when $q \geq 1$, the problem is convex and can be solved using any standard convex optimization solver. So we choose the smallest value $q = 1$ for which the objective function is convex, and the resulting optimization problem is an approximation to solve (3.10),

$$\arg \min_{\boldsymbol{u}\in\mathbb{R}^{nm}}\ \ \|\boldsymbol{u}\|_1 \quad \text{s. t.} \quad \bar{\boldsymbol{x}} = \boldsymbol{\Phi}\boldsymbol{u}, \tag{3.12}$$

The above principle is usually called ℓ_1-minimization or basis pursuit, and the algorithm is summarized in Algorithm 1 (Foucart *et al.*, 2013).

Algorithm 1 Sparse input design via basis pursuit

Input: Initial state $\boldsymbol{x}_1 \in \mathbb{R}^n$, final state $\boldsymbol{x}_\mathrm{f} \in \mathbb{R}^n$, linear dynamical system $(\boldsymbol{A} \in \mathbb{R}^{n\times n}, \boldsymbol{B} \in \mathbb{R}^{n\times m})$

Set $\bar{\boldsymbol{x}} = \boldsymbol{x}_\mathrm{f} - \boldsymbol{A}^n\boldsymbol{x}_1$

Compute $\boldsymbol{\Phi} = \begin{bmatrix}\boldsymbol{A}^{n-1}\boldsymbol{B} & \boldsymbol{A}^{n-2}\boldsymbol{B} & \ldots & \boldsymbol{B}\end{bmatrix}$

Determine $\boldsymbol{u}$ using (3.12)

Output: Control inputs $\boldsymbol{u}$

It is worth noting that the equivalence of ℓ_1 and ℓ_0 norm depends on the properties of $\boldsymbol{\Phi}$. We have added a small discussion on these properties in the "Notes" section.

3.3.2 Greedy Algorithms

We next present two greedy algorithms to solve (3.10), namely orthogonal matching pursuit and compressive sampling matching pursuit (Foucart *et al.*, 2013). Both algorithms are iterative in nature and update the support set $\mathcal{U}$ of $\boldsymbol{u}$ in every iteration using greedy heuristics.

The orthogonal matching pursuit algorithm starts with an empty support set $\mathcal{U}$ and $\boldsymbol{u} = \mathbf{0}$. In every iteration, it computes the residue, $\bar{\boldsymbol{x}} - \boldsymbol{\Phi}\boldsymbol{u}$, and matches it with the columns of $\boldsymbol{\Phi}$ that are not indexed by $\mathcal{U}$. The index i^* of the column with the highest inner product (in absolute value) is added to the target support $\mathcal{U}$,

$$i^* = \arg\max_{i \notin \mathcal{U}} \frac{\left|\boldsymbol{\Phi}_i^\mathsf{T}(\bar{\boldsymbol{x}} - \boldsymbol{\Phi}\boldsymbol{u})\right|}{\|\boldsymbol{\Phi}_i\|}, \tag{3.13}$$

where $\boldsymbol{\Phi}_i$ is the ith column of $\boldsymbol{\Phi}$. Using the updated support, the corresponding estimate of $\boldsymbol{u}$ is computed. These steps are continued until (3.6) is satisfied. The resulting algorithm is summarized in Algorithm 2, where $(\cdot)^\dagger$ denotes the pseudoinverse of a matrix.

Algorithm 2 Sparse input design via orthogonal matching pursuit

Input: Initial state $\boldsymbol{x}_1 \in \mathbb{R}^n$, final state $\boldsymbol{x}_\mathrm{f} \in \mathbb{R}^n$, linear dynamical system $(\boldsymbol{A} \in \mathbb{R}^{n\times n}, \boldsymbol{B} \in \mathbb{R}^{n\times m})$

Initialization: $\mathcal{U} = \emptyset$, $\boldsymbol{u} = \mathbf{0}$

1: Set $\bar{\boldsymbol{x}} = \boldsymbol{x}_\mathrm{f} - \boldsymbol{A}^n\boldsymbol{x}_1$
2: Compute $\boldsymbol{\Phi} = \begin{bmatrix}\boldsymbol{A}^{n-1}\boldsymbol{B} & \boldsymbol{A}^{n-2}\boldsymbol{B} & \dots & \boldsymbol{B}\end{bmatrix}$
3: **repeat**
4: Compute $i^* = \arg\max_{i\notin\mathcal{U}} \frac{\left|\boldsymbol{\Phi}_i^\mathsf{T}(\bar{\boldsymbol{x}}-\boldsymbol{\Phi}\boldsymbol{u})\right|}{\|\boldsymbol{\Phi}_i\|}$
5: Update support $\mathcal{U} \leftarrow \mathcal{U} \cup \{i^*\}$
6: Determine $\boldsymbol{u}_\mathcal{U} \leftarrow \boldsymbol{\Phi}_\mathcal{U}^\dagger\bar{\boldsymbol{x}}$ and $\boldsymbol{u}_{\mathcal{U}^\complement} \leftarrow \mathbf{0}$
7: **until** $\bar{\boldsymbol{x}} = \boldsymbol{\Phi}\boldsymbol{u}$

Output: Control inputs $\boldsymbol{u}$

The orthogonal matching pursuit algorithm generally works well. However, a drawback of the algorithm is that once an incorrect index is added to the target support $\mathcal{U}$, it can not be removed. The second greedy algorithm, called the compressive sampling matching pursuit algorithm, addresses this issue. Every iteration of compressive sampling matching pursuit first adds the indices of top $2ns$ columns of $\boldsymbol{\Phi}$ that match with the residue. Then, the corresponding estimate of $\boldsymbol{u}$ is computed using the extended target support. Finally, it keeps the estimate's ns largest absolute entries and sets the remaining entries to zero. In effect, every iteration of the algorithm returns an ns-sparse vector. The iterations are stopped when (3.6) holds true. The resulting algorithm is summarized in Algorithm 3.

Algorithm 3 Sparse input design via compressive sampling matching pursuit

Input: Initial state $\boldsymbol{x}_1 \in \mathbb{R}^n$, final state $\boldsymbol{x}_\mathrm{f} \in \mathbb{R}^n$, linear dynamical system $(\boldsymbol{A} \in \mathbb{R}^{n\times n}, \boldsymbol{B} \in \mathbb{R}^{n\times m})$, sparsity level s

Initialization: $\mathcal{U} = \emptyset$, $\boldsymbol{u} = \boldsymbol{0}$

1: Set $\bar{\boldsymbol{x}} = \boldsymbol{x}_\mathrm{f} - \boldsymbol{A}^n\boldsymbol{x}_1$

2: Compute $\boldsymbol{\Phi} = \begin{bmatrix}\boldsymbol{A}^{n-1}\boldsymbol{B} & \boldsymbol{A}^{n-2}\boldsymbol{B} & \dots & \boldsymbol{B}\end{bmatrix}$

3: **repeat**

4: Compute $\mathcal{I}^* = \underset{\substack{\mathcal{I}\subseteq\{1,2,\dots,nm\}\\ |\mathcal{I}|\leq 2ns}}{\arg\max} \sum_{i\in\mathcal{I}} \frac{\left|\boldsymbol{\Phi}_i^\top(\bar{\boldsymbol{x}}-\boldsymbol{\Phi}\boldsymbol{u})\right|}{\|\boldsymbol{\Phi}_i\|}$

5: Update support $\mathcal{U} \leftarrow \mathcal{U} \cup \mathcal{I}^*$

6: Determine $\boldsymbol{u}_\mathcal{U} \leftarrow \boldsymbol{\Phi}_\mathcal{U}^\dagger\bar{\boldsymbol{x}}$ and $\boldsymbol{u}_{\mathcal{U}^\complement} \leftarrow \boldsymbol{0}$

7: Determine new support $\mathcal{U} \leftarrow \underset{\substack{\mathcal{I}\subseteq\{1,2,\dots,nm\}\\ |\mathcal{I}|\leq ns}}{\arg\max} \sum_{i\in\mathcal{I}} |\boldsymbol{u}_i|$

8: Update the estimate $\boldsymbol{u}_{\mathcal{U}^\complement} \leftarrow \boldsymbol{0}$

9: **until** $\bar{\boldsymbol{x}} = \boldsymbol{\Phi}\boldsymbol{u}$

Output: Control inputs $\boldsymbol{u}$

We note that the optimization problems in the orthogonal matching pursuit (Step 4) and compressive sampling matching pursuit (Steps 4 and 7) algorithms need not be unique. In that case, we follow a predefined rule, for example, the lexicographic order.

There are numerous other sparse recovery algorithms in the compressed sensing literature, like iterative hard thresholding (Foucart *et al.*, 2013), sparse Bayesian learning (Wipf and Rao, 2004), approximate message passing (Donoho *et al.*, 2009), deep learning (Mousavi *et al.*, 2015) and generative models (Bora *et al.*, 2017). We omit the details here.

3.4 Design via Piecewise Sparse Recovery Algorithms

We recall the piecewise sparse recovery problem in (3.9), and present the variant of orthogonal matching pursuit and compressive sampling matching pursuit algorithms that solve (3.9).

3.4.1 Piecewise Orthogonal Matching Pursuit

The piecewise orthogonal matching pursuit algorithm is similar to the classical orthogonal matching pursuit (Li *et al.*, 2016). The only difference between the two algorithms is that the piecewise orthogonal matching pursuit algorithm ensures that $\|\boldsymbol{u}_k\|_0 \leq s$ holds for all values of k by not adding elements to $\mathcal{U}$ corresponding to $\boldsymbol{u}_k$ if $\|\boldsymbol{u}_k\|_0 = s$. To this end, piecewise orthogonal matching pursuit tracks the time index k for which $\mathcal{S}_k = s$ and excludes them from the subsequent iterations. This set of "abandoned" values of k is given by the set $\{k \in \{1, 2, \ldots, n\} : |\mathcal{S}_k| = s\}$. When this set is empty, i.e., $|\mathcal{S}_k| < s$ for $k = 1, 2, \ldots, n$, piecewise orthogonal matching pursuit behaves like the regular orthogonal matching pursuit algorithm in Algorithm 2. The algorithm is summarized in Algorithm 4. We note that piecewise orthogonal matching pursuit is not the same as applying orthogonal matching pursuit n times because the vector $\bar{\boldsymbol{x}}$ is common to all the inputs $\boldsymbol{u}_k$'s.

3.4.2 Piecewise Compressive Sampling Matching Pursuit

Similar to piecewise orthogonal matching pursuit, we can modify the compressive sampling matching pursuit to get the piecewise compressive sampling matching pursuit algorithm (Zhong and Li, 2018). Instead of choosing the top $2ns$ indices in Step 4, the piecewise counterpart chooses the top $2s$ corresponding to each of the n time index,

$$\mathcal{I}_k^* = \underset{\substack{\mathcal{I} \subseteq \{1,2,\ldots,m\} \\ |\mathcal{I}| \leq 2s}}{\arg\max} \sum_{i \in \mathcal{I}} \frac{\left| \left[\boldsymbol{\Phi}^\top (\bar{\boldsymbol{x}} - \boldsymbol{\Phi}\boldsymbol{u}) \right]_{(k-1)m+i} \right|}{\left\| \boldsymbol{\Phi}_{(k-1)m+i} \right\|}. \tag{3.14}$$

Algorithm 4 Sparse input design via piecewise orthogonal matching pursuit

---

Input: Initial state $\boldsymbol{x}_1 \in \mathbb{R}^n$, final state $\boldsymbol{x}_\mathrm{f} \in \mathbb{R}^n$, linear dynamical system ($\boldsymbol{A} \in \mathbb{R}^{n\times n}, \boldsymbol{B} \in \mathbb{R}^{n\times m}$), sparsity level $s < m$

Initialization: $\mathcal{U} = \mathcal{S}_k = \emptyset$, for $k = 1, 2, \ldots, n$, $\boldsymbol{u} = \boldsymbol{0}$

1: Set $\bar{\boldsymbol{x}} = \boldsymbol{x}_\mathrm{f} - \boldsymbol{A}^n\boldsymbol{x}_1$ and $\boldsymbol{\Phi} = \begin{bmatrix}\boldsymbol{A}^{n-1}\boldsymbol{B} & \boldsymbol{A}^{n-2}\boldsymbol{B} & \ldots & \boldsymbol{B}\end{bmatrix}$

2: **repeat**

3: Compute $(k^*, i^*) = \underset{\substack{k:|\mathcal{S}_k|<s\\ i\notin\mathcal{S}_k}}{\arg\max} \frac{\left|\boldsymbol{\Phi}_{(k-1)m+i}^\top(\bar{\boldsymbol{x}}-\boldsymbol{\Phi}\boldsymbol{u})\right|}{\left\|\boldsymbol{\Phi}_{(k-1)m+i}\right\|}$

4: Update the support set of $\boldsymbol{u}_{k^*}$ as $\mathcal{S}_{k^*} \leftarrow \mathcal{S}_{k^*} \cup \{i^*\}$

5: Update the support set of $\boldsymbol{u}$ as $\mathcal{U} \leftarrow \mathcal{U} \cup \{(k^*-1)m + i^*\}$

6: Determine $\boldsymbol{u}_\mathcal{U} \leftarrow \boldsymbol{\Phi}_\mathcal{U}^\dagger \bar{\boldsymbol{x}}$ and $\boldsymbol{u}_{\mathcal{U}^\complement} \leftarrow \boldsymbol{0}$

7: **until** $\bar{\boldsymbol{x}} = \boldsymbol{\Phi}\boldsymbol{u}$

Output: Control inputs $\boldsymbol{u}$

Likewise, the top ns indices in Step 7 are replaced with the top s indices corresponding to each time index,

$$\mathcal{S}_k = \underset{\substack{\mathcal{I}\subseteq\{1,2,\ldots,m\}\\ |\mathcal{I}|\leq s}}{\arg\max} \sum_{i\in\mathcal{I}} \left|\boldsymbol{u}_{(k-1)m+i}\right|. \tag{3.15}$$

In other words, the support index identification steps are replaced by piecewise support identification to ensure the extra constraint: $\|\boldsymbol{u}_k\|_0 \leq s$ in (3.9). The pseudocode of the algorithm is summarized in Algorithm 5.

3.5 State-independent Actuator Scheduling

The goal of sparse input design with state-independent actuator schedule is to find a sparse actuator schedule $\{\mathcal{S}_k \subseteq \{1, 2, \ldots, m\}\}_{k=1}^n$ with , $|\mathcal{S}_k| \leq s$ such that the resulting controllability matrix has full rank. The actuator schedule gives the indices $\mathcal{U}$ of the nonzero entries of $\boldsymbol{u}$ in (3.3), i.e.,

$$\mathcal{U} = \bigcup_{k=1}^{n} \{i + (k-1)m : i \in \mathcal{S}_k\}. \tag{3.16}$$

Algorithm 5 Sparse input design via piecewise compressive sampling matching pursuit

Input: Initial state $\boldsymbol{x}_1 \in \mathbb{R}^n$, final state $\boldsymbol{x}_\mathrm{f} \in \mathbb{R}^n$, linear dynamical system $(\boldsymbol{A} \in \mathbb{R}^{n\times n}, \boldsymbol{B} \in \mathbb{R}^{n\times m})$, sparsity level $s < m$

Initialization: $\mathcal{S}_k = \emptyset$, for $k = 1, 2, \ldots, n$, $\boldsymbol{u} = \boldsymbol{0}$

Set $\bar{\boldsymbol{x}} = \boldsymbol{x}_\mathrm{f} - \boldsymbol{A}^n \boldsymbol{x}_1$ and $\boldsymbol{\Phi} = \begin{bmatrix} \boldsymbol{A}^{n-1}\boldsymbol{B} & \boldsymbol{A}^{n-2}\boldsymbol{B} & \ldots & \boldsymbol{B} \end{bmatrix}$

repeat

Update the support set for $k = 1, 2, \ldots, n$ as $\mathcal{S}_k \leftarrow \mathcal{S}_k \cup I_k^*$ using

$$\mathcal{I}_k^* = \underset{\substack{\mathcal{I} \subseteq \{1,2,\ldots,m\} \\ |\mathcal{I}| \leq 2s}}{\arg\max} \sum_{i \in \mathcal{I}} \frac{\left| \boldsymbol{\Phi}_{(k-1)m+i}^\mathsf{T} (\bar{\boldsymbol{x}} - \boldsymbol{\Phi}\boldsymbol{u}) \right|}{\left\| \boldsymbol{\Phi}_{(k-1)m+i} \right\|}$$

Update the support set of $\boldsymbol{u}$ as $\mathcal{U} = \bigcup_{k=1}^n \{i + (k-1)m : i \in \mathcal{S}_k\}$

Determine $\boldsymbol{u}_\mathcal{U} \leftarrow \boldsymbol{\Phi}_\mathcal{U}^\dagger \bar{\boldsymbol{x}}$ and $\boldsymbol{u}_{\mathcal{U}^\complement} \leftarrow \boldsymbol{0}$

Determine new support for $k = 1, 2, \ldots, n$ as

$$\mathcal{S}_k \leftarrow \underset{\substack{\mathcal{I} \subseteq \{1,2,\ldots,m\} \\ |\mathcal{I}| \leq s}}{\arg\max} \sum_{i \in \mathcal{I}} \left| \boldsymbol{u}_{(k-1)m+i} \right|$$

Update the support set of $\boldsymbol{u}$ as $\mathcal{U} = \bigcup_{k=1}^n \{i + (k-1)m : i \in \mathcal{S}_k\}$

Update the estimate $\boldsymbol{u}_{\mathcal{U}^\complement} \leftarrow \boldsymbol{0}$

until $\bar{\boldsymbol{x}} = \boldsymbol{\Phi}\boldsymbol{u}$

Output: Control inputs $\boldsymbol{u}$

Therefore, the resulting controllability matrix is

$$\boldsymbol{\Phi}_\mathcal{U} = \begin{bmatrix} \boldsymbol{A}^{n-1}\boldsymbol{B}_{\mathcal{S}_1} & \boldsymbol{A}^{n-2}\boldsymbol{B}_{\mathcal{S}_2} & \ldots & \boldsymbol{B}_{\mathcal{S}_n} \end{bmatrix}, \tag{3.17}$$

and the desired condition is

$$\operatorname{rank}\{\boldsymbol{\Phi}_\mathcal{U}\} = n. \tag{3.18}$$

Therefore, we derive $\boldsymbol{u}_{\mathcal{U}^\complement} = \boldsymbol{0}$, and the system of equations (3.6) simplifies as

$$\bar{\boldsymbol{x}} = \boldsymbol{\Phi}_\mathcal{U} \boldsymbol{u}_\mathcal{U}, \tag{3.19}$$

which does not have any sparsity constraint on the solution. Hence, for a given pair of initial and final states $(\boldsymbol{x}_1, \boldsymbol{x}_{\mathbf{f}})$, the nonzero entries of $\boldsymbol{u}$ are given by the least-square solution of (3.19),

$$\boldsymbol{u}_{\mathcal{U}} = \boldsymbol{\Phi}_{\mathcal{U}}^{\dagger}\bar{\boldsymbol{x}} \quad \text{and} \quad \boldsymbol{u}_{\mathcal{U}^{\complement}} = \mathbf{0}. \tag{3.20}$$

In the sequel, we discuss how to find a sparse actuator schedule; given the actuator schedule, the control inputs are given by (3.20).

As discussed in Section 3.2, we can solve for a sparse schedule using the following average sparsity constraint.

$$\begin{aligned} &\underset{\substack{\mathcal{S}_k \subseteq \{1,2,\ldots,m\} \\ k=1,2,\ldots,n}}{\arg\min} \sum_{k=1}^{n} |\mathcal{S}_k| \\ &\qquad \text{s. t.} \quad \operatorname{rank}\left\{\left[\boldsymbol{A}^{n-1}\boldsymbol{B}_{\mathcal{S}_1} \quad \boldsymbol{A}^{n-2}\boldsymbol{B}_{\mathcal{S}_2} \quad \ldots \quad \boldsymbol{B}_{\mathcal{S}_n}\right]\right\} = n. \end{aligned} \tag{3.21}$$

Alternatively, we can solve for a sparse schedule by imposing a sparsity constraint on each control input $\boldsymbol{u}_k$, i.e.,

$$\begin{aligned} &\underset{\substack{\mathcal{S}_k \subseteq \{1,2,\ldots,m\} \\ k=1,2,\ldots,n}}{\arg\min} \sum_{k=1}^{n} |\mathcal{S}_k| \quad \text{s. t.} \quad |\mathcal{S}_k| \leq s, \quad \text{for} \quad k = 1, 2, \ldots, n, \\ &\qquad \operatorname{rank}\left\{\left[\boldsymbol{A}^{n-1}\boldsymbol{B}_{\mathcal{S}_1} \quad \boldsymbol{A}^{n-2}\boldsymbol{B}_{\mathcal{S}_2} \quad \ldots \quad \boldsymbol{B}_{\mathcal{S}_n}\right]\right\} = n. \end{aligned} \tag{3.22}$$

The above combinatorial problems are hard to solve, and unlike the state-dependent sparse actuator scheduling problem, we do not optimize the control input here. Also, the solution to the problem may not be unique, and the different solution schedules can require different energy for the control inputs to drive the system to a desired state. For example, from (3.20), for a given controllability Gramian matrix $\boldsymbol{W} = \boldsymbol{\Phi}_{\mathcal{U}}\boldsymbol{\Phi}_{\mathcal{U}}^{\top}$, the worst-case energy of the control inputs is given by

$$\max_{\bar{\boldsymbol{x}}:\|\bar{\boldsymbol{x}}\|=1} \left\|\boldsymbol{\Phi}_{\mathcal{U}}^{\dagger}\bar{\boldsymbol{x}}\right\|^2 = \max_{\bar{\boldsymbol{x}}:\|\bar{\boldsymbol{x}}\|=1} \bar{\boldsymbol{x}}^{\top}\boldsymbol{W}^{-1}\bar{\boldsymbol{x}} = \lambda_{\max}\left(\boldsymbol{W}^{-1}\right) = \frac{1}{\lambda_{\min}\left(\boldsymbol{W}\right)}, \tag{3.23}$$

where $\lambda_{\max}$ and $\lambda_{\min}$ denote a matrix's maximum and minimum absolute eigenvalues, respectively. So, one approach to solving for a sparse schedule is to minimize the inverse of the minimum eigenvalue of the resulting controllability matrix. Similarly, we can design other controllability metrics to quantify the required control input energy in different

Table 3.1: Some popular controllability metrics, with $\boldsymbol{W}$ denoting the resulting controllability Gramian

Controllability Measure	Controllability metric
Average control energy	$\mathrm{tr}\left\{\boldsymbol{W}^{-1}\right\}$
The volume of the ellipsoid	$\lvert\boldsymbol{W}\rvert^{-1/n}$
Inverse of the trace	$\mathrm{tr}\left\{\boldsymbol{W}\right\}^{-1}$

ways. Table 3.1 summarizes a few popular control metrics from the literature (Siami *et al.*, 2020).

Based on the above observations, we reformulate the sparse scheduling problem in (3.21) as follows:

$$\underset{\substack{\mathcal{S}_k\subseteq\{1,2,\ldots,m\}\\ k=1,2,\ldots,n}}{\arg\min}\ \rho(\boldsymbol{W}) \ \text{ s. t. } \sum_{k=1}^{n}|\mathcal{S}_k|\leq ns, \quad \boldsymbol{W}=\boldsymbol{\Phi}_{\mathcal{U}}\boldsymbol{\Phi}_{\mathcal{U}}^{\mathsf{T}}$$
$$\text{and } \boldsymbol{\Phi}_{\mathcal{U}}=\begin{bmatrix}\boldsymbol{A}^{n-1}\boldsymbol{B}_{\mathcal{S}_1} & \boldsymbol{A}^{n-2}\boldsymbol{B}_{\mathcal{S}_2} & \ldots & \boldsymbol{B}_{\mathcal{S}_n}\end{bmatrix}, \quad (3.24)$$

where ρ can be a controllability metric (see Table 3.1 for examples). Moreover, we can rewrite $\boldsymbol{W}$ in the above equation as

$$\boldsymbol{W}=\sum_{k=1}^{n}\sum_{i\in\mathcal{S}_k}\boldsymbol{A}^{n-k}\boldsymbol{B}_i\boldsymbol{B}_i^{\mathsf{T}}\boldsymbol{A}^{(n-k)\mathsf{T}}. \quad (3.25)$$

As a result, (3.24) is equivalent to the following optimization problem:

$$\underset{\substack{\mathcal{S}_k\subseteq\{1,2,\ldots,m\}\\ k=1,2,\ldots,n}}{\arg\min}\ \rho\left(\sum_{k=1}^{n}\sum_{i\in\mathcal{S}_k}\boldsymbol{A}^{n-k}\boldsymbol{B}_i\boldsymbol{B}_i^{\mathsf{T}}\boldsymbol{A}^{(n-k)\mathsf{T}}\right) \text{ s. t. } \sum_{k=1}^{n}|\mathcal{S}_k|\leq ns. \quad (3.26)$$

Similarly, we can formulate the sparse scheduling problem with the sparsity constraint on each control input as follows:

$$\underset{\substack{\mathcal{S}_k\subseteq\{1,2,\ldots,m\}\\ k=1,2,\ldots,n}}{\arg\min}\ \rho\left(\sum_{k=1}^{n}\sum_{i\in\mathcal{S}_k}\boldsymbol{A}^{n-k}\boldsymbol{B}_i\boldsymbol{B}_i^{\mathsf{T}}\boldsymbol{A}^{(n-k)\mathsf{T}}\right) \text{ s. t. } |\mathcal{S}_k|\leq s, k=1,2,\ldots,n. \quad (3.27)$$

3.6 Sparse Actuator Scheduling Algorithms

The greedy heuristic sequentially selects one control input index in every iteration to maximize the controllability metric.

Here, the controllability metrics like average control energy $(\operatorname{tr}\left\{\boldsymbol{W}^{-1}\right\})$ and worst-case control energy $(1/\lambda_{\min}(\boldsymbol{W}))$ are undefined if $\boldsymbol{W}$ is not invertible. So, to avoid singularity, we use a small constant $\epsilon > 0$ to ensure that $\boldsymbol{W}+\epsilon\boldsymbol{I}$ is invertible. So the greedy cost function is $\rho(\boldsymbol{W}+\epsilon\boldsymbol{I})$, where ρ is the controllability metric.

The algorithms are similar in spirit to the orthogonal matching pursuit and piecewise orthogonal matching pursuit presented in Algorithm 6 and Algorithm 7, respectively.

Algorithm 6 Greedy sparse actuator scheduling

Input: Linear dynamical system $(\boldsymbol{A} \in \mathbb{R}^{n\times n}, \boldsymbol{B} \in \mathbb{R}^{n\times m})$
Initialization: $\mathcal{U} = \emptyset$, $\boldsymbol{W} = \boldsymbol{0}$, $\epsilon > 0$
Compute $\boldsymbol{\Phi} = \begin{bmatrix}\boldsymbol{A}^{n-1}\boldsymbol{B} & \boldsymbol{A}^{n-2}\boldsymbol{B} & \ldots & \boldsymbol{B}\end{bmatrix}$
repeat
  Compute $i^* = \arg\min_{i\notin\mathcal{U}} \rho\left(\boldsymbol{W} + \boldsymbol{\Phi}_i\boldsymbol{\Phi}_i^\top + \epsilon\boldsymbol{I}\right)$
  Update support $\mathcal{U} \leftarrow \mathcal{U} \cup \{i^*\}$
  Update the Gramian matrix $\boldsymbol{W} \leftarrow \boldsymbol{W} + \boldsymbol{\Phi}_{i^*}\boldsymbol{\Phi}_{i^*}^\top$
until $\operatorname{rank}\{\boldsymbol{W}\} = n$
Output: Actuator schedule $\mathcal{U}$

3.6.1 Greedy Sparse Actuator Scheduling

The greedy algorithm adds one control input index per iteration. Also, once an index is added, it is not removed in the subsequent iterations. The iterations are continued until the resulting controllability matrix is full rank. The resulting algorithm for the average sparsity problem in (3.26) is summarized in Algorithm 6 (Siami *et al.*, 2020).

Although the greedy algorithm is not guaranteed to converge to the optimal solution, it can be shown to provide a near-optimal solution using the submodularity property.

Algorithm 7 Greedy piecewise sparse actuator scheduling algorithm

Input: Linear dynamical system ($\boldsymbol{A} \in \mathbb{R}^{n\times n}, \boldsymbol{B} \in \mathbb{R}^{n\times m}$), sparsity level $s < m$

Initialization: $\mathcal{S}_k = \emptyset, \quad \text{for } k = 1, 2, \dots, n, \quad \boldsymbol{W} = \boldsymbol{0}, \ \epsilon > 0$

Set $\boldsymbol{\Phi} = \begin{bmatrix} \boldsymbol{A}^{n-1}\boldsymbol{B} & \boldsymbol{A}^{n-2}\boldsymbol{B} & \dots & \boldsymbol{B} \end{bmatrix}$

repeat

  Compute $(k^*, i^*) = \underset{k:|\mathcal{S}_k|<s, i\notin\mathcal{S}_k}{\arg\min}\ \rho\left(\boldsymbol{W} + \boldsymbol{\Phi}_{(k-1)m+i}\boldsymbol{\Phi}_{(k-1)m+i}^{\mathsf{T}} + \epsilon\boldsymbol{I}\right)$

  Update the support set as $\mathcal{S}_{k^*} \leftarrow \mathcal{S}_{k^*} \cup \{i^*\}$

  Update the Gramian matrix $\boldsymbol{W} \leftarrow \boldsymbol{W} + \boldsymbol{\Phi}_{(k^*-1)m+i^*}\boldsymbol{\Phi}_{(k^*-1)m+i^*}^{\mathsf{T}}$

until rank $\{\boldsymbol{W}\} = n$

Output: Actuator schedule $\{\mathcal{S}_k\}_{k=1}^{n}$

Definition 3.1. A set function f that maps $\mathcal{P}$ to $\mathbb{R}$ where $\mathcal{P}$ denotes the powerset of $\{1, 2, \dots, nm\}$ is said to be submodular if for all $\mathcal{A} \subseteq \mathcal{B} \subseteq \{1, 2, \dots, nm\}$ and for all $e \in \{1, 2, \dots, nm\} \setminus \mathcal{B}$, we have

$$f(\mathcal{A} \cup \{e\}) - f(\mathcal{A}) \geq f(\mathcal{B} \cup \{e\}) - f(\mathcal{B}). \tag{3.28}$$

Using the submodularity property, we can discuss the theoretical guarantee for the greedy algorithm.

Theorem 3.1 (Nemhauser *et al.*, 1978)**.** Consider the greedy algorithm in Algorithm 6 for a given linear dynamical system $(\boldsymbol{A}, \boldsymbol{B})$, sparsity level $s > n - \text{rank}\{\boldsymbol{A}\}$, and parameter ϵ. We define a function based on the cost function of as (3.26)

$$\rho_{\text{set}}(\mathcal{S}) = -\rho\left(\sum_{k=1}^{n} \sum_{\substack{i=j-(k-1)m \\ 1\leq i\leq m, j\in\mathcal{S}}} \boldsymbol{A}^{n-k}\boldsymbol{B}_i\boldsymbol{B}_i^{\mathsf{T}}\boldsymbol{A}^{(n-k)\mathsf{T}}\right), \tag{3.29}$$

If the function $\rho_{\text{set}}(\mathcal{S})$ is monotone and submodular, then the greedy algorithm outputs a schedule $\mathcal{S}^*$ that satisfies

$$\rho(\boldsymbol{\Phi}_{\mathcal{S}^*}) \leq (1 - 1/e) \min_{\mathcal{S}:|\mathcal{S}|\leq ns} \rho(\boldsymbol{\Phi}_{\mathcal{S}}). \tag{3.30}$$

While ρ_{set} of the average control energy cost $\text{tr}\left\{\boldsymbol{W}^{-1}\right\}$ does not satisfy submodularity, ρ_{set} of the volume of ellipsoid function $|\boldsymbol{W}|^{-1/n}$

is submodular. Therefore, the greedy algorithm has guaranteed performance with the volume of the ellipsoid function as the controllability metric (Olshevsky, 2017).

Furthermore, the performance guarantees for greedy algorithms with non-submodular controllability metrics can be obtained via the concepts of approximate submodularity (Chamon *et al.*, 2017), submodularity ratio, and curvature (Summers and Kamgarpour, 2019), which quantifies how far a set function is from being supermodular. These results show that average control energy cost $\operatorname{tr}\left\{\boldsymbol{W}^{-1}\right\}$ exhibits close to the submodularity property, and the greedy algorithm can reach a near-optimal solution.

3.6.2 Greedy Piecewise Sparse Actuator Scheduling

A greedy algorithm similar to Algorithm 6 to tackle the per-time instant sparsity-constrained problem in (3.27) is summarized in Algorithm 7. This algorithm obeys the piecewise sparsity constraints while adding an index to the support.

We note that the greedy piecewise sparse actuator scheduling algorithm is a naive method to solve the problem and is not guaranteed to converge to the optimal solution all the time. For example, consider the following system matrices

$$\boldsymbol{A} = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & \frac{3}{4} \\ 0 & 0 & 0 \end{bmatrix} \quad \text{and} \quad \boldsymbol{B} = \begin{bmatrix} \frac{3}{4} & 0 & 0 \\ 0 & \frac{1}{2} & 0 \\ 0 & 0 & \frac{1}{4} \end{bmatrix}. \tag{3.31}$$

The controllability matrix is given by

$$\boldsymbol{\Phi} = \begin{bmatrix} 0 & 0 & \frac{3}{16} & 0 & \frac{1}{2} & 0 & \frac{3}{4} & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & \frac{3}{16} & 0 & \frac{1}{2} & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \frac{1}{4} \end{bmatrix}. \tag{3.32}$$

Given that the rank of matrix $\boldsymbol{A}$ is 2, a sparsity level of $s = 1$ ensures controllability. Consequently, the first iteration of the greedy algorithm in Algorithm 7 consistently selects (k^*, i^*) as (3,3) for any ρ function choice from Table 3.1. Consequently, column 9 of $\boldsymbol{\Phi}$ cannot be chosen in subsequent iterations. Now, regardless of the subsequent column selections made by the algorithm, the resulting controllability matrix fails to meet the rank condition, leading to an incorrect termination of the algorithm. Therefore, the naive greedy algorithm proves ineffective in certain scenarios and needs refinement, an aspect that remains unaddressed in the existing literature.

As a concluding remark, we note that extending the compressive sampling matching pursuit algorithms for state-independent actuator scheduling is non-trivial and is an open problem.

3.7 Summary

- For sparse controllable linear dynamical systems, sparse inputs can be designed using actuator scheduling that can either be dependent or independent of initial and final states.
- The state-dependent sparse actuator scheduling with average sparsity constraint uses the standard sparse signal recovery algorithms.
- The state-dependent sparse actuator scheduling with per-time instant sparsity constraint uses the piecewise sparse signal recovery algorithms.
- The state-independent sparse actuator scheduling relies on suitable controllability metrics to also optimize the control energy and uses greedy algorithms to optimize the schedule.

3.8 Notes

An important aspect we did not cover in this section is the guarantee of the success of the compressed sensing algorithms for sparse input design with state-dependent actuator schedule in Section 3.2. We briefly discuss the related compressed sensing literature below.

Null Space Property

Consider the basis pursuit algorithm in Section 3.3.1. The ℓ_0-norm and ℓ_1-norm-based optimizations (3.10) and (3.12) are identical only under an assumption called the null-space property of the controllability matrix $\mathbf{\Phi}$. Specifically, every ns-sparse vector $\boldsymbol{u}$ satisfying $\bar{\boldsymbol{x}} = \mathbf{\Phi}\boldsymbol{u}$ is a solution to the optimization problem (3.12) if and only if $\mathbf{\Phi}$ satisfies the nullspace property with order $2ns$, i.e., if for all index sets $\mathcal{S}$, with $|\mathcal{S}| \leq 2ns$, the matrix $\mathbf{\Phi}$ satisfies,

$$\|z_{\mathcal{S}}\|_1 < \left\|z_{\mathcal{S}^{\complement}}\right\|_1 \tag{3.33}$$

for all vectors $\boldsymbol{z}$ belonging to the null space of $\mathbf{\Phi}$.

Restricted Isometric Property

Another important concept in compressed sensing is the restricted isometry property, which is commonly used as a sufficient condition for the null space property. A given matrix $\mathbf{\Phi}$ satisfies the restricted isometry property with restricted isometric constant δ_s of order s if all s-sparse vectors $\boldsymbol{z}$ satisfies

$$(1-\delta)\|z\|^2 \leq \|\boldsymbol{A}z\|^2 \leq (1+\delta)\|z\|^2, \tag{3.34}$$

for all $\delta \geq \delta_s$.

If the restricted isometric constant satisfies $\delta_{2ns} < \sqrt{2} - 1$, then $\mathbf{\Phi}$ satisfies the nullspace property with order $2ns$, implying the equivalence of (3.10) and (3.12).

Additionally, if the restricted isometric constant is such that $\delta_{13ns} < \frac{1}{6}$, the orthogonal matching pursuit algorithm in Algorithm 2 is guaranteed to succeed. Similarly if $\delta_{13ns} < \frac{\sqrt{\sqrt{11/3}-1}}{2} \approx 0.4782$, the compressive sampling matching pursuit algorithm in Algorithm 3 is guaranteed to succeed.

Coherence

Coherence is another simple measure used for the analysis of recovery algorithms. The ℓ_1-coherence function of $\mathbf{\Phi} \in \mathbb{R}^{n \times nm}$ is defined as

$$\mu_1(s) = \max_{i=1,2,\ldots,nm} \max\left\{\sum_{j\in\mathcal{S}} \frac{\mathbf{\Phi}_i^\top \mathbf{\Phi}_j}{\|\mathbf{\Phi}_i\|\left\|\mathbf{\Phi}_j\right\|},\ \ \mathcal{S} \subset \{1, 2, \ldots, nm\},\ \ |\mathcal{S}| \leq s\right\} \tag{3.35}$$

In general, the smaller the coherence, the better the recovery algorithms perform. The basis pursuit algorithm in Algorithm 1 and orthogonal matching pursuit algorithm in Algorithm 2 succeed if

$$\mu_1(ns) + \mu_1(ns - 1) < 1. \tag{3.36}$$

Furthermore, verifying whether a given matrix holds the null space property or restricted isometric property is difficult. Typically, random (for example, Gaussian) matrices satisfy these properties, but analyzing the null space property of $\mathbf{\Phi}$ even when $\boldsymbol{A}$ or $\boldsymbol{B}$ is random is an open problem. However, there has been a study on the restricted isometric property of the observability matrix when the system matrix is random (Joseph and Murthy, 2018; Joseph and Murthy, 2019). A similar analysis for the controllability matrix is missing in the literature.

For further details on these properties, the construction of compressive sensing matrices, and the role of randomness in establishing the above properties, please refer to Foucart *et al.* (2013) and references therein.

4

Variants of Controllability for Sparse Inputs with Time-Varying Support

This section builds upon the notion of sparse controllability and presents three related notions of controllability under input sparsity constraints: stabilizability, output controllability, and nonnegative controllability.

4.1 Sparse Stabilizability

We recall the state space model of the linear dynamical system $(\boldsymbol{A}, \boldsymbol{B})$, with state $\boldsymbol{x}_k \in \mathbb{R}^n$ and input $\boldsymbol{u}_k \in \mathbb{R}^m$,

$$\boldsymbol{x}_{k+1} = \boldsymbol{A}\boldsymbol{x}_k + \boldsymbol{B}\boldsymbol{u}_k, \tag{4.1}$$

for discrete-time indices $k = 1, 2, \ldots$. The linear dynamical system $(\boldsymbol{A}, \boldsymbol{B})$ in (4.1) is said to be (non-sparse) stabilizable if, for any initial state $\boldsymbol{x}_1$, there exists a sequence of inputs $\{\boldsymbol{u}_k\}_{k>0}$ such that $\lim_{k\to\infty} \|\boldsymbol{x}_k\| = 0$. The classical test for stabilizability is as follows:

Theorem 4.1 (Hespanha, 2018, Chapter 8)**.** The linear dynamical system $(\boldsymbol{A} \in \mathbb{R}^{n\times n}, \boldsymbol{B} \in \mathbb{R}^{n\times m})$ in (4.1) is stabilizable if and only if

$$\operatorname{rank}\left\{\begin{bmatrix}\boldsymbol{A} - \lambda \boldsymbol{I} & \boldsymbol{B}\end{bmatrix}\right\} = n, \tag{4.2}$$

for all $\lambda \in \mathbb{C}$ such that $|\lambda| \geq 1$.

We extend the notion of stabilizability to the case of sparse inputs and derive the stabilizability test as follows.

Definition 4.1. The linear dynamical system $(\boldsymbol{A}, \boldsymbol{B})$ in (4.1) is said to be s-sparse stabilizable for a given $s \geq 1$ if, for any initial state $\boldsymbol{x}_1$, there exists a sequence of s-sparse inputs $\{\boldsymbol{u}_k : \|\boldsymbol{u}_k\|_0 \leq s\}_{k>0}$ such that $\lim_{k\to\infty} \|\boldsymbol{x}_k\| = 0$.

As in the case of sparse controllability, sparse stabilizability is a stronger notion than stabilizability and a weaker notion than s-sparse controllability. So, the conditions for stabilizability in Theorem 4.1 are necessary for sparse stabilizability, and the conditions for sparse controllability in Theorem 2.8 are sufficient for s-sparse stabilizability. Further, we recall that Theorem 4.1 asserts that the system is stabilizable if and only if the unstable part of the system is controllable. Therefore, the system is s-sparse stabilizable if and only if the unstable part of the system is s-sparse controllable. However, the eigenvalues of the state matrix corresponding to the unstable part of the system are greater than or equal to 1; hence, the unstable part is always s-sparse controllable. Therefore, we conclude that stabilizable systems are sparse stabilizable. This result is summarized below.

Theorem 4.2 (Sriram *et al.*, 2022)**.** For any sparsity level, $s \geq 1$, the linear dynamical system $(\boldsymbol{A}, \boldsymbol{B})$ in (4.1) is s-sparse stabilizable if and only if it is stabilizable.

Proof. Since any sparse stabilizable linear dynamical system is also stabilizable, it is enough to show that stabilizability implies sparse stabilizability. To prove this claim, suppose that the system is stabilizable. We then construct the canonical form using the real Jordan decomposition of the state transition matrix $\boldsymbol{A}$ as follows:

$$\boldsymbol{A} = \boldsymbol{P}^{-1} \begin{bmatrix} \boldsymbol{S} & \boldsymbol{0} \\ \boldsymbol{0} & \boldsymbol{T} \end{bmatrix} \boldsymbol{P} = \begin{bmatrix} \boldsymbol{P}^{(1)} \\ \boldsymbol{P}^{(2)} \end{bmatrix}^{-1} \begin{bmatrix} \boldsymbol{S} & \boldsymbol{0} \\ \boldsymbol{0} & \boldsymbol{T} \end{bmatrix} \begin{bmatrix} \boldsymbol{P}^{(1)} \\ \boldsymbol{P}^{(2)} \end{bmatrix}, \tag{4.3}$$

where $\boldsymbol{P} = \begin{bmatrix} \boldsymbol{P}^{(1)\mathsf{T}} & \boldsymbol{P}^{(2)\mathsf{T}} \end{bmatrix}^\mathsf{T}$ is an invertible matrix, and the square matrices $\boldsymbol{S}$ and $\boldsymbol{T}$ are formed by the Jordan blocks of $\boldsymbol{A}$ corresponding to the eigenvalues whose absolute value is greater than 1 and the eigenvalues whose absolute value is less than 1, respectively.

Premultiplying (4.1) with $\boldsymbol{P}$ gives

$$\boldsymbol{P}^{(1)}\boldsymbol{x}_{k+1} = \boldsymbol{S}\boldsymbol{P}^{(1)}\boldsymbol{x}_k + \boldsymbol{P}^{(1)}\boldsymbol{B}\boldsymbol{u}_k \tag{4.4}$$

$$\boldsymbol{P}^{(2)}\boldsymbol{x}_{k+1} = \boldsymbol{T}\boldsymbol{P}^{(2)}\boldsymbol{x}_k + \boldsymbol{P}^{(2)}\boldsymbol{B}\boldsymbol{u}_k, \tag{4.5}$$

Let $\boldsymbol{x}_k^{(1)} = \boldsymbol{P}^{(1)}\boldsymbol{x}_k$ and $\boldsymbol{x}_k^{(2)} = \boldsymbol{P}^{(2)}\boldsymbol{x}_k$ for all values of $k \geq 0$. Then, we arrive at

$$\boldsymbol{x}_{k+1}^{(1)} = \boldsymbol{S}\boldsymbol{x}_k^{(1)} + \boldsymbol{P}^{(1)}\boldsymbol{B}\boldsymbol{u}_k \tag{4.6}$$

$$\boldsymbol{x}_{k+1}^{(2)} = \boldsymbol{T}\boldsymbol{x}_k^{(2)} + \boldsymbol{P}^{(2)}\boldsymbol{B}\boldsymbol{u}_k. \tag{4.7}$$

Then, we have two linear dynamical systems $(\boldsymbol{S}, \boldsymbol{P}^{(1)}\boldsymbol{B})$ and $(\boldsymbol{T}, \boldsymbol{P}^{(2)}\boldsymbol{B})$ with a common input $\boldsymbol{u}_k$.

Further, since the linear dynamical system $(\boldsymbol{A}, \boldsymbol{B})$ is stabilizable, Theorem 4.1 ensures that for any $\lambda \in \mathbb{C}$ such that $|\lambda| \geq 1$

$$\text{rank}\left\{\begin{bmatrix}\boldsymbol{A} - \lambda\boldsymbol{I} & \boldsymbol{B}\end{bmatrix}\right\} = n. \tag{4.8}$$

Also, from (4.3), we obtain

$$n = \text{rank}\left\{\boldsymbol{P}\begin{bmatrix}\boldsymbol{A} - \lambda\boldsymbol{I} & \boldsymbol{B}\end{bmatrix}\begin{bmatrix}\boldsymbol{P}^{-1} & \boldsymbol{0}\\ \boldsymbol{0} & \boldsymbol{I}\end{bmatrix}\right\} \tag{4.9}$$

$$= \text{rank}\left\{\begin{bmatrix}\boldsymbol{S} - \lambda\boldsymbol{I} & \boldsymbol{0} & \boldsymbol{P}^{(1)}\boldsymbol{B}\\ \boldsymbol{0} & \boldsymbol{T} - \lambda\boldsymbol{I} & \boldsymbol{P}^{(2)}\boldsymbol{B}\end{bmatrix}\right\}. \tag{4.10}$$

Hence, $\begin{bmatrix}\boldsymbol{S} - \lambda\boldsymbol{I} & \boldsymbol{P}^{(1)}\boldsymbol{B}\end{bmatrix}$ has full row rank for any $\lambda \in \mathbb{C}$ such that $|\lambda| \geq 1$. Furthermore, from (4.3), all eigenvalues of $\boldsymbol{S}$ are greater than 1. So, $\boldsymbol{S}$ is invertible and $\begin{bmatrix}\boldsymbol{S} - \lambda\boldsymbol{I} & \boldsymbol{P}^{(1)}\boldsymbol{B}\end{bmatrix}$ has full row rank for any $\lambda \in \mathbb{C}$ such that $|\lambda| < 1$. In other words, $\begin{bmatrix}\boldsymbol{S} - \lambda\boldsymbol{I} & \boldsymbol{P}^{(1)}\boldsymbol{B}\end{bmatrix}$ has full row rank for any $\lambda \in \mathbb{C}$. From Theorem 2.9, we conclude that the linear dynamical system $(\boldsymbol{S}, \boldsymbol{P}^{(1)}\boldsymbol{B})$ is s-sparse controllable for any $s > 0$. Hence, for any initial state $\boldsymbol{x}_1$, we can find s-sparse inputs $\{\boldsymbol{u}_k\}_{k=1}^n$ such that the linear dynamical system $(\boldsymbol{S}, \boldsymbol{P}^{(1)}\boldsymbol{B})$ is driven to the all-zero state, i.e., $\boldsymbol{x}_{n+1}^{(1)} = \boldsymbol{P}^{(1)}\boldsymbol{x}_{n+1} = \boldsymbol{0}$. However, these inputs do ensure that $\boldsymbol{P}^{(2)}\boldsymbol{x}_n = \boldsymbol{0}$.

Furthermore, if we choose $\boldsymbol{u}_k = 0$ for $k > n$, from (4.6), we get

$$\boldsymbol{P}^{(1)}\boldsymbol{x}_k = \boldsymbol{0} \quad \forall \;\; k > n. \tag{4.11}$$

Hence, from (4.7), for any $k > n+1$,

$$\|\boldsymbol{x}_{k+1}\| = \left\|\boldsymbol{P}^{-1}\boldsymbol{P}\boldsymbol{x}_{k+1}\right\| \leq \left\|\boldsymbol{P}^{-1}\right\| \|\boldsymbol{P}\boldsymbol{x}_{k+1}\| \tag{4.12}$$

$$= \left\|\boldsymbol{P}^{-1}\right\| \left\|\boldsymbol{P}^{(2)}\boldsymbol{x}_{k+1}\right\|, \tag{4.13}$$

where we use (4.11). Using the fact that $\boldsymbol{u}_k = 0$ for $k > n+1$, we have

$$\boldsymbol{P}^{(2)}\boldsymbol{x}_{k+1} = \boldsymbol{T}\boldsymbol{P}^{(2)}\boldsymbol{x}_k = \boldsymbol{T}^{k-n}\boldsymbol{P}^{(2)}\boldsymbol{x}_{n+1}. \tag{4.14}$$

Consequently, we can further upper bound (4.13) as

$$\|\boldsymbol{x}_{k+1}\| \leq \left\|\boldsymbol{P}^{-1}\right\| \left\|\boldsymbol{T}^{k-n}\boldsymbol{P}^{(2)}\boldsymbol{x}_{n+1}\right\| \leq \|\boldsymbol{T}\|^{k-n} \left\|\boldsymbol{P}^{-1}\right\| \left\|\boldsymbol{P}^{(2)}\boldsymbol{x}_{n+1}\right\|. \tag{4.15}$$

Since all the eigenvalues of $\boldsymbol{T}$ are less than 1, we deduce

$$\lim_{k\to\infty} \|\boldsymbol{x}_k\| = 0. \tag{4.16}$$

Hence, the linear dynamical system $(\boldsymbol{A}, \boldsymbol{B})$ is s-sparse stabilizable for any $s > 0$. The proof is complete. □

The above result establishes that sparse stabilizability is equivalent to stabilizability, unlike its controllability counterpart. The resulting relationship between sparse controllability, controllability, and sparse stabilizability is depicted by the Venn diagram in Figure 4.1.

The proof of Theorem 4.2 provides an approach to finding sparse inputs that stabilize the system. Specifically, for any initial state $\boldsymbol{x}_1$, we determine sparse inputs $\{\boldsymbol{u}_k\}_{k=1}^{n}$ that drive the system $(\boldsymbol{S}, \boldsymbol{P}^{(1)}\boldsymbol{B})$ from $\boldsymbol{P}^{(1)}\boldsymbol{x}_1$ to the all-zero state. Here, $\boldsymbol{S}$ and $\boldsymbol{P}^{(1)}$ are defined in (4.3). The sparse inputs can be determined using any of the algorithms (Algorithms 1 to 7) described in Section 3.

4.2 Sparse Output Controllability

To define the notion of output controllability, we consider a linear dynamical system whose state and observations are as follows:

$$\boldsymbol{x}_{k+1} = \boldsymbol{A}\boldsymbol{x}_k + \boldsymbol{B}\boldsymbol{u}_k \quad \text{and} \quad \boldsymbol{y}_k = \boldsymbol{C}\boldsymbol{x}_k, \tag{4.17}$$

where $\boldsymbol{y}_k$ is the observation at time k and $\boldsymbol{C}$ is the observation matrix. The linear dynamical system $(\boldsymbol{A}, \boldsymbol{B}, \boldsymbol{C})$ in (4.17) is said to be (non-sparse) output controllable if, for any initial state $\boldsymbol{x}_1$ and final output

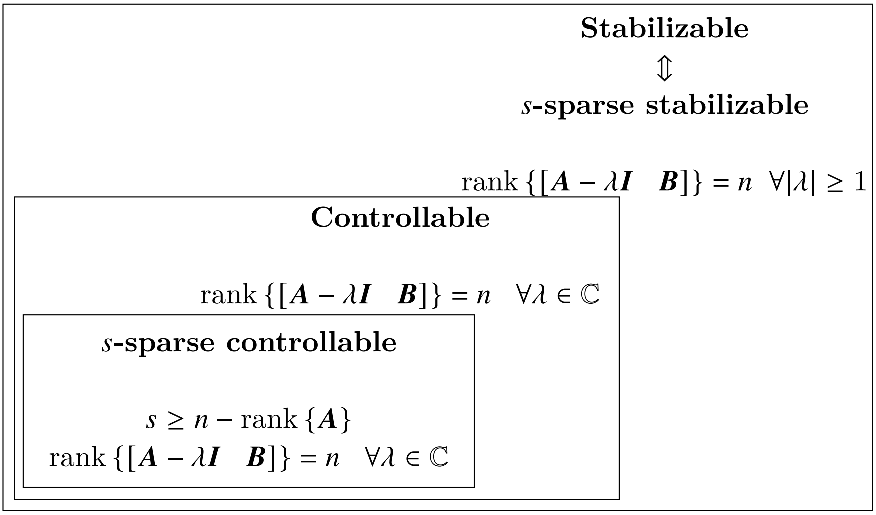

Figure 4.1: Venn diagram showing the relation between sparse controllability, controllability, sparse stabilizability, and stabilizability, and the necessary and sufficient conditions that each set satisfies.

$\boldsymbol{y}_{\mathrm{f}}$, there exist inputs $\{\boldsymbol{u}_k\}_{k=1}^{n}$ that drive the system from $\boldsymbol{x}_1$ to the output $\boldsymbol{y}_{n+1} = \boldsymbol{y}_{\mathrm{f}}$. The classical Kalman rank test for output controllability is as follows:

Theorem 4.3. The linear dynamical system $(\boldsymbol{A} \in \mathbb{R}^{n\times n}, \boldsymbol{B} \in \mathbb{R}^{n\times m}, \mathbb{C} \in \mathbb{R}^{p\times n})$ in (4.17) is output controllable if and only if

$$\operatorname{rank}\left\{\boldsymbol{C}\left[\boldsymbol{A}^{n-1}\boldsymbol{B} \quad \boldsymbol{A}^{n-2}\boldsymbol{B} \quad \dots \quad \boldsymbol{B}\right]\right\} = p. \tag{4.18}$$

We extend the notion of output controllability to the case of sparse inputs and derive the necessary and sufficient conditions for output controllability as follows.

Definition 4.2. The linear dynamical system $(\boldsymbol{A}, \boldsymbol{B}, \boldsymbol{C})$ in (4.17) is said to be s-sparse output controllable for a given $s \geq 1$ if, for any initial state $\boldsymbol{x}_1$ and final output $\boldsymbol{y}_{\mathrm{f}}$, there exist an integer $K < \infty$ and s-sparse inputs $\{\boldsymbol{u}_k : \|\boldsymbol{u}_k\|_0 \leq s\}_{k=1}^{K}$ that drives the system from $\boldsymbol{x}_1$ to the output $\boldsymbol{y}_{K+1} = \boldsymbol{y}_{\mathrm{f}}$.

The extension of the Kalman rank test for sparse output controllability is as follows.

Theorem 4.4 (Joseph, 2022b)**.** The linear dynamical system ($\boldsymbol{A} \in \mathbb{R}^{n\times n}, \boldsymbol{B} \in \mathbb{R}^{n\times m}, \boldsymbol{C} \in \mathbb{R}^{p\times n}$) in (4.17) is s-sparse output controllable if and only if there exist an integer $K < \infty$ and sets $\{\mathcal{S}_k \subseteq \{1, 2, \ldots, m\},\ |\mathcal{S}_k| \leq s\}_{k=1}^{K}$ such that

$$\text{rank}\left\{\boldsymbol{C}\begin{bmatrix}\boldsymbol{A}^{K-1}\boldsymbol{B}_{\mathcal{S}_1} & \boldsymbol{A}^{K-2}\boldsymbol{B}_{\mathcal{S}_2} & \ldots & \boldsymbol{B}_{\mathcal{S}_K}\end{bmatrix}\right\} = p. \tag{4.19}$$

Proof. The proof is similar to that of Theorem 2.3, and we omit it here. □

Although Theorem 4.4 gives the necessary and sufficient condition for sparse output controllability, verifying the condition is computationally expensive due to its combinatorial nature. Therefore, we seek a result similar to Theorem 2.8 for output controllability. However, Theorem 2.8 is based on the PBH test for controllability, and an analogous PBH test is not available for output sparse controllability. So, the extension of Theorem 2.8 to output sparse controllability is not straightforward and needs new proof techniques.

As in the case of sparse controllability, sparse output controllability is a stronger notion than output controllability. So, the condition $\text{rank}\{\boldsymbol{C}\boldsymbol{\Phi}\} = p$ for output controllability in Theorem 4.3 is necessary for sparse output controllability, where

$$\boldsymbol{\Phi} = \begin{bmatrix}\boldsymbol{A}^{n-1}\boldsymbol{B} & \boldsymbol{A}^{n-2}\boldsymbol{B} & \ldots & \boldsymbol{B}\end{bmatrix} \in \mathbb{R}^{n\times nm}. \tag{4.20}$$

We first provide weaker but non-combinatorial conditions to check sparse output controllability. The first result establishes the necessary conditions for sparse output controllability of a controllable system.

Proposition 4.1. The linear dynamical system ($\boldsymbol{A} \in \mathbb{R}^{n\times n}, \boldsymbol{B} \in \mathbb{R}^{n\times m}, \boldsymbol{C} \in \mathbb{R}^{p\times n}$) in (4.17) is s-sparse output controllable only if

$$\text{rank}\{\boldsymbol{C}\boldsymbol{\Phi}\} = p \text{ and } s \geq \max_{1\leq i\leq n} \frac{1}{i}\left(p - \text{rank}\left\{\boldsymbol{C}\boldsymbol{A}^i\right\}\right), \tag{4.21}$$

where $\boldsymbol{\Phi}$ is the controllability matrix defined in (4.20).

Proof. The necessity of $\text{rank}\{\boldsymbol{C}\boldsymbol{\Phi}\} = p$ is trivial from Theorem 4.3. Next, if the linear dynamical system $(\boldsymbol{A}, \boldsymbol{B}, \boldsymbol{C})$ in (4.17) is s-sparse output controllable, from Theorem 4.4, there exist an integer $K < \infty$ and sets $\{\mathcal{S}_k \subseteq \{1, 2, \ldots, m\},\ |\mathcal{S}_k| = s\}_{k=1}^{K}$ such that

$$\text{ColSpace}\{\boldsymbol{C}\left[\boldsymbol{A}^{K-1}\boldsymbol{B}_{\mathcal{S}_1} \quad \boldsymbol{A}^{K-2}\boldsymbol{B}_{\mathcal{S}_2} \quad \ldots \quad \boldsymbol{B}_{\mathcal{S}_K}\right]\} = \mathbb{R}^p. \tag{4.22}$$

Also, for any $1 \leq i \leq K-1$ we have

$$\text{ColSpace}\left\{\boldsymbol{C}\left[\boldsymbol{A}^{K-1}\boldsymbol{B}_{\mathcal{S}_1} \quad \boldsymbol{A}^{K-2}\boldsymbol{B}_{\mathcal{S}_2} \quad \ldots \quad \boldsymbol{A}^{i}\boldsymbol{B}_{\mathcal{S}_{K-i}}\right]\right\} \subseteq \text{ColSpace}\left\{\boldsymbol{C}\boldsymbol{A}^i\right\}. \tag{4.23}$$

Therefore, from (4.22), we derive

$$\text{rank}\left\{\boldsymbol{C}\left[\boldsymbol{A}^i \quad \boldsymbol{A}^{i-1}\boldsymbol{B}_{\mathcal{S}_{K-i+1}} \quad \boldsymbol{A}^{i-2}\boldsymbol{B}_{\mathcal{S}_{K-i+2}} \quad \ldots \quad \boldsymbol{B}_{\mathcal{S}_K}\right]\right\} = p. \tag{4.24}$$

Hence, we arrive at

$$\text{rank}\left\{\boldsymbol{C}\boldsymbol{A}^i\right\} + \text{rank}\left\{\boldsymbol{C}\left[\boldsymbol{A}^{i-1}\boldsymbol{B}_{\mathcal{S}_{K-i+1}} \quad \boldsymbol{A}^{i-2}\boldsymbol{B}_{\mathcal{S}_{K-i+2}} \quad \ldots \quad \boldsymbol{B}_{\mathcal{S}_K}\right]\right\} \geq p. \tag{4.25}$$

Also, since $\boldsymbol{C}\left[\boldsymbol{A}^{i-1}\boldsymbol{B}_{\mathcal{S}_{K-i+1}} \quad \boldsymbol{A}^{i-2}\boldsymbol{B}_{\mathcal{S}_{K-i+2}} \quad \ldots \quad \boldsymbol{B}_{\mathcal{S}_K}\right] \in \mathbb{R}^{p \times is}$, we get

$$\text{rank}\left\{\boldsymbol{C}\boldsymbol{A}^i\right\} + is \geq p. \tag{4.26}$$

Rewriting the above expression, we obtain for all $i = 1, 2 \ldots,$

$$s \geq \frac{1}{i}\left(p - \text{rank}\left\{\boldsymbol{C}\boldsymbol{A}^i\right\}\right), \tag{4.27}$$

which leads to the desired necessary condition. □

We next provide a set of sufficient conditions for sparse output controllability. We first derive the results for controllable linear dynamical systems, i.e., we assume that the controllability matrix $\boldsymbol{\Phi}$ in (4.20) has full rank, i.e., $\text{rank}\{\boldsymbol{\Phi}\} = n$. We relax later this controllability assumption in Theorem 4.8.

Proposition 4.2. Suppose that the linear dynamical system ($\boldsymbol{A} \in \mathbb{R}^{n \times n}, \boldsymbol{B} \in \mathbb{R}^{n \times m}, \boldsymbol{C} \in \mathbb{R}^{p \times n}$) in (4.17) is controllable. Then, the system is s-sparse output controllable if

$$\text{rank}\{\boldsymbol{C}\boldsymbol{\Phi}\} = p \text{ and } s \geq \max_{1 \leq i \leq n} \text{rank}\left\{\boldsymbol{C}\boldsymbol{A}^i\right\} - \text{rank}\left\{\boldsymbol{C}\boldsymbol{A}^{i-1}\right\}, \tag{4.28}$$

where $\boldsymbol{\Phi}$ is the controllability matrix defined in (4.20).

We isolate three lemmas to prove that the conditions are sufficient.

Lemma 4.5. Suppose that the linear dynamical system ($\boldsymbol{A} \in \mathbb{R}^{n\times n}, \boldsymbol{B} \in \mathbb{R}^{n\times m}, \boldsymbol{C} \in \mathbb{R}^{p\times n}$) in (4.17) is controllable. Then, for $0 \le i \le n-1$,

$$\mathrm{ColSpace}\left\{\boldsymbol{C}^{(i)} - \boldsymbol{C}^{(i+1)}\right\} = \mathrm{ColSpace}\left\{\left[\boldsymbol{I} - \boldsymbol{C}^{(i+1)}\right]\boldsymbol{C}\boldsymbol{A}^{i}\right\}, \tag{4.29}$$

where $\boldsymbol{C}^{(i)} = \boldsymbol{C}\boldsymbol{A}^{i}\left(\boldsymbol{C}\boldsymbol{A}^{i}\right)^{\dagger}$. Also,

$$\mathrm{rank}\left\{\boldsymbol{C}^{(i)} - \boldsymbol{C}^{(i+1)}\right\} = \mathrm{rank}\left\{\boldsymbol{C}\boldsymbol{A}^{i}\right\} - \mathrm{rank}\left\{\boldsymbol{C}\boldsymbol{A}^{i+1}\right\}. \tag{4.30}$$

Proof. We start by noting that for any $0 \le i \le n-1$, $\boldsymbol{C}^{(i)}$ and $\boldsymbol{C}^{(i+1)}$ are symmetric and

$$\boldsymbol{C}^{(i+1)} = \left[\boldsymbol{C}\boldsymbol{A}^{i+1}\left(\boldsymbol{C}\boldsymbol{A}^{i+1}\right)^{\dagger}\right]^{\mathsf{T}} = \left[\boldsymbol{C}\boldsymbol{A}^{i}\boldsymbol{A}\left(\boldsymbol{C}\boldsymbol{A}^{i+1}\right)^{\dagger}\right]^{\mathsf{T}} \tag{4.31}$$

$$= \left[\boldsymbol{C}\boldsymbol{A}^{i}\left(\boldsymbol{C}\boldsymbol{A}^{i}\right)^{\dagger}\boldsymbol{C}\boldsymbol{A}^{i}\boldsymbol{A}\left(\boldsymbol{C}\boldsymbol{A}^{i+1}\right)^{\dagger}\right]^{\mathsf{T}} = \left[\boldsymbol{C}^{(i)}\boldsymbol{C}^{(i+1)}\right]^{\mathsf{T}} = \boldsymbol{C}^{(i+1)}\boldsymbol{C}^{(i)}. \tag{4.32}$$

Therefore, we derive

$$\begin{aligned}\mathrm{ColSpace}\left\{\boldsymbol{C}^{(i)} - \boldsymbol{C}^{(i+1)}\right\} &= \mathrm{ColSpace}\left\{\boldsymbol{C}^{(i)} - \boldsymbol{C}^{(i+1)}\boldsymbol{C}^{(i)}\right\} &(4.33)\\ &= \mathrm{ColSpace}\left\{\left[\boldsymbol{I} - \boldsymbol{C}^{(i+1)}\right]\boldsymbol{C}^{(i)}\right\} &(4.34)\\ &= \mathrm{ColSpace}\left\{\left[\boldsymbol{I} - \boldsymbol{C}^{(i+1)}\right]\boldsymbol{C}\boldsymbol{A}^{i}\right\}, &(4.35)\end{aligned}$$

which follows because $\mathrm{ColSpace}\left\{\boldsymbol{C}^{(i)}\right\} = \mathrm{ColSpace}\left\{\boldsymbol{C}\boldsymbol{A}^{i}\right\}$.

Furthermore, since $\mathrm{ColSpace}\left\{\boldsymbol{C}\boldsymbol{A}^{i+1}\right\} \subseteq \mathrm{ColSpace}\left\{\boldsymbol{C}\boldsymbol{A}^{i}\right\}$, we also deduce

$$\begin{aligned}\mathrm{rank}\left\{\boldsymbol{C}\boldsymbol{A}^{i}\right\} &= \mathrm{rank}\left\{\boldsymbol{C}\boldsymbol{A}^{i+1}\right\} + \mathrm{rank}\left\{\left[\boldsymbol{I} - \boldsymbol{C}\boldsymbol{A}^{i+1}\left(\boldsymbol{C}\boldsymbol{A}^{i+1}\right)^{\dagger}\right]\boldsymbol{C}\boldsymbol{A}^{i}\right\} &(4.36)\\ &= \mathrm{rank}\left\{\boldsymbol{C}\boldsymbol{A}^{i+1}\right\} + \mathrm{rank}\left\{\left[\boldsymbol{I} - \boldsymbol{C}^{(i+1)}\right]\boldsymbol{C}\boldsymbol{A}^{i}\right\} &(4.37)\\ &= \mathrm{rank}\left\{\boldsymbol{C}\boldsymbol{A}^{i+1}\right\} + \mathrm{rank}\left\{\boldsymbol{C}^{(i)} - \boldsymbol{C}^{(i+1)}\right\}, &(4.38)\end{aligned}$$

where we use (4.35). Thus, we have the rank condition.

□

Lemma 4.6. Suppose that the linear dynamical system ($\boldsymbol{A} \in \mathbb{R}^{n\times n}, \boldsymbol{B} \in \mathbb{R}^{n\times m}, \boldsymbol{C} \in \mathbb{R}^{p\times n}$) in (4.17) is controllable. Then, for $0 \le i \le n-1$, there exists a set $\mathcal{S} \subseteq \{1, 2, \ldots, m\}$ such that $|\mathcal{S}| = \text{rank}\left\{\boldsymbol{C}\boldsymbol{A}^i\right\} - \text{rank}\left\{\boldsymbol{C}\boldsymbol{A}^{i+1}\right\}$ and

$$\text{ColSpace}\left\{\boldsymbol{C}^{(i)} - \boldsymbol{C}^{(i+1)}\right\} = \text{ColSpace}\left\{\left[\boldsymbol{I} - \boldsymbol{C}^{(i+1)}\right]\boldsymbol{C}\boldsymbol{A}^i\boldsymbol{B}_{\mathcal{S}}\right\}. \quad (4.39)$$

where $\boldsymbol{C}^{(i)} = \boldsymbol{C}\boldsymbol{A}^i\left(\boldsymbol{C}\boldsymbol{A}^i\right)^{\dagger}$.

Proof. Since the system is controllable, from Theorem 2.1 and (4.20), we have $\text{ColSpace}\{\boldsymbol{\Phi}\} = \mathbb{R}^n$. Therefore, from Theorem 4.5, we have

$$\begin{aligned}\text{ColSpace}\left\{\boldsymbol{C}^{(i)} - \boldsymbol{C}^{(i+1)}\right\} &= \text{ColSpace}\left\{\left[\boldsymbol{I} - \boldsymbol{C}^{(i+1)}\right]\boldsymbol{C}\boldsymbol{A}^i\boldsymbol{\Phi}\right\} &(4.40)\\ &= \text{ColSpace}\left\{\left[\boldsymbol{I} - \boldsymbol{C}^{(i+1)}\right]\boldsymbol{C}\boldsymbol{A}^i\boldsymbol{B}\right\}, &(4.41)\end{aligned}$$

which follows from the fact that for any $k \geq 1$

$$\left[\boldsymbol{I} - \boldsymbol{C}^{(i+1)}\right]\boldsymbol{C}\boldsymbol{A}^i\boldsymbol{A}^k = \boldsymbol{C}\boldsymbol{A}^{i+k} - \boldsymbol{C}\boldsymbol{A}^{i+1}\left(\boldsymbol{C}\boldsymbol{A}^{i+1}\right)^{\dagger}\boldsymbol{C}\boldsymbol{A}^{i+k} = \boldsymbol{0}. \quad (4.42)$$

Also, from Theorem 4.5, we obtain

$$\text{rank}\left\{\boldsymbol{C}^{(i)} - \boldsymbol{C}^{(i+1)}\right\} = \text{rank}\left\{\boldsymbol{C}\boldsymbol{A}^i\right\} - \text{rank}\left\{\boldsymbol{C}\boldsymbol{A}^{i+1}\right\} = |\mathcal{S}|. \quad (4.43)$$

Thus, from (4.41), there exist $|\mathcal{S}|$ columns in $\left[\boldsymbol{I} - \boldsymbol{C}^{(i+1)}\right]\boldsymbol{C}\boldsymbol{A}^i\boldsymbol{B}$ that span $\text{ColSpace}\left\{\boldsymbol{C}^{(i)} - \boldsymbol{C}^{(i+1)}\right\}$, and the proof is complete. □

Lemma 4.7. Suppose that the linear dynamical system ($\boldsymbol{A} \in \mathbb{R}^{n\times n}, \boldsymbol{B} \in \mathbb{R}^{n\times m}, \boldsymbol{C} \in \mathbb{R}^{p\times n}$) in (4.17) is controllable. Then, for any sparsity level $s \leq m$, there exist an integer $K < \infty$ and sets $\{\mathcal{S}_k \subseteq \{1, 2, \ldots, m\} : |\mathcal{S}_k| = s\}_{k=1}^K$ such that

$$\text{ColSpace}\{\boldsymbol{C}\boldsymbol{A}^n\} \subseteq \text{ColSpace}\left\{\boldsymbol{C}\boldsymbol{A}^n\left[\boldsymbol{A}^{K-1}\boldsymbol{B}_{\mathcal{S}_1} \quad \boldsymbol{A}^{K-2}\boldsymbol{B}_{\mathcal{S}_2} \quad \ldots \quad \boldsymbol{B}_{\mathcal{S}_K}\right]\right\}. \quad (4.44)$$

Proof. Let the real Jordan canonical of $\boldsymbol{A}$ be

$$\boldsymbol{A} = \boldsymbol{P}^{-1}\begin{bmatrix}\boldsymbol{J} & \boldsymbol{0}\\ \boldsymbol{0} & \boldsymbol{N}\end{bmatrix}\boldsymbol{P}, \quad (4.45)$$

where $\boldsymbol{P} \in \mathbb{R}^{n\times n}$ is an invertible matrix and the square matrices $\boldsymbol{J}$ and $\boldsymbol{N}$ are formed by the Jordan blocks of $\boldsymbol{A}$ corresponding to its nonzero and zero eigenvalues, respectively. Here, $\boldsymbol{J}$ is an invertible matrix, and

$\boldsymbol{N}$ is a nilpotent matrix, i.e., $\boldsymbol{N}^n = \boldsymbol{0}$. Consequently, for any $n \leq P$, we deduce

$$\boldsymbol{A}^P = \boldsymbol{P}^{-1} \begin{bmatrix} \boldsymbol{J}^P & \boldsymbol{0} \\ \boldsymbol{0} & \boldsymbol{0} \end{bmatrix} \boldsymbol{P}. \tag{4.46}$$

Also, for any $n \leq P \leq Q$, by the Cayley-Hamilton theorem, we know that there exist real numbers $\{\alpha_j\}_{j=0}^{n-1}$ such that $\boldsymbol{J}^{P-Q} = \sum_{j=0}^{n-1} \alpha_j \boldsymbol{J}^j$. Substituting this relation into (4.46) yields

$$\boldsymbol{A}^P = \boldsymbol{P}^{-1} \begin{bmatrix} \boldsymbol{J}^Q \sum_{j=0}^{n-1} \alpha_j \boldsymbol{J}^j & \boldsymbol{0} \\ \boldsymbol{0} & \boldsymbol{0} \end{bmatrix} \boldsymbol{P} = \sum_{j=0}^{n-1} \alpha_j \boldsymbol{P}^{-1} \begin{bmatrix} \boldsymbol{J}^{Q+j} & \boldsymbol{0} \\ \boldsymbol{0} & \boldsymbol{0} \end{bmatrix} \boldsymbol{P} = \sum_{j=0}^{n-1} \alpha_j \boldsymbol{A}^{Q+j}. \tag{4.47}$$

Therefore, for any $n \leq P \leq Q$, and set $\mathcal{T} \subseteq \{1, 2, \ldots, m\}$, we arrive at

$$\text{ColSpace}\left\{\boldsymbol{A}^P \boldsymbol{B}_{\mathcal{T}}\right\} = \text{ColSpace}\left\{\sum_{j=0}^{n-1} \alpha_j \boldsymbol{A}^{Q+j} \boldsymbol{B}_{\mathcal{T}}\right\} \tag{4.48}$$

$$\subseteq \text{ColSpace}\left\{\boldsymbol{A}^Q \begin{bmatrix} \boldsymbol{A}^{n-1}\boldsymbol{B}_{\mathcal{T}} & \boldsymbol{A}^{n-2}\boldsymbol{B}_{\mathcal{T}} & \ldots & \boldsymbol{B}_{\mathcal{T}} \end{bmatrix}\right\}. \tag{4.49}$$

Hence, for any $\{\mathcal{T}_l\}_{l=1}^m$ such that $|\mathcal{T}_l| = s$ and $\cup_{l=1}^m \mathcal{T}_l = \{1, 2, \ldots, m\}$, we deduce

$$\text{ColSpace}\left\{\boldsymbol{A}^P \boldsymbol{B}\right\} = \text{ColSpace}\left\{\begin{bmatrix} \boldsymbol{A}^P \boldsymbol{B}_{\mathcal{T}_1} & \boldsymbol{A}^P \boldsymbol{B}_{\mathcal{T}_2} & \ldots & \boldsymbol{A}^P \boldsymbol{B}_{\mathcal{T}_m} \end{bmatrix}\right\}. \tag{4.50}$$

Here, substituting $\mathcal{T} = \mathcal{T}_l$ and $Q = (P-n)nm + n + n(m-l)$ in (4.49), we get

$$\begin{aligned} &\text{ColSpace}\left\{\boldsymbol{A}^P \boldsymbol{B}_{\mathcal{T}_l}\right\} \\ &\quad \subseteq \text{ColSpace}\Big\{\boldsymbol{A}^{(P-n)nm+n} \\ &\quad \times \begin{bmatrix} \boldsymbol{A}^{n(m-l)+n-1}\boldsymbol{B}_{\mathcal{T}_l} & \boldsymbol{A}^{n(m-l)+n-2}\boldsymbol{B}_{\mathcal{T}_l} & \ldots & \boldsymbol{A}^{n(m-l)}\boldsymbol{B}_{\mathcal{T}_l} \end{bmatrix}\Big\}. \end{aligned} \tag{4.51}$$

Combining the above relation with (4.50) gives

$$\begin{aligned} &\text{ColSpace}\left\{\boldsymbol{A}^P \boldsymbol{B}\right\} \\ &\subseteq \text{ColSpace}\Big\{\boldsymbol{A}^{(P-n)nm+n}\big[\boldsymbol{A}^{nm-1}\boldsymbol{B}_{\mathcal{T}_1} \quad \boldsymbol{A}^{nm-2}\boldsymbol{B}_{\mathcal{T}_1} \quad \ldots \quad \boldsymbol{A}^{n(m-1)}\boldsymbol{B}_{\mathcal{T}_1} \\ &\quad \boldsymbol{A}^{n(m-1)-1}\boldsymbol{B}_{\mathcal{T}_2} \quad \boldsymbol{A}^{n(m-1)-2}\boldsymbol{B}_{\mathcal{T}_2} \quad \ldots \quad \boldsymbol{A}^{n(m-2)}\boldsymbol{B}_{\mathcal{T}_2} \ldots \\ &\quad \boldsymbol{A}^{n-1}\boldsymbol{B}_{\mathcal{T}_m} \quad \boldsymbol{A}^{n-2}\boldsymbol{B}_{\mathcal{T}_m} \quad \ldots \quad \boldsymbol{B}_{\mathcal{T}_m}\big]\Big\}. \end{aligned} \tag{4.52}$$

Further, substituting for $P = n, n+1, \ldots, 2n-1$ in the above relation, we obtain that for any sparsity level $s \leq m$, with $K = n^2m$ and $\mathcal{S}_k = \mathcal{T}_{m-(\lfloor k/n \rfloor \mod m)}$,

$$\begin{aligned} &\text{ColSpace}\left\{\left[\boldsymbol{A}^{2n-1}\boldsymbol{B} \quad \boldsymbol{A}^{2n-2}\boldsymbol{B} \ \ldots \ \boldsymbol{A}^{n}\boldsymbol{B}\right]\right\} \\ &\qquad \subseteq \text{ColSpace}\left\{\boldsymbol{A}^{n}\left[\boldsymbol{A}^{K-1}\boldsymbol{B}_{\mathcal{S}_1} \quad \boldsymbol{A}^{K-2}\boldsymbol{B}_{\mathcal{S}_2} \ \ldots \ \boldsymbol{B}_{\mathcal{S}_K}\right]\right\}. \end{aligned} \tag{4.53}$$

Finally, the above relation implies that

$$\begin{aligned} &\text{ColSpace}\left\{\boldsymbol{C}\boldsymbol{A}^{n}\left[\boldsymbol{A}^{K-1}\boldsymbol{B}_{\mathcal{S}_1} \quad \boldsymbol{A}^{K-2}\boldsymbol{B}_{\mathcal{S}_2} \ \ldots \ \boldsymbol{B}_{\mathcal{S}_K}\right]\right\} \\ &\qquad \supseteq \text{ColSpace}\left\{\boldsymbol{C}\left[\boldsymbol{A}^{2n-1}\boldsymbol{B} \quad \boldsymbol{A}^{2n-2}\boldsymbol{B} \ \ldots \ \boldsymbol{A}^{n}\boldsymbol{B}\right]\right\} && (4.54) \\ &\qquad = \text{ColSpace}\left\{\boldsymbol{C}\boldsymbol{A}^{n}\left[\boldsymbol{A}^{n-1}\boldsymbol{B} \quad \boldsymbol{A}^{n-2}\boldsymbol{B} \ \ldots \ \boldsymbol{B}\right]\right\} && (4.55) \\ &\qquad = \text{ColSpace}\left\{\boldsymbol{C}\boldsymbol{A}^{n}\right\}, && (4.56) \end{aligned}$$

where we use the fact that the system is controllable and

$$\text{ColSpace}\left\{\left[\boldsymbol{A}^{n-1}\boldsymbol{B} \quad \boldsymbol{A}^{n-2}\boldsymbol{B} \quad \ldots \quad \boldsymbol{B}\right]\right\} = \mathbb{R}^n. \tag{4.57}$$

Hence, the proof is completed. □

We are now ready to prove the sufficient conditions for sparse output controllability.

Proof of Proposition 4.2. Suppose that the sufficient conditions in Proposition 4.2 hold. Then, for any $\boldsymbol{z} \in \mathbb{R}^p$ and $i = 0, 1, \ldots, n-1$, we have

$$\boldsymbol{C}^{(i)}\boldsymbol{z} = \left[\boldsymbol{C}^{(i)} - \boldsymbol{C}^{(i+1)}\right]\boldsymbol{z} + \boldsymbol{C}^{(i+1)}\boldsymbol{z}, \tag{4.58}$$

where $\boldsymbol{C}^{(i)} = \boldsymbol{C}\boldsymbol{A}^{i}\left(\boldsymbol{C}\boldsymbol{A}^{i}\right)^{\dagger}$. Then, using Theorem 4.6, there exists a vector $\boldsymbol{u} \in \mathbb{R}^m$ such that $\|\boldsymbol{u}\|_0 = \text{rank}\left\{\boldsymbol{C}^{(i)}\right\} - \text{rank}\left\{\boldsymbol{C}^{(i+1)}\right\} \leq s$ such that

$$\boldsymbol{C}^{(i)}\boldsymbol{z} = \left[\boldsymbol{I} - \boldsymbol{C}^{(i+1)}\right]\boldsymbol{C}\boldsymbol{B}\boldsymbol{u} + \boldsymbol{C}^{(i+1)}\boldsymbol{z} = \boldsymbol{C}\boldsymbol{B}\boldsymbol{u} + \boldsymbol{C}^{(i+1)}\left[\boldsymbol{z} - \boldsymbol{C}\boldsymbol{B}\boldsymbol{u}\right]. \tag{4.59}$$

Therefore, recursively using (4.59) with $\boldsymbol{z} = \boldsymbol{y} - \sum_{k=1}^{l}\boldsymbol{C}\boldsymbol{A}^{k-1}\boldsymbol{B}\boldsymbol{u}_k$, we conclude that for any $\boldsymbol{y} \in \mathbb{R}^p$,

$$\begin{aligned} \boldsymbol{C}^{(0)}\boldsymbol{y} &= \boldsymbol{C}\boldsymbol{B}\boldsymbol{u}_1 + \boldsymbol{C}^{(1)}\left[\boldsymbol{y} - \boldsymbol{C}\boldsymbol{B}\boldsymbol{u}_1\right] && (4.60) \\ &= \boldsymbol{C}\boldsymbol{B}\boldsymbol{u}_1 + \boldsymbol{C}\boldsymbol{A}\boldsymbol{B}\boldsymbol{u}_2 + \boldsymbol{C}^{(2)}\left[\boldsymbol{y} - \boldsymbol{C}\boldsymbol{B}\boldsymbol{u}_1 - \boldsymbol{C}\boldsymbol{A}\boldsymbol{B}\boldsymbol{u}_2\right] && (4.61) \\ &= \sum_{k=1}^{n}\boldsymbol{C}\boldsymbol{A}^{k-1}\boldsymbol{B}\boldsymbol{u}_k + \boldsymbol{C}^{(n)}\left[\boldsymbol{y} - \sum_{k=1}^{n}\boldsymbol{C}\boldsymbol{A}^{k-1}\boldsymbol{B}\boldsymbol{u}_k\right], && (4.62) \end{aligned}$$

for some $\boldsymbol{u}_k$ such that $\|\boldsymbol{u}_k\|_0 \leq s$, for $k = 1, 2, \ldots, n$. Here, we note that

$$p \geq \operatorname{rank}\{\boldsymbol{C}\} \geq \operatorname{rank}\{\boldsymbol{C\Phi}\} = p. \tag{4.63}$$

So, we have $\boldsymbol{C}^{(0)} = \boldsymbol{C}\boldsymbol{C}^{\dagger} = \boldsymbol{I}$, and hence, for any $\boldsymbol{y} \in \mathbb{R}^p$, there exist s-sparse vectors $\{\boldsymbol{u}_k\}_{k=1}^n$ such that

$$\boldsymbol{y} = \sum_{k=1}^{n} \boldsymbol{C}\boldsymbol{A}^{k-1}\boldsymbol{B}\boldsymbol{u}_k + \boldsymbol{C}^{(n)}\left[\boldsymbol{y} - \sum_{k=1}^{n} \boldsymbol{C}\boldsymbol{A}^{k-1}\boldsymbol{B}\boldsymbol{u}_k\right]. \tag{4.64}$$

Here, the last term on the right-hand side belongs to ColSpace $\{\boldsymbol{C}\boldsymbol{A}^n\}$. Consequently, using Theorem 4.7, we derive that for any $\boldsymbol{y} \in \mathbb{R}^p$, there exist an integer $K < \infty$ and s-sparse vectors $\{\boldsymbol{u}_k\}_{k=1}^K$ such that

$$\boldsymbol{y} = \sum_{k=1}^{K} \boldsymbol{C}\boldsymbol{A}^{k-1}\boldsymbol{B}\boldsymbol{u}_k. \tag{4.65}$$

Hence, from Theorem 4.4, we conclude that the linear dynamical system $(\boldsymbol{A}, \boldsymbol{B}, \boldsymbol{C})$ is s-sparse output controllable. □

As promised earlier, we now relax the assumption that the system is controllable to present the most general result for sparse output controllability.

Theorem 4.8 (Joseph, 2022b). The linear dynamical system $(\boldsymbol{A} \in \mathbb{R}^{n\times n}, \boldsymbol{B} \in \mathbb{R}^{n\times m}, \boldsymbol{C} \in \mathbb{R}^{p\times n})$ in (4.17) is s-sparse output controllable only if

$$\operatorname{rank}\{\boldsymbol{C\Phi}\} = p \text{ and } s \geq \max_{1\leq i\leq n} \frac{1}{i}\left(p - \operatorname{rank}\left\{\boldsymbol{C}\boldsymbol{A}^i\boldsymbol{\Phi}\right\}\right), \tag{4.66}$$

where $\boldsymbol{\Phi}$ is the controllability matrix defined in (4.20). The system is s-sparse output controllable if

$$\operatorname{rank}\{\boldsymbol{C\Phi}\} = p \text{ and } s \geq \max_{0\leq i\leq n-1} \operatorname{rank}\left\{\boldsymbol{C}\boldsymbol{A}^i\boldsymbol{\Phi}\right\} - \operatorname{rank}\left\{\boldsymbol{C}\boldsymbol{A}^{i+1}\boldsymbol{\Phi}\right\} \tag{4.67}$$

.

Proof. Let $\boldsymbol{Q} \in \mathbb{R}^{n\times R}$ be a matrix whose columns are orthonormal $(\boldsymbol{Q}\boldsymbol{Q}^{\mathsf{T}} = \boldsymbol{I})$ and ColSpace $\{\boldsymbol{Q}\}$ = ColSpace $\{\boldsymbol{\Phi}\}$ with $R = \operatorname{rank}\{\boldsymbol{\Phi}\}$. Then, from (4.17), we obtain

$$\boldsymbol{Q}^{\mathsf{T}}\boldsymbol{x}_k = \left(\boldsymbol{Q}^{\mathsf{T}}\boldsymbol{A}\boldsymbol{Q}\right)\boldsymbol{Q}^{\mathsf{T}}\boldsymbol{x}_{k-1} + \left(\boldsymbol{Q}^{\mathsf{T}}\boldsymbol{B}\right)\boldsymbol{u}_k \tag{4.68}$$

$$\boldsymbol{y}_k = (\boldsymbol{C}\boldsymbol{Q})\,\boldsymbol{Q}^{\mathsf{T}}\boldsymbol{x}_k. \tag{4.69}$$

The modified system thus obtained is $(\boldsymbol{Q}^\top \boldsymbol{A}\boldsymbol{Q} \in \mathbb{R}^{R\times R}, \boldsymbol{Q}^\top \boldsymbol{B} \in \mathbb{R}^{R\times m}, \boldsymbol{C}\boldsymbol{Q} \in \mathbb{R}^{p\times R})$. From Theorem 4.4, the modified system is s-sparse output controllable if and only if there exist an integer $K < \infty$ and sets $\{\mathcal{S}_k \subseteq \{1,2,\ldots,m\},\ |\mathcal{S}_k| \leq s\}_{k=1}^K$ such that

$$\operatorname{rank}\left\{\boldsymbol{C}\left[\boldsymbol{A}^{K-1}\boldsymbol{B}_{\mathcal{S}_1} \quad \boldsymbol{A}^{K-2}\boldsymbol{B}_{\mathcal{S}_2} \quad \ldots \quad \boldsymbol{B}_{\mathcal{S}_K}\right]\right\} = p. \tag{4.70}$$

So, we see that the modified system is s-sparse output controllable if and only if the system $(\boldsymbol{A},\boldsymbol{B},\boldsymbol{C})$ is also s-sparse output controllable.

Applying Proposition 4.1 to the modified system, we prove that the system $(\boldsymbol{A},\boldsymbol{B},\boldsymbol{C})$ is s-sparse output controllable only if $\operatorname{rank}\{\boldsymbol{C}\boldsymbol{\Phi}\} = p$ and

$$s \geq \max_{1\leq i\leq n} \frac{1}{i}\left(p - \operatorname{rank}\left\{\boldsymbol{C}\boldsymbol{A}^i\boldsymbol{Q}\right\}\right) = \max_{1\leq i\leq n} \frac{1}{i}\left(p - \operatorname{rank}\left\{\boldsymbol{C}\boldsymbol{A}^i\boldsymbol{\Phi}\right\}\right). \tag{4.71}$$

Therefore, the necessary conditions are proved.

To prove the sufficient conditions, assume that (4.67) holds. Also, the controllability matrix of the modified system satisfies

$$\operatorname{rank}\left\{\left[\boldsymbol{Q}^\top\boldsymbol{A}^{n-1}\boldsymbol{B} \quad \boldsymbol{Q}^\top\boldsymbol{A}^{n-2}\boldsymbol{B} \quad \boldsymbol{Q}^\top\boldsymbol{B}\right]\right\} = \operatorname{rank}\left\{\boldsymbol{Q}^\top\boldsymbol{\Phi}\right\} = R. \tag{4.72}$$

Hence, the modified system is controllable, and it also satisfies the rank condition,

$$\operatorname{rank}\left\{(\boldsymbol{C}\boldsymbol{Q})\left(\boldsymbol{Q}^\top\boldsymbol{\Phi}\right)\right\} = \operatorname{rank}\{\boldsymbol{C}\boldsymbol{\Phi}\} = p, \tag{4.73}$$

and the sparsity level-based condition,

$$\begin{aligned}\operatorname{rank}&\left\{(\boldsymbol{C}\boldsymbol{Q})\left(\boldsymbol{Q}^\top\boldsymbol{A}\boldsymbol{Q}\right)^i\right\} - \operatorname{rank}\left\{(\boldsymbol{C}\boldsymbol{Q})\left(\boldsymbol{Q}^\top\boldsymbol{A}\boldsymbol{Q}\right)^{i+1}\right\}\\ &= \operatorname{rank}\left\{(\boldsymbol{C}\boldsymbol{A})^i\,\boldsymbol{Q}\right\} - \operatorname{rank}\left\{(\boldsymbol{C}\boldsymbol{A})^{i+1}\,\boldsymbol{Q}\right\} &(4.74)\\ &= \operatorname{rank}\left\{(\boldsymbol{C}\boldsymbol{A})^i\,\boldsymbol{\Phi}\right\} - \operatorname{rank}\left\{(\boldsymbol{C}\boldsymbol{A})^{i+1}\,\boldsymbol{\Phi}\right\} \leq s. &(4.75)\end{aligned}$$

Then, by Proposition 4.2, the modified system is s-sparse output controllable, and consequently, the system $(\boldsymbol{A},\boldsymbol{B},\boldsymbol{C})$ is also s-sparse output controllable. □

Some observations from Theorem 4.8 are as follows. In Theorem 4.8, when $\boldsymbol{C} = \boldsymbol{I}$, the condition $\operatorname{rank}\{\boldsymbol{C}\boldsymbol{\Phi}\} = p = n$ reduces to $\operatorname{rank}\{\boldsymbol{\Phi}\} = n$. Also, the sufficient bound on the sparsity level becomes

$$\max_{0\leq i\leq n-1} \operatorname{rank}\left\{\boldsymbol{C}\boldsymbol{A}^{i}\boldsymbol{\Phi}\right\} - \operatorname{rank}\left\{\boldsymbol{C}\boldsymbol{A}^{i+1}\boldsymbol{\Phi}\right\}$$

$$= \max_{0\leq i\leq n-1} \operatorname{rank}\left\{\boldsymbol{A}^{i}\boldsymbol{\Phi}\right\} - \operatorname{rank}\left\{\boldsymbol{A}^{i+1}\boldsymbol{\Phi}\right\} \tag{4.76}$$

$$= \max_{0\leq i\leq n-1} \operatorname{rank}\left\{\boldsymbol{A}^{i}\right\} - \operatorname{rank}\left\{\boldsymbol{A}^{i+1}\right\} \leq n - \operatorname{rank}\left\{\boldsymbol{A}\right\} \tag{4.77}$$

$$= \operatorname{rank}\left\{\boldsymbol{C}\boldsymbol{\Phi}\right\} - \operatorname{rank}\left\{\boldsymbol{C}\boldsymbol{A}\boldsymbol{\Phi}\right\}, \tag{4.78}$$

where we use the Sylvester rank inequality in (4.77). Therefore, we conclude that

$$\max_{0\leq i\leq n-1} \operatorname{rank}\left\{\boldsymbol{C}\boldsymbol{A}^{i}\boldsymbol{\Phi}\right\} - \operatorname{rank}\left\{\boldsymbol{C}\boldsymbol{A}^{i+1}\boldsymbol{\Phi}\right\} = n - \operatorname{rank}\left\{\boldsymbol{A}\right\}. \tag{4.79}$$

So, the bound on the sparsity level is equivalent to $s \geq n - \operatorname{rank}\{\boldsymbol{A}\}$. Similarly, the necessary bound on the sparsity level is

$$\max_{1\leq i\leq n} \frac{1}{i}\left(p - \operatorname{rank}\left\{\boldsymbol{C}\boldsymbol{A}^{i}\boldsymbol{\Phi}\right\}\right) = \max_{1\leq i\leq n} \frac{1}{i}\left(\operatorname{rank}\left\{\boldsymbol{A}^{0}\right\} - \operatorname{rank}\left\{\boldsymbol{A}^{i}\right\}\right) \tag{4.80}$$

$$= \max_{1\leq i\leq n} \frac{1}{i}\sum_{j=0}^{i-1} \operatorname{rank}\left\{\boldsymbol{A}^{j}\right\} - \operatorname{rank}\left\{\boldsymbol{A}^{j+1}\right\} \tag{4.81}$$

$$\leq \max_{1\leq i\leq n} \operatorname{rank}\left\{\boldsymbol{A}^{i-1}\right\} - \operatorname{rank}\left\{\boldsymbol{A}^{i+1}\right\} \tag{4.82}$$

$$\leq n - \operatorname{rank}\left\{\boldsymbol{A}\right\} \tag{4.83}$$

$$= n - \operatorname{rank}\left\{\boldsymbol{C}\boldsymbol{A}\boldsymbol{\Phi}\right\}. \tag{4.84}$$

Hence, the necessary bound in Theorem 4.8 reduces to $s \geq n - \operatorname{rank}\{\boldsymbol{A}\}$. Consequently, Theorem 4.8 reduces to Theorem 2.8 when $\boldsymbol{C} = \boldsymbol{I}$, implying Theorem 4.8 is a tight result.

Finally, the s-sparse control inputs that are required to drive the system to a desired output starting from a given initial state can be cast as a sparse signal recovery problem, similar to the ones discussed in Section 3. The problem can be solved using (Algorithms 1 to 7) described in Section 3.

4.3 Sparse Nonnegative Controllability

The linear dynamical system $(\boldsymbol{A}, \boldsymbol{B})$ in (4.1) is said to be (non-sparse) nonnegative controllable if, for any initial state $\boldsymbol{x}_1$ and any final state $\boldsymbol{x}_{\mathrm{f}}$, there exist a time index $K < \infty$ and inputs $\{\boldsymbol{u}_k \in \mathbb{R}_+^m,\ k = 1, 2, \ldots, K-1\}$ that steer the system from the state $\boldsymbol{x}_1$ to $\boldsymbol{x}_{K+1} = \boldsymbol{x}_{\mathrm{f}}$. The classical test for nonnegative controllability is as follows:

Theorem 4.9 (Yoshida *et al.*, 1994)**.** The linear dynamical system ($\boldsymbol{A} \in \mathbb{R}^{n\times n}, \boldsymbol{B} \in \mathbb{R}^{n\times m}$) in (4.1) is nonnegative controllable if and only if the following conditions hold:

(a) $\operatorname{rank}\left\{\begin{bmatrix}\boldsymbol{A} - \lambda \boldsymbol{I} & \boldsymbol{B}\end{bmatrix}\right\} = n$, for all $\lambda \in \mathbb{C}$

(b) For every left eigenvalue $\lambda \geq 0$ of $\boldsymbol{A}$, the corresponding eigenvector $\boldsymbol{z}$ is such that $\boldsymbol{z}^\top \boldsymbol{B}$ has at least one strictly negative entry, i.e., there exists no tuples ($\lambda \geq 0, \boldsymbol{z} \neq \boldsymbol{0}$) such $\boldsymbol{z}^\top \boldsymbol{A} = \lambda \boldsymbol{z}^\top$ and $\boldsymbol{z}^\top \boldsymbol{B} \in \mathbb{R}_+^m$.

We extend the notion of nonnegative controllability to the case of sparse inputs and derive a nonnegative controllability test as follows.

Definition 4.3. The linear dynamical system $(\boldsymbol{A}, \boldsymbol{B})$ in (4.1) is said to be s-sparse nonnegative controllable for a given $s \geq 1$ if, for any initial state $\boldsymbol{x}_1$ and final state $\boldsymbol{x}_{\mathrm{f}}$, there exist an integer $K < \infty$ and s-sparse inputs $\{\boldsymbol{u}_k \in \mathbb{R}_+^m : \quad \|\boldsymbol{u}_k\|_0 \leq s\}_{k=1}^K$ that steer the system from $\boldsymbol{x}_1$ to $\boldsymbol{x}_{K+1} = \boldsymbol{x}_{\mathrm{f}}$.

We note that sparse nonnegative controllability is a stronger notion than sparse controllability and nonnegative controllability. So, the conditions for s-sparse controllability in Theorem 2.8 and the conditions for nonnegative controllability in Theorem 4.9 are necessary for sparse nonnegative controllability. The next main result of this section shows that the two sets of conditions are also sufficient for sparse nonnegative controllability.

Before presenting the main result, we look at a special case where $\boldsymbol{A}$ is invertible.

Proposition 4.3. For any sparsity level, $s \geq 1$, the linear dynamical system ($\boldsymbol{A} \in \mathbb{R}^{n\times n}, \boldsymbol{B} \in \mathbb{R}^{n\times m}$) in (4.1), with nonsingular $\boldsymbol{A}$, is s-sparse nonnegative controllable if it is nonnegative controllable.

Proof. For a given sparsity level, suppose that the system is not s-sparse nonnegative controllable. Then, for any set $\mathcal{S} \subseteq \{1, 2, \ldots, m\}$ and

$|\mathcal{S}| \leq s$, we can construct a set $\{\mathcal{T}_k \subseteq \{1, 2, \ldots, m\} : |\mathcal{T}_k| = s\}_{k=1}^{m-1}$ such that

$$\bigcup_{k=1}^{m-1} \mathcal{T}_k \bigcup \mathcal{S} = \{1, 2 \ldots, m\}. \tag{4.85}$$

Then, for any $K < \infty$, we have

$$\begin{aligned} \mathrm{Span}_+ \Big\{ \big[& \boldsymbol{A}^{Km-1}\boldsymbol{B}_{\mathcal{T}_1} \quad \boldsymbol{A}^{Km-2}\boldsymbol{B}_{\mathcal{T}_1} \quad \ldots \quad \boldsymbol{A}^{K(m-1)}\boldsymbol{B}_{\mathcal{T}_1} \\ & \boldsymbol{A}^{K(m-1)-1}\boldsymbol{B}_{\mathcal{T}_2} \quad \boldsymbol{A}^{K(m-1)-2}\boldsymbol{B}_{\mathcal{T}_2} \quad \ldots \quad \boldsymbol{A}^{K(m-2)}\boldsymbol{B}_{\mathcal{T}_2} \\ & \ldots \quad \boldsymbol{A}^{2K-1}\boldsymbol{B}_{\mathcal{T}_{m-1}} \quad \boldsymbol{A}^{2K-2}\boldsymbol{B}_{\mathcal{T}_{m-1}} \quad \ldots \quad \boldsymbol{A}^{K}\boldsymbol{B}_{\mathcal{T}_{m-1}} \\ & \boldsymbol{A}^{K-1}\boldsymbol{B}_{\mathcal{S}} \quad \boldsymbol{A}^{K-2}\boldsymbol{B}_{\mathcal{S}} \quad \ldots \quad \boldsymbol{B}_{\mathcal{S}} \big] \Big\} \subset \mathbb{R}^n, \end{aligned} \tag{4.86}$$

where Span_+ is the positive span of the columns of the matrix, i.e.,

$$\mathrm{Span}_+ \{\boldsymbol{M}\} = \left\{ \sum_{i=1}^{M} \alpha_i \boldsymbol{M}_i : \ \ \alpha_i \geq 0 \right\}, \tag{4.87}$$

where M is the number of columns of $\boldsymbol{M}$. Rearranging the columns of the matrix, from (4.86), we derive

$$\begin{aligned} \mathrm{Span}_+ \Big\{ \big[& \boldsymbol{A}^{Km-1}\boldsymbol{B}_{\mathcal{T}_1} \quad \boldsymbol{A}^{K(m-1)-1}\boldsymbol{B}_{\mathcal{T}_2} \quad \ldots \quad \boldsymbol{A}^{2K-1}\boldsymbol{B}_{\mathcal{T}_{m-1}} \quad \boldsymbol{A}^{K-1}\boldsymbol{B}_{\mathcal{S}} \\ & \boldsymbol{A}^{Km-2}\boldsymbol{B}_{\mathcal{T}_1} \quad \boldsymbol{A}^{K(m-1)-2}\boldsymbol{B}_{\mathcal{T}_2} \quad \ldots \quad \boldsymbol{A}^{2K-2}\boldsymbol{B}_{\mathcal{T}_{m-1}} \quad \boldsymbol{A}^{K-2}\boldsymbol{B}_{\mathcal{S}} \\ & \ldots \quad \boldsymbol{A}^{K(m-1)}\boldsymbol{B}_{\mathcal{T}_1} \quad \boldsymbol{A}^{K(m-2)}\boldsymbol{B}_{\mathcal{T}_2} \quad \ldots \quad \boldsymbol{A}^{K}\boldsymbol{B}_{\mathcal{T}_{m-1}} \quad \boldsymbol{B}_{\mathcal{S}} \big] \Big\} \subset \mathbb{R}^n. \end{aligned} \tag{4.88}$$

We can further rewrite the above equation as

$$\mathrm{Span}_+ \left\{ \left[\boldsymbol{A}^{K-1}\hat{\boldsymbol{B}} \quad \boldsymbol{A}^{K-2}\hat{\boldsymbol{B}} \quad \ldots \quad \hat{\boldsymbol{B}} \right] \right\} \subset \mathbb{R}^n, \tag{4.89}$$

where we define

$$\hat{\boldsymbol{B}} = \left[\boldsymbol{A}^{K(m-1)}\boldsymbol{B}_{\mathcal{T}_1} \quad \boldsymbol{A}^{K(m-2)}\boldsymbol{B}_{\mathcal{T}_2} \quad \ldots \quad \boldsymbol{A}^{K}\boldsymbol{B}_{\mathcal{T}_{m-1}} \quad \boldsymbol{B}_{\mathcal{S}} \right]. \tag{4.90}$$

Since (4.89) holds for any $K < \infty$, we conclude that the system $(\boldsymbol{A}, \hat{\boldsymbol{B}})$ is not negative controllable. By Theorem 4.9, at least one of the following statements is violated:

(a) There exists no tuple $(\lambda \in \mathbb{C}, \boldsymbol{z} \neq \boldsymbol{0})$ such that $\boldsymbol{z}^\top \boldsymbol{A} = \lambda \boldsymbol{z}^\top$ and $\boldsymbol{z}^\top \hat{\boldsymbol{B}} = \boldsymbol{0}$.

(b) There exists no tuple $(\lambda \geq 0, \boldsymbol{z} \neq \boldsymbol{0})$ such that $\boldsymbol{z}^\top \boldsymbol{A} = \lambda \boldsymbol{z}^\top$ and $\boldsymbol{z}^\top \hat{\boldsymbol{B}} \in \mathbb{R}_+^m$.

Nonetheless, when $\boldsymbol{z}^\top \boldsymbol{A} = \lambda \boldsymbol{z}^\top$, we get

$$\boldsymbol{z}^\top \hat{\boldsymbol{B}} = \begin{bmatrix} \lambda^{K(m-1)} \boldsymbol{z}^\top \boldsymbol{B}_{\mathcal{T}_1} & \lambda^{K(m-2)} \boldsymbol{z}^\top \boldsymbol{B}_{\mathcal{T}_2} & \dots & \lambda^{K} \boldsymbol{z}^\top \boldsymbol{B}_{\mathcal{T}_{m-1}} & \boldsymbol{z}^\top \boldsymbol{B}_{\mathcal{S}} \end{bmatrix} \tag{4.91}$$

$$= \boldsymbol{z}^\top \begin{bmatrix} \boldsymbol{B}_{\mathcal{T}_1} & \boldsymbol{B}_{\mathcal{T}_2} & \dots \boldsymbol{B}_{\mathcal{T}_{m-1}} & \boldsymbol{B}_{\mathcal{S}} \end{bmatrix} \begin{bmatrix} \lambda^{K(m-1)} \boldsymbol{I} & & & \\ & \lambda^{K(m-2)} \boldsymbol{I} & & \\ & & \ddots & \\ & & & \boldsymbol{I} \end{bmatrix}. \tag{4.92}$$

Also, since $\boldsymbol{A}$ is nonsingular, $\lambda \neq 0$, and as a consequence, using (4.85), at least one of the following statements are violated:

(a) There exists no tuple $(\lambda \in \mathbb{C}, \boldsymbol{z} \neq \boldsymbol{0})$ such that $\boldsymbol{z}^\top \boldsymbol{A} = \lambda \boldsymbol{z}^\top$ and $\boldsymbol{z}^\top \boldsymbol{B} = \boldsymbol{0}$.

(b) There exists no tuple $(\lambda \geq 0, \boldsymbol{z} \neq \boldsymbol{0})$ such that $\boldsymbol{z}^\top \boldsymbol{A} = \lambda \boldsymbol{z}^\top$ and $\boldsymbol{z}^\top \boldsymbol{B} \in \mathbb{R}_+^m$.

Therefore, by Theorem 4.9, the system $(\boldsymbol{A}, \boldsymbol{B})$ is not nonnegative controllable.

In short, we showed that if the system is not s-sparse nonnegative controllable, then it is not nonnegative controllable. Hence, the proof. □

Building on the above intermediate result, we present the next main result of this section.

Theorem 4.10 (Joseph, 2022a)**.** For any sparsity level, $s \geq 1$, the linear dynamical system $(\boldsymbol{A} \in \mathbb{R}^{n \times n}, \boldsymbol{B} \in \mathbb{R}^{n \times m})$ in (4.1) is s-sparse nonnegative controllable if and only if it is nonnegative controllable and s-sparse controllable.

We prove the main result for the general system using the following two lemmas.

Lemma 4.11. Suppose that the linear dynamical system ($\boldsymbol{A} \in \mathbb{R}^{n\times n}, \boldsymbol{B} \in \mathbb{R}^{n\times m}$) in (4.1) is nonnegative controllable and $s \geq N - \operatorname{rank}\{\boldsymbol{A}\}$. Then, for any $\boldsymbol{z} \in \mathbb{R}^n$ and integer $i = 0, 1, \ldots, n-1$, there exists an s-sparse vector $\boldsymbol{u} \in \mathbb{R}_+^m$ such that

$$\left[\boldsymbol{A}^i \left(\boldsymbol{A}^i\right)^\dagger - \boldsymbol{A}^{i+1}\left(\boldsymbol{A}^{i+1}\right)^\dagger\right]\boldsymbol{z} = \left[\boldsymbol{I} - \boldsymbol{A}^{i+1}\left(\boldsymbol{A}^{i+1}\right)^\dagger\right]\boldsymbol{A}^i\boldsymbol{B}\boldsymbol{u}. \tag{4.93}$$

Proof. Since the system is nonnegative controllable, for any $\boldsymbol{z} \in \mathbb{R}^n$, there exist an integer $K < \infty$ and (non-sparse) inputs $\{\boldsymbol{u}_k\}_{k=1}^K \in \mathbb{R}_+^m$ such that

$$\boldsymbol{z} = \sum_{k=1}^{K} \boldsymbol{A}^{K-k}\boldsymbol{B}\boldsymbol{u}_k. \tag{4.94}$$

Multiplying both sides by $\left[\boldsymbol{I} - \boldsymbol{A}^{i+1}\left(\boldsymbol{A}^{i+1}\right)^\dagger\right]\boldsymbol{A}^i$ yields

$$\left[\boldsymbol{I} - \boldsymbol{A}^{i+1}\left(\boldsymbol{A}^{i+1}\right)^\dagger\right]\boldsymbol{A}^i\boldsymbol{z} = \sum_{k=1}^{K}\left[\boldsymbol{I} - \boldsymbol{A}^{i+1}\left(\boldsymbol{A}^{i+1}\right)^\dagger\right]\boldsymbol{A}^i\boldsymbol{A}^{K-k}\boldsymbol{B}\boldsymbol{u}_k \tag{4.95}$$

$$= \left[\boldsymbol{I} - \boldsymbol{A}^{i+1}\left(\boldsymbol{A}^{i+1}\right)^\dagger\right]\boldsymbol{A}^i\boldsymbol{B}\boldsymbol{u}_K, \tag{4.96}$$

using the fact that for $K - k \geq 1$,

$$\left[\boldsymbol{I} - \boldsymbol{A}^{i+1}\left(\boldsymbol{A}^{i+1}\right)^\dagger\right]\boldsymbol{A}^{K-k+i} = \left[\boldsymbol{A}^{i+1} - \boldsymbol{A}^{i+1}\left(\boldsymbol{A}^{i+1}\right)^\dagger\boldsymbol{A}^{i+1}\right]\boldsymbol{A}^{K-k-1} = \boldsymbol{0}. \tag{4.97}$$

Also, from Theorem 4.5 with $\boldsymbol{C} = \boldsymbol{I}$, we know that

$$\operatorname{ColSpace}\left\{\left[\boldsymbol{I} - \boldsymbol{A}^{i+1}\left(\boldsymbol{A}^{i+1}\right)^\dagger\right]\boldsymbol{A}^i\right\} = \operatorname{ColSpace}\left\{\boldsymbol{A}^i\left(\boldsymbol{A}^i\right)^\dagger - \boldsymbol{A}^{i+1}\left(\boldsymbol{A}^{i+1}\right)^\dagger\right\}. \tag{4.98}$$

As a result, from (4.96), for every $\boldsymbol{z} \in \mathbb{R}^n$, there exists a vector $\boldsymbol{v} \in \mathbb{R}_+^m$ such that

$$\left[\boldsymbol{A}^i \left(\boldsymbol{A}^i\right)^\dagger - \boldsymbol{A}^{i+1}\left(\boldsymbol{A}^{i+1}\right)^\dagger\right]\boldsymbol{z} = \left[\boldsymbol{I} - \boldsymbol{A}^{i+1}\left(\boldsymbol{A}^{i+1}\right)^\dagger\right]\boldsymbol{A}^i\boldsymbol{B}\boldsymbol{v}. \tag{4.99}$$

Using this formulation, we define a non-empty set

$$\mathcal{V}(\boldsymbol{z}) = \left\{\boldsymbol{v} \in \mathbb{R}_+^n : \left[\boldsymbol{A}^i \left(\boldsymbol{A}^i\right)^\dagger - \boldsymbol{A}^{i+1}\left(\boldsymbol{A}^{i+1}\right)^\dagger\right]\boldsymbol{z} = \left[\boldsymbol{I} - \boldsymbol{A}^{i+1}\left(\boldsymbol{A}^{i+1}\right)^\dagger\right]\boldsymbol{A}^i\boldsymbol{B}\boldsymbol{v}\right\} \tag{4.100}$$

$$= \left\{\boldsymbol{v} \in \mathbb{R}_+^n : \bar{\boldsymbol{z}} = \bar{\boldsymbol{Z}}\boldsymbol{v}\right\}, \tag{4.101}$$

where $\bar{z} = \bar{Z}v$ is the row-reduced echelon form of the system of equations with variable v in (4.99) after removing the zero rows.

By the fundamental theorem in linear programming, the system $\bar{z} = \bar{Z}v$ has a basic feasible solution $v \in \mathbb{R}_+^n$ with at most rank $\{\bar{Z}\}$ nonzero elements. However, invoking Theorem 4.5 with $C = I$,

$$\operatorname{rank}\{\bar{Z}\} = \operatorname{rank}\left\{A^i (A^i)^\dagger - A^{i+1}\left(A^{i+1}\right)^\dagger\right\} = \operatorname{rank}\{A^i\} - \operatorname{rank}\{A^{i+1}\} \tag{4.102}$$

$$\leq n - \operatorname{rank}\{A\} \leq s, \tag{4.103}$$

where we use the Sylvester rank inequality. Hence, there exists an s-sparse vector $u \in \mathbb{R}_+^m$ such that

$$\left[A^i (A^i)^\dagger - A^{i+1}\left(A^{i+1}\right)^\dagger\right] z = \left[I - A^{i+1}\left(A^{i+1}\right)^\dagger\right] A^i B u. \tag{4.104}$$

The proof is complete. □

Lemma 4.12. Consider the linear dynamical system ($A \in \mathbb{R}^{n\times n}, B \in \mathbb{R}^{n\times m}$) in (4.1), and any sparsity level $s \leq m$. Then, for any $z \in \operatorname{ColSpace}\{CA^n\}$, there exist an integer $K < \infty$ and s-sparse inputs $\{u_k \in \mathbb{R}_+^m\}_{k=1}^K$ such that

$$z = \sum_{k=1}^{K} A^{n+K-k} B u_k. \tag{4.105}$$

Proof. Let the real Jordan canonical of A be

$$A = P^{-1} \begin{bmatrix} J & 0 \\ 0 & N \end{bmatrix} P, \tag{4.106}$$

where $P \in \mathbb{R}^{n\times n}$ is an invertible matrix and the square matrices $J \in \mathbb{R}^{J\times J}$ and $N \in \mathbb{R}^{(n-J)\times(n-J)}$ are formed by the Jordan blocks of A corresponding to its nonzero and zero eigenvalues, respectively. Here, J is an invertible matrix, and N is a nilpotent matrix, i.e., $N^n = 0$. Consequently, for any $P \geq n$, we deduce

$$A^P = P^{-1} \begin{bmatrix} J^P & 0 \\ 0 & 0 \end{bmatrix} P. \tag{4.107}$$

Also, due to the assumption that the system $(\boldsymbol{A}, \boldsymbol{B})$ is nonnegative controllable, we deduce that there exist an integer $K < \infty$ and (non-sparse) inputs $\{\boldsymbol{v}_k \in \mathbb{R}^m\}_{k=1}^K$ for any vector $\boldsymbol{x}$ such that

$$\boldsymbol{x} = \sum_{k=1}^{K} \boldsymbol{A}^{K-k} \boldsymbol{B} \boldsymbol{v}_k. \tag{4.108}$$

Premultiplying with $\boldsymbol{P}$ gives

$$\boldsymbol{P}\boldsymbol{x} = \sum_{k=1}^{K} \begin{bmatrix} \boldsymbol{J}^{K-k} & \boldsymbol{0} \\ \boldsymbol{0} & \boldsymbol{N}^{K-k} \end{bmatrix} \boldsymbol{P}\boldsymbol{B}\boldsymbol{v}_k. \tag{4.109}$$

Let $\boldsymbol{P}^{(J)}$ be the submatrix of $\boldsymbol{P}$ formed using its first J rows. Thus, we arrive at

$$\boldsymbol{P}^{(J)}\boldsymbol{x} = \sum_{k=1}^{K} \boldsymbol{J}^{K-k} \boldsymbol{P}^{(J)} \boldsymbol{B} \boldsymbol{v}_k. \tag{4.110}$$

Therefore, the system $(\boldsymbol{J}, \boldsymbol{P}^{(J)}\boldsymbol{B})$ is nonnegative controllable. By Proposition 4.3, $(\boldsymbol{J}, \boldsymbol{P}^{(J)}\boldsymbol{B})$ is s-sparse nonnegative controllable. As a result, for any $\boldsymbol{z}^{(1)} \in \mathbb{R}^J$, there exist an integer $K < \infty$ and inputs $\{\boldsymbol{u}_k \in \mathbb{R}^m\}_{k=1}^K$ such that

$$\boldsymbol{z}^{(1)} = \sum_{k=1}^{K} \boldsymbol{J}^{K-k} \boldsymbol{P}^{(J)} \boldsymbol{B} \boldsymbol{u}_k. \tag{4.111}$$

Therefore, we get for any $\boldsymbol{z} = \begin{bmatrix} \boldsymbol{z}^{(1)} \\ \boldsymbol{z}^{(2)} \end{bmatrix} \in \mathbb{R}^n$,

$$\boldsymbol{P}^{-1} \begin{bmatrix} \boldsymbol{J}^n & \boldsymbol{0} \\ \boldsymbol{0} & \boldsymbol{0} \end{bmatrix} \boldsymbol{z} = \boldsymbol{P}^{-1} \begin{bmatrix} \boldsymbol{J}^n & \boldsymbol{0} \\ \boldsymbol{0} & \boldsymbol{0} \end{bmatrix} \begin{bmatrix} \boldsymbol{z}^{(1)} \\ \boldsymbol{z}^{(2)} \end{bmatrix} = \boldsymbol{P}^{-1} \begin{bmatrix} \boldsymbol{J}^n \boldsymbol{z}^{(1)} \\ \boldsymbol{0} \end{bmatrix} \tag{4.112}$$

$$= \boldsymbol{P}^{-1} \begin{bmatrix} \boldsymbol{J}^n \sum_{k=1}^{K} \boldsymbol{J}^{K-k} \boldsymbol{P}^{(J)} \boldsymbol{B} \boldsymbol{u}_k \\ \boldsymbol{0} \end{bmatrix} \tag{4.113}$$

$$= \sum_{k=1}^{K} \boldsymbol{P}^{-1} \begin{bmatrix} \boldsymbol{J}^{n+K-k} & \boldsymbol{0} \\ \boldsymbol{0} & \boldsymbol{0} \end{bmatrix} \boldsymbol{P}\boldsymbol{B}\boldsymbol{u}_k \tag{4.114}$$

$$= \boldsymbol{P}^{-1} \sum_{k=1}^{K} \boldsymbol{A}^{n+K-k} \boldsymbol{B} \boldsymbol{u}_k, \tag{4.115}$$

where we use (4.107). Finally, since (4.107) also implies that

$$\text{ColSpace}\left\{\boldsymbol{A}^n\right\} = \text{ColSpace}\left\{\boldsymbol{P}^{-1}\begin{bmatrix}\boldsymbol{J}^n & \boldsymbol{0}\\ \boldsymbol{0} & \boldsymbol{0}\end{bmatrix}\right\}, \tag{4.116}$$

and we arrive at the desired result. □

Proof of Theorem 4.10. The necessity of the condition is straightforward because s-sparse nonnegative controllability is more stringent than nonnegative controllability and s-sparse controllability.

To prove the sufficiency, we suppose that the conditions in Theorem 4.10 hold. So, the sparsity level $s \geq n - \text{rank}\left\{\boldsymbol{A}\right\}$. Then, by Theorem 4.11, for any $\boldsymbol{z} \in \mathbb{R}^p$ and $i = 0, 1, \ldots, n-1$, there exists a vector $\boldsymbol{u}_{i+1} \in \mathbb{R}_+^n$ with $\|\boldsymbol{u}_{i+1}\|_0 \leq s$ such that

$$\boldsymbol{A}^i\left(\boldsymbol{A}^i\right)^\dagger \boldsymbol{z} = \left[\boldsymbol{A}^i\left(\boldsymbol{A}^i\right)^\dagger - \boldsymbol{A}^{i+1}\left(\boldsymbol{A}^{i+1}\right)^\dagger\right]\boldsymbol{z} + \boldsymbol{A}^{i+1}\left(\boldsymbol{A}^{i+1}\right)^\dagger \boldsymbol{z} \tag{4.117}$$

$$= \left[\boldsymbol{I} - \boldsymbol{A}^{i+1}\left(\boldsymbol{A}^{i+1}\right)^\dagger\right]\boldsymbol{A}^i\boldsymbol{B}\boldsymbol{u}_{i+1} + \boldsymbol{A}^{i+1}\left(\boldsymbol{A}^{i+1}\right)^\dagger \boldsymbol{z} \tag{4.118}$$

$$= \boldsymbol{A}^i\boldsymbol{B}\boldsymbol{u}_{i+1} + \boldsymbol{A}^{i+1}\left(\boldsymbol{A}^{i+1}\right)^\dagger\left[\boldsymbol{z} - \boldsymbol{A}^i\boldsymbol{B}\boldsymbol{u}_{i+1}\right]. \tag{4.119}$$

Therefore, recursively using (4.119) for $i = 0, 1, \ldots, n-1$, we get for any $\boldsymbol{y} \in \mathbb{R}^n$,

$$\boldsymbol{y} = \boldsymbol{B}\boldsymbol{u}_1 + \boldsymbol{A}\boldsymbol{A}^\dagger\left[\boldsymbol{y} - \boldsymbol{B}\boldsymbol{u}_1\right] \tag{4.120}$$

$$= \boldsymbol{B}\boldsymbol{u}_1 + \boldsymbol{A}\boldsymbol{B}\boldsymbol{u}_2 + \boldsymbol{A}^2\left(\boldsymbol{A}^2\right)^\dagger\left[\boldsymbol{y} - \boldsymbol{B}\boldsymbol{u}_1 - \boldsymbol{A}\boldsymbol{B}\boldsymbol{u}_2\right] \tag{4.121}$$

$$= \sum_{k=1}^{n}\boldsymbol{A}^{k-1}\boldsymbol{B}\boldsymbol{u}_k + \boldsymbol{A}^n\left(\boldsymbol{A}^n\right)^\dagger\left[\boldsymbol{y} - \sum_{k=1}^{n}\boldsymbol{A}^{k-1}\boldsymbol{B}\boldsymbol{u}_k\right], \tag{4.122}$$

where we substitute with $\boldsymbol{z} = \boldsymbol{y} - \sum_{k=1}^{l}\boldsymbol{A}^{k-1}\boldsymbol{B}\boldsymbol{u}_k$ in (4.119) for $l = 0, 1, \ldots, n-1$, and $\boldsymbol{u}_k \in \mathbb{R}_+^n$ has $\|\boldsymbol{u}_k\|_0 \leq s$, for $k = 1, 2, \ldots, n$.

Furthermore, the last term on the right-hand side of (4.122) belongs to $\text{ColSpace}\left\{\boldsymbol{A}^n\right\}$. So by Theorem 4.12, there exist an integer $K < \infty$ and input $\{\boldsymbol{u}_k \in \mathbb{R}^m\}_{k=n+1}^{n+K}$ such that

$$\boldsymbol{A}^n\left(\boldsymbol{A}^n\right)^\dagger\left[\boldsymbol{y} - \sum_{k=1}^{n}\boldsymbol{A}^{k-1}\boldsymbol{B}\boldsymbol{u}_k\right] = \sum_{k=1}^{K}\boldsymbol{A}^{n+K-k}\boldsymbol{B}\boldsymbol{u}_k. \tag{4.123}$$

Combining (4.122) and (4.123), we prove the desired result. □

We can restate the above theorem using Theorem 2.8 and Theorem 4.9 as follows:

Corollary 4.13. For any sparsity level, $s \geq 1$, the linear dynamical system ($\boldsymbol{A} \in \mathbb{R}^{n\times n}, \boldsymbol{B} \in \mathbb{R}^{n\times m}$) in (4.1) is s-sparse nonnegative controllable if and only if the following conditions hold:

(a) There exists no tuple ($\lambda \in \mathbb{C}, \boldsymbol{z} \neq \boldsymbol{0}$) such that $\boldsymbol{z}^\mathsf{T}\boldsymbol{A} = \lambda\boldsymbol{z}^\mathsf{T}$ and $\boldsymbol{z}^\mathsf{T}\boldsymbol{B} = \boldsymbol{0}$.

(b) There exists no tuple ($\lambda \geq 0, \boldsymbol{z} \neq \boldsymbol{0}$) such that $\boldsymbol{z}^\mathsf{T}\boldsymbol{A} = \lambda\boldsymbol{z}^\mathsf{T}$ and $\boldsymbol{z}^\mathsf{T}\boldsymbol{B} \in \mathbb{R}_+^m$.

(c) The sparsity level $s \geq n - \text{rank}\{\boldsymbol{A}\}$.

The above result implies that if the system $(\boldsymbol{A}, \boldsymbol{B})$ is s-sparse nonnegative controllable, it is also s-sparse nonpositive controllable. This is because if the system $(\boldsymbol{A}, \boldsymbol{B})$ satisfies the conditions of Theorem 4.10, the system $(\boldsymbol{A}, -\boldsymbol{B})$ also satisfies them.

We close this section by highlighting the main difference between sparse controllability and sparse nonnegative controllability. Sparse nonnegative controllability does not ensure that there exists a common actuator schedule that works for any initial and final states, as guaranteed by Theorem 2.3 for sparse controllability. Consequently, to design sparse nonnegative control to reach a desired final state, based on the above results, we need to resort to the state-dependent actuator scheduling approach in Section 3.2. Some compressed sensing algorithms to design nonnegative sparse control can be found in Wu *et al.* (2014), Nguyen *et al.* (2019), and Bruckstein *et al.* (2008) and their references and cited articles. We skip the details of the algorithms in this monograph.

4.4 Summary

- All stabilizable linear dynamical systems are s-sparse stabilizable, for any sparsity level $s \geq 1$.

- If a linear dynamical system is s-sparse output controllable, there exists an actuator schedule independent of the initial and final states, which can guarantee the output controllability of the system.

- The minimum sparsity level $s = s^*$ that guarantees s-sparse output controllability of an output controllable system $(\boldsymbol{A}, \boldsymbol{B}, \boldsymbol{C})$ is

$$\max_{1 \leq i \leq n} \frac{1}{i} \left(p - \operatorname{rank}\left\{\boldsymbol{C}\boldsymbol{A}^i\boldsymbol{\Phi}\right\}\right) \leq s^* \leq \max_{0 \leq i \leq n-1} \operatorname{rank}\left\{\boldsymbol{C}\boldsymbol{A}^i\boldsymbol{\Phi}\right\} - \operatorname{rank}\left\{\boldsymbol{C}\boldsymbol{A}^{i+1}\boldsymbol{\Phi}\right\},$$

where $\boldsymbol{C} \in \mathbb{R}^{p \times n}$ and $\boldsymbol{\Phi}$ is the controllability matrix.

- A linear dynamical system is s-sparse nonnegative controllable if and only if it is nonnegative controllable and s-sparse controllable.

4.5 Notes

The results and proofs concerning stabilizability are attributed to Sriram *et al.* (2022). Furthermore, for systems that are (sparse) stabilizable, this paper presents an algorithm designed to ascertain sparse control inputs steering the system state toward zero. This methodology diverges fundamentally from the classical stabilization approach via state feedback. Intriguingly, both methods involve control inputs that are linear functions of the initial state. However, while Sriram *et al.* (2022) employs a finite number of nonzero control inputs, state feedback can potentially entail infinitely many nonzero inputs.

Expanding the scope of applicability, the study also demonstrates that all stabilizable LDSs can achieve sparse mean square stabilization, especially when the process noise possesses zero mean and bounded second moment. In such scenarios, employing a sequential estimation technique to identify sparse control inputs can ensure the mean square stabilization of the system. Moreover, the paper explores how a detectable and stabilizable LDS can realize sparse stabilization through output feedback, accompanied by the development of an algorithm for identifying the corresponding sparse control inputs.

5

Sparse Control With Time-Invariant Support

This section extends the idea of sparse control to finding a small set of actuators so that the resulting system is controllable. We discuss the controllability tests and algorithms to design sparse control inputs with fixed support. The results discussed in this section heavily rely on Sections 2 to 4.

5.1 Controllability via Control With Time-Invariant Support

We recall the state space model of the linear dynamical system $(\boldsymbol{A}, \boldsymbol{B})$, with state $\boldsymbol{x}_k \in \mathbb{R}^n$, input $\boldsymbol{u}_k \in \mathbb{R}^m$, and $\boldsymbol{y}_k \in \mathbb{R}^p$,

$$\boldsymbol{x}_{k+1} = \boldsymbol{A}\boldsymbol{x}_k + \boldsymbol{B}\boldsymbol{u}_k \quad \text{and} \quad \boldsymbol{y}_k = \boldsymbol{C}\boldsymbol{x}_k, \tag{5.1}$$

for discrete-time indices $k = 1, 2, \ldots$. Here, we impose the constraint that the inputs are jointly sparse i.e.,

$$\text{supp}(\boldsymbol{u}_k) = \mathcal{S} \quad \text{and} \quad |\mathcal{S}| \leq s. \tag{5.2}$$

for $k = 1, 2, \ldots$. Therefore, $\boldsymbol{u}_k$ is s-sparse, and the control inputs have common support $\mathcal{S}$.

The notions of sparse controllability in Definition 2.2, sparse stabilizability in Definition 4.1, sparse output controllability in Definition 4.2

and nonnegative sparse control in Definition 4.3 can be trivially extended to sparse control inputs with time-invariant support. Similarly, we can extend Theorems 2.1, 4.1, 4.3 and 4.9 for the sparse control with time-invariant support case as follows:

Theorem 5.1 (Joseph and Murthy, 2021)**.** Consider the linear dynamical system ($\boldsymbol{A} \in \mathbb{R}^{n\times n}, \boldsymbol{B} \in \mathbb{R}^{n\times m}, \boldsymbol{C} \in \mathbb{R}^{p\times n}$) in (5.1).

(a) The system is controllable using s-sparse inputs with time-invariant support if and only if there exists a set $\mathcal{S} \subseteq \{1, 2, \ldots, m\}$ with $|\mathcal{S}| \leq s$ such that

$$\operatorname{rank}\left\{\begin{bmatrix}\boldsymbol{A} - \lambda\boldsymbol{I} & \boldsymbol{B}_{\mathcal{S}}\end{bmatrix}\right\} = n, \quad \forall\, \lambda \in \mathbb{C}. \tag{5.3}$$

(b) The system is stabilizable using s-sparse inputs with time-invariant support if and only if there exists a set $\mathcal{S} \subseteq \{1, 2, \ldots, m\}$ with $|\mathcal{S}| \leq s$ such that

$$\operatorname{rank}\left\{\begin{bmatrix}\boldsymbol{A} - \lambda\boldsymbol{I} & \boldsymbol{B}_{\mathcal{S}}\end{bmatrix}\right\} = n, \quad \forall\, \lambda : \ |\lambda| \geq 1. \tag{5.4}$$

(c) The system is output controllable using s-sparse inputs with time-invariant support if and only if there exists a set $\mathcal{S} \subseteq \{1, 2, \ldots, m\}$ with $|\mathcal{S}| \leq s$ such that

$$\operatorname{rank}\left\{\boldsymbol{C}\begin{bmatrix}\boldsymbol{A}^{n-1}\boldsymbol{B}_{\mathcal{S}} & \boldsymbol{A}^{n-2}\boldsymbol{B}_{\mathcal{S}} & \ldots & \boldsymbol{B}_{\mathcal{S}}\end{bmatrix}\right\} = p. \tag{5.5}$$

(d) The system is nonnegative controllable using s-sparse inputs with time-invariant support if and only if there exists a set $\mathcal{S} \subseteq \{1, 2, \ldots, m\}$ with $|\mathcal{S}| \leq s$ such that

(i) $\operatorname{rank}\left\{\begin{bmatrix}\boldsymbol{A} - \lambda\boldsymbol{I} & \boldsymbol{B}_{\mathcal{S}}\end{bmatrix}\right\} = n$, for all $\lambda \in \mathbb{C}$

(ii) For every left eigenvalue $\lambda \geq 0$ of $\boldsymbol{A}$ and the corresponding eigenvector $\boldsymbol{z}$, we have $\boldsymbol{z}^\mathsf{T}\boldsymbol{B}_{\mathcal{S}} \notin \mathbb{R}_+^m$.

Furthermore, we note that sparse control with time-invariant support is more restricted than sparse control with time-varying support. So, the necessary and sufficient conditions for sparse control with time-varying support are necessary for sparse control with time-invariant support. We skip the results here to avoid repetition.

5.2 Minimum Sparsity Level for Controllability

We start with a simple result from Theorem 5.1 to bound the sparsity level.

Proposition 5.1. Consider the linear dynamical system ($\boldsymbol{A} \in \mathbb{R}^{n\times n}, \boldsymbol{B} \in \mathbb{R}^{n\times m}, \boldsymbol{C} \in \mathbb{R}^{p\times n}$) in (5.1). The system is controllable using s-sparse inputs with time-invariant support only if

$$\operatorname{rank}\left\{\begin{bmatrix}\boldsymbol{A}-\lambda\boldsymbol{I} & \boldsymbol{B}\end{bmatrix}\right\} = n, \;\; \forall\, \lambda \in \mathbb{C} \quad \text{and} \quad s \geq n - \min_{\lambda\in\mathbb{C}} \operatorname{rank}\left\{\boldsymbol{A}-\lambda\boldsymbol{I}\right\}. \tag{5.6}$$

Proof. From Theorem 5.1, there exists $\mathcal{S} \subseteq \{1, 2, \ldots, m\}$ with $|\mathcal{S}| \leq s$ such that

$$n = \min_{\lambda\in\mathbb{C}} \operatorname{rank}\left\{\begin{bmatrix}\boldsymbol{A}-\lambda\boldsymbol{I} & \boldsymbol{B}_{\mathcal{S}}\end{bmatrix}\right\} \tag{5.7}$$

$$\leq \min_{\lambda\in\mathbb{C}} \operatorname{rank}\left\{\boldsymbol{A}-\lambda\boldsymbol{I}\right\} + \operatorname{rank}\left\{\boldsymbol{B}_{\mathcal{S}}\right\} \tag{5.8}$$

$$\leq \min_{\lambda\in\mathbb{C}} \operatorname{rank}\left\{\boldsymbol{A}-\lambda\boldsymbol{I}\right\} + s. \tag{5.9}$$

Rearranging the above relation, we arrive at the desired result. □

The above proposition is a weak result, but finding the minimum sparsity level for ensuring controllability via sparse control with time-invariant support is an NP-hard problem. To prove the result, we first (informally) introduce the necessary terminology from computational complexity. Decision problems fall into different categories based on their complexity:

1. The class of P-problems (P stands for polynomial time) refers to the decision problems for which there exists a polynomial-time algorithm that solves the problem. Here, a polynomial-time algorithm is one whose computational time complexity is a polynomial function of the size of the input.

2. The class of NP-problems (NP stands for nondeterministic polynomial time) refers to the decision problems for which there exists a polynomial-time algorithm that verifies a given solution.

3. The class of NP-hard problems refers to the (not necessarily decision) problems for which a solving algorithm can be transformed into a solving algorithm for an NP-problem in polynomial time.

4. The class of NP-complete problems consists of all problems that are both NP and NP-hard.

Clearly, every P-problem is NP, but every NP-hard problem is not NP. If a problem is proved to be NP-hard, then its solution cannot be found in polynomial time unless P = NP. There are numerous NP-hard problems, and we introduce one problem below.

Theorem 5.2 (Karp, 2010)**.** Given a set $\mathcal{C} = \{\mathcal{C}_1, \mathcal{C}_2, \ldots \mathcal{C}_N\}$, where $\mathcal{C}_i \subseteq \{1, 2, \ldots, m\}$, the minimum hitting set problem seeks for the set $\mathcal{S}^*$ such that

$$\mathcal{S}^* = \min_{\mathcal{S} \subseteq \{1,2,\ldots,m\}} |\mathcal{S}| \text{ s. t. } \mathcal{S} \cap \mathcal{C}_i \neq \emptyset, \ \forall\, i = 1, 2, \ldots, N. \tag{5.10}$$

The minimum hitting set problem is NP-hard.

Taking the above result for granted, we prove the next main result of this section.

Theorem 5.3 (Sriram *et al.*, 2022)**.** For any general linear dynamical system ($\boldsymbol{A} \in \mathbb{R}^{n\times n}, \boldsymbol{B} \in \mathbb{R}^{n\times m}$) in (5.1), finding the set $\mathcal{S} \subseteq \{1, 2, \ldots, m\}$ with smallest cardinality the that satisfies

$$\operatorname{rank}\left\{\begin{bmatrix}\boldsymbol{A} - \lambda\boldsymbol{I} & \boldsymbol{B}_{\mathcal{S}}\end{bmatrix}\right\} = n, \ \ \forall\, \lambda \in \mathbb{C}. \tag{5.11}$$

is NP-hard.

Proof. We solve for

$$\mathcal{S}^* = \arg\min{}_{\mathcal{S} \subseteq \{1,2,\ldots,m\}} |\mathcal{S}| \text{ s. t. } \operatorname{rank}\left\{\begin{bmatrix}\boldsymbol{A} - \lambda\boldsymbol{I} & \boldsymbol{B}_{\mathcal{S}}\end{bmatrix}\right\} = n, \ \forall \lambda \in \mathbb{C}. \tag{5.12}$$

Let $\{\lambda_i\}_{i=1}^N$ be the distinct left eigenvalues of $\boldsymbol{A}$. We can rewrite (5.12) as

$$\mathcal{S}^* = \arg\min_{\mathcal{S}\subseteq\{1,2,\ldots,m\}} |\mathcal{S}| \text{ s. t.}$$
$$\forall\lambda \in \{\lambda_i\}_{i=1}^N \;\; \exists\, \boldsymbol{v} \text{ for which } \boldsymbol{v}^\top \begin{bmatrix}\boldsymbol{A}-\lambda\boldsymbol{I} & \boldsymbol{B}_{\mathcal{S}}\end{bmatrix} \neq \boldsymbol{0} \tag{5.13}$$
$$= \arg\min_{\mathcal{S}\subseteq\{1,2,\ldots,m\}} |\mathcal{S}| \text{ s. t.}$$
$$\forall\lambda \in \{\lambda_i\}_{i=1}^N \;\exists\, (\boldsymbol{v}, j \in \mathcal{S}) \text{ for which } \boldsymbol{v}^\top \begin{bmatrix}\boldsymbol{A}-\lambda\boldsymbol{I} & \boldsymbol{B}_j\end{bmatrix} \neq \boldsymbol{0} \tag{5.14}$$
$$= \arg\min_{\mathcal{S}\subseteq\{1,2,\ldots,m\}} |\mathcal{S}| \text{ s. t. } \forall i = 1,2,\ldots,N \;\; , \mathcal{S}\cap C_i \neq \emptyset, \tag{5.15}$$

where we define N sets

$$C_i = \{j : \;\exists\, \boldsymbol{v} \text{ s. t. } \boldsymbol{v}^\top(\boldsymbol{A}-\lambda_i\boldsymbol{I}) = \boldsymbol{0} \text{ and } \boldsymbol{v}^\top\boldsymbol{B}_j \neq 0\}. \tag{5.16}$$

Since $\boldsymbol{A}$ and $\boldsymbol{B}$ can be any matrix, the sets C_i can be arbitrary. Consequently, solving the above problem enables one to solve the minimum hitting set problem, and the problem is NP-hard. □

Theorem 5.3 seems pessimistic, but it shows the intractability of the sparse control problem for general systems. So it is not surprising that we only have a weak bound in Proposition 5.1 on the sparsity level rather than the exact value similar to Theorem 2.8.

Theorem 5.4. For any general linear dynamical system ($\boldsymbol{A} \in \mathbb{R}^{n\times n}, \boldsymbol{B} \in \mathbb{R}^{n\times m}$) in (5.1), finding the set $\mathcal{S} \subseteq \{1,2,\ldots,m\}$ with smallest cardinality that satisfies

$$\operatorname{rank}\left\{\begin{bmatrix}\boldsymbol{A}-\lambda\boldsymbol{I} & \boldsymbol{B}_{\mathcal{S}}\end{bmatrix}\right\} = n, \;\; \forall\,\lambda \in \mathbb{C} : |\lambda| \geq 1. \tag{5.17}$$

is NP-hard.

Proof. The proof is similar to that of Theorem 5.3 and hence omitted. □

5.3 Sparse Input Design

From Theorem 2.2, we note that a system, which is controllable using s-sparse inputs with common support, needs at most n sparse inputs to drive the system from any initial to any final state. So, for a given initial state $\boldsymbol{x}_1 \in \mathbb{R}^n$ and final state $\boldsymbol{x}_{\mathrm{f}} \in \mathbb{R}^n$, we seek the n jointly sparse inputs $\{\boldsymbol{u}_k\}_{k=1}$ such that

$$\boldsymbol{x}_{\mathrm{f}} - \boldsymbol{A}^n \boldsymbol{x}_1 = \sum_{k=1}^{n} \boldsymbol{A}^{n-k} \boldsymbol{B} \boldsymbol{u}_k = \sum_{j=1}^{m} \sum_{k=1}^{n} \boldsymbol{A}^{n-k} \boldsymbol{B}_j \boldsymbol{u}_k[j]. \tag{5.18}$$

where $\boldsymbol{u}_k[j]$ is the jth entry of $\boldsymbol{u}_k$. Let the vector of sparse unknowns be

$$\hat{\boldsymbol{u}} = \begin{bmatrix} \hat{\boldsymbol{u}}_1 \\ \hat{\boldsymbol{u}}_2 \\ \vdots \\ \hat{\boldsymbol{u}}_m \end{bmatrix}, \tag{5.19}$$

where $\hat{\boldsymbol{u}}_j = \begin{bmatrix} \boldsymbol{u}_1[j] & \boldsymbol{u}_2[j] & \dots & \boldsymbol{u}_n[j] \end{bmatrix}^{\mathsf{T}} \in \mathbb{R}^n$. Then, we rewrite (5.18) as

$$\bar{\boldsymbol{x}} = \boldsymbol{x}_{\mathrm{f}} - \boldsymbol{A}^n \boldsymbol{x}_1 = \hat{\boldsymbol{\Phi}} \hat{\boldsymbol{u}}, \tag{5.20}$$

where we define the reordered controllability matrix $\hat{\boldsymbol{\Phi}}$ as

$$\hat{\boldsymbol{\Phi}} = \big[\boldsymbol{A}^{n-1}\boldsymbol{B}_1 \quad \boldsymbol{A}^{n-2}\boldsymbol{B}_1 \quad \dots \quad \boldsymbol{B}_1 \quad \boldsymbol{A}^{n-1}\boldsymbol{B}_2 \quad \boldsymbol{A}^{n-2}\boldsymbol{B}_2 \quad \dots \quad \boldsymbol{B}_2 \quad \dots \quad \boldsymbol{A}^{n-1}\boldsymbol{B}_m \quad \boldsymbol{A}^{n-2}\boldsymbol{B}_m \quad \dots \quad \boldsymbol{B}_m \big]. \tag{5.21}$$

Hence, solving for sparse inputs is the same as solving the system of equations

$$\bar{\boldsymbol{x}} = \hat{\boldsymbol{\Phi}} \hat{\boldsymbol{u}}, \tag{5.22}$$

for a block sparse solution $\hat{\boldsymbol{u}}$, i.e., each block $\hat{\boldsymbol{u}}_i$ is either zero or nonzero. We note that the block $\hat{\boldsymbol{u}}_i$ is nonzero if i belongs to the common support of the sparse vectors. In other words, if the control inputs are s-sparse, at most s blocks of $\hat{u}$ are nonzero. This special sparsity structure is called *block sparsity* or *group sparsity* or *clustered sparsity.*

As in the case of time-varying support, we divide the sparse input design algorithms for the time-invariant support case into two categories: state-dependent actuator scheduling and state-independent actuator scheduling.

5.4 State-dependent Actuator Scheduling

In the state-dependent actuator scheduling, we directly solve (5.22) with the constraint that $\hat{\boldsymbol{u}}$ is block sparse. There are numerous sparse recovery algorithms used in compressive sensing to estimate a block sparse vector, and we present convex optimization methods and greedy methods.

5.4.1 Convex Optimization Methods

The convex optimization methods modify the basis pursuit algorithm in Algorithm 1 to promote the block sparse structure. We note that each block of $\hat{\boldsymbol{u}}$ can either be zero or nonzero. Within a nonzero block, the values are unrestricted. The convex optimization approach minimizes the energy or ℓ_2 norm of each block, i.e., it considers the vector $\begin{bmatrix} \|\hat{\boldsymbol{u}}_1\| & \|\hat{\boldsymbol{u}}_2\| & \dots & \|\hat{\boldsymbol{u}}_m\| \end{bmatrix}$. Since $\hat{\boldsymbol{u}}$ is block sparse, the vector of ℓ_2 norm is sparse. Thus, we minimize the ℓ_1 norm of the ℓ_2 norm vector to obtain

$$\arg\min_{\hat{\boldsymbol{u}} \in \mathbb{R}^{nm}} \sum_{i=1}^{m} \|\hat{\boldsymbol{u}}_i\| \quad \text{s. t.} \quad \bar{\boldsymbol{x}} = \boldsymbol{\Phi}\hat{\boldsymbol{u}}, \tag{5.23}$$

The above principle is usually called the $\ell_{2,1}$-minimization or group sparse basis pursuit, and the algorithm is summarized in Algorithm 8 (Yuan and Lin, 2006; Huang and Zhang, 2010; Kolar *et al.*, 2011).

Algorithm 8 Sparse input design via group basis pursuit

Input: Initial state $\boldsymbol{x}_1 \in \mathbb{R}^n$, final state $\boldsymbol{x}_\mathrm{f} \in \mathbb{R}^n$, linear dynamical system $(\boldsymbol{A} \in \mathbb{R}^{n\times n}, \boldsymbol{B} \in \mathbb{R}^{n\times m})$

1: Set $\bar{\boldsymbol{x}} = \boldsymbol{x}_\mathrm{f} - \boldsymbol{A}^n\boldsymbol{x}_1$
2: Compute $\hat{\boldsymbol{\Phi}}$ using (5.21)
3: Determine $\hat{\boldsymbol{u}}$ using (5.23)

Output: Control inputs $\hat{\boldsymbol{u}}$

5.4.2 Greedy Algorithms

We next present two greedy algorithms to solve (5.22), namely block orthogonal matching pursuit and block compressive sampling matching pursuit (Zhang *et al.*, 2019).

The block orthogonal matching pursuit algorithm is similar to orthogonal matching pursuit in Algorithm 2. The main difference is that in every iteration, it chooses a block index instead of an entry index of the sparse vector. Further, we note that the part of the rearranged controllability matrix in (5.21) corresponding to the ith block is

$$\hat{\boldsymbol{\Phi}}_{\mathcal{U}_i} = \begin{bmatrix} \boldsymbol{A}^{n-1}\boldsymbol{B}_i & \boldsymbol{A}^{n-2}\boldsymbol{B}_i & \dots & \boldsymbol{B}_i \end{bmatrix}, \tag{5.24}$$

where $\mathcal{U}_i = \{(i-1)n+1, (i-1)n+2, \dots, in\}$. So, the block index chosen in every iteration is the one with the largest ℓ_2 norm of the residue when projected onto the matrix $\hat{\boldsymbol{\Phi}}_{\mathcal{U}_i}$. Once an entry is added to the block support set, the corresponding control inputs are calculated using the least square solution. The resulting algorithm is summarized in Algorithm 9.

Algorithm 9 Sparse input design via block orthogonal matching pursuit

Input: Initial state $\boldsymbol{x}_1 \in \mathbb{R}^n$, final state $\boldsymbol{x}_{\mathrm{f}} \in \mathbb{R}^n$, linear dynamical system $(\boldsymbol{A} \in \mathbb{R}^{n\times n}, \boldsymbol{B} \in \mathbb{R}^{n\times m})$

Initialization: $\mathcal{S} = \mathcal{U} = \emptyset$, $\hat{\boldsymbol{u}} = \boldsymbol{0}$

Set $\hat{\boldsymbol{x}} = \boldsymbol{x}_{\mathrm{f}} - \boldsymbol{A}^n\boldsymbol{x}_1$
Set $\mathcal{U}_i = \{(i-1)n+1, (i-1)n+2, \dots, in\}$, for $i = 1, 2, \dots, m$
Compute $\hat{\boldsymbol{\Phi}}$ using (5.21)
repeat
    Compute $i^* = \arg\max_{i\notin\mathcal{S}} \frac{\left\|\hat{\boldsymbol{\Phi}}_{\mathcal{U}_i}^{\top}(\bar{\boldsymbol{x}}-\hat{\boldsymbol{\Phi}}\hat{\boldsymbol{u}})\right\|}{\left\|\hat{\boldsymbol{\Phi}}_{\mathcal{U}_i}\right\|}$
    Update support $\mathcal{S} \leftarrow \mathcal{S} \cup \{i^*\}$ and $\mathcal{U} \leftarrow \mathcal{U} \cup \mathcal{U}_{i^*}$
    Determine $\hat{\boldsymbol{u}}_{\mathcal{U}} \leftarrow \hat{\boldsymbol{\Phi}}_{\mathcal{U}}^{\dagger}\hat{\boldsymbol{x}}$ and $\hat{\boldsymbol{u}}_{\mathcal{U}^{\complement}} \leftarrow \boldsymbol{0}$
until $\bar{\boldsymbol{x}} = \hat{\boldsymbol{\Phi}}\hat{\boldsymbol{u}}$

Output: Control inputs $\hat{\boldsymbol{u}}$

The second greedy algorithm, called the block compressive sampling matching pursuit algorithm, is a modified version of compressive sampling matching pursuit in Algorithm 3. Every iteration of block compressive sampling matching pursuit first adds the block indices of top $2s$ blocks of $\boldsymbol{\Phi}$ that match with the residue. Then, the corresponding estimate of $\hat{\boldsymbol{u}}$ is computed using the extended target support. Finally, it keeps the estimate's s blocks with the largest ℓ_2 norm and sets the remaining entries to zero. The resulting algorithm is summarized in Algorithm 10.

Algorithm 10 Sparse input design via block compressive sampling matching pursuit

Input: Initial state $\boldsymbol{x}_1 \in \mathbb{R}^n$, final state $\boldsymbol{x}_\mathrm{f} \in \mathbb{R}^n$, linear dynamical system ($\boldsymbol{A} \in \mathbb{R}^{n\times n}, \boldsymbol{B} \in \mathbb{R}^{n\times m}$), sparsity level s

Initialization: $\mathcal{S} = \mathcal{U} = \emptyset$, $\hat{\boldsymbol{u}} = \boldsymbol{0}$

Set $\bar{\boldsymbol{x}} = \boldsymbol{x}_\mathrm{f} - \boldsymbol{A}^n \boldsymbol{x}_1$

Set $\mathcal{U}_i = \{(i-1)n+1, (i-1)n+2, \ldots, in\}$, for $i = 1, 2, \ldots, m$

Compute $\hat{\boldsymbol{\Phi}}$ using (5.21)

repeat

  Compute $\mathcal{I}^* = \underset{\mathcal{I}\subseteq\{1,2,\ldots,m\},\,|\mathcal{I}|\leq 2s}{\arg\max} \sum_{i\in\mathcal{I}} \frac{\left\|\hat{\boldsymbol{\Phi}}_{\mathcal{U}_i}^\mathsf{T}(\bar{\boldsymbol{x}}-\hat{\boldsymbol{\Phi}}\hat{\boldsymbol{u}})\right\|}{\|\hat{\boldsymbol{\Phi}}_{\mathcal{U}_i}\|}$

  Update support $\mathcal{S} \leftarrow \mathcal{S} \cup \mathcal{I}^*$ and $\mathcal{U} \leftarrow \bigcup_{i\in\mathcal{S}} \mathcal{U}_i$

  Determine $\hat{\boldsymbol{u}}_\mathcal{S} \leftarrow \boldsymbol{\Phi}_\mathcal{U}^\dagger \bar{\boldsymbol{x}}$ and $\hat{\boldsymbol{u}}_{\mathcal{U}^\complement} \leftarrow \boldsymbol{0}$

  Determine new block support $\mathcal{S} \leftarrow \underset{\mathcal{I}\subseteq\{1,2,\ldots,m\},\,|\mathcal{I}|\leq s}{\arg\max} \sum_{i\in\mathcal{I}} \left\|\hat{\boldsymbol{u}}_{\mathcal{U}_i}\right\|$

  Update support $\mathcal{U} \leftarrow \bigcup_{i\in\mathcal{S}} \mathcal{U}_i$

  Update the estimate $\boldsymbol{u}_{\mathcal{U}^\complement} \leftarrow \boldsymbol{0}$

until $\bar{\boldsymbol{x}} = \boldsymbol{\Phi}\hat{\boldsymbol{u}}$

Output: Control inputs $\hat{\boldsymbol{u}}$

The block sparse recovery algorithms can be found in the standard compressed sensing literature. Please refer to (Yuan and Lin, 2006; Huang and Zhang, 2010; Kolar *et al.*, 2011) for block basis pursuit, Eldar *et al.* (2010) and Swirszcz *et al.* (2009) for block orthogonal matching pursuit, and Zhang *et al.* (2019) or block compressive sampling orthogonal matching pursuit.

5.5 State-independent Actuator Scheduling

The goal of sparse input design with state-independent actuator schedule is to find a sparse actuator schedule $\mathcal{S} \subseteq \{1, 2, \ldots, m\}$ with , $|\mathcal{S}| \leq s$ such that the resulting controllability matrix is full rank. The actuator schedule gives the indices of the nonzero entries $\mathcal{U}$ of $\hat{\boldsymbol{u}}$ in (5.19), i.e.,

$$\mathcal{U} = \bigcup_{i\in\mathcal{S}} \{(i-1)n+1, (i-1)n+2, \ldots, in\}. \tag{5.25}$$

Therefore, we derive $\hat{\boldsymbol{u}}_{\mathcal{U}^\complement} = \mathbf{0}$. Hence, for a given pair of initial and final states $(\boldsymbol{x}_1, \boldsymbol{x}_\mathrm{f})$ the nonzero entries of $\boldsymbol{u}$ are given by the least-square solution.

As discussed in Section 3.5, we can solve for a sparse schedule using any controllability metric ρ that quantifies various required control input energy, as given in Table 3.1. Hence, we reformulate the sparse scheduling problem as

$$\underset{\mathcal{S}\subseteq\{1,2,\ldots,m\}}{\arg\min}\ \rho\left(\sum_{k=1}^{n}\sum_{i\in\mathcal{S}}\boldsymbol{A}^{n-k}\boldsymbol{B}_i\boldsymbol{B}_i^\top\boldsymbol{A}^{(n-k)\top}\right)\ \text{ s. t. }\ |\mathcal{S}|\leq s. \tag{5.26}$$

The above problem can be solved using a greedy heuristic similar to Algorithm 6 (Siami *et al.*, 2020). Similar to the block orthogonal matching pursuit and block compressive sampling matching pursuit Algorithms 9 and 10, the greedy algorithm chooses one block index in every iteration until the resulting controllability matrix is full rank. The greedy block sparse scheduling algorithm is summarized in Algorithm 11.

Algorithm 11 Greedy block sparse actuator scheduling

Input: Linear dynamical system $(\boldsymbol{A}\in\mathbb{R}^{n\times n}, \boldsymbol{B}\in\mathbb{R}^{n\times m})$
Initialization: $\mathcal{S}=\emptyset$, $\boldsymbol{W}=\mathbf{0}$, $\epsilon>0$
Set $\mathcal{U}_i=\{(i-1)n+1,(i-1)n+2,\ldots,in\}$, for $i=1,2,\ldots,m$
Compute $\boldsymbol{\Phi}=\begin{bmatrix}\boldsymbol{A}^{n-1}\boldsymbol{B} & \boldsymbol{A}^{n-2}\boldsymbol{B} & \ldots & \boldsymbol{B}\end{bmatrix}$
repeat
    Compute $i^*=\arg\min_{i\notin\mathcal{S}}\rho\left(\boldsymbol{W}+\boldsymbol{\Phi}_{\mathcal{U}_i}\boldsymbol{\Phi}_{\mathcal{U}_i}^\top+\epsilon\boldsymbol{I}\right)$
    Update support $\mathcal{S}\leftarrow\mathcal{S}\cup\{i^*\}$
    Update Gramian matrix $\boldsymbol{W}\leftarrow\boldsymbol{W}+\boldsymbol{\Phi}_{\mathcal{U}_i}\boldsymbol{\Phi}_{\mathcal{U}_i}^\top$
until rank $\{\boldsymbol{W}\}=n$
Output: Actuator schedule $\mathcal{S}$

5.6 Summary

- Finding the minimum sparsity level for controllability via sparse inputs with common support is an NP-hard problem.
- Designing sparse inputs with state-dependent actuator scheduling can be solved using block sparse recovery algorithms.

- The state-independent sparse actuator scheduling relies on suitable controllability metrics to also optimize the control energy and uses greedy algorithms to optimize the schedule.

5.7 Notes

The concept of designing control inputs with time-varying support shares some connection with the actuator placement or actuator subset selection problems explored in existing literature (Ruckman and Fuller, 1995; Silva *et al.*, 2019; Argha *et al.*, 2019; Manohar *et al.*, 2021; Taha *et al.*, 2017; Dhingra *et al.*, 2014). However, it's important to note that the majority of existing works concentrate on continuous-time systems, whereas our focus lies on discrete-time systems. The analysis and block sparse recovery algorithms discussed are specific to discrete-time linear systems, and extending them to continuous-time systems is yet to be studied.

6

Future Directions and Emerging Topics

Building on the studies discussed in the previous section, this section outlines the main challenges in the field of sparse actuator control, examines the connections and contrasts with other related areas, and suggests future research directions in this area.

6.1 Theory of Controllability With Sparsity

In Sections 2, 4 and 5, we presented the theoretical underpinnings of sparse actuator control, focusing mainly on the existence results. Other research directions and extensions are listed below.

Continuous-time systems

The theory and algorithms discussed here are tailored to discrete-time systems, posing a challenge in extending them to continuous-time systems due to inherent differences between the two domains. Defining sparse actuator utilization within continuous systems is not straightforward. One approach involves establishing continuous-time scheduling while imposing restrictions on schedule or actuator use alterations. Investigating how these theories can be adapted to continuous-time linear dynamical systems presents an enticing avenue for research.

Maximum hands-off

Another related area of research is sparsity in time or maximum hands-off presented in Section 1.4.1. Both approaches aim to reduce the control effort and share several advantages, such as reduced communication overhead and minimized emissions, vibrations, or noise due to inactivity due to non-activity. However, the theoretical connection or comparison between sparsity in time (maximum hands-off) and sparsity in actuator use has not been established yet and presents an interesting direction for future work.

Switched control systems

Sparsity constraints can be seen as a special case of switched control systems (Skafidas *et al.*, 1999; Sun and Zhao, 2001; Sun and Ge, 2005) where we switch between different discrete-time linear dynamical systems defined by the following set:

$$\{(\boldsymbol{A} \in \mathbb{R}^{n \times n}, \boldsymbol{B}_{\mathcal{S}} \in \mathbb{R}^{n \times s}) : \ \mathcal{S} \subset \{1, 2, \ldots, m\}, \ \text{ and } \ |\mathcal{S}| \leq s\} \tag{6.1}$$

Since switched systems are thoroughly investigated, connecting sparsity constraints to switched systems and borrowing ideas from their theory and design is a fascinating topic to explore.

Other notions of controllability

Controllability with sparsity constraints has been studied only in the context of a few related concepts, such as stabilizability, output controllability, and non-negative controllability. There have been no studies on how the theory changes with additional constraints on actuator use, such as limited repeated use of certain actuators or limited changes in the schedule over time. It is also interesting to explore other notions of controllability, such as (strong) structural controllability and target controllability for structural systems.

Connections to sensing and observability

Observability is the dual of controllability. Therefore, sparse controllability theory and algorithms can provide insights into sensing with a limited number of sensors from the available set. Although different formulations have been studied, the state-of-the-art algorithms are greedy in nature, and an explicit connection between sparse observability and sparse controllability is yet to be established.

6.2 Sparse Actuator Control Design Algorithms

In Sections 3 and 5, we discussed different approaches to designing sparse control inputs. The currently available state-dependent and state-independent actuator scheduling actuator scheduling can be improved in several ways. A few examples are discussed below.

State-dependent actuator scheduling

As mentioned in Section 3, the actuator scheduling algorithms adapted from the compressed sensing literature lack theoretical guarantees regarding control performance optimality. While these algorithms perform well in practice, analyzing their effectiveness for sparse control design remains an open problem.

Additionally, the sparsity constraint on individual inputs leads to a new sparsity structure called piecewise sparsity (see (3.9)). Unlike the standard sparse recovery problem (see (3.8)), the piecewise sparse recovery problem is not well-studied, and only limited greedy algorithms (Li *et al.*, 2016; Zhong and Li, 2018) are available in the literature. More efficient algorithms beyond the existing greedy heuristics are still an open problem in sparse signal processing literature.

Another important aspect that has not been utilized is the special structure in the controllability matrix $\boldsymbol{\Phi}$. In the standard compressed sensing problem, the measurement matrix is either random or a submatrix of a well-conditioned basis matrix of $\mathbb{R}^n$. However, for the sparse control design problem, the controllability matrix has a unique structure due to the different powers of $\boldsymbol{A}$. Although this structure has been exploited to derive the theoretical results in Sections 2, 4 and 5, it has not been leveraged in algorithm development, presenting a new direction for exploration.

Finally, although we discuss various block sparse recovery algorithms for designing control inputs with time-invariant support, these methods have not undergone comprehensive scrutiny within the control design literature. Examining both the empirical and theoretical performance of these approaches is an open area.

State-independent actuator scheduling

The second approach for design is state-independent actuator scheduling. While the current work mostly uses greedy or heuristic-based algorithms, we can consider more systematic designs for actuator scheduling. Some examples include convex optimization-based or machine learning-based algorithms. Moreover, similar to state-dependent actuator scheduling, the special structure of the controllability matrix $\mathbf{\Phi}$ can be exploited to design more efficient algorithms. Analysis of the algorithms beyond submodularity-related concepts is also overlooked.

6.3 Sparse Control With System Constraints

Since sparse actuator scheduling is a relatively new research area, the available algorithms are limited and mostly borrowed from the existing literature. The current approaches of sparse control design focus on finding the sparsest solution. However, it may not be the optimal solution from the control system perspective. Therefore, future research could involve designing sparse inputs adhering to control system constraints.

Energy constraints

The sparsest solution may not always be the most energy-efficient. A denser (or less sparse) solution within the maximum sparsity constraint can offer flexibility and often lower energy costs. An important question is how to design sparse inputs that steer the system to the desired state with minimal energy expenditure while obeying the sparsity constraint. Studying the trade-off between sparsity and energy constraints is also of interest.

Time-to-control constraints

In some cases, it is crucial to reach the desired state in a minimal number of time steps. Optimizing time-to-control is challenging as the number of sparse inputs must be optimized. So, it can not directly rely on solving a system of equations (see (3.6)). Jointly optimizing sparsity and time-to-control is an intriguing area for further exploration.

Bandwidth constraints

Bandwidth constraints are important in applications like cyber-physical systems where the controller and the plant are geographically separated. Optimizing the bandwidth constraint involves models from communication theory and quantization, opening an exciting direction of research at the intersection of these diverse areas.

Constrained control

Practical control systems often have constraints due to safe operating conditions or limitations of the controller/actuator. These constraints include limited input values, states, and outputs within a certain range or states following a specific trajectory. Designing sparse control with these additional constraints is another interesting research avenue.

6.4 Extensions to Other Control Models

While the state-of-the-art research looks only at centralized sparse actuator control, but it can be extended to other models.

Decentralized or distributed control

One of the motivations for sparse actuator control is the difficulty of optimizing dense control in large systems with geographically separated actuators. Decentralized or distributed control strategies can be useful but often come at the cost of control performance. Designing sparse actuator schedules offline can lead to smaller dimension control input design problems, making sparse actuation a compromise between centralized and decentralized control. This approach combines the good control performance of centralized control while being computationally efficient like decentralized control. Choosing the right sparsity constraints and balancing the two approaches is an upcoming research opportunity.

Other types of control systems

Some immediate extensions of the sparse actuator model are to look at non-linear systems, structured systems, or stochastic systems. We can look at whether it is possible to control such systems with sparsity constraints and develop algorithms for designing such control.

Further, compressed sensing with adaptive measurements is a well-studied research area (Malloy and Nowak, 2014; Zhang *et al.*, 2017; Nakos *et al.*, 2018). Drifting inspirations from the area, we can develop new algorithms for adaptive control, particularly useful for stochastic systems. Similarly, dictionary learning (Kreutz-Delgado *et al.*, 2003; Rubinstein *et al.*, 2012; Joseph and Murthy, 2020) is yet another compressed sensing-related area where the goal is to learn both measurement matrix and sparse vectors from a set of linear measurements. We can adapt these algorithms for joint system identification and sparse control design. There is also an increasing interest in joint actuator-sensor design, which can also be combined with sparsity.

6.5 Real-world Applications

The existing work thus far is limited to mathematical results and simulation results using toy examples. Although the research on sparse control is motivated by several real-world applications as listed in Section 1.2, an important next step is testing them in real-world settings or practical applications to demonstrate the effectiveness of sparse control. Bridging the gap between theory and real-world application will enable stakeholders to use sparse actuator control as a standard tool of control engineering practice, making a tangible impact on society and contributing to positive change.

A good starting point for the social network application is to rely on several standard simulation models that are available. For example, social and information networks often follow power laws (Stephen and

Toubia, 2009; Muchnik *et al.*, 2013; Csányi and Szendrői, 2004), meaning that a few nodes have many edges, and many nodes have a few edges. Therefore, certain nodes can act as hubs, influencing most of the other nodes. Sparse control can potentially apply nonzero control over those hub nodes and zero control over the other nodes yet control the entire system. Once the algorithms are tested on simulation models, real-world datasets, such as the Stanford large network dataset collection (Leskovec and Sosič, 2016) or Friendster social network (Boyd, 2004; Seki and Nakamura, 2017), can be used.

It is also worthwhile to explore the differences between time-varying and time-invariant support cases in practical applications. Examining whether time-invariant models offer sufficient performance and investigating the scenarios where time-invariant models might fall short could help determine the practicality and effectiveness of the two approaches. This comparison can lead to a deeper understanding of how these models can be optimized and applied across various applications.

6.6 Summary

The motivation for this monograph arises from understanding how using a limited number of actuators can still achieve excellent control performance. The fundamental tools employed in this work are well-established within the electrical engineering research community, particularly in the fields of control and signal processing. However, sparse actuator control is an emerging area that is still in its early stages. Initial findings are promising, and the potential applications are diverse and far-reaching in high-dimensional control.

While this section highlights only a few future directions, it suggests that the entire field of control theory can be reconsidered with a focus on sparsity. This perspective opens up new avenues for research and innovation. The most compelling and impactful results in this area are yet to come, holding great promise for advancements in control systems.

List of Symbols

Vector		
$\|\cdot\|$	:	Euclidean norm of a vector
$\|\cdot\|_0$	:	Number of nonzero entries of a vector
$\|\cdot\|_1$	:	ℓ_1−norm of a vector
$\mathrm{supp}(\cdot)$	:	Support set of a vector

Matrix		
$\boldsymbol{A}_{ij}$	:	(i, j)th entry of the matrix $\boldsymbol{A}$
$\boldsymbol{A}_i$	:	ith column of the matrix $\boldsymbol{A}$
$\boldsymbol{A}_{\mathcal{S}}$	:	Submatrix of $\boldsymbol{A}$ with the columns indexed by $\mathcal{S}$
$(\cdot)^{\mathsf{T}}$	:	Transpose of a matrix
$(\cdot)^{-1}$	:	Inverse of a matrix
$\lvert\cdot\rvert$	:	Determinant of a matrix
$(\cdot)^{\dagger}$	:	Pseudo-inverse of a matrix
$\mathrm{tr}\{\cdot\}$	:	Trace of a matrix
$\mathrm{rank}\{\cdot\}$	:	Rank of a matrix
$\|\cdot\|$	:	Spectral norm of a matrix
$\lambda_{\max}(\cdot)$	:	Largest eigenvalue of a matrix
$\lambda_{\min}(\cdot)$	:	Smallest eigenvalue of a matrix
$\mathrm{ColSpace}\{\cdot\}$	:	Column space of a matrix

Field

$\mathbb{R}$	:	Field of real numbers
$\mathbb{R}_+$	:	Field of non-negative real numbers
$\mathbb{C}$	:	Field of complex numbers

Set

$\lvert\cdot\rvert$	:	Cardinality of a set
$(\cdot)^{\complement}$	:	Complement of a set
$\cup$	:	Union of two sets
$\cap$	:	Intersection of two sets

Miscellaneous

$\mathbf{0}$	:	All zero vector or matrix
$\boldsymbol{I}$	:	Identity matrix
$\emptyset$	:	Empty set

References

Abbas, W., M. Shabbir, Y. Yazıcıoğlu, and X. Koutsoukos. (2023). "On zero forcing sets and network controllability–computation and edge augmentation". *IEEE Transactions on Control of Network Systems.*

Anderson, B. and J. Moore. (2005). *Optimal filtering.* Courier Dover.

Anguluri, R., O. Kosut, and L. Sankar. (2023). "Localization and estimation of unknown forced inputs: A group LASSO approach". *IEEE Transactions on Control of Network Systems.*

Anguluri, R., N. Taghipourbazargani, O. Kosut, and L. Sankar. (2022). "A complex-LASSO approach for localizing forced oscillations in power systems". In: *Proceedings of IEEE Power & Energy Society General Meeting.* IEEE. 01–05.

Arastoo, R., Y. GhaedSharaf, M. V. Kothare, and N. Motee. (2016). "Optimal state feedback controllers with strict row sparsity constraints". In: *Proceedings of American Control Conference.* IEEE. 1948–1953.

Argha, A., S. W. Su, A. Savkin, and B. Celler. (2019). "A framework for optimal actuator/sensor selection in a control system". *International Journal of Control.* 92(2): 242–260.

Athans, M. (1972). "On the determination of optimal costly measurement strategies for linear stochastic systems". *Automatica.* 8(4): 397–412.

Babazadeh, M. and A. Nobakhti. (2016). “Sparsity promotion in state feedback controller design”. *IEEE Transactions on Automatic Control*. 62(8): 4066–4072.

Ballotta, L., L. Schenato, and L. Carlone. (2020). “Computation-communication trade-offs and sensor selection in real-time estimation for processing networks”. *IEEE Transactions on Network Science and Engineering*. 7(4): 2952–2965.

Baraniuk, R. G. (2007). “Compressive sensing [lecture notes]”. *IEEE Signal Processing Magazine*. 24(4): 118–121.

Bora, A., A. Jalal, E. Price, and A. G. Dimakis. (2017). “Compressed sensing using generative models”. In: *Proceedings of International Conference on Machine Learning*. PMLR. 537–546.

Boyd, D. M. (2004). “Friendster and publicly articulated social networking”. In: *Proceedings of CHI’04 extended abstracts on Human factors in computing systems*. 1279–1282.

Brockwell, P. J., R. A. Davis, and M. V. Calder. (2002). *Introduction to time series and forecasting*. Vol. 2. Springer.

Bruckstein, A. M., M. Elad, and M. Zibulevsky. (2008). “On the uniqueness of nonnegative sparse solutions to underdetermined systems of equations”. *IEEE Transactions on Information Theory*. 54(11): 4813–4820.

Candès, E. J., J. Romberg, and T. Tao. (2006). “Robust uncertainty principles: Exact signal reconstruction from highly incomplete frequency information”. *IEEE Transactions on information theory*. 52(2): 489–509.

Cárdenas, A. A., S. Amin, and S. Sastry. (2008). “Research challenges for the security of control systems”. *HotSec*. 5(15): 1158.

Chakraborty, R. K., G. Joseph, and C. R. Murthy. (2024). “Bayesian learning-based Kalman smoothing For linear dynamical systems with unknown sparse inputs”. In: *Proceedings of IEEE International Conference on Acoustics, Speech and Signal Processing*. IEEE. 13431–13435.

Chamon, L. F., G. J. Pappas, and A. Ribeiro. (2017). “The mean square error in Kalman filtering sensor selection is approximately supermodular”. In: *Proceedings of IEEE Conference on Decision and Control*. IEEE. 343–350.

Chapman, A. and M. Mesbahi. (2013). “On strong structural controllability of networked systems: A constrained matching approach”. In: *Proceedings of American Control Conference*. IEEE. 6126–6131.

Chatterjee, D., M. Nagahara, D. E. Quevedo, and K. M. Rao. (2016). “Characterization of maximum hands-off control”. *Systems & Control Letters*. 94: 31–36.

Chen, C.-T. (1984). *Linear system theory and design*. Saunders College publishing.

Chen, R., X. Li, H. Zhong, and M. Fei. (2019). “A novel online detection method of data injection attack against dynamic state estimation in smart grid”. *Neurocomputing*. 344: 73–81.

Csányi, G. and B. Szendrői. (2004). “Structure of a large social network”. *Physical Review E*. 69(3): 036131.

Dai, W. and S. Yüksel. (2013). “Observability of a linear system under sparsity constraints”. *IEEE Transactions on Automatic Control*. 58(9): 2372–2376.

Dhingra, N. K., M. R. Jovanović, and Z.-Q. Luo. (2014). “An ADMM algorithm for optimal sensor and actuator selection”. In: *Proceedings of IEEE Conference on Decision and Control*. IEEE. 4039–4044.

Donoho, D. L. (2006). “Compressed sensing”. *IEEE Transactions on Information Theory*. 52(4): 1289–1306.

Donoho, D. L., A. Maleki, and A. Montanari. (2009). “Message-passing algorithms for compressed sensing”. *Proceedings of the National Academy of Sciences*. 106(45): 18914–18919.

Eldar, Y. C., P. Kuppinger, and H. Bolcskei. (2010). “Block-sparse signals: Uncertainty relations and efficient recovery”. *IEEE Transactions on Signal Processing*. 58(6): 3042–3054.

Fardad, M. and M. R. Jovanović. (2014). “On the design of optimal structured and sparse feedback gains via sequential convex programming”. In: *Proceedings of American Control Conference*. IEEE. 2426–2431.

Fornasier, M. and H. Rauhut. (2015). “Compressive sensing”. *Handbook of Mathematical Methods in Imaging*. 1: 187–229.

Fosson, S. M., F. Garin, S. Gracy, A. Y. Kibangou, and D. Swart. (2019). “Input and state estimation exploiting input sparsity”. In: *Proceedings of European Control Conference*. IEEE. 2344–2349.

Foucart, S., H. Rauhut, S. Foucart, and H. Rauhut. (2013). *An invitation to compressive sensing*. Springer.

Hao, J., R. J. Piechocki, D. Kaleshi, W. H. Chin, and Z. Fan. (2015). “Sparse malicious false data injection attacks and defense mechanisms in smart grids”. *IEEE Transactions on Industrial Informatics*. 11(5): 1–12.

Hashemi, A., M. Ghasemi, H. Vikalo, and U. Topcu. (2020). “Randomized greedy sensor selection: Leveraging weak submodularity”. *IEEE Transactions on Automatic Control*. 66(1): 199–212.

Heemels, W. M. H., A. R. Teel, N. Van de Wouw, and D. Nešić. (2010). “Networked control systems with communication constraints: Tradeoffs between transmission intervals, delays and performance”. *IEEE Transactions on Automatic Control*. 55(8): 1781–1796.

Hespanha, J. P. (2018). *Linear systems theory*. Princeton University Press.

Huang, J. and T. Zhang. (2010). “The benefit of group sparsity”. *The Annals of Statistics*. 38(4): 1978–2004.

Hvistendahl, M., D. Normile, and J. Cohen. (2013). “Despite large research effort, H7N9 continues to baffle”. *Science*. 340(6131): 414–415.

Ikeda, T. (2024). “Non-convex optimization problems for maximum hands-off control”. *arXiv preprint arXiv:2402.10402*.

Ikeda, T. and K. Kashima. (2018). “On sparse optimal control for general linear systems”. *IEEE Transactions on Automatic Control*. 64(5): 2077–2083.

Ito, K., T. Ikeda, and K. Kashima. (2021). “Sparse optimal stochastic control”. *Automatica*. 125: 109438.

Joseph, G. (2022a). “Controllability of a linear system with nonnegative sparse controls”. *IEEE Transactions on Automatic Control*. 67(1): 468–473.

Joseph, G. (2022b). “Output controllability of a linear dynamical system with sparse controls”. *IEEE Transactions on Control of Network Systems*. 10(1): 147–156.

Joseph, G., S. Moothedath, and J. Lin. (2023). “Minimal input structural modifications for strongly structural controllability”. *arXiv preprint arXiv:2311.05653*.

Joseph, G. and C. R. Murthy. (2018). “On the observability of a linear system with a sparse initial state”. *IEEE Signal Processing Letters*. 25(7): 994–998.

Joseph, G. and C. R. Murthy. (2020). “On the convergence of a Bayesian algorithm for joint dictionary learning and sparse recovery”. *IEEE Transactions on Signal Processing*. 68: 343–358.

Joseph, G. and C. R. Murthy. (2021). “Controllability of linear dynamical systems under input sparsity constraints”. *IEEE Transactions on Automatic Control*. 66(2): 924–931.

Joseph, G. and C. R. Murthy. (2019). “Measurement bounds for observability of linear dynamical systems under sparsity constraints”. *IEEE Transactions on Signal Processing*. 67(8): 1992–2006.

Joseph, G., B. Nettasinghe, V. Krishnamurthy, and P. K. Varshney. (2021). “Controllability of network opinion in Erdős–Rényi graphs using sparse control inputs”. *SIAM Journal on Control and Optimization*. 59(3): 2321–2345.

Kafashan, M., A. Nandi, and S. Ching. (2016). “Relating observability and compressed sensing of time-varying signals in recurrent linear networks”. *Neural Networks*. 83: 11–20.

Karp, R. M. (2010). *Reducibility among combinatorial problems*. Springer.

Kolar, M., J. Lafferty, and L. Wasserman. (2011). “Union support recovery in multi-task learning”. *Journal of Machine Learning Research*. 12(7).

Kreutz-Delgado, K., J. F. Murray, B. D. Rao, K. Engan, T.-W. Lee, and T. J. Sejnowski. (2003). “Dictionary learning algorithms for sparse representation”. *Neural computation*. 15(2): 349–396.

Kutyniok, G. (2013). “Theory and applications of compressed sensing”. *GAMM-Mitteilungen*. 36(1): 79–101.

Leskovec, J. and R. Sosič. (2016). “Snap: A general-purpose network analysis and graph-mining library”. *ACM Transactions on Intelligent Systems and Technology*. 8(1): 1–20.

Li, K., C. R. Rojas, T. Yang, H. Hjalmarsson, K. H. Johansson, and S. Cong. (2016). “Piecewise sparse signal recovery via piecewise orthogonal matching pursuit”. In: *Proceedings of IEEE International Conference on Acoustics, Speech and Signal Processing*. IEEE. 4608–4612.

Lin, F., M. Fardad, and M. R. Jovanović. (2013). “Design of optimal sparse feedback gains via the alternating direction method of multipliers”. *IEEE Transactions on Automatic Control*. 58(9): 2426–2431.

Liu, K., E. Fridman, and Y. Xia. (2020). *Networked control under communication constraints*. Springer.

Liu, Y.-Y., J.-J. Slotine, and A.-L. Barabási. (2013). “Observability of complex systems”. *Proceedings of the National Academy of Sciences*. 110(7): 2460–2465.

Ma, R. and P. Shi. (2022). “Secure state estimation for cyber-physical systems under sparse data injection attacks: a switched counteraction approach”. *International Journal of Control*. 95(4): 963–974.

Mai, H. and Z. Yin. (2023). “A new zero-order algorithm to solve the maximum hands-off control”. *IEEE Transactions on Automatic Control*.

Mai, H. and Z. Yin. (2024). “Some mathematical characteristics in sparse control for linear time-varying systems”. *International Journal of Robust and Nonlinear Control*.

Malloy, M. L. and R. D. Nowak. (2014). “Near-optimal adaptive compressed sensing”. *IEEE Transactions on Information Theory*. 60(7): 4001–4012.

Manohar, K., J. N. Kutz, and S. L. Brunton. (2021). “Optimal sensor and actuator selection using balanced model reduction”. *IEEE Transactions on Automatic Control*. 67(4): 2108–2115.

Marucci, L., D. A. Barton, I. Cantone, M. A. Ricci, M. P. Cosma, S. Santini, D. di Bernardo, and M. di Bernardo. (2009). “How to turn a genetic circuit into a synthetic tunable oscillator, or a bistable switch”. *PloS one*. 4(12): e8083.

Mousavi, A., A. B. Patel, and R. G. Baraniuk. (2015). “A deep learning approach to structured signal recovery”. In: *Proceedings of Allerton Conference on Communication, Control, and Computing*. IEEE. 1336–1343.

Muchnik, L., S. Pei, L. C. Parra, S. D. Reis, J. S. Andrade Jr, S. Havlin, and H. A. Makse. (2013). “Origins of power-law degree distribution in the heterogeneity of human activity in social networks”. *Scientific Reports*. 3(1): 1783.

Nagahara, M. (2020). *Sparsity methods for systems and control*. now Publishers.

Nagahara, M. (2023). “Sparse control for continuous-time systems”. *International Journal of Robust and Nonlinear Control*. 33(1): 6–22.

Nagahara, M., J. Østergaard, and D. E. Quevedo. (2016). “Discrete-time hands-off control by sparse optimization”. *EURASIP Journal on Advances in Signal Processing*. 2016: 1–8.

Nagahara, M. and D. E. Quevedo. (2011). “Sparse representations for packetized predictive networked control”. *IFAC Proceedings Volumes*. 44(1): 84–89.

Nagahara, M., D. E. Quevedo, and D. Nešić. (2015). “Maximum hands-off control: a paradigm of control effort minimization”. *IEEE Transactions on Automatic Control*. 61(3): 735–747.

Nakos, V., X. Shi, D. P. Woodruff, and H. Zhang. (2018). “Improved algorithms for adaptive compressed sensing”. *arXiv preprint arXiv:1804.09673*.

Nemhauser, G. L., L. A. Wolsey, and M. L. Fisher. (1978). “An analysis of approximations for maximizing submodular set functions-I”. *Mathematical Programming*. 14: 265–294.

Neumann, G., T. Noda, and Y. Kawaoka. (2009). “Emergence and pandemic potential of swine-origin H1N1 influenza virus”. *Nature*. 459(7249): 931–939.

Nguyen, T. T., J. Idier, C. Soussen, and E.-H. Djermoune. (2019). “Non-negative orthogonal greedy algorithms”. *IEEE Transactions on Signal Processing*. 67(21): 5643–5658.

Olshevsky, A. (2017). “On (non) supermodularity of average control energy”. *IEEE Transactions on Control of Network Systems*. 5(3): 1177–1181.

Polyak, B. T., M. V. Khlebnikov, and P. S. Shcherbakov. (2014). "Sparse feedback in linear control systems". *Automation and Remote Control*. 75: 2099–2111.

Pope III, C. A., R. T. Burnett, M. J. Thun, E. E. Calle, D. Krewski, K. Ito, and G. D. Thurston. (2002). "Lung cancer, cardiopulmonary mortality, and long-term exposure to fine particulate air pollution". *Journal of the American Medical Association*. 287(9): 1132–1141.

Prasad, R., C. Murthy, and B. Rao. (2014). "Joint approximately sparse channel estimation and data detection in OFDM systems using sparse Bayesian learning". *IEEE Transactions on Signal Processing*. 62(14): 3591–3603.

Rajapakse, I., M. Groudine, and M. Mesbahi. (2012). "What can systems theory of networks offer to biology?" *PLoS computational biology*. 8(6): e1002543.

Rubinstein, R., T. Peleg, and M. Elad. (2012). "Analysis K-SVD: A dictionary-learning algorithm for the analysis sparse model". *IEEE Transactions on Signal Processing*. 61(3): 661–677.

Ruckman, C. E. and C. R. Fuller. (1995). "Optimizing actuator locations in active noise control systems using subset selection". *Journal of Sound and Vibration*. 186(3): 395–406.

Sanandaji, B. M., M. B. Wakin, and T. L. Vincent. (2014). "Observability with random observations". *IEEE Transactions on Automatic Control*. 59(11): 3002–3007.

Sefati, S., N. J. Cowan, and R. Vidal. (2015). "Linear systems with sparse inputs: Observability and input recovery". In: *Proceedings of American Control Conference*. IEEE. 5251–5257.

Seki, K. and M. Nakamura. (2017). "The mechanism of collapse of the Friendster network: what can we learn from the core structure of Friendster:?" *Social Network Analysis and Mining*. 7: 1–21.

Shao, M., X. Tang, Y. Zhang, and W. Li. (2006). "City clusters in China: air and surface water pollution". *Frontiers in Ecology and the Environment*. 4(7): 353–361.

Siami, M., A. Olshevsky, and A. Jadbabaie. (2020). "Deterministic and randomized actuator scheduling with guaranteed performance bounds". *IEEE Transactions on Automatic Control*. 66(4): 1686–1701.

Silva, V. L., L. F. Chamon, and A. Ribeiro. (2019). "Model predictive selection: A receding horizon scheme for actuator selection". In: *Proceedings of American Control Conference*. IEEE. 347–353.

Skafidas, E., R. J. Evans, A. V. Savkin, and I. R. Petersen. (1999). "Stability results for switched controller systems". *Automatica*. 35(4): 553–564.

Sriram, C., G. Joseph, and C. R. Murthy. (2022). "Stabilizability of linear dynamical systems using sparse control inputs". *IEEE Transactions on Automatic Control*.

Stephen, A. T. and O. Toubia. (2009). "Explaining the power-law degree distribution in a social commerce network". *Social Networks*. 31(4): 262–270.

Summers, T. and M. Kamgarpour. (2019). "Performance guarantees for greedy maximization of non-submodular controllability metrics". In: *Proceedings of European Control Conference*. IEEE. 2796–2801.

Sun, H. and J. Zhao. (2001). "Control Lyapunov functions for switched control systems". In: *Proceedings of American Control Conference*. Vol. 3. IEEE. 1890–1891.

Sun, K. and Z. Li. (2022). "Sparse data injection attacks on smart grid: An information-theoretic approach". *IEEE Sensors Journal*. 22(14): 14553–14562.

Sun, Z. and S. S. Ge. (2005). "Analysis and synthesis of switched linear control systems". *Automatica*. 41(2): 181–195.

Swirszcz, G., N. Abe, and A. C. Lozano. (2009). "Grouped orthogonal matching pursuit for variable selection and prediction". *Advances in Neural Information Processing Systems*. 22.

Taha, A. F., N. Gatsis, T. Summers, and S. Nugroho. (2017). "Actuator selection for cyber-physical systems". In: *Proceedings of American Control Conference*. IEEE. 5300–5305.

Tatikonda, S. and S. Mitter. (2004). "Control under communication constraints". *IEEE Transactions on Automatic Control*. 49(7): 1056–1068.

Trefois, M. and J.-C. Delvenne. (2015). "Zero forcing number, constrained matchings and strong structural controllability". *Linear Algebra and its Applications*. 484: 199–218.

Tsang, K. F. E., M. Huang, K. H. Johansson, and L. Shi. (2020). "Sparse linear injection attack on multi-agent consensus control systems". *IEEE Control Systems Letters*. 5(2): 665–670.

Vafaee, R. and M. Siami. (2024). "Real-time sensor selection for time-varying networks with guaranteed performance". *Automatica*. 163: 111550.

Wakin, M. B. *et al.* (2008). "An introduction to compressive sampling". *IEEE Signal Processing Magazine*. 25(2): 21–30.

Wipf, D. P. and B. D. Rao. (2004). "Sparse Bayesian learning for basis selection". *IEEE Transactions on Signal processing*. 52(8): 2153–2164.

Wu, L., Y. Yang, and H. Liu. (2014). "Nonnegative-LASSO and application in index tracking". *Computational Statistics & Data Analysis*. 70: 116–126.

Ye, L., M. Chi, Z.-W. Liu, and V. Gupta. (2022). "Online actuator selection and controller design for linear quadratic regulation with unknown system model". *arXiv preprint arXiv:2201.10197*.

Yoshida, H., Y. Tanada, T. Tanaka, and K. Yunokuchi. (1994). "Controllability of multiple input discrete-time linear systems with positive controls". *Transactions of the Society of Instrument and Control Engineers*. 30(3): 243–252.

Yuan, M. and Y. Lin. (2006). "Model selection and estimation in regression with grouped variables". *Journal of the Royal Statistical Society Series B: Statistical Methodology*. 68(1): 49–67.

Zhang, J., Q. Xiang, Y. Yin, C. Chen, and X. Luo. (2017). "Adaptive compressed sensing for wireless image sensor networks". *Multimedia Tools and Applications*. 76: 4227–4242.

Zhang, X., W. Xu, Y. Cui, L. Lu, and J. Lin. (2019). "On recovery of block sparse signals via block compressive sampling matching pursuit". *IEEE Access*. 7: 175554–175563.

Zhong, J. and C. Li. (2018). "Piecewise sparse recovery via piecewise greedy method". *Journal of Mathematical Research with Applications*. 6: 643–658.

Zhou, K., J. C. Doyle, and K. Glover. (1996). *Robust and optimal control*. Vol. 40. Prentice Hall New Jersey.