Multilinear Control Systems Theory and its Applications

by

Can Chen

A dissertation submitted in partial fulfillment
of the requirements for the degree of
Doctor of Philosophy
(Applied and Interdisciplinary Mathematics)
in The University of Michigan
2021

Doctoral Committee:

Professor Anthony M. Bloch, Co-Chair
Associate Professor Indika Rajapakse, Co-Chair
Professor Daniel Burns
Professor Lindsey Muir
Doctor Amit Surana

Can Chen

canc@umich.edu

ORCID ID: 0000-0003-2310-0074

To my family

ACKNOWLEDGEMENTS

I would like to thank my two advisors, Anthony M. Bloch and Indika Rajapakse, for their continued support and encouragement throughout the years. I would also like to thank Amit Surana for his scientific guidance during my early graduate career.

I owe a deep debt of gratitude to the Rajapakse lab members, former and present, for our scientific discussions that guided my work: Stephen Lindsly, Samuel Dilworth, Gabrielle Dotson, Walter Meixner, Sivakumar Jeyarajan, Christopher York, Scott Ronquist, Emily Crosette, and Sijia Liu.

I would like to thank the Math graduate program coordinators and staff for keeping me on track to graduate. I would also like to thank Anthony M. Bloch, Indika Rajapakse, Daniel Burns, Lindsey Muir, and Amit Surana for serving on my thesis committee.

TABLE OF CONTENTS

LIST OF FIGURES

Figure

LIST OF TABLES

ABSTRACT

In biological and engineering systems, structure, function, and dynamics are highly coupled. Such multiway interactions can be naturally and compactly captured via tensor-based representations. Exploiting recent advances in tensor algebraic methods, we develop novel theoretical and computational approaches for data-driven model learning, analysis, and control of such tensor-based representations. In one line of work, we extend classical linear time-invariant (LTI) system notions including stability, reachability, and observability to multilinear time-invariant (MLTI) systems, in which the state, inputs, and outputs are represented as tensors, and express these notions in terms of more standard concepts of tensor ranks/decompositions. We also introduce a tensor decomposition-based model reduction framework which can significantly reduce the number of MLTI system parameters. In another line of work, we develop the notion of tensor entropy for uniform hypergraphs, which can capture higher order interactions between entities than classical graphs. We show that this tensor entropy is an extension of von Neumann entropy for graphs and can be used as a measure of regularity for uniform hypergraphs. Moreover, we employ uniform hypergraphs for studying controllability of high-dimensional networked systems. We propose another tensor-based multilinear system representation for characterizing the multidimensional state dynamics of uniform hypergraphs, and derive a Kalman-rank-like condition to identify the minimum number of control nodes (MCN) needed to achieve full control of the whole hypergraph. We demonstrate

these new tensor-based theoretical and computational developments in a variety of biological and engineering examples.

CHAPTER I

Preliminaries

Controlling high-dimensional systems remains an extremely challenging task as many control strategies do not scale well with the dimension of the systems. Of particular interest in this thesis are complex biological and engineering systems in which structure, function, and dynamics are highly coupled. Such interactions can be naturally and compactly captured by tensors. Tensors are multidimensional arrays generalized from vectors and matrices, and have wide applications in many domains such as social networks, biology, cognitive science, applied mechanics, scientific computation, and signal processing [28, 41, 50, 66, 70]. For example, the organization of the interphase nucleus in the human genome reflects a dynamical interaction between 3D genome structure, function, and its relationship to phenotype, a concept known as the 4D Nucleome (4DN) [28]. 4DN research requires a comprehensive view of genome-wide structure, gene expression, the proteome, and phenotype, which fits naturally with a tensorial representation [103, 120]. In order to apply the standard system and control framework in applications such as these, tensors need to be vectorized, leading to an extremely high-dimensional system representation in which the number of states/parameters scale exponentially with the number of dimensions of the tensors involved [120]. With the vectorization of tensors, hidden patterns/struc-

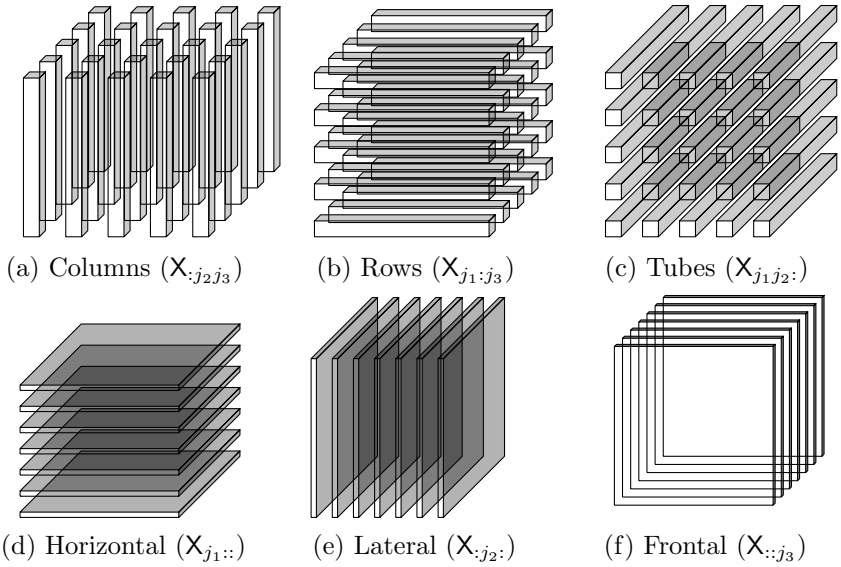
tures, e.g., redundancy/correlations, can get lost, and thus one cannot exploit such inherent structures for efficient representations and computations.

Moreover, most real world data representations are multidimensional, and using graph (or even multilayer graph) models to describe them may result in a loss of higher-order information [23, 129]. A hypergraph is a generalization of a graph in which its hyperedges can join any number of nodes [10, 33, 46, 123]. Thus, hypergraphs can capture multidimensional relationships unambiguously [129]. Examples of hypergraphs include co-authorship networks, film actor/actress networks, and protein-protein interaction networks [89]. In addition, a hypergraph can be represented by a tensor if its hyperedges contain the same number of nodes, referred to as a uniform hypergraph [23]. The dynamics of uniform hypergraphs thus can be naturally described by a tensor-based multilinear system.

In this chapter, we provide detailed literature reviews on tensors and hypergraphs which are the main objects of this thesis.

1.1 Tensors

We take most of the concepts and notations for tensor algebra from the comprehensive works of Kolda *et al.* [69, 70] and Ragnarsson *et al.* [99, 100]. A *tensor* is a multidimensional array. The *order* of a tensor is the number of its dimensions, and each dimension is called a *mode*. An kth order tensor usually is denoted by $\mathsf{X} \in \mathbb{R}^{n_1 \times n_2 \times \cdots \times n_k}$. The sets of indexed indices and size of X are denoted by $\mathbf{j} = \{j_1, j_2, \ldots, j_k\}$ and $\mathcal{N} = \{n_1, n_2, \ldots, n_k\}$, respectively. Let $\Pi_{\mathcal{N}}$ represent the product of all elements in $\mathcal{N}$, i.e., $\Pi_{\mathcal{N}} = \prod_{p=1}^{k} n_p$. It is therefore reasonable to consider scalars $x \in \mathbb{R}$ as zero-order tensors, vectors $\mathbf{v} \in \mathbb{R}^n$ as first-order tensors, and matrices $\mathbf{A} \in \mathbb{R}^{n \times m}$ as second-order tensors. For a third-order tensor, *fibers* are



(a) Columns ($\mathsf{X}_{:j_2j_3}$) (b) Rows ($\mathsf{X}_{j_1:j_3}$) (c) Tubes ($\mathsf{X}_{j_1j_2:}$)

(d) Horizontal ($\mathsf{X}_{j_1::}$) (e) Lateral ($\mathsf{X}_{:j_2:}$) (f) Frontal ($\mathsf{X}_{::j_3}$)

Figure 1.1: **Fibers and slices of a third-order tensor.** The figure is adapted from [69].

named as *column* ($\mathsf{X}_{:j_2j_3}$), *row* ($\mathsf{X}_{j_1:j_3}$), and *tube* ($\mathsf{X}_{j_1j_2:}$), while *slices* are named as *horizontal* ($\mathsf{X}_{j_1::}$), *lateral* ($\mathsf{X}_{:j_2:}$), and *frontal* ($\mathsf{X}_{::j_3}$), see Figure 1.1. A tensor is called *cubical* if every mode is the same size, i.e., $\mathsf{X} \in \mathbb{R}^{n\times n\times\cdots\times n}$. A cubical tensor X is called *supersymmetric* if $\mathsf{X}_{j_1j_2\ldots j_k}$ is invariant under any permutation of the indices, and is called *diagonal* if $\mathsf{X}_{j_1j_2\ldots j_k} = 0$ except $j_1 = j_2 = \cdots = j_k$.

1.1.1 Tensor Products

By extending the notion of vector outer product, the *outer product* of two tensors $\mathsf{X} \in \mathbb{R}^{n_1\times n_2\times\cdots\times n_k}$ and $\mathsf{Y} \in \mathbb{R}^{m_1\times m_2\times\cdots\times m_s}$ is defined as

$$(\mathsf{X} \circ \mathsf{Y})_{j_1j_2\ldots j_ki_1i_2\ldots i_s} = \mathsf{X}_{j_1j_2\ldots j_k}\mathsf{Y}_{i_1i_2\ldots i_s}.$$

In contrast, the *inner product* of two tensors $\mathsf{X}, \mathsf{Y} \in \mathbb{R}^{n_1\times n_2\times\cdots\times n_k}$ is defined as

$$\langle \mathsf{X}, \mathsf{Y}\rangle = \sum_{j_1=1}^{n_1}\cdots\sum_{j_k=1}^{n_k}\mathsf{X}_{j_1j_2\ldots j_k}\mathsf{Y}_{j_1j_2\ldots j_k},$$

leading to the tensor Frobenius norm $\|\mathsf{X}\|^2 = \langle \mathsf{X}, \mathsf{X}\rangle$. We say two tensors X and Y are *orthogonal* if the inner product $\langle \mathsf{X}, \mathsf{Y}\rangle = 0$.

The *tensor vector multiplication* $\mathsf{X} \times_p \mathbf{v}$ along mode p for a vector $\mathbf{v} \in \mathbb{R}^{n_p}$ is defined by

$$(\mathsf{X} \times_p \mathbf{v})_{j_1 j_2 \dots j_{p-1} j_{p+1} \dots j_k} = \sum_{j_p=1}^{n_p} \mathsf{X}_{j_1 j_2 \dots j_p \dots j_k} \mathbf{v}_{j_p}, \tag{1.1}$$

which can be extended to

$$\mathsf{X} \times_1 \mathbf{v}_1 \times_2 \mathbf{v}_2 \times_3 \cdots \times_k \mathbf{v}_k = \mathsf{X}\mathbf{v}_1\mathbf{v}_2 \dots \mathbf{v}_k \in \mathbb{R} \tag{1.2}$$

for $\mathbf{v}_p \in \mathbb{R}^{n_p}$. The expression (1.2) is also known as the homogeneous polynomial associated with X. If $\mathbf{v}_p = \mathbf{v}$ for all p, we write (1.2) as $\mathsf{X}\mathbf{v}^k$ for simplicity. Similarly, the *matrix tensor multiplication* $\mathsf{X} \times_p \mathbf{A}$ along mode p for a matrix $\mathbf{A} \in \mathbb{R}^{m \times n_p}$ is defined by

$$(\mathsf{X} \times_p \mathbf{A})_{j_1 j_2 \dots j_{p-1} i j_{p+1} \dots j_k} = \sum_{j_p=1}^{n_p} \mathsf{X}_{j_1 j_2 \dots j_p \dots j_k} \mathbf{A}_{i j_p}. \tag{1.3}$$

This product can be generalized to what is known as the *Tucker product*, for $\mathbf{A}_p \in \mathbb{R}^{m_p \times n_p}$,

$$\mathsf{X} \times_1 \mathbf{A}_1 \times_2 \mathbf{A}_2 \times_3 \cdots \times_k \mathbf{A}_k = \mathsf{X} \times \{\mathbf{A}_1, \mathbf{A}_2, \dots, \mathbf{A}_k\} \in \mathbb{R}^{m_1 \times m_2 \times \cdots \times m_k}. \tag{1.4}$$

1.1.2 Tensor Unfolding

Tensor unfolding is considered as a critical operation in tensor computations [69, 70, 99]. In order to unfold a tensor $\mathsf{X} \in \mathbb{R}^{n_1 \times n_2 \times \cdots \times n_k}$ into a vector or a matrix, we use an index mapping function $ivec(\cdot, \mathcal{N}) : \mathbb{Z}^+ \times \mathbb{Z}^+ \times \overset{k}{\cdots} \times \mathbb{Z}^+ \to \mathbb{Z}^+$ as defined by Ragnarsson *et al.* [99, 100], which is given by

$$ivec(\mathbf{j}, \mathcal{N}) = j_1 + \sum_{p=2}^{k} (j_p - 1) \prod_{l=1}^{p-1} n_l.$$

The index mapping function *ivec* returns the index for tensor vectorization, i.e., $\mathbf{x} \in \mathbb{R}^{\Pi_{\mathcal{N}}}$ is the vectorization of X such that $\mathbf{x}_{ivec(\mathbf{j}, \mathcal{N})} = \mathsf{X}_{j_1 j_2 \dots j_N}$. If $k = 2$, *ivec* will stack all the columns of X.

For tensor matricization, let z be an integer such that $1 \leq z < k$, and $\mathbb{S}$ be a permutation of $\{1, 2, \ldots, k\}$. If $\mathbf{r} = \{\mathbb{S}(1), \mathbb{S}(2), \ldots, \mathbb{S}(z)\}$ and $\mathbf{c} = \{\mathbb{S}(z+1), \mathbb{S}(z+2), \ldots, \mathbb{S}(k)\}$ with $\mathcal{P} = \{n_{\mathbb{S}(1)}, n_{\mathbb{S}(2)}, \ldots, n_{\mathbb{S}(z)}\}$ and $\mathcal{Q} = \{n_{\mathbb{S}(z+1)}, n_{\mathbb{S}(z+2)}, \ldots, n_{\mathbb{S}(k)}\}$, respectively, the **rc**-unfolding matrix of X, denoted by $\mathbf{X}_{(\mathbf{rc})} \in \mathbb{R}^{\Pi_{\mathcal{P}} \times \Pi_{\mathcal{Q}}}$, is given by

$$(\mathbf{X}_{(\mathbf{rc})})_{pq} = \mathsf{X}^{\mathbb{S}}_{p_1 p_2 \ldots p_z q_1 q_2 \ldots q_{k-z}}, \tag{1.5}$$

where $p = ivec(\mathbf{p}, \mathcal{P})$, $q = ivec(\mathbf{q}, \mathcal{Q})$, and $\mathsf{X}^{\mathbb{S}}$ is the $\mathbb{S}$-*transpose* of X defined as

$$\mathsf{X}^{\mathbb{S}}_{j_{\mathbb{S}(1)} j_{\mathbb{S}(2)} \cdots j_{\mathbb{S}(k)}} = \mathsf{X}_{j_1 j_2 \ldots j_k}.$$

When $z = 1$ and $\mathbb{S} = \begin{pmatrix} 1 & 2 & \ldots & p & p+1 & \ldots & k \\ p & 1 & \ldots & p-1 & p+1 & \ldots & k \end{pmatrix}$, the tensor unfolding is called the *p-mode matricization*, denoted by $\mathbf{X}_{(p)}$.

1.1.3 Tensor Decompositions

There are several definitions of tensor ranks [40, 69, 70], which are intimately related to different notions of tensor decompositions.

Higher-Order Singular Value Decomposition

The *multilinear ranks* or the *p-ranks* of X are the ranks of the p-mode matricizations, denoted by $\text{rank}_p(\mathsf{X})$. The multilinear ranks are related to the so-called higher-order singular value decomposition (HOSVD), a multilinear generalization of the matrix singular value decomposition (SVD) [11, 39].

Theorem I.1 (HOSVD). *A tensor $\mathcal{X} \in \mathbb{R}^{n_1 \times n_2 \times \cdots \times n_k}$ can be written as*

$$\mathcal{X} = \mathcal{S} \times_1 \boldsymbol{U}_1 \times_2 \boldsymbol{U}_2 \times_3 \cdots \times_k \boldsymbol{U}_k, \tag{1.6}$$

where $\boldsymbol{U}_p \in \mathbb{R}^{n_p \times n_p}$ are orthogonal matrices, and $\mathcal{S} \in \mathbb{R}^{n_1 \times n_2 \times \cdots \times n_k}$ is a tensor of which the subtensors $\mathcal{S}_{j_p = \alpha}$, obtained by fixing the pth index to α, have the properties:

- *all-orthogonality: two subtensors* $\mathsf{S}_{j_p=\alpha}$ *and* $\mathsf{S}_{j_p=\beta}$ *are orthogonal for all possible values of* p, α *and* β *subject to* $\alpha \neq \beta$;
- *ordering:* $\|\mathsf{S}_{j_p=1}\| \geq \cdots \geq \|\mathsf{S}_{j_p=n_p}\| \geq 0$ *for all possible values of* p.

The Frobenius norms $\|\mathsf{S}_{j_p=j}\|$, *denoted by* $\gamma_j^{(p)}$, *are the* p*-mode singular values of* X.

De Lathauwer *et al.* [39] showed that the number of nonvanishing p-mode singular values from the HOSVD of a tensor is equal to its p-mode multilinear rank. In addition, the error bound of the low mutilinear rank approximation is provided in [39]. Unlike the matrix SVD, the approximation fails to obtain the best rank approximation of X. Nevertheless, it still can provide a "good" estimate with appropriate p-mode singular values truncated [39].

CANDECOMP/PARAFAC Decomposition

Analogous to rank-one matrices, a tensor $\mathsf{X} \in \mathbb{R}^{n_1 \times n_2 \times \cdots \times n_k}$ is rank-one if it can be written as the outer product of k vectors, i.e., $\mathsf{X} = \mathbf{a}^{(1)} \circ \mathbf{a}^{(2)} \circ \cdots \circ \mathbf{a}^{(k)}$. The CANDECOMP/PARAFAC decomposition (CPD) decomposes a tensor $\mathsf{X} \in \mathbb{R}^{n_1 \times n_2 \times \cdots \times n_k}$ into a sum of rank-one tensors as form of outer products. It is often useful to normalize all the vectors and have weights $\lambda_r > 0$ in descending order in front:

$$\mathsf{X} = \sum_{r=1}^{R} \lambda_r \mathbf{a}_r^{(1)} \circ \mathbf{a}_r^{(2)} \circ \cdots \circ \mathbf{a}_r^{(k)}, \tag{1.7}$$

where $\mathbf{a}_r^{(p)} \in \mathbb{R}^{n_p}$ have unit length, and R is called the *CP rank* of X if it is the minimum integer that achieves (1.7). The factor matrices $\mathbf{A}^{(p)} \in \mathbb{R}^{n_p \times R}$ are the combination of the vectors from the rank-one components for $p = 1, 2, \ldots, k$, i.e., $\mathbf{A}^{(p)} = \begin{bmatrix} \mathbf{a}_1^{(p)} & \mathbf{a}_2^{(p)} & \cdots & \mathbf{a}_R^{(p)} \end{bmatrix}$. The CPD is unique up to scaling and permutation under a weak condition: for $k \geq 2$ and $R \geq 2$, $\sum_{p=1}^{k} \mathcal{K}_{\mathbf{A}^{(p)}} \geq 2R + (k-1)$, where $\mathcal{K}_{\mathbf{A}^{(p)}}$, called the *k-rank* of a matrix, is the maximum number of columns of $\mathbf{A}^{(p)}$ that are linearly independent with each other [73, 113, 117].

The CP rank of a tensor is always greater than or equal to its multilinear ranks [40]. In fact, it is greater than or equal to any unfolding matrix rank [93] (which can be used in TT-ranks and unfolding rank defined later too). The best CP rank approximation is ill-posed [40], but carefully truncating the CP rank will yield a good estimate of the original tensor. Both CPD and HOSVD are special cases of Tucker decomposition, which decomposes a tensor into the form of Tucker product (1.4), i.e., $\mathsf{Y} = \mathsf{X} \times \{\mathbf{A}_1, \mathbf{A}_2, \ldots, \mathbf{A}_k\}$ [72].

Tensor Train Decomposition

The tensor train decomposition (TTD) of a tensor $\mathsf{X} \in \mathbb{R}^{n_1 \times n_2 \times \cdots \times n_k}$ is given by

$$\mathsf{X} = \sum_{r_k=1}^{R_k} \cdots \sum_{r_0=1}^{R_0} \mathsf{X}^{(1)}_{r_0:r_1} \circ \mathsf{X}^{(2)}_{r_1:r_2} \circ \cdots \circ \mathsf{X}^{(k)}_{r_{k-1}:r_k}, \tag{1.8}$$

where $\{R_0, R_1, \ldots, R_k\}$ is the set of *TT-ranks* with $R_0 = R_k = 1$, and $\mathsf{X}^{(p)} \in \mathbb{R}^{R_{p-1} \times n_p \times R_p}$ are called the *core tensors* of the TTD. Here we have used : for brevity of notation, see Appendix .2.1. There exist optimal TT-ranks for the TTD such that

$$R_p = \texttt{rank}(\texttt{reshape}(\mathsf{X}, \prod_{j=1}^{p} n_j, \prod_{j=p+1}^{k} n_j)),$$

for $p = 1, 2, \ldots, k-1$ [91]. A core tensor $\mathsf{X}^{(p)}$ is called *left-orthonormal* if $(\bar{\mathbf{X}}^{(p)})^\top \bar{\mathbf{X}}^{(p)} = \mathbf{I} \in \mathbb{R}^{R_p \times R_p}$, and is called *right-orthonormal* if $\underline{\mathbf{X}}^{(p)} (\underline{\mathbf{X}}^{(p)})^\top = \mathbf{I} \in \mathbb{R}^{R_{p-1} \times R_{p-1}}$ where

$$\bar{\mathbf{X}}^{(p)} = \texttt{reshape}(\mathsf{X}^{(p)}, R_{p-1} n_p, R_p),$$

$$\underline{\mathbf{X}}^{(p)} = \texttt{reshape}(\mathsf{X}^{(p)}, R_{p-1}, n_p R_p),$$

are the *left-* and *right-unfoldings* of the core tensor, respectively. Here $\mathbf{I}$ denotes the identity matrix, and `rank` and `reshape` refer to the rank and reshape operations in MATLAB, respectively (see details in Appendix .2.2 and .2.3.). Detailed algorithms for left- and right-orthonormalization can be found in [68]. TTD is advantageous in

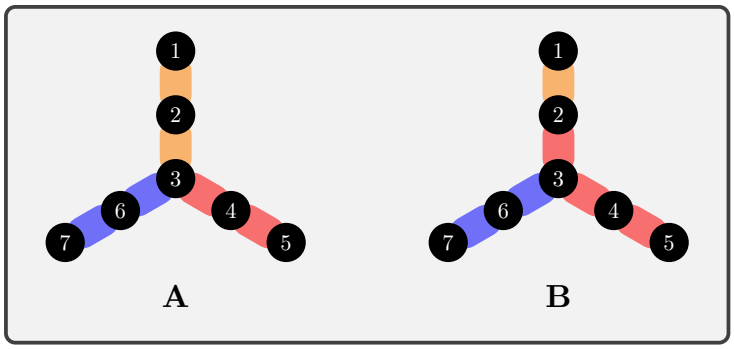


Figure 1.2: **Hypergraphs.** (A) A 3-uniform hypergraph with hyperedges $e_1 = \{1, 2, 3\}$, $e_2 = \{3, 4, 5\}$, and $e_3 = \{3, 6, 7\}$. (B) A non-uniform hypergrah with hyperedges $e_1 = \{1, 2\}$, $e_2 = \{2, 3, 4, 5\}$, and $e_3 = \{3, 6, 7\}$.

that it provides better compression, i.e., truncating the TT-ranks results in a quasi-optimal approximation of X, and is computationally more robust [91]. Moreover, basic tensor algebra, such as addition, tensor products, and norms, can be done in the TT-format without requiring to recover back to the full tensor representation [91].

1.2 Hypergraphs

We borrow some fundamental concepts of hypergraphs from the works [7, 36, 58, 97]. An *undirected hypergraph* G is a pair such that $\mathsf{G} = \{\mathbf{V}, \mathbf{E}\}$ where $\mathbf{V} = \{1, 2, \ldots, n\}$ is the node set and $\mathbf{E} = \{e_1, e_2, \ldots, e_s\}$ is the *hyperedge* set with $e_l \subseteq \mathbf{V}$ for $l = 1, 2, \ldots, s$. Two nodes are called *adjacent* if they are in the same hyperedge. A hypergraph is called *connected* if given two nodes, there is a path connecting them through hyperedges. If all hyperedges contain the same number of nodes, i.e., $|e_p| = k$ ($|\cdot|$ denotes the cardinality of a set), G is called a *k-uniform hypergraph*, see Figure 1.2A. Significantly, every k-uniform hypergraph can be represented by a tensor.

Definition I.2. Let $\mathsf{G} = \{\mathbf{V}, \mathbf{E}\}$ be a k-uniform hypergraph with n nodes. The *adjacency tensor* $\mathsf{A} \in \mathbb{R}^{n \times n \times \cdots \times n}$, which is a kth order n-dimensional supersymmetric

tensor, is defined as

$$\mathsf{A}_{j_1 j_2 \cdots j_k} = \begin{cases} \frac{1}{(k-1)!} \text{ if } (j_1, j_2, \ldots, j_k) \in \mathbf{E} \\ \\ 0, \text{ otherwise} \end{cases} . \tag{1.9}$$

Similarly to standard graphs, the *degree* of node j of a uniform hypergraph is defined as

$$d_j = \sum_{j_2=1}^{n} \sum_{j_3=1}^{n} \cdots \sum_{j_k=1}^{n} \mathsf{A}_{j j_2 j_3 \cdots j_k}. \tag{1.10}$$

Note that the choice of the nonzero coefficient $\frac{1}{(k-1)!}$ in (1.9) guarantees that the degree of each node is equal to the number of hyperedges that contain that node, which is consistent with the notion of degree in standard graphs. The *degree tensor* D of a hypergraph G, associated with A, is a kth order n-dimensional diagonal tensor with $\mathsf{D}_{jj\ldots j}$ equal to (1.10). If $\mathsf{D}_{jj\ldots j} = d$ for all j, then G is called *d-regular*. Given any k nodes, if they are contained in one hyperedge, then G is called *complete*.

Definition I.3. Let $\mathsf{G} = \{\mathbf{V}, \mathbf{E}\}$ be a k-uniform hypergraph with n node. The *Laplacian tensor* $\mathsf{L} \in \mathbb{R}^{n \times n \times \cdots \times n}$ of G, which is a kth order n-dimensional supersymmetric tensor, is defined as

$$\mathsf{L} = \mathsf{D} - \mathsf{A}, \tag{1.11}$$

where D and A are the degree and adjacency tensors of G, respectively.

The Laplacian tensors of uniform hypergraphs possess many similar properties as Laplacian matrices. For example, the smallest *H-eigenvalue* of L is always zero corresponding to the all-one *H-eigenvector* [97]. Moreover, Chen *et al.* [31] showed

that the *Z-eigenvector* associated with the second smallest *Z-eigenvalue* of a normalized Laplacian tensor can be used for hypergraph partition. Detailed descriptions of tensor eigenvalues can be found in Appendix .1.

CHAPTER II

Multilinear Control Systems Theory

In order to take advantage of tensor algebraic computations, recently a new class of multilinear time-invariant (MLTI) system has been introduced [107, 120], in which the states and outputs are preserved as tensors. The system evolution is generated by the action of multilinear operators which are formed using Tucker products of matrices. By using tensor unfolding, an operation that transforms a tensor into a matrix, Rogers *et al.* [107] and Surana *et al.* [120] developed methods for model identification/reduction from tensor time series data. An application of such tensor-based representation/identification for skeleton-based human behavior recognition from videos demonstrated significant improvements in classification accuracy compared to standard linear time-invariant (LTI)-based approaches [41]. However, the MLTI system representation is limited because it assumes the multilinear operators are formed from the Tucker products of matrices (and thus precludes more general tensorial representations) and does not incorporate control inputs.

The role of tensor algebra has also been explored for modeling and simulation of nonlinear dynamics, where the vector field is a multilinear function of states [72]. Tensor decomposition techniques such as CANDECOMP/PARAFAC decomposition (CPD) and tensor train decomposition (TTD) can reduce system size, thus reducing

memory usage and enabling efficient computation during simulations. Note that in contrast to the MLTI systems framework of [107, 120], in this application, tensor algebra is applied to the system represented in the conventional vector form. The author in [50] exploits tensor decompositions to compute numerical solutions of master equations associated with Markov processes on extremely large state spaces. The Einstein product and even-order paired tensors, along with TTD, were utilized for developing tensor representations for operators based on nearest-neighbor interactions, construction of pseudoinverses for dimensionality reduction methods, and the approximation of transfer operators of dynamical systems. The key contributions of this chapter are as follows:

- We propose new tensor notions for positive definiteness, tensor inversion, and a new way of concatenation of tensors to create block tensors. Using these tensor constructs, we develop tensor algebraic conditions for stability, reachability, and observability for the generalized MLTI systems.

- We establish new results relating the unfolding rank to other more standard notions of tensor ranks, including multilinear ranks, CP rank, and TT-ranks. Using such relations, we provide criteria for reachability and observability which do not require tensor unfolding, and can be computed using efficient tensor algebraic methods. Similarly, we express MLTI system stability conditions using higher-order singular value decomposition (HOSVD), CPD, and TTD.

- Using generalized CPD/TTD, we develop a framework for model reduction of MLTI systems. This approach takes advantage of tensor decompositions which otherwise cannot be exploited after unfolding the MLTI systems to obtain a standard LTI form. It also successfully realizes the tensor decomposition-based

criteria for stability, reachability, and observability. Furthermore, we establish new stability results by utilizing the factor matrices from tensor decompositions for this reduced model with lesser computational costs.

- We provide computational and memory complexity analysis for the CPD- and TTD-based methods in comparison to unfolding-based matrix methods and demonstrate our framework in four numerical examples.

This chapter is organized into eight sections. In Section 2.1, we introduce the notion of even-order paired tensors and their properties. In Section 2.2, we build new results relating the unfolding rank of a tensor to other more standard notions of tensor ranks. A new notion of block tensors is discussed in Section 2.3. Section 2.4 introduces the MLTI system representation using the Einstein product and even-order paired tensors in detail, and we establish stability, reachability, and observability conditions for MLTI systems in Section 2.5. The application of generalized CPD/TTD for model reduction is discussed in Section 2.6. Four numerical examples are presented in Section 2.7. Finally, we summarize different numerical approaches associated with MLTI systems in Section 2.8. All the content of this chapter has published in [24, 27].

2.1 Even-Order Paired Tensors

We first discuss the notion of even-order paired tensors and the Einstein product which will play an important role in developing the MLTI systems theory.

Definition II.1. *Even-order paired tensors* are $2k$th order tensors with elements specified using a pairwise index notation, i.e., $\mathsf{A}_{j_1 i_1 \dots j_k i_k}$ for $\mathsf{A} \in \mathbb{R}^{n_1 \times m_1 \times \cdots \times n_k \times m_k}$.

Definition II.2. Given an even-order paired tensor $\mathsf{A} \in \mathbb{R}^{n_1 \times m_1 \times \cdots \times n_k \times m_k}$, the *Einstein product* between A and an kth order tensor $\mathsf{X} \in \mathbb{R}^{m_1 \times m_2 \times \cdots \times m_k}$ is the contraction

along the second index in each pair from A, i.e.,

$$(\mathsf{A} * \mathsf{X})_{j_1 j_2 \ldots j_k} = \sum_{i_1=1}^{m_1} \cdots \sum_{i_k=1}^{m_k} \mathsf{A}_{j_1 i_1 \ldots j_k i_k} \mathsf{X}_{i_1 i_2 \ldots i_k}. \tag{2.1}$$

We use the pairwise index notation for even-order tensors because it is convenient for defining the unfolding transformation ψ, see Definition II.4, and for representing core matrices/tensors in generalized tensor decompositions, see Section 2.1.2. Note that even-order paired tensors and the Einstein product (2.1) can be viewed as multidimensional generalizations of matrices and the standard matrix-vector product, respectively [50]. Similar to the standard matrix-matrix product, one can also define a generalized form of the Einstein product between two even-order paired tensors. We will see later that the Einstein product can be efficiently computed using tensor decompositions of even-order paired tensors.

Definition II.3. Given two even-order paired tensors $\mathsf{A} \in \mathbb{R}^{n_1 \times m_1 \times \cdots \times n_k \times m_k}$ and $\mathsf{B} \in \mathbb{R}^{m_1 \times l_1 \times \cdots \times m_k \times l_k}$, the Einstein product $\mathsf{A} * \mathsf{B} \in \mathbb{R}^{n_1 \times l_1 \times \cdots \times n_k \times l_k}$ is defined by

$$(\mathsf{A} * \mathsf{B})_{j_1 i_1 \ldots j_k i_k} = \sum_{s_1=1}^{m_1} \cdots \sum_{s_k=1}^{m_k} \mathsf{A}_{j_1 s_1 \ldots j_k s_k} \mathsf{B}_{s_1 i_1 \ldots s_k i_k}. \tag{2.2}$$

2.1.1 Isomorphism to General Linear Group

Brazell *et al.* [16] investigated properties of even-order tensors under the Einstein product (different from (2.2)) through construction of an isomorphism to $\mathrm{GL}(n, \mathbb{R})$, i.e., the set of $n \times n$ real valued invertible matrices. The existence of the isomorphism enables one to generalize several matrix concepts, such as invertibility and eigenvalue decomposition to the tensor case [16, 37, 54, 77, 119]. We can establish an analogous isomorphism for even-order paired tensors by a permutation of indices.

Definition II.4. Define the map ψ: $\mathbb{T}_{n_1 m_1 \ldots n_k m_k}(\mathbb{R}) \to \mathbb{M}_{\Pi_{\mathcal{N}} \Pi_{\mathcal{M}}}(\mathbb{R})$ with $\psi(\mathsf{A}) = \mathbf{A}$

defined component-wise as

$$\mathsf{A}_{j_1 i_1 \dots j_k i_k} \xrightarrow{\psi} \mathbf{A}_{ivec(\mathbf{j},\mathcal{N})ivec(\mathbf{i},\mathcal{M})}, \tag{2.3}$$

where $\mathbb{T}_{n_1 m_1 \dots n_k m_k}(\mathbb{R})$ is the set of all real $n_1 \times m_1 \times \cdots \times n_k \times m_k$ even-order paired tensors, and $\mathbb{M}_{\Pi_{\mathcal{N}} \Pi_{\mathcal{M}}}(\mathbb{R})$ is set of all real $\Pi_{\mathcal{N}} \times \Pi_{\mathcal{M}}$ matrices.

The map ψ can be viewed as a tensor unfolding discussed in (1.5) with $z = k$ and $\mathbb{S} = \begin{pmatrix} 1 & 2 & \dots & k & k+1 & k+2 & \dots & 2k \\ 1 & 3 & \dots & 2k-1 & 2 & 4 & \dots & 2k \end{pmatrix}$, so the Frobenius norm is preserved through ψ, i.e., $\|\mathsf{A}\| = \|\psi(\mathsf{A})\|$. More significantly, ψ is bijective, and the restriction of ψ^{-1} on the general linear group produces a group isomorphism.

Corollary II.5. *Let $n_p = m_p$ for all p and $\mathbb{G}_{n_1 m_1 \dots n_k m_k}(\mathbb{R}) = \psi^{-1}(GL(\Pi_{\mathcal{N}}, \mathbb{R}))$, i.e., $\mathbb{G}_{n_1 m_1 \dots n_k m_k}$ is the space of all even-order paired tensors which maps to the general linear group under ψ. Then $\mathbb{G}_{n_1 m_1 \dots n_k m_k}(\mathbb{R})$ is a group equipped with the Einstein product (2.2), and ψ is a group isomorphism.*

Detailed proofs can be found in [16, 54]. Based on the unfolding property, we can define several tensor notations analogous to matrices as follows:

- For an even-order paired tensor $\mathsf{A} \in \mathbb{R}^{n_1 \times m_1 \times \cdots \times n_k \times m_k}$, $\mathsf{T} \in \mathbb{R}^{m_1 \times n_1 \times \cdots \times m_k \times n_k}$ is called the *U-transpose* of A if $\mathsf{T}_{i_1 j_1 \dots i_k j_k} = \mathsf{A}_{j_1 i_1 \dots j_k i_k}$, and is denoted by $\mathsf{A}^\top$. We refer to an even-order paired tensor that is identical to its U-transpose as *weakly symmetric*.

- For an even-order paired tensor $\mathsf{A} \in \mathbb{R}^{n_1 \times m_1 \times \cdots \times n_k \times m_k}$, the *unfolding rank* of A is defined as $\text{rank}_U(\mathsf{A}) = \text{rank}(\psi(\mathsf{A}))$ [77].

- An even-order "square" tensor $\mathsf{D} \in \mathbb{R}^{n_1 \times n_1 \times \cdots \times n_k \times n_k}$ is called the *U-diagonal* tensor if all its entries are zeros except for $\mathsf{D}_{j_1 j_1 \dots j_k j_k}$. If all the diagonal entires $\mathsf{D}_{j_1 j_1 \dots j_k j_k} = 1$, then D is the *U-identity* tensor, denoted by I.

- For an even-order square tensor $\mathsf{A} \in \mathbb{R}^{n_1 \times n_1 \times \cdots \times n_k \times n_k}$, if there exists a tensor $\mathsf{B} \in \mathbb{R}^{n_1 \times n_1 \times \cdots \times n_k \times n_k}$ such that $\mathsf{A} * \mathsf{B} = \mathsf{B} * \mathsf{A} = \mathsf{I}$, then B is called the *U-inverse* of A, denoted by A^{-1}.

- An even-order square tensor $\mathsf{A} \in \mathbb{R}^{n_1 \times n_1 \times \cdots \times n_k \times n_k}$ is called *U-positive definite* if $\mathsf{X}^\top * \mathsf{A} * \mathsf{X} > 0$ for any nonzero tensor $\mathsf{X} \in \mathbb{R}^{n_1 \times n_2 \times \cdots \times n_k}$.

- For an even-order square tensor $\mathsf{A} \in \mathbb{R}^{n_1 \times n_1 \times \cdots \times n_k \times n_k}$, the *unfolding determinant* of A is defined as $\det_U(\mathsf{A}) = \det(\psi(\mathsf{A}))$ [77].

We show that the notion of U-positive definiteness is a generalization of M-positive definiteness and rank-one positive definiteness proposed in [59, 98] for the even-order elasticity tensors.

Definition II.6. An even-order square tensor $\mathsf{A} \in \mathbb{R}^{n_1 \times n_1 \times \cdots \times n_k \times n_k}$ is called *M-positive definite* if the multilinear functional

$$\mathsf{A} \times \{\mathbf{x}_1^\top, \mathbf{x}_1^\top, \ldots, \mathbf{x}_k^\top, \mathbf{x}_k^\top\} > 0, \tag{2.4}$$

for any nonzero vector $\mathbf{x}_p$. If all $\mathbf{x}_p$ are equal, A is called *rank-one positive definite*.

Lemma II.7. *Let $\mathsf{A} \in \mathbb{R}^{n_1 \times m_1 \times \cdots \times n_k \times m_k}$ be an even-order paired tensor. Then the product $\mathsf{A} \times \{\boldsymbol{U}_1, \boldsymbol{V}_1, \ldots, \boldsymbol{U}_k, \boldsymbol{V}_k\} = \mathsf{U} * \mathsf{A} * \mathsf{V}^\top \in \mathbb{R}^{q_1 \times s_1 \times \cdots \times q_k \times s_k}$ for $\mathsf{U} = \boldsymbol{U}_1 \circ \boldsymbol{U}_2 \circ \cdots \circ \boldsymbol{U}_k$ and $\mathsf{V} = \boldsymbol{V}_1 \circ \boldsymbol{V}_2 \circ \cdots \circ \boldsymbol{V}_k$ where $\boldsymbol{U}_p \in \mathbb{R}^{q_p \times n_p}$ and $\boldsymbol{V}_p \in \mathbb{R}^{s_p \times m_p}$.*

Proof. This follows from the definitions of the Tucker and Einstein products. □

Proposition II.8. *If an even-order square tensor $\mathsf{A} \in \mathbb{R}^{n_1 \times n_1 \times \cdots \times n_k \times n_k}$ is U-positive definite, it is M-positive definite. Moreover, if $n_1 = n_2 = \cdots = n_k$, U-positive definiteness also implies rank-one positive definiteness.*

Proof. By Lemma II.7, it follows that $\mathsf{A} \times \{\mathbf{x}_1^\top, \mathbf{x}_1^\top, \ldots, \mathbf{x}_k^\top, \mathbf{x}_k^\top\} = \mathsf{X}^\top * \mathsf{A} * \mathsf{X}$ for $\mathsf{X} = \mathbf{x}_1 \circ \mathbf{x}_2 \circ \cdots \circ \mathbf{x}_k$, i.e., X is a rank-one tensor. Moreover, if $n_1 = n_2 = \cdots = n_k$,

M-positive definiteness implies rank-one positive definiteness [98]. Therefore, the results follow immediately. □

Eigenvalue problems for tensors were first explored by Qi [96] and Lim [74] independently. Brazell *et al.* [16] formulated a new tensor eigenvalue problem through the isomorphism ψ for fourth-order tensors, and Cui *et al.* [37] extended the tensor eigenvalue problem to even-order tensors.

Definition II.9. Let $\mathsf{A} \in \mathbb{R}^{n_1 \times n_1 \times \cdots \times n_k \times n_k}$ be an even-order square tensor. If $\mathsf{X} \in \mathbb{C}^{n_1 \times n_2 \times \cdots n_k}$ is a nonzero kth order tensor, $\lambda \in \mathbb{C}$, and X and λ satisfy $\mathsf{A} * \mathsf{X} = \lambda \mathsf{X}$, then we call λ and X as the *U-eigenvalue* and *U-eigentensor* of A, respectively.

The algebraic and geometric multiplicities of U-eigenvalues can be defined as for matrices. The generalization of the Caley-Hamilton theorem for the tensor case can be obtained by the isomorphism property, i.e., an even-order square tensor A satisfies its own characteristic polynomial $p(\lambda) = \det_U(\lambda \mathsf{I} - \mathsf{A})$. Moreover, it can be shown that the notion of U-eigenvalues is a generalization of Z-eigenvalues and M-eigenvalues as proposed in [59, 74, 96]. Detailed proofs are omitted here.

2.1.2 Generalized Tensor Decompositions

We introduce the notion of generalized CPD/TTD for even-order paired tensors described in [50], in which the generalized CPD can also be viewed as the extension of the Kronecker rank approximation proposed by Van Loan [124]. Generalized CPD and TTD share a similar format and possess many analogous properties.

Definition II.10. Given an even-order paired tensor $\mathsf{A} \in \mathbb{R}^{n_1 \times m_1 \times \cdots \times n_k \times m_k}$, the generalized CPD of A is given by

$$\mathsf{A} = \sum_{r=1}^{R} \mathsf{A}^{(1)}_{r::} \circ \mathsf{A}^{(2)}_{r::} \circ \cdots \circ \mathsf{A}^{(k)}_{r::}, \tag{2.5}$$

where $\mathsf{A}^{(p)} \in \mathbb{R}^{R \times n_p \times m_p}$. Extending Van Loan's definition [124], we call the smallest R that achieves (2.5) the Kronecker rank of A.

Definition II.11. Given an even-order paired tensor $\mathsf{A} \in \mathbb{R}^{n_1 \times m_1 \times \cdots \times n_k \times n_k}$, the generalized TTD of A is given by

$$\mathsf{A} = \sum_{r_0=1}^{R_0} \cdots \sum_{r_k=1}^{R_k} \mathsf{A}^{(1)}_{r_0::r_1} \circ \mathsf{A}^{(2)}_{r_1::r_2} \circ \cdots \circ \mathsf{A}^{(k)}_{r_{k-1}::r_k}, \tag{2.6}$$

where $\mathsf{A}^{(p)} \in \mathbb{R}^{R_{p-1} \times n_p \times m_p \times R_p}$, and $\{R_0, R_1, \ldots, R_k\}$ is the set of TT-ranks with $R_0 = R_k = 1$.

Please refer to Appendix .2.1 for the use of : notation. Given two even-order paired tensors in the generalized CPD/TTD format, the Einstein product (2.2) between the two can be computed without having to reconstruct the full tensors, i.e., keeping the original format [50]. The following proposition states the case for generalized CPD, which also applies to generalized TTD.

Proposition II.12. *Given two even-order paired tensors $\mathsf{A} \in \mathbb{R}^{n_1 \times m_1 \times \ldots n_k \times m_k}$ and $\mathsf{B} \in \mathbb{R}^{m_1 \times l_1 \times \cdots \times m_k \times l_k}$ in the format of (2.5) with Kronekcer ranks R and S, respectively, the Einstein product $\mathsf{A} * \mathsf{B}$ is given by*

$$\mathsf{A} * \mathsf{B} = \sum_{t=1}^{T} \mathsf{E}^{(1)}_{t::} \circ \mathsf{E}^{(2)}_{t::} \circ \cdots \circ \mathsf{E}^{(k)}_{t::}, \tag{2.7}$$

where $\mathsf{E}^{(p)}_{t::} = \mathsf{A}^{(p)}_{r::} \mathsf{B}^{(p)}_{s::} \in \mathbb{R}^{n_p \times m_p}$, and $t = ivec(\{r, s\}, \{R, S\})$ with $T = RS$.

Remark. The computational complexity of the Einstein product (2.7) is about $\mathcal{O}(kn^3R^2)$ assuming that $n_p = m_p = l_p = n$ and $R = S$, which is much lower than $\mathcal{O}(n^{3k})$ from the Einstein product (2.2) if R is small.

The generalized CPD can be recovered from the standard CPD, and similarly for generalized TTD. The algorithm below is extended from the results by Van Loan [124]

about the Kronecker rank approximation. Thus, one can easily obtain generalized CPD by using any technique for computing the standard CPD including alternating least square (ALS) and modified ALS methods [69, 70].

Algorithm 1 Generalized CPD

1: Given an even-order paired tensors $\mathsf{A} \in \mathbb{R}^{n_1 \times m_1 \times \cdots \times n_k \times m_k}$
2: Set $\check{\mathsf{A}} = \texttt{reshape}(\mathsf{A}, n_1 m_1, n_2 m_2, \ldots, n_k m_k)$
3: Apply CPD algorithms on $\check{\mathsf{A}}$ such that $\check{\mathsf{A}} = \sum_{r=1}^{R} \lambda_r \mathbf{a}_1^{(r)} \circ \mathbf{a}_2^{(r)} \circ \cdots \circ \mathbf{a}_k^{(r)}$
4: Set $\mathsf{A}_{r::}^{(p)} = \lambda_r^{\frac{1}{k}} \texttt{reshape}(\mathbf{a}_p^{(r)}, n_p, m_p)$ for $p = 1, 2, \ldots, k$
5: **return** Component tensors $\mathsf{A}^{(p)}$ for $p = 1, 2, \ldots, k$.

Algorithm 2 Generalized TTD

1: Given an even-order paired tensors $\mathsf{A} \in \mathbb{R}^{n_1 \times m_1 \times \cdots \times n_k \times m_k}$
2: Set $\check{\mathsf{A}} = \texttt{reshape}(\mathsf{A}, n_1 m_1, n_2 m_2, \ldots, n_k m_k)$
3: Apply the standard TTD algorithm on $\check{\mathsf{A}}$ such that

$$\check{\mathsf{A}} = \sum_{r_0=1}^{R_0} \cdots \sum_{r_k=1}^{R_k} \check{\mathsf{A}}_{r_0:r_1}^{(1)} \circ \check{\mathsf{A}}_{r_1:r_2}^{(2)} \circ \cdots \circ \check{\mathsf{A}}_{r_{k-1}:r_k}^{(k)}$$

4: Set $\mathsf{A}_{r_{p-1}::r_p}^{(p)} = \texttt{reshape}(\check{\mathsf{A}}_{r_{p-1}:r_p}^{(p)}, n_p, m_p)$ for $p = 1, 2, \ldots, k$
5: **return** Component tensors $\mathsf{A}^{(p)}$ for $p = 1, 2, \ldots, k$.

2.2 Tensor Rank Relations

We establish new results relating the unfolding rank of an even-order paired tensor to its multilinear ranks, CP rank, and TT-ranks. These relationships are useful for checking multilinear generalizations of reachability and observability rank conditions.

Proposition II.13. *Let $\mathsf{A} \in \mathbb{R}^{n_1 \times m_1 \times \cdots \times n_k \times m_k}$ be an even-order paired tensor. If $\text{rank}_U(\mathsf{A}) = \Pi_{\mathcal{N}}$ (or $\text{rank}_U(\mathsf{A}) = \Pi_{\mathcal{M}}$), then $\text{rank}_{2p-1}(\mathsf{A}) = n_p$ (or $\text{rank}_{2p}(\mathsf{A}) = m_p$) for $p = 1, 2, \ldots, k$.*

Proof. Without loss of generality, assume that $\Pi_{\mathcal{M}} \leq \Pi_{\mathcal{N}}$ and $\text{rank}_U(\mathsf{A}) = \Pi_{\mathcal{N}}$. Then $\psi(\mathsf{A})$ has $\Pi_{\mathcal{M}}$ linearly independent columns. The goal here is to construct a transformation from $\psi(\mathsf{A})$ to $\mathbf{A}_{(2p)}^{\top}$, which can be easily visualized through the

representation $(z, \mathbb{S})$ defined in (1.5). Let

$$\mathbb{S}_1 = \begin{pmatrix} 1 & 2 & \dots & k & k+1 & k+2 & \dots & 2k \\ 1 & 3 & \dots & 2k-1 & 2 & 4 & \dots & 2k \end{pmatrix},$$

$$\mathbb{S}_2 = \begin{pmatrix} 1 & 2 & \dots & 2p-1 & 2p & \dots & 2k-1 & 2k \\ 1 & 2 & \dots & 2p-1 & 2p+1 & \dots & 2k & 2p \end{pmatrix},$$

$$\mathbb{S}_3 = \begin{pmatrix} 1 & 2 & \dots & k & k+1 & k+2 & \dots & k+p & k+p+1 & \dots & 2k \\ 1 & 3 & \dots & 2N-1 & 2n & 2 & \dots & 2n-2 & 2n+2 & \dots & 2N \end{pmatrix},$$

$$\mathbb{S}_4 = \begin{pmatrix} 1 & 2 & \dots & k & k+1 & \dots & k+p-1 & k+p & \dots & 2k-1 & 2k \\ 1 & 3 & \dots & 2k-1 & 2 & \dots & 2p-2 & 2p+2 & \dots & 2k & 2p \end{pmatrix}.$$

Clearly, $\psi(\mathsf{A})$ and $\mathbf{A}_{(2p)}^{\top}$ can be represented by $(k, \mathbb{S}_1)$ and $(2k-1, \mathbb{S}_2)$, respectively. According to the definition of the index mapping function $ivec(\mathbf{i}, \mathcal{M})$, we first require a column permutation matrix $\mathbf{P}$ such that $\psi(\mathsf{A})\mathbf{P}$ is represented by $(k, \mathbb{S}_3)$. Every m_p columns of $\psi(\mathsf{A})\mathbf{P}$ correspond to the columns of $\mathbf{A}_{(2p)}^{\top}$. Collect each set of m_p columns of $\psi(\mathsf{A})\mathbf{P}$ and stack them vertically to form a tall matrix $\tilde{\mathbf{A}}$ with the representation $(2k-1, \mathbb{S}_4)$. Since the columns of $\psi(\mathsf{A})\mathbf{P}$ are linearly independent, $\operatorname{rank}(\tilde{\mathbf{A}}) = m_p$. Finally, according to the definition of the index mapping function $ivec(\mathbf{j}, \mathcal{N})$, we require a row permutation matrix $\mathbf{Q}$ such that $\mathbf{Q}\tilde{\mathbf{A}} = \mathbf{A}_{(2p)}^{\top}$. Hence, $\operatorname{rank}_{2p}(\mathsf{A}) = \operatorname{rank}(\mathbf{A}_{(2p)}^{\top}) = m_p$. Note that the converse of the statement is incorrect. □

Proposition II.14. *Let $\mathsf{A} \in \mathbb{R}^{n_1 \times m_1 \times \cdots \times n_k \times m_k}$ be an even-order paired tensor given in the CPD format (2.5) with CP rank equal to R. If the following conditions*

$$\sum_{p=1:2}^{2k} \mathcal{K}_{\boldsymbol{A}^{(p)}} \geq R + k - 1, \ \sum_{p=2:2}^{2k} \mathcal{K}_{\boldsymbol{A}^{(p)}} \geq R + k - 1 \tag{2.8}$$

are satisfied for every $\mathcal{K}_{\boldsymbol{A}^{(p)}} \geq 1$, then $\operatorname{rank}_U(\mathsf{A}) = R$. The notations $\sum_{p=1:2}^{2k}$ and $\sum_{p=2:2}^{2k}$ represent the sums of all odd indices and all even indices, respectively.

In order to prove Proposition II.14, we need to introduce the concept of Khatri-Rao product.

Definition II.15. Given two matrices $\mathbf{A} \in \mathbb{R}^{n \times m}$ and $\mathbf{B} \in \mathbb{R}^{l \times m}$, the *Khatri-Rao*

product, denoted by $\mathbf{A} \odot \mathbf{B}$, results in a $nl \times m$ matrix:

$$\mathbf{A} \odot \mathbf{B} = \begin{bmatrix} \mathbf{a}_1 \otimes \mathbf{b}_1 & \mathbf{a}_2 \otimes \mathbf{b}_2 & \dots & \mathbf{a}_m \otimes \mathbf{b}_m \end{bmatrix},$$

where $\otimes$ denotes the Kronecker product, and $\mathbf{a}_p$ and $\mathbf{b}_p$ are the column vectors of $\mathbf{A}$ and $\mathbf{B}$, respectively.

The following lemma provided by Sidiropoulos *et al.* [114, 117] gives some properties of rank and k-rank of the Khatri-Rao product $\mathbf{A} \odot \mathbf{B}$.

Lemma II.16. *Given two matrices $\boldsymbol{A} \in \mathbb{R}^{n \times R}, \boldsymbol{B} \in \mathbb{R}^{m \times R}$, the Khatri-Rao product $\boldsymbol{A} \odot \boldsymbol{B}$ has column rank R if $\mathcal{K}_{\boldsymbol{A}} + \mathcal{K}_{\boldsymbol{B}} \geq R + 1$ for $\mathcal{K}_{\boldsymbol{A}}, \mathcal{K}_{\boldsymbol{B}} \geq 1$. Moreover, $\mathcal{K}_{\boldsymbol{A} \odot \boldsymbol{B}} \geq \min\{\mathcal{K}_{\boldsymbol{A}} + \mathcal{K}_{\boldsymbol{B}} - 1, R\}$.*

Proposition II.17. *Given matrices $\boldsymbol{A}^{(p)} \in \mathbb{R}^{n_p \times R}$, the Khatri-Rao product $\boldsymbol{A}^{(1)} \odot \boldsymbol{A}^{(2)} \odot \cdots \odot \boldsymbol{A}^{(k)}$ has column rank R if $\sum_{p=1}^{k} \mathcal{K}_{\boldsymbol{A}^{(p)}} \geq R + k - 1$ for $\mathcal{K}_{\boldsymbol{A}^{(p)}} \geq 1$.*

Proof. Suppose that $k = 3$. By Lemma II.16, the Khatri-Rao product $\mathbf{A}^{(1)} \odot \mathbf{A}^{(2)} \odot \mathbf{A}^{(3)}$ has full column rank R if $\mathcal{K}_{\mathbf{A}^{(1)} \odot \mathbf{A}^{(2)}} + \mathcal{K}_{\mathbf{A}^{(3)}} \geq R + 1$. Since we know that $\mathcal{K}_{\mathbf{A} \odot \mathbf{B}} \geq \min\{\mathcal{K}_{\mathbf{A}} + \mathcal{K}_{\mathbf{B}} - 1, R\}$, the above inequality can be satisfied if

$$\min\{\mathcal{K}_{\mathbf{A}^{(1)}} + \mathcal{K}_{\mathbf{A}^{(2)}} - 1, R\} + \mathcal{K}_{\mathbf{A}^{(3)}} \geq R + 1.$$

When $\mathcal{K}_{\mathbf{A}^{(1)}} + \mathcal{K}_{\mathbf{A}^{(2)}} > R + 1$, the condition is reduced to $\mathcal{K}_{\mathbf{A}^{(3)}} \geq 1$, and when $\mathcal{K}_{\mathbf{A}^{(1)}} + k_{\mathbf{A}^{(2)}} \leq R + 1$, the condition becomes $\mathcal{K}_{\mathbf{A}^{(1)}} + \mathcal{K}_{\mathbf{A}^{(2)}} + \mathcal{K}_{\mathbf{A}^{(3)}} \geq R + 2$. Therefore, the Khatri-Rao product $\mathbf{A}^{(1)} \odot \mathbf{A}^{(2)} \odot \mathbf{A}^{(3)}$ has full column rank R if $\mathcal{K}_{\mathbf{A}^{(1)}} + \mathcal{K}_{\mathbf{A}^{(2)}} + \mathcal{K}_{\mathbf{A}^{(3)}} \geq R + 2$. The result can be easily extended to $p = k$ using the same approach. □

Now, we can prove Proposition II.14.

Proof. Suppose that A has the CPD format (2.5) with CP rank equal to R. Applying the unfolding transformation ψ yields

$$\psi(\mathsf{A}) = (\mathbf{A}^{(2k-1)} \odot \cdots \odot \mathbf{A}^{(1)})\mathbf{S}(\mathbf{A}^{(2k)} \odot \cdots \odot \mathbf{A}^{(2)})^\top,$$

where $\mathbf{S} \in \mathbb{R}^{R\times R}$ is a diagonal matrix containing the weights of the CPD on its diagonal. By Proposition II.17, the two Khatri-Rao products $\mathbf{A}^{(2k-1)} \odot \cdots \odot \mathbf{A}^{(1)}$ and $\mathbf{A}^{(2k)} \odot \cdots \odot \mathbf{A}^{(2)}$ have full column rank R if the two conditions $\sum_{p=1:2}^{2k} \mathcal{K}_{\mathbf{A}^{(p)}} \geq R + k - 1$, and $\sum_{p=2:2}^{2k} \mathcal{K}_{\mathbf{A}^{(p)}} \geq R + k - 1$ are satisfied. Hence, $\text{rank}_U(\mathsf{A}) = R$. Note that we do not require the CPD of A to be unique in the statement. □

Proposition II.18. *Let $\mathsf{A} \in \mathbb{R}^{n_1\times m_1\times\cdots\times n_k\times m_k}$ be an even-order paired tensor. Then $\text{rank}_U(\mathsf{A}) = \tilde{R}_k$ where $\tilde{R}_k$ is the kth optimal TT-rank of $\tilde{\mathsf{A}}$, the $\mathbb{S}$-transpose of A with* $\mathbb{S} = \begin{pmatrix} 1 & 2 & \dots & k & k+1 & k+2 & \dots & 2k \\ 1 & 3 & \dots & 2k-1 & 2 & 4 & \dots & 2k \end{pmatrix}$.

Proof. The result follows from the definition of optimal TT-ranks. □

Remark. Given the TTD of A, the TTD of $\tilde{\mathsf{A}} \in \mathbb{R}^{n_1\times\cdots\times n_k\times m_1\times\cdots\times m_k}$ can be constructed by manipulating the core tensors $\mathsf{A}^{(p)}$ without converting back to the full format. Assume that $n_p = m_p = n$ for all p, and R is the average of the TT-ranks of A. If R remains unchanged or decreases during this conversion, the computational complexity is estimated to be at most $\mathcal{O}(k^2n^3R^3)$[1]. A detailed algorithm for the TTD-based permutation can be found in [25].

2.3 Block Tensors

We next discuss notions of block tensors and tensor decompositions which will form the basis for developing tensor algebra-based concepts of stability, reachability, and observability of MLTI systems. Analogously to block matrices, one can define

[1]Big O notation: $f(x) = \mathcal{O}(g(x))$ as $x \to \infty$ if and only if there exists a positive real number M and a real number x_0 such that $|f(x)| \leq Mg(x)$ for all $x \geq x_0$.

the notion of block tensors. For tensors of the same size, we propose a block tensor construction which does not introduce any wasteful zeros compared to the block tensors proposed in [119], and thus offers computational advantages.

Definition II.19. Let $\mathsf{A}, \mathsf{B} \in \mathbb{R}^{n_1 \times m_1 \times \cdots \times n_k \times m_k}$ be two even-order paired tensors. The *p-mode row block tensor* $\left|\begin{array}{cc} \mathsf{A} & \mathsf{B} \end{array}\right|_p \in \mathbb{R}^{n_1 \times m_1 \times \cdots \times n_p \times 2m_p \times \cdots \times n_k \times m_k}$ is defined by

$$(\left|\begin{array}{cc} \mathsf{A} & \mathsf{B} \end{array}\right|_p)_{j_1 l_1 \ldots j_k l_k} = \begin{cases} \mathsf{A}_{j_1 l_1 \ldots j_k l_k}, & j_s = 1, \ldots, n_s,\ l_s = 1, \ldots, m_s\ \forall s \\ \mathsf{B}_{j_1 l_1 \ldots j_k l_k}, & j_s = 1, \ldots, n_s\ \forall s, l_s = 1, \ldots, m_s \text{ for } s \neq p \cdot \\ & \text{and } l_s = m_s + 1, \ldots, 2m_s \text{ for } s = p \end{cases}$$

For example, if $\mathsf{A}, \mathsf{B} \in \mathbb{R}^{2\times 2\times 2\times 2}$, then the 1-mode row block tensor is given by $\left|\begin{array}{cc} \mathsf{A} & \mathsf{B} \end{array}\right|_1 \in \mathbb{R}^{2\times 4\times 2\times 2}$ such that $(\left|\begin{array}{cc} \mathsf{A} & \mathsf{B} \end{array}\right|_1)_{:i_1::} = \mathsf{A}$ for $i_1 = 1, 2$ and $(\left|\begin{array}{cc} \mathsf{A} & \mathsf{B} \end{array}\right|_1)_{:i_1::} = \mathsf{B}$ for $i_1 = 3, 4$. Similarly for $\left|\begin{array}{cc} \mathsf{A} & \mathsf{B} \end{array}\right|_2 \in \mathbb{R}^{2\times 2\times 2\times 4}$. Detailed explanations of the MATLAB colon operation : can be found in Appendix .2.1. When $k = 1$, it reduces to the row block matrices. The *p-mode column block tensor*

$$\left|\begin{array}{c} \mathsf{A} \\ \mathsf{B} \end{array}\right|_p \in \mathbb{R}^{n_1 \times m_1 \times \cdots \times 2n_p \times m_p \times \cdots \times n_k \times m_k}$$

can be defined in a similar manner. The p-mode block tensors exhibit many properties analogous to block matrix computations, e.g., the Einstein product can distribute over block tensors, and the blocks of p-mode row block tensors map to contiguous blocks under ψ up to some permutations [99].

Proposition II.20. *Let $\mathcal{A}, \mathcal{B} \in \mathbb{R}^{n_1 \times m_1 \times \cdots \times n_k \times m_k}$ and $\mathcal{C}, \mathcal{D} \in \mathbb{R}^{m_1 \times l_1 \times \cdots \times m_k \times l_k}$. Then the following properties hold:*

- $\mathcal{P} * \left|\begin{array}{cc} \mathcal{A} & \mathcal{B} \end{array}\right|_p = \left|\begin{array}{cc} \mathcal{P} * \mathcal{A} & \mathcal{P} * \mathcal{B} \end{array}\right|_p$ *for any* $\mathcal{P} \in \mathbb{R}^{s_1 \times m_1 \times \cdots \times s_k \times m_k}$;

- $\left|\begin{matrix} C \\ D \end{matrix}\right|_p * Q = \left|\begin{matrix} C * Q \\ D * Q \end{matrix}\right|_p$ *for any* $Q \in \mathbb{R}^{l_1 \times r_1 \times \cdots \times l_k \times r_k}$;
- $\left|\begin{matrix} A & B \end{matrix}\right|_p * \left|\begin{matrix} C \\ D \end{matrix}\right|_p = A * C + B * D.$

Proof. The proof follows immediately from the definition of p-mode row/column block tensors and the Einstein product. □

Proposition II.21. *Let* $\mathcal{A}, \mathcal{B} \in \mathbb{R}^{n_1 \times m_1 \times \cdots \times n_k \times m_k}$ *be two even-order paired tensors. Then* $\psi(\left|\begin{matrix} \mathcal{A} & \mathcal{B} \end{matrix}\right|_p) = \left[\begin{matrix} \psi(\mathcal{A}) & \psi(\mathcal{B}) \end{matrix}\right] \boldsymbol{P}$, *where* $\boldsymbol{P}$ *is a column permutation matrix. In particular, when* $m_p = 1$ *for all* p *or* $p = k$, $\boldsymbol{P}$ *is the identity matrix.*

Proof. We consider the case for $k = 2$. Since the size of the odd modes of the block tensor remains the same, we only need to consider the even modes' unfolding transformation. When $p = 1$, the index mapping function for the even modes is

$$ivec(\mathbf{i}, \mathcal{M}) = i_1 + 2(i_2 - 1)m_1,$$

for $i_1 = 1, 2, \ldots, 2m_1$. Based on the definition of p-mode row block tensors, the first m_1 columns of $\psi(\left|\begin{matrix} \mathsf{A} & \mathsf{B} \end{matrix}\right|_1)$ are the vectorizations of $\mathsf{A}_{:i_1:i_2}$ for $i_1 = 1, 2, \ldots, m_1$ and $i_2 = 1$, and the second m_1 columns are the vectorizations of $\mathsf{B}_{:i_1:i_2}$ for $i_1 = m_1+1, m_1+2, \ldots, 2m_1$ and $i_2 = 1$. The alternating pattern continues for all m_2 pairs of m_1 columns. Hence, $\psi(\left|\begin{matrix} \mathsf{A} & \mathsf{B} \end{matrix}\right|_1) = \left[\begin{matrix} \psi(\mathsf{A}) & \psi(\mathsf{B}) \end{matrix}\right] \mathbf{P}$ for some column permutation matrix $\mathbf{P}$. When $p = 2$, the index mapping function for the even modes is given by

$$ivec(\mathbf{i}, \mathcal{M}) = i_1 + (i_2 - 1)m_1,$$

for $i_2 = 1, 2, \ldots, 2m_2$. Similarly, the first $m_1 m_2$ columns of $\psi(\left|\begin{matrix} \mathsf{A} & \mathsf{B} \end{matrix}\right|_2)$ are the vectorizations of $\mathsf{A}_{:i_1:i_2}$ for $i_1 = 1, 2, \ldots, m_1$ and $i_2 = 1, 2, \ldots, m_2$, and the second

$m_1 m_2$ columns are the vectorizations of $\mathsf{B}_{:i_1:i_2}$ for $i_1 = 1, 2, \ldots, m_1$ and $i_2 = m_2 + 1, m_2 + 2, \ldots, 2m_2$. Hence, $\psi(\left|\mathsf{A} \quad \mathsf{B}\right|_2) = \begin{bmatrix}\psi(\mathsf{A}) & \psi(\mathsf{B})\end{bmatrix}$. A similar analysis can be used to prove the case for $k > 2$. Moreover, when $m_p = 1$ for all p, $\psi(\mathsf{A})$ and $\psi(\mathsf{B})$ are vectors, so no permutation needs to be considered. The proposition can be considered as a special case of Theorem 3.3 in [99]. □

Given q even-order paired tensors $\mathsf{X}_n \in \mathbb{R}^{n_1 \times m_1 \times \cdots \times n_k \times m_k}$, one can apply Definition II.19 successively to create a $n_1 \times m_1 \times \cdots \times n_p \times m_p q \times \cdots \times n_k \times m_k$ even-order p-mode row block tensor. However, a more general concatenation approach can be defined for multiple blocks.

Definition II.22. Given q even-order paired tensors $\mathsf{X}_n \in \mathbb{R}^{n_1 \times m_1 \times \cdots \times n_k \times m_k}$, if $q = q_1 q_2 \ldots q_k$, the $n_1 \times m_1 q_1 \times \cdots \times n_k \times m_k q_k$ even-order *mode row block tensor* Y can be constructed in the following way:

- Compute the 1-mode row block tensor concatenation over $\{\mathsf{X}_1, \cdots, \mathsf{X}_{q_1}\}$, $\{\mathsf{X}_{q_1+1}, \cdots, \mathsf{X}_{2q_1}\}$ and so on to obtain $q_2 q_3 \ldots q_k$ block tensors denoted by $\mathsf{X}^{(1)}_1, \mathsf{X}^{(1)}_2, \ldots, \mathsf{X}^{(1)}_{q_2 q_3 \ldots q_k}$;
- Compute the 2-mode row block tensors concatenation over $\{\mathsf{X}^{(1)}_1, \cdots, \mathsf{X}^{(1)}_{q_2}\}$, $\{\mathsf{X}^{(1)}_{q_2+1}, \cdots, \mathsf{X}^{(1)}_{2q_2}\}$ and so on to obtain $q_3 q_4 \ldots q_k$ block tensors denoted by $\mathsf{X}^{(2)}_1, \mathsf{X}^{(2)}_2, \ldots, \mathsf{X}^{(2)}_{q_3 q_4 \ldots q_k}$;
- Keep repeating the process until the last k-mode row block tensor is obtained.

We denote the mode row block tensor as $\mathsf{Y} = \left|\mathsf{X}_1 \quad \mathsf{X}_2 \quad \ldots \quad \mathsf{X}_q\right|$.

For example, suppose that $\mathsf{X}_n \in \mathbb{R}^{2 \times 2 \times 2 \times 2 \times 2 \times 2}$ for $n = 1, 2, \ldots, q$ and $q = 8$. Let $q = q_1 q_2 q_3$ with $q_1 = q_2 = q_3 = 2$. Given this factorization of q, the mode row block tensor $\mathsf{Y} \in \mathbb{R}^{2 \times 4 \times 2 \times 4 \times 2 \times 4}$ is constructed in the manner shown in Figure 2.3,

in which $\mathsf{X}_p^{(1)} \in \mathbb{R}^{2\times4\times2\times2\times2\times2}$ and $\mathsf{X}_p^{(2)} \in \mathbb{R}^{2\times4\times2\times4\times2\times2}$. Another factorization with $q_1 = 2$, $q_2 = 4$ and $q_3 = 1$ would return $\mathsf{Y} \in \mathbb{R}^{2\times4\times2\times8\times2\times2}$. The generalized *mode column block tensors* with multiple blocks can be constructed in a similar manner. When $m_p = 1$ for all p, the above generalized mode row block tensor maps exactly to contiguous blocks in its unfolding under ψ, which could be beneficial in many block tensor applications. Furthermore, the choices of q_p may affect the structure of mode block tensors, which can be significant in tensor ranks/decompositions [25].

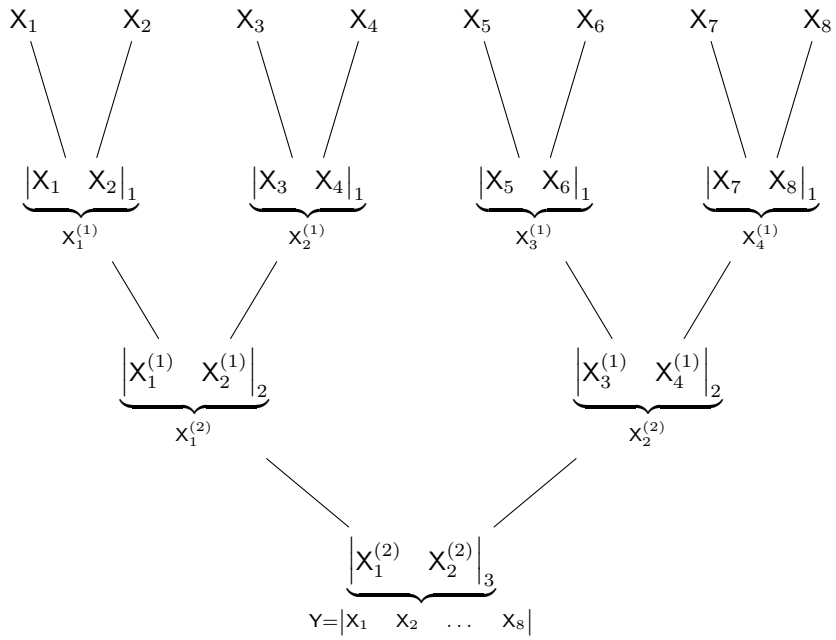


Figure 2.1: **An example of mode row block tensor.**

2.4 MLTI System Representation

To describe the evolution of tensor time series, the authors in [107, 120] introduced a MLTI system using the Tucker product, which can be generalized by incorporating

control inputs as follows:

$$\begin{cases} \mathsf{X}_{t+1} = \mathsf{X}_t \times \{\mathbf{A}_1, \ldots, \mathbf{A}_k\} + \mathsf{U}_t \times \{\mathbf{B}_1, \ldots, \mathbf{B}_k\} \\ \mathsf{Y}_t = \mathsf{X}_t \times \{\mathbf{C}_1, \ldots, \mathbf{C}_k\} \end{cases}, \tag{2.9}$$

where $\mathsf{X}_t \in \mathbb{R}^{n_1 \times n_2 \times \cdots \times n_k}$ is the latent state space tensor, $\mathsf{Y}_t \in \mathbb{R}^{m_1 \times m_2 \times \cdots \times m_k}$ is the output tensor, and $\mathsf{U}_t \in \mathbb{R}^{q_1 \times q_2 \times \cdots \times q_k}$ is the control tensor. $\mathbf{A}_p \in \mathbb{R}^{n_p \times n_p}$, $\mathbf{B}_p \in \mathbb{R}^{n_p \times q_p}$, and $\mathbf{C}_n \in \mathbb{R}^{m_p \times n_p}$ are real valued matrices for $p = 1, 2, \ldots, k$. The Tucker product provides a suitable way to deal with MLTI systems because it allows one to exploit matrix computations. However, we find that (2.9) can be replaced by a more general representation using the notion of even-order paired tensors and the Einstein product. Moreover, the new representation is more concise and systematic compared to the tensor-based linear system proposed in [41].

Definition II.23. A more general representation of MLTI system is given by

$$\begin{cases} \mathsf{X}_{t+1} = \mathsf{A} * \mathsf{X}_t + \mathsf{B} * \mathsf{U}_t \\ \mathsf{Y}_t = \mathsf{C} * \mathsf{X}_t \end{cases}, \tag{2.10}$$

where $\mathsf{A} \in \mathbb{R}^{n_1 \times n_1 \times \cdots \times n_k \times n_k}$, $\mathsf{B} \in \mathbb{R}^{n_1 \times q_1 \times \cdots \times n_k \times q_k}$, and $\mathsf{C} \in \mathbb{R}^{m_1 \times n_1 \times \cdots \times m_k \times n_k}$ are even-order paired tensors.

Proposition II.24. *The governing equations (2.10) can be obtained from (2.9) by setting A, B and C to be the outer products of component matrices $\{\boldsymbol{A}_1, \boldsymbol{A}_2, \ldots, \boldsymbol{A}_k\}$, $\{\boldsymbol{B}_1, \boldsymbol{B}_2, \ldots, \boldsymbol{B}_k\}$, and $\{\boldsymbol{C}_1, \boldsymbol{C}_2, \ldots, \boldsymbol{C}_k\}$, respectively.*

Proof. The result follows from II.7 with $m_p = 1$ and $\mathbf{V}_p = 1$ for all p. □

The main advantages of the MLTI system (2.10) are as follows:

- The Einstein product representation (2.10) of MLTI systems is indeed the generalization of (2.9). While Proposition II.24 shows that MLTI systems in the

form of (2.9) can always be transformed into the form of (2.10), the converse is not always true, see (2.20) for example. It is true only when $R_1 = R_2 = R_3 = 1$.

- The MLTI system (2.10) takes a form similar to the standard LTI system model with matrix product replaced with the Einstein product, so the representation is more natural for developing the MLTI systems theory including notions of stability, reachability, and observability. Moreover, the concept of transfer functions which is commonly used in modern control theory can be extended for MLTI systems, see Definition II.25.

- We can exploit tensor decompositions of the even-order paired tensors A, B, and C to accelerate computations in MLTI systems theory. In particular, if A, B, and C possess low tensor rank structures, we can obtain a low-parameter MLTI representation. In addition, many operations such as the Einstein product and unfolding rank can be achieved efficiently in the tensor decomposition format compared to unfolding-based matrix methods, see remarks in Section 2.5 and 2.6.

- Traditional LTI model reduction and identification techniques such as balanced truncation and eigensystem realization algorithm can be extended using the form of (2.10).

Definition II.25. The *transfer function* $\mathsf{G}(z)$ of (2.10) is given by

$$\mathsf{G}(z) = \mathsf{C} * (z\mathsf{I} - \mathsf{A})^{-1} * \mathsf{B}, \tag{2.11}$$

where z is a complex variable.

We first investigate the elementary solution to the MLTI system (2.10), which is crucial in the analysis of stability, reachability, and observability.

Proposition II.26. *For an unforced MLTI system $\mathcal{X}_{t+1} = \mathcal{A} * \mathcal{X}_t$, the solution for $\mathcal{X}$ at time s, given initial condition $\mathcal{X}_0$, is $\mathcal{X}_s = \mathcal{A}^s * \mathcal{X}_0$ where $\mathcal{A}^s = \mathcal{A} * \mathcal{A} * \overset{s}{\cdots} * \mathcal{A}$.*

The proof is straightforward using the notion of even-order paired tensors and the Einstein product. Applying Proposition II.26, we can write down the explicit solution of (2.10) which takes an analogous form to the LTI system

$$\mathsf{X}_s = \mathsf{A}^s * \mathsf{X}_0 + \sum_{j=0}^{s-1} \mathsf{A}^{s-j-1} * \mathsf{B} * \mathsf{U}_j. \tag{2.12}$$

Lastly, we want to note that one can always transform the MLTI system (2.10) into a LTI system using ψ, i.e., $\mathbf{x}_{t+1} = \psi(\mathsf{A})\mathbf{x}_t + \psi(\mathsf{B})\mathbf{u}_t$, and determine the stability, reachability, and observability using classical matrix techniques.

2.5 MLTI Systems Theory

We now introduce the concepts of stability, reachability, and observability for MLTI systems.

2.5.1 Stability

There are many notions of stability for dynamical systems [17, 64, 110]. For LTI systems, it is conventional to investigate so-called internal stability. Generalizing from LTI systems, the equilibrium point $\mathsf{X} = \mathsf{O}$ (O denotes the zero tensors) of an unforced MLTI system is called *stable* if $\|\mathsf{X}_t\| \leq \gamma \|\mathsf{X}_0\|$ for some $\gamma > 0$, *asymptotically stable* if $\|\mathsf{X}_t\| \to 0$ as $t \to \infty$, and *unstable* if it is not stable.

Lemma II.27. *The tensor eigenvalue problem in Definition II.9 can be represented by $\mathcal{A} = \mathcal{V} * \mathcal{D} * \mathcal{V}^{-1}$ where $\mathcal{D} \in \mathbb{R}^{n_1 \times n_1 \times \cdots \times n_k \times n_k}$ is an U-diagonal tensor with U-eigenvalues on its diagonal, and $\mathcal{V} \in \mathbb{R}^{n_1 \times n_1 \times \cdots \times n_k \times n_k}$ is a mode row block tensor consisting of all the U-eigentensors, i.e., $\mathcal{V} = \left| \mathcal{X}_1 \quad \mathcal{X}_2 \quad \ldots \quad \mathcal{X}_{\Pi_{\mathcal{J}}} \right|$. We have chosen*

$q_p = n_p$ in applying the mode row block tensor operation which enables to express the tensor eigenvalue decomposition in the form analogous to the matrix case.

Proof. The proof follows immediately from Proposition II.20. □

Proposition II.28. *Let λ_j be the U-eigenvalues of A for $j = ivec(\boldsymbol{j}, \mathcal{N})$. For an unforced MLTI system, the equilibrium point $\mathsf{X} = \mathsf{O}$ is:*

- *stable if and only if $|\lambda_j| \leq 1$ for all $j = 1, 2, \ldots, \Pi_{\mathcal{N}}$; for those equal to 1, its algebraic and geometry multiplicities must be equal;*
- *asymptotically stable if $|\lambda_j| < 1$ for all $j = 1, 2, \ldots, \Pi_{\mathcal{N}}$;*
- *unstable if $|\lambda_j| > 1$ for some $j = 1, 2, \ldots, \Pi_{\mathcal{N}}$.*

Proof. We only focus on the case when A has a full set of U-eigentensors. It follows from Proposition II.26 and Lemma II.27 that $\mathsf{A}^s = \sum_{j_1=1}^{n_1} \cdots \sum_{j_k=1}^{n_k} \lambda_j^s \mathsf{W}_{j_1 j_1 \cdots j_k j_k}$ for some even-order square tensors $\mathsf{W}_{j_1 j_1 \cdots j_k j_k}$. Then the results follow immediately. □

Corollary II.29. *Suppose that the HOSVD of A is provided with p-mode singular values. For an unforced MLTI system, the equilibrium point $\mathsf{X} = \mathsf{O}$ is asymptotically stable if the sum of the p-mode singular values square is less than one for any p.*

Proof. Without loss of generality, suppose that $p = 1$. Based on Property 8 in [39], $\sum_{j=1}^{n_1} (\gamma_j^{(1)})^2 = \|\mathsf{A}\|^2 = \|\psi(\mathsf{A})\|^2$. In addition, we know that the magnitude of the maximal eigenvalue of a matrix is less than or equal to its Frobenius norm. Hence, the proof follows immediately from Proposition II.28. □

Corollary II.30. *Suppose that the CPD of A is provided and its factor matrices $\boldsymbol{A}^{(p)}$ and $\boldsymbol{A}^{(q)}$ have all the column vectors orthonormal for at least one odd p and even q. For an unforced MLTI system, the equilibrium point $\mathsf{X} = \mathsf{O}$ is asymptotically stable if the first weight element $\lambda_1 < 1$.*

Lemma II.31. *Given matrices $\boldsymbol{A}^{(p)} \in \mathbb{R}^{n_p \times R}$, the Khatri-Rao product $\boldsymbol{A}^{(1)} \odot \boldsymbol{A}^{(2)} \odot \cdots \odot \boldsymbol{A}^{(k)}$ has all the column vectors orthogonal if at least one of $\boldsymbol{A}^{(p)}$ has all the column vectors orthogonal for $p = 1, 2, \ldots, k$.*

Proof. Suppose that $k = 2$. Based on the properties of Kronecker product, for any $1 \leq p, q \leq R$, the inner product between $\mathbf{a}_p^{(1)} \otimes \mathbf{a}_p^{(2)}$ and $\mathbf{a}_q^{(1)} \otimes \mathbf{a}_q^{(2)}$ is given by

$$(\mathbf{a}_p^{(1)} \otimes \mathbf{a}_p^{(2)})^\top (\mathbf{a}_q^{(1)} \otimes \mathbf{a}_q^{(2)}) = ((\mathbf{a}_p^{(1)})^\top \mathbf{a}_q^{(1)})) \otimes ((\mathbf{a}_p^{(2)})^\top \mathbf{a}_q^{(2)})).$$

Therefore, if $\mathbf{A}^{(1)}$ or $\mathbf{A}^{(2)}$ has all column vectors orthogonal, then the inner product between $\mathbf{a}_p^{(1)} \otimes \mathbf{a}_p^{(2)}$ and $\mathbf{a}_q^{(1)} \otimes \mathbf{a}_q^{(2)}$ is zero for any p, q. □

Now we can prove Proposition II.30.

Proof. Suppose that A has the CPD format (2.5). Applying the unfolding transformation ψ yields

$$\psi(\mathsf{A}) = (\mathbf{A}^{(2k-1)} \odot \cdots \odot \mathbf{A}^{(1)}) \mathbf{S} (\mathbf{A}^{(2k)} \odot \cdots \odot \mathbf{A}^{(2)})^\top,$$

where $\mathbf{S} \in \mathbb{R}^{R \times R}$ is a diagonal matrix containing the weights of the CPD on its diagonal. By Lemma II.31, the two Khatri-Rao products $\mathbf{A}^{(2k-1)} \odot \cdots \odot \mathbf{A}^{(1)}$ and $\mathbf{A}^{(2k)} \odot \cdots \odot \mathbf{A}^{(2)}$ have all the column vectors orthonormal if $\mathbf{A}^{(p)}$ and $\mathbf{A}^{(q)}$ have all the column vectors orthonormal for at least one odd p and even q. Thus, λ_1 will be the largest singular value of $\psi(\mathsf{A})$. In addition, we know that the magnitude of the maximal eigenvalue of a matrix is less than or equal to its largest singular value. Hence, the proof follows immediately from Proposition II.28. Note that there is one special case when the CPD uniqueness condition fails, i.e., $\sum_{p=1}^{2k} \mathcal{K}_{\mathbf{A}^{(p)}} = 2R + 2k - 2$. However, different CPDs, satisfying the orthonormal condition, correspond to the same matrix SVD under ψ up to some orthogonal transformations. □

Corollary II.32. *Suppose that the TTD of* $\tilde{\mathsf{A}} \in \mathbb{R}^{n_1 \times \cdots \times n_k \times n_1 \times \cdots \times n_k}$*, defined in Proposition II.18, is provided with the first* $k-1$ *core tensors left-orthonormal and the last* k *core tensors right-orthonormal. For an unforced MLTI system, the equilibrium point* $\mathsf{X} = \mathsf{O}$ *is asymptotically stable if the largest singular value of* $\bar{\bar{\boldsymbol{A}}}^{(k)}$ *is less than one, where* $\bar{\bar{\boldsymbol{A}}}^{(k)} = \texttt{reshape}(\tilde{A}^{(k)}, R_{k-1}n_k, R_k)$.

Proof. Based on the results of [68], the singular values of $\bar{\bar{\mathbf{A}}}^{(k)}$ are the singular values of $\psi(\mathsf{A})$. In addition, we know that the magnitude of the maximal eigenvalue of a matrix is less than or equal to its largest singular value. Hence, the proof follows immediately from Proposition II.28. □

Remark. Although Proposition II.28 offers strong stability results for unforced MLTI systems, computing U-eigenvalues usually requires an order of $\mathcal{O}(\Pi_{\mathcal{N}}^3)$ number of operations through tensor unfolding and matrix eigenvalue decomposition. To the contrary, Corollary II.29, II.30, and II.32 can be used to determine the stability of MLTI systems much faster. In particular, if the TTD of $\tilde{\mathsf{A}}$ is provided, the time complexity of left- and right-orthonormalization is about $\mathcal{O}(knR^3)$ assuming that $n_p = n$ for all p, and R is the average of the TT-ranks of $\tilde{\mathsf{A}}$ [91]. Moreover, truncating the TT-rank $\tilde{R}_k$ of $\tilde{\mathsf{A}}$ would not alter the largest singular values of $\bar{\bar{\mathbf{A}}}^{(k)}$. Therefore, setting $\tilde{R}_k = 1$ and computing the vector 2-norm of $\bar{\bar{\mathbf{A}}}^{(k)}$ will return the largest singular value of $\psi(\mathsf{A})$.

2.5.2 Reachability

In this and the following subsections, we introduce the definitions of reachability and observability for MLTI systems which are similar to analogous concepts for the LTI systems [17, 64, 110]. We then establish sufficient and necessary conditions for reachability and observability for MLTI systems.

Definition II.33. The MLTI system (2.10) is said to be *reachable* on $[t_0, t_1]$ if, given any initial condition X_0 and any final state X_1, there exists a sequence of inputs U_t that steers the state of the system from $\mathsf{X}_{t_0} = \mathsf{X}_0$ to $\mathsf{X}_{t_1} = \mathsf{X}_1$.

Theorem II.34. *The pair $(\mathcal{A}, \mathcal{B})$ is reachable on $[t_0, t_1]$ if and only if the reachability Gramian*

$$\mathcal{W}_r(t_0, t_1) = \sum_{t=t_0}^{t_1-1} \mathcal{A}^{t_1-t-1} * \mathcal{B} * \mathcal{B}^\top * (\mathcal{A}^\top)^{t_1-t-1}, \tag{2.13}$$

which is a weakly symmetric even-order square tensor, is U-positive definite.

Proof. Suppose $\mathsf{W}_r(t_0, t_1)$ is U-positive definite, and let X_0 be the initial state and X_1 be the desired final state. Choose $\mathsf{U}_t = \mathsf{B}^\top * (\mathsf{A}^\top)^{t_1-t-1} * \mathsf{W}_r^{-1}(t_0, t_1) * \mathsf{V}$ for some constant tensor V. It follows from the solution of the system (2.10) that $\mathsf{X}_{t_1} = \mathsf{A}^{t_1} * \mathsf{X}_0 + \sum_{j=0}^{t_1-1} \mathsf{A}^{t_1-j-1} * \mathsf{B} * \mathsf{U}_t = \mathsf{A}^{t_1} * \mathsf{X}_0 + \mathsf{W}_r(t_0, t_1) * \mathsf{W}_r^{-1}(t_0, t_1) * \mathsf{V} = \mathsf{A}^{t_1} * \mathsf{X}_0 + \mathsf{V}$. Taking $\mathsf{V} = -\mathsf{A}^{t_1} * \mathsf{X}_0 + \mathsf{X}_1$, we have $\mathsf{X}_{t_1} = \mathsf{X}_1$.

We show the converse by contradiction. Suppose $\mathsf{W}_r(t_0, t_1)$ is not U-positive definite. Then there exists $\mathsf{X}_a \neq \mathsf{O}$ such that $\mathsf{X}_a^\top * \mathsf{A}^{t_1-t-1} * \mathsf{B} = \mathsf{O}$ for any t. Take $\mathsf{X}_1 = \mathsf{X}_a + \mathsf{A}^{t_1} * \mathsf{X}_0$, and it follows that $\mathsf{X}_a + \mathsf{A}^{t_1} * \mathsf{X}_0 = \mathsf{A}^{t_1} * \mathsf{X}_0 + \sum_{j=t_0}^{t_1-1} \mathsf{A}^{t_1-j-1} * \mathsf{B} * \mathsf{U}_j$. Multiplying from the left by $\mathsf{X}_a^\top$ yields $\mathsf{X}_a^\top * \mathsf{X}_a = \sum_{j=t_0}^{t_1-1} \mathsf{X}_a^\top * \mathsf{A}^{t_1-j-1} * \mathsf{B} * \mathsf{U}_j = 0$, which implies that $\mathsf{X}_a = \mathsf{O}$, a contradiction. □

Corollary II.35. *If the reachability Gramian $\mathcal{W}_r(t_0, t_1)$ is not M-positive definite, the pair $(\mathcal{A}, \mathcal{B})$ is not reachable on $[t_0, t_1]$.*

Proof. The proof follows immediately from Proposition II.8 and Theorem II.34. □

The reachability Gramian assesses to what degree each state is affected by an input [109]. The infinite horizon reachability Gramian can be computed from the

tensor Lyapunov equation which is defined by

$$\mathsf{W}_r - \mathsf{A} * \mathsf{W}_r * \mathsf{A}^\top = \mathsf{B} * \mathsf{B}^\top. \tag{2.14}$$

By the unfolding property, if the pair (A, B) is reachable over an infinite horizon and all the U-eigenvalues of A have magnitude less than one, one can show that there exists a unique weakly symmetric U-positive definite solution W_r. Solving the infinite horizon reachability Gramian from the tensor Lyapunov equation may be computationally intensive, so a tensor version of the Kalman rank condition is also provided.

Proposition II.36. *The pair (A, B) is reachable if and only if the $n_1 \times m_1 q_1 \times \cdots \times n_k \times m_k q_k$ even-order reachability tensor*

$$\mathcal{R} = \left| \mathsf{B} \quad \mathsf{A} * \mathsf{B} \quad \dots \quad \mathsf{A}^{\Pi_{\mathcal{N}}-1} * \mathsf{B} \right| \tag{2.15}$$

spans $\mathbb{R}^{n_1 \times n_2 \times \cdots \times n_k}$. In other words, $\text{rank}_U(\mathcal{R}) = \Pi_{\mathcal{N}}$.

Proof. The proof follows from Proposition II.26 and the generalized Cayley Hamilton theorem discussed in the tensor eigenvalue problem. □

First, any choice of construction for the mode row block tensor works for the reachability tensor. Second, when $k = 1$, Proposition II.36 simplifies to the famous Kalman rank condition for reachability of LTI systems. The following corollaries involving with HOSVD (multilinear ranks), CPD (CP rank), and TTD (TT-ranks) provide useful necessary or sufficient conditions for reachability of MLTI systems if the reachability tensor $\mathcal{R}$ is given in the HOSVD, CPD, or TTD format.

Corollary II.37. *Given the reachability tensor $\mathcal{R}$ in (2.15), if $\text{rank}_{2p-1}(\mathcal{R}) \neq n_p$ for some p, the pair (A, B) is not reachable.*

Proof. The proof follows immediately from Proposition II.13 and II.36. □

Corollary II.38. *Given the reachability tensor $\mathcal{R}$ in (2.15), if the set of p-mode singular values of $\mathcal{R}$ obtained from the HOSVD contains zero for odd p, the pair (A, B) is not reachable.*

Proof. We know that the number of nonvanishing p-mode singular values equals to its corresponding p-mode multilinear rank. Hence, the result follows immediately from Proposition II.36 and Corollary II.37. □

Corollary II.39. *Given the reachability tensor $\mathcal{R}$ in (2.15), if the CPD of $\mathcal{R}$ satisfies (2.8) with CP rank equal to $\Pi_{\mathcal{N}}$, the pair (A, B) is reachable. Conversely, if the pair (A, B) is reachable, then the CP rank of $\mathcal{R}$ is greater than or equal to $\Pi_{\mathcal{N}}$.*

Proof. The first part of the proof follows immediately from Proposition II.14 and II.36. The second part of the proof follows from the fact that the CP rank of a tensor is greater than or equal to its unfolding rank. □

Corollary II.40. *Given the reachability tensor $\mathcal{R}$ in (2.15), the pair (A, B) is reachable if and only if the kth optimal TT-rank of $\tilde{\mathcal{R}} \in \mathbb{R}^{n_1 \times \cdots \times n_k \times n_1 q_1 \times \cdots \times n_k q_k}$, defined in Proposition II.18, is equal to $\Pi_{\mathcal{N}}$.*

Proof. The proof follows immediately from Proposition II.18 and II.36. □

Remark. Finding the unfolding rank of the reachability tensor $\mathcal{R}$ through tensor unfolding and matrix QR decomposition is computationally expensive, and has a $\mathcal{O}(\Pi_{\mathcal{N}}^3 \Pi_{\mathcal{Q}})$ time complexity. However, if the reachability tensor $\mathcal{R}$ is already given in the tensor decomposition format, computing the unfolding rank can be achieved efficiently based on Corollary II.37, II.39, and II.40. Particularly, if the TTD of $\tilde{\mathcal{R}}$ is provided, we do not need any additional computation to obtain the unfolding rank.

2.5.3 Observability

The results of observability can be simply obtained by the duality principle, similarly to LTI systems.

Definition II.41. The MLTI system (2.10) is said to be *observable* on $[t_0, t_1]$ if any initial state $\mathsf{X}_{t_0} = \mathsf{X}_0$ can be uniquely determined by Y_t on $[t_0, t_1]$.

Theorem II.42. *The pair (A, C) is observable on $[t_0, t_1]$ if and only if the observability Gramian*

$$\mathsf{W}_o(t_0, t_1) = \sum_{t=t_0}^{t_1-1} (\mathsf{A}^\top)^{t-t_0} * \mathsf{C}^\top * \mathsf{C} * \mathsf{A}^{t-t_0}, \tag{2.16}$$

which is a weakly symmetric even-order square tensor, is U-positive definite.

Proof. Suppose that $\mathsf{W}_o(t_0, t_1)$ is U-positive definite and let X_0 be the initial state such that $\mathsf{Y}_t = \mathsf{C} * \mathsf{X}_t = \mathsf{C} * \mathsf{A}^{t-t_0} * \mathsf{X}_0$ for any $t \in [t_0, t_1]$. Multiplying from the left by $(\mathsf{A}^\top)^{t-t_0} * \mathsf{C}^\top$ yields $(\mathsf{A}^\top)^{t-t_0} * \mathsf{C}^\top * \mathsf{Y}_t = (\mathsf{A}^\top)^{t-t_0} * \mathsf{C}^\top * \mathsf{C} * \mathsf{A}^{t-t_0} * \mathsf{X}_0$, which implies that $\sum_{t=t_0}^{t_1-1} (\mathsf{A}^\top)^{t-t_0} * \mathsf{C}^\top * \mathsf{Y}_t = \sum_{t=t_0}^{t_1-1} (\mathsf{A}^\top)^{t-t_0} * \mathsf{C}^\top * \mathsf{C} * \mathsf{A}^{t-t_0} * \mathsf{X}_0 = \mathsf{W}_o(t_0, t_1) * \mathsf{X}_0$. Since $\mathsf{W}_o(t_0, t_1)$ is U-invertible, this equation has a unique solution $\mathsf{X}_0 = \mathsf{W}_o^{-1}(t_0, t_1) \sum_{t=t_0}^{t_1-1} (\mathsf{A}^\top)^{t-t_0} * \mathsf{C}^\top * \mathsf{Y}_t$. Hence, (A, C) is observable on $[t_0, t_1]$.

Again, we show the converse by contradiction. Suppose that $\mathsf{W}_o(t_0, t_1)$ is not U-positive definite. Then there exists $\mathsf{X}_a \neq \mathsf{O}$ such that $\mathsf{C} * \mathsf{A}^{t-t_0} * \mathsf{X}_a = \mathsf{O}$ for any t. Take $\mathsf{X}_{t_0} = \mathsf{X}_0 + \mathsf{X}_a$ for some initial state X_0. Then $\mathsf{Y}_t = \mathsf{C} * \mathsf{A}^{t-t_0} * \mathsf{X}_0 + \mathsf{C} * \mathsf{A}^{t-t_0} * \mathsf{X}_a = \mathsf{C} * \mathsf{A}^{t-t_0} * \mathsf{X}_0$ for any $t \in [t_0, t_1]$. The initial states X_0 and $\mathsf{X}_0 + \mathsf{X}_a$ produce the same output, which implies that (A, C) is not observable on $[t_0, t_1]$, a contradiction. □

Corollary II.43. *If the observability Gramian $\mathsf{W}_o(t_0, t_1)$ is not M-positive definite, the pair (A, C) is not observable on $[t_0, t_1]$.*

The observability Gramian assesses to what degree each state affects future outputs [109]. The infinite horizon observability Gramian can be computed from the tensor Lyapunov equation defined by

$$\mathsf{A}^\top * \mathsf{W}_o * \mathsf{A} - \mathsf{W}_o = -\mathsf{C}^\top * \mathsf{C}. \tag{2.17}$$

If the pair (A, C) is observable and all the U-eigenvalues of A have magnitude less than one, there exists a unique weakly symmetric U-positive definite solution W_o.

The following results can be proved similarly to those in the reachability section.

Proposition II.44. *The pair $(\mathcal{A}, \mathcal{C})$ is observable if and only if the $m_1 n_1 \times n_1 \times \cdots \times m_k n_k \times n_k$ even-order observability tensor*

$$\mathcal{O} = \begin{vmatrix} \mathcal{C} & \mathcal{C} * \mathcal{A} & \dots & \mathcal{C} * \mathcal{A}^{\Pi_{\mathcal{N}}-1} \end{vmatrix}^\top \tag{2.18}$$

spans $\mathbb{R}^{n_1 \times n_2 \times \cdots \times n_k}$. In other words, $\text{rank}_U(\mathcal{O}) = \Pi_{\mathcal{N}}$.

Corollary II.45. *Given the observability tensor $\mathcal{O}$ in (2.18), if $\text{rank}_{2p}(\mathcal{O}) \neq n_p$ for some p, the pair $(\mathcal{A}, \mathcal{C})$ is not observable.*

Corollary II.46. *Given the observability tensor $\mathcal{O}$ in (2.18), if the set of n-mode singular values of $\mathcal{O}$ obtained from the HOSVD contains zero for even p, the pair $(\mathcal{A}, \mathcal{C})$ is not observable.*

Corollary II.47. *Given the observability tensor $\mathcal{O}$ in (2.18), if the CPD of $\mathcal{O}$ satisfies (2.8) with CP rank equal to $\Pi_{\mathcal{N}}$, the pair $(\mathcal{A}, \mathcal{C})$ is observable. Conversely, if the pair $(\mathcal{A}, \mathcal{C})$ is observable, then the CP rank of $\mathcal{O}$ is greater than or equal to $\Pi_{\mathcal{N}}$.*

Corollary II.48. *Given the observability tensor $\mathcal{O}$ in (2.18), the pair $(\mathcal{A}, \mathcal{C})$ is observable if and only if the kth optimal TT-rank of $\tilde{\mathcal{O}} \in \mathbb{R}^{m_1 n_1 \times \cdots \times m_k n_k \times n_1 \times \cdots \times n_k}$, defined in Proposition II.18, is equal to $\Pi_{\mathcal{N}}$.*

2.6 Model Reduction for MLTI Systems

Based on the observations in Section 2.5, it is more natural to manipulate MLTI systems in the tensor decomposition format so that all the computational advantages can be realized. This may also result in a more compressed representation. The problem of model reduction has been studied heavily in the framework of classical control [42, 47, 90]. Methods including proper orthogonal decomposition (POD), scale-separation and averaging, and balanced truncation are applied in many engineering applications when dealing with high-dimensional linear/nonlinear systems [85]. Using generalized CPD/TTD, we propose a new MLTI representation with fewer parameters. Note that we omit colons in each component tensor in this and the following subsections for simplicity (e.g., $\mathsf{A}_r^{(p)} = \mathsf{A}_{r::}^{(p)}$).

Proposition II.49. *The MLTI system (2.10) is equivalent to*

$$\begin{cases} \mathcal{X}_{t+1} = \sum_{r=1}^{R_1} \mathcal{X}_t \times \{\mathcal{A}_r^{(1)}, \ldots, \mathcal{A}_r^{(k)}\} + \sum_{r=1}^{R_2} \mathcal{U}_t \times \{\mathcal{B}_r^{(1)}, \ldots, \mathcal{B}_r^{(k)}\} \\ \mathcal{Y}_t = \sum_{r=1}^{R_3} \mathcal{X}_t \times \{\mathcal{C}_r^{(1)}, \ldots, \mathcal{C}_r^{(k)}\} \end{cases}, \tag{2.19}$$

where R_1, R_2, R_3 are the Kronecker ranks of the system, and $\mathcal{A}^{(p)} \in \mathbb{R}^{R_1 \times n_p \times n_p}$, $\mathcal{B}^{(p)} \in \mathbb{R}^{R_2 \times n_p \times q_p}$, and $\mathcal{C}^{(p)} \in \mathbb{R}^{R_3 \times m_p \times n_p}$.

Proof. The proof follows from Definition II.10 and Proposition II.24. □

Remark. The number of parameters of the MLTI system representation (2.19) is $R_1 \sum_{p=1}^k n_p^2 + R_2 \sum_{p=1}^k n_p q_p + R_3 \sum_{p=1}^k m_p n_p$. If the Kronecker ranks R_1, R_2, R_3 are relatively small, the total number of parameters is much less than that of the MLTI system model (2.10) which is given by $\prod_{p=1}^k n_p^2 + \prod_{p=1}^k n_p q_p + \prod_{p=1}^k m_p n_p$.

The MLTI system representation (2.19) is attractive for systems captured by sparse tensors or tensors with low Kronecker ranks where the two advantages, model

reduction and computational efficiency, can be exploited. In particular, if A, B, and C are fourth-order paired tensors, the generalized CPDs are reduced to matrix SVD problems, see Section 9.2 in [124]. However, there are two major drawbacks. First, for $k > 2$, there is no exact method to compute the Kronecker rank of a tensor [69], and truncating the rank does not ensure a good estimate. Second, current CPD algorithms are not numerically stable, which could result in ill-conditioning during the tensor decomposition and low rank approximation. One way to fix these issues is to replace generalized CPD by generalized TTD in (2.19), which takes a similar form. The algorithms for computing generalized TTD are numerically stable with unique optimal TT-ranks [91]. Most importantly, the TTD-based results obtained in Section 2.5 can be realized in the form of (2.19). For example, we can determine the stability of MLTI systems from the TTD of $\tilde{\mathsf{A}}$ defined in Proposition II.18, which can be obtained from the generalized TTD of A efficiently.

Recall from Section 2.5 that one can always convert the MLTI system (2.10) to an equivalent LTI form and then apply traditional model reduction approaches, e.g., balanced truncation. However, after converting to a matrix form, the low tensor rank structure exploited in the form of (2.19) may not be preserved, and thus low memory requirements cannot be achieved, see Section 2.7. Furthermore, as shown in [25], the MLTI system (2.19) can be used to further develop a higher-order balanced truncation framework directly in the TTD format, which can provide additional computation and memory benefits over unfolding-based model reduction methods.

2.6.1 Explicit Solution and Stability

In addition to using tensor decompositions, we can exploit matrix calculations of the factor matrices $\mathsf{A}_r^{(p)}$ to develop notions including explicit solution and stability for the MLTI system (2.19) which also have lower computational costs compared to

unfolding-based methods.

Proposition II.50 (Solution)**.** *For an unforced MLTI system* $\mathcal{X}_{t+1} = \sum_{r=1}^{R_1} \mathcal{X}_t \times \{\mathcal{A}_r^{(1)}, \mathcal{A}_r^{(2)}, \ldots, \mathcal{A}_r^{(k)}\}$*, the solution for* $\mathcal{X}$ *at time* s*, given initial condition* $\mathcal{X}_0$*, is*

$$\mathcal{X}_s = \sum_{r=1}^{R_1^s} \mathcal{X}_0 \times \{\bar{\mathcal{A}}_r^{(1)}, \bar{\mathcal{A}}_r^{(2)}, \ldots, \bar{\mathcal{A}}_r^{(k)}\}, \tag{2.20}$$

where $\bar{\mathcal{A}}_r^{(p)} = \mathcal{A}_{r_1}^{(n)} \mathcal{A}_{r_2}^{(p)} \ldots \mathcal{A}_{r_s}^{(p)}$ *for* $r = ivec(\{r_1, r_2, \ldots, r_s\}, \{R_1, R_1, \overset{s}{\cdots}, R_1\})$.

Proof. The result follows immediately from Proposition II.12 and II.26. □

If the Kronecker rank R_1 is small, computing the explicit solution using (2.20) can be faster than using the Einstein product (2.9). Additionally, we can assess the stability of the unforced MLTI system of (2.19) based upon the Lyapunov approach.

Proposition II.51 (Stability)**.** *For the unforced MLTI system of (2.19), the equilibrium point* $\mathcal{X} = \mathcal{O}$ *is*

- *stable (i.s.L) if* $\sum_{r=1}^{R_1} \prod_{p=1}^{k} \alpha_r^{(p)} = 1$*;*
- *asymptotically stable (i.s.L) if* $\sum_{r=1}^{R_1} \prod_{p=1}^{k} \alpha_r^{(p)} < 1$*,*

where $\alpha_r^{(p)}$ *denote the largest singular values of* $\mathcal{A}_r^{(p)}$*.*

Proof. Let's consider $V(\mathcal{X}) = \|\mathcal{X}\|$ as the Lyapunov function candidate and let $f(\mathcal{X}) = \sum_{r=1}^{R_1} \mathcal{X} \times \{\mathcal{A}_r^{(1)}, \mathcal{A}_r^{(2)}, \ldots, \mathcal{A}_r^{(k)}\}$. Then it follows that $V(f(\mathcal{X})) - V(\mathcal{X}) = \|\sum_{r=1}^{R_1} \mathcal{X} \times \{\mathcal{A}_r^{(1)}, \mathcal{A}_r^{(2)}, \ldots, \mathcal{A}_r^{(k)}\}\| - \|\mathcal{X}\| \leq \sum_{r=1}^{R_1} \|\mathcal{X} \times \{\mathcal{A}_r^{(1)}, \mathcal{A}_r^{(2)}, \ldots, \mathcal{A}_r^{(k)}\}\| - \|\mathcal{X}\| \leq (\sum_{r=1}^{R_1} \prod_{p=1}^{k} \alpha_r^{(p)} - 1)\|\mathcal{X}\|$, where the last inequality is based on Theorem 6 in [61]. Then the results follow immediately. □

Remark. The computational complexity of finding the matrix SVDs of the factor matrices can be estimated as $\mathcal{O}(kn^3 R_1)$ assuming that $n_p = n$ for all p.

When all the Kronecker ranks of the system $R_1 = R_2 = R_3 = 1$, the MLTI system (2.20) reduces to the Tucker product representation proposed by Surana *et al.* [120], which provides a more direct way to see that the Tucker-based MLTI model is only a special case of the MLTI system (2.10). Additionally, we can obtain stronger stability conditions for the unforced MLTI system in this case.

Proposition II.52 (Stability)**.** *Suppose that $R_1 = 1$ in (2.20), and $\rho^{(p)}$ are the spectral radii of $\mathsf{A}_1^{(p)}$. Then the unforced MLTI system of (2.20) is*

- *stable if and only if $\prod_{p=1}^{k} \rho^{(p)} \leq 1$, and when $\prod_{p=1}^{k} \rho^{(p)} = 1$, their corresponding eigenvalues must have equal algebraic and geometric multiplicity;*
- *asymptotically stable if $\prod_{p=1}^{k} \rho^{(p)} < 1$;*
- *unstable if $\prod_{p=1}^{k} \rho^{(p)} > 1$.*

Proof. Based on Equation (2.25) in [99], $\psi(\mathsf{A}) = \mathsf{A}_1^{(k)} \otimes \mathsf{A}_1^{(k-1)} \otimes \cdots \otimes \mathsf{A}_1^{(1)}$ where the operation $\otimes$ denotes the Kronecker product. Moreover, the U-eigenvalues of A are equal to the products of eigenvalues of these component matrices $\mathsf{A}_1^{(p)}$, and the U-eigenvalues have equal algebraic and geometric multiplicities if and only if the factor eigenvalues have equal multiplicities [18]. Then the results follow immediately from Proposition II.28. □

The above results including Proposition II.50, II.51, and II.52 can be reformulated by replacing the Kronecker rank summation by a series of TT-ranks summations if A, B, and C are given in the generalized TTD format. Finally, the Kronecker product can be used to unfold the MLTI system (2.20) into a LTI system, i.e.,

$$\psi(\mathsf{A}) = \sum_{r=1}^{R_1} \mathsf{A}_r^{(k)} \otimes \mathsf{A}_r^{(k-1)} \otimes \cdots \otimes \mathsf{A}_r^{(1)},$$

and similarly for $\psi(\mathsf{B})$ and $\psi(\mathsf{C})$. Hence, one can apply traditional control theory techniques to determine the MLTI system properties.

2.7 Numerical Examples

We provide four examples to illustrate the MLTI systems theory and model reduction using the techniques developed above. All the numerical examples presented were performed on a Linux machine with 8 GB RAM and a 2.4 GHz Intel Core i5 processor and were conducted in MATLAB R2018a with the Tensor Toolbox 2.6 [112] and the TT toolbox [92].

2.7.1 Reachability and Observability Tensors

In this example, we consider a simple single-input and single-output (SISO) system that is given by (2.9) with $\mathbf{A}_1 = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 0.2 & 0.5 & 0.8 \end{bmatrix}$, $\mathbf{A}_2 = \begin{bmatrix} 0 & 1 \\ 0.5 & 0 \end{bmatrix}$, $\mathbf{B}_1 = \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}$, $\mathbf{B}_2 = \begin{bmatrix} 0 \\ 1 \end{bmatrix}$, $\mathbf{C}_1 = \begin{bmatrix} 1 & 0 & 0 \end{bmatrix}$, $\mathbf{C}_2 = \begin{bmatrix} 1 & 0 \end{bmatrix}$, and the states $\mathsf{X}_t \in \mathbb{R}^{3\times 2}$ are second-order tensors, i.e., matrices. The product of the two spectral radii of $\mathbf{A}_1$ and $\mathbf{A}_2$ is 0.9207, which implies that the system is asymptotically stable. In addition, the reachability and observability tensors based on (2.15) and (2.18) are given by

$$\mathcal{R}_{::11} = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0.8 & 0 \end{bmatrix}, \qquad \mathcal{R}_{::21} = \begin{bmatrix} 0 & 0 & 0.5 \\ 0 & 0 & 0.4 \\ 1 & 0 & 0.57 \end{bmatrix},$$

$$\mathcal{R}_{::12} = \begin{bmatrix} 0.4 & 0 & 0.378 \\ 0.57 & 0 & 0.4849 \\ 0.756 & 0 & 0.6339 \end{bmatrix}, \mathcal{R}_{::22} = \begin{bmatrix} 0 & 0.285 & 0 \\ 0 & 0.378 & 0 \\ 0 & 0.4849 & 0 \end{bmatrix},$$

and

$$\mathcal{O}_{::11} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0.5 \end{bmatrix}, \mathcal{O}_{::21} = \begin{bmatrix} 0 & 0 & 0 \\ 0.04 & 0.15 & 0.285 \\ 0 & 0 & 0 \end{bmatrix},$$

$$\mathcal{O}_{::12} = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix}, \quad \mathcal{O}_{::22} = \begin{bmatrix} 0.1 & 0.25 & 0.4 \\ 0 & 0 & 0 \\ 0.057 & 0.1825 & 0.378 \end{bmatrix},$$

respectively. We compute the TTDs of the permuted tensors $\tilde{\mathcal{R}}$ and $\tilde{\mathcal{O}}$, respectively and observe that $\text{rank}_U(\mathcal{R}) = 6$ and $\text{rank}_U(\mathcal{O}) = 6$. The system therefore is both reachable and observable.

2.7.2 Kronecker Rank/TT-Rank Approximation

In this example, we consider a SISO MLTI system (2.10) with random sparse tensors $\mathsf{A} \in \mathbb{R}^{3\times3\times3\times3\times3\times3}$, $\mathsf{B} \in \mathbb{R}^{3\times3\times3}$, and $\mathsf{C} \in \mathbb{R}^{3\times3\times3}$. According to Algorithm 1, we compute the generalized CPDs of A, B, and C using the tensor toolbox function `cp_als` with estimated Kronecker ranks $R_1 = 49$, $R_2 = 2$, and $R_3 = 2$, respectively. Note that the number of parameters in the system with full Kronekcer ranks could be greater than that for the original system. We then fix R_2 and R_3 and gradually truncate R_1, since R_1 is most critical in determining the number of parameters of the reduced system. As we can see in the table, the number of parameters decreases dramatically as R_1 decreases. In order to assess the approximation error resulting from this truncation, we compute the relative error using the $\mathcal{H}$-infinity norm $\|\cdot\|_\infty$ between the full system and reduced system transfer functions based on (2.11). In particular, we find that when $R_1 = 10$, the reduced MLTI system is still close to the original system with $\mathcal{H}$-infinity norm relative error of 0.0888.

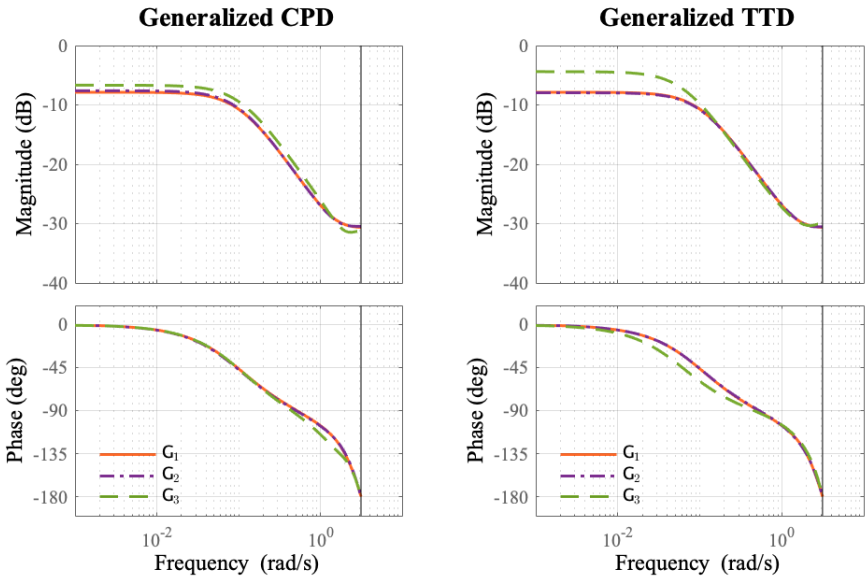


Figure 2.2: **Bode diagrams.** G_1, G_2, and G_3 are the transfer functions for the three reduced MLTI systems corresponding to Table 2.1, respectively. One may view G_1 as the transfer function of the original system. Since the function `cp_als` is not numerically stable, the results may not be exactly consistent with Table 2.1 for those obtained by generalized CPD.

Table 2.1: **Kronecker rank/TT-ranks approximations of the MLTI system.** We omit the first and last trivial TT-ranks in the generalized TTDs of A, B, and C.

	Reduced Ranks	**# Parameters**	$\frac{\|\mathbf{G}_{\mathbf{full}}-\mathbf{G}_{\mathbf{red}}\|_\infty}{\|\mathbf{G}_{\mathbf{full}}\|_\infty}$
Full System	-	783	-
Generalized CPD	$49, 2, 2$	1359	1.58×10^{-10}
	$20, 2, 2$	576	0.0223
	$10, 2, 2$	306	0.0888
Generalized TTD	$\{7, 8\}, \{1, 2\}, \{2, 2\}$	678	4.39×10^{-15}
	$\{7, 6\}, \{1, 2\}, \{2, 2\}$	534	0.0099
	$\{7, 5\}, \{1, 2\}, \{2, 2\}$	462	0.4911

We repeat a similar process for TT-ranks approximation through generalized TTD. The results are shown in the same table. We find that both generalized CPD and TTD can achieve efficient model reduction while keeping the approximation errors low. Generalized TTD in particular achieves better accuracy for a similar number of reduced parameters as compared to generalized CPD, but the latter can maintain a resonable approximation error with an even lower number of parameters. The Bode diagrams for the reduced MLTI systems are shown in Figure 2.2. Note that in this example, we manually selected the truncation to study the tradeoff between

Table 2.2: **Memory consumption comparison between the generalized TTD- and balanced truncation-based methods.** We reported the TT-ranks of A, B, and C (ignoring the first and last trivial TT-ranks) and the number of singular values retained in the Hankel matrix during the balanced truncation.

	Ranks	**# Parameters**	$\frac{\|\mathsf{G}_{\mathbf{full}}-\mathsf{G}_{\mathbf{red}}\|_\infty}{\|\mathsf{G}_{\mathbf{full}}\|_\infty}$
Full System	-	139968	-
Generalized TTD	{6,6}, {6,6}, {6,6}	5184	3.98×10^{-15}
Balanced Truncation	200	120000	0.0169
	100	30000	0.1001
	40	4800	0.2360

number of parameters in the reduced system and the approximation error.

2.7.3 Memory Consumption Comparison

In this example, we consider a multiple-input and multiple-output (MIMO) MLTI system (2.10) with random even-order paired tensors $\mathsf{A}, \mathsf{B}, \mathsf{C} \in \mathbb{R}^{6\times6\times6\times6\times6\times6}$ that possess low TT-ranks. We compare the memory consumptions of the generalized TTD-based representation (2.20) with the reduced models obtained from the unfolding-based balanced truncation. The results are shown in Table 2.2. One can clearly see that if the MLTI systems possess low TT-ranks structure, the generalized TTD-based approach achieves much better accuracy for a similar number of parameters as compared to balanced truncation.

2.7.4 Computational Time Comparison

In this example, we consider unforced MLTI systems (2.20) with random sparse even-order paired tensors $\mathsf{A} \in \mathbb{R}^{2\times2\times\overset{p}{\cdots}\times2\times2}$ in the generalized TTD format such that $\psi(\mathsf{A}) \in \mathbb{R}^{2^p\times2^p}$. We compare the run time of Corollary II.32 with the matrix SVD of $\psi(\mathsf{A})$ for determining the stability of the systems. The results are shown in Table 2.3. When $p \geq 10$, the TTD-based method for finding the largest singular value of $\psi(\mathsf{A})$ exhibits a signficant time advantage compared to the matrix SVD-based method for which the time increases exponentially.

Table 2.3: **Run time comparison between the TTD- and SVD-based methods in finding the largest singular value of $\psi(\mathbf{A})$.** For the TTD-based method, we reported computational time includes conversion from the generalized TTD of A to the TTD of $\tilde{\mathsf{A}}$ and left- and right-orthonormalization.

p	TTD(s)	SVD(s)	$\sigma_{\max}$	Relative error	Stability
6	0.0399	6.8551×10^{-4}	0.8082	1.3738×10^{-16}	asy. stable
8	0.0491	0.0439	0.9626	4.1523×10^{-15}	asy. stable
10	0.0591	0.4979	0.8645	3.8527×10^{-15}	asy. stable
12	0.0909	30.7663	0.8485	5.7573×10^{-15}	asy. stable
14	0.2623	2115.1	0.9984	1.3566×10^{-14}	asy. stable

2.8 Discussion

While tensor unfolding to a matrix form provides the advantage of leveraging highly optimized matrix algebra libraries, in doing so however one may not be able to exploit the higher-order hidden patterns/structures, e.g., redundancy/correlations, present in the tensor. For instance, in the context of solving PDEs, Brazell *et al.* [16] found that higher-order tensor representations preserve low bandwidth, thereby keeping the computational cost and memory requirement low. As shown in Section 2.7, TTD-based methods are more efficient in terms of computational speed and memory requirements compared to unfolding-based methods when the MLTI systems have low TT-ranks structure. Although CPD typically offers better compression than TTD, the computation of CP rank is NP-hard, and the lower rank approximations can be ill-posed [40]. TTD is more suitable for numerical computations with well developed TT-algebra [91]. Basic tensor operations such as addition, the Einstein product, Frobenius norm, block tensor, solution to multilinear equations, and tensor pseudoinverse, can be computed and maintained in the TTD format, without requiring full tensor representation. This can provide significant computational advantages in finding the reachability/observability tensors and associated unfolding ranks according to Corollary II.40 and II.48, and in obtaining solution of the tensor Lyapunov equations. For details, we refer the reader to [25] and the references therein.

Another line of approach is to exploit the isomorphism property to build algorithms directly in the full tensor format from existing methods. For example, Brazell *et al.* [16] proposed higher-order biconjugate gradient (HOBG) method for solving multilinear systems which can be used for computing U-inverses and MLTI system transfer functions. Analogously, one can generalize the matrix-based Rayleigh quotient iteration method for computing U-eigenvalues (which can be used for determining MLTI system stability) directly in the tensor form. However, the computational efficiency of this type of method remains to be investigated. Finally, one can combine tensor algebra-based and matrix-based methods to provide the advantages of both approaches as hybrid methods, see some examples in [25] in the context of MLTI model reduction. In future, it would be worthwhile to systematically explore which of the above mentioned approaches or combination thereof is best given the problem structure.

CHAPTER III

Tensor Entropy for Uniform Hypergraphs

Many real world complex systems can be decomposed and analyzed using networks. There are two classical types of complex networks, scale-free networks and small world networks, which have provided insights in social sciences, cell biology, neuroscience, and computer science [4, 8, 128]. Recent advancements in genomics technology, such as genome-wide chromosome conformation capture (Hi-C), have inspired us to consider the human genome as a dynamic graph [103, 106]. Studying such dynamic graphs often requires identifying the changes in network properties, such as degree distribution, path lengths, and clustering coefficients [65, 87, 104].

Numerous techniques have been developed for anomaly detection based on evaluating similarities between graphs [3, 94]. A classic approach for detecting anomalous timestamps during the evolution of dynamic graphs is comparing two consecutive graphs using distance or similarity functions. A comprehensive survey on similarity measures can be found in [22]. Two popular measures, Hamming distance [52] and Jaccard distance [75], are often problem-specific and sensitive to small perturbations or scaling, thus providing limited understanding of variations between graphs [44]. Model-agnostic approaches, such as eigenvalue-based/spectral measures, are more flexible in their representations and interpretations. Therefore, these approaches can

more appropriately quantify the global structural complexity of graphs [43, 76, 115].

The von Neumann entropy of a graph, first introduced by Braunstein *et al.* [14], is a spectral measure used in structural pattern recognition. The intuition behind the measure is linking the graph Laplacian to density matrices from quantum mechanics, and measuring the complexity of the graphs via the von Neumman entropy of the corresponding density matrices [86]. Additionally, the measure can be viewed as the information theoretic Shannon entropy, i.e.,

$$\mathrm{S} = -\sum_{j} \eta_j \ln \eta_j, \tag{3.1}$$

where η_j are the normalized eigenvalues of the Laplacian matrix of a graph such that $\sum_j \eta_j = 1$. Passerini and Severini [95] observed that the von Neumman entropy of a graph tends to grow with the number of connected components, the reduction of long paths, and the increase of nontrivial symmetricity, and suggested that it can be viewed as a measure of regularity. They also showed that the entropy (3.1) is upper bounded by $\ln(n-1)$ where n is the number of nodes of a graph.

Hypergraph entropy has been recently explored by Hu *et al.* [57], Bloch *et al.* [13], and Wang *et al.* [125]. In [57], Hu *et al.* utilized the probability distribution of the node degrees to fit into the Shannon entropy formula and established its lower and upper bounds for special hypergraphs. The degree-based entropy solely depends on the degree distributions of hypergraphs, thus failing to capture comprehensive information, such as path lengths and clustering patterns. Similarly, [13] defined a hypergraph entropy using incidence matrices, but this formulation may lose higher-order structural information hidden in the hypergraphs, such as nontrivial symmetricity. Furthermore, [125] constructed a hypergraph entropy using node weighting scores, calculated from a density estimate technique, to select significant lines for model fitting. The score of a node (i.e, a fitting line) relies on residuals

measured with the Sampson distance under some kernel functions from the line to the data points. Hence, the entropy is difficult to compute and cannot be directly applied to general forms of hypergraphs.

Based on the works [39, 86, 95], we present a new spectral measure called tensor entropy, which can decipher topological attributes of uniform hypergraphs. The key contributions of this chapter are as follows:

- We introduce a new notion of entropy for uniform hypergraphs based on the higher-order singular value decomposition (HOSVD) of the corresponding Laplacian tensors. We establish results on the lower and upper bounds of the proposed tensor entropy, and provide a formula for computing the entropy of complete uniform hypergraphs.

- We adapt a fast and memory efficient tensor train decomposition (TTD)-based computational framework in computing the proposed tensor entropy for uniform hypergraphs.

- We create two simulated datasets, a hyperedge growth model and a Watts-Strogatz model for uniform hypergraphs. We demonstrate that the proposed tensor entropy is a measure of regularity relying on the node degrees, path lengths, clustering coefficients, and nontrivial symmetricity for uniform hypergraphs. Further, we present applications to three real world examples: a primary school contact dataset, a mouse neuron endomicroscopy dataset, and a cellular reprogramming dataset. The final example demonstrates the efficacy of the TTD-based computational framework in computing the proposed tensor entropy.

- We perform preliminary explorations of tensor eigenvalues in the entropy com-

putation and the notion of robustness for uniform hypergraphs.

This chapter is organized into five sections. We first extend graph-based definitions to describe uniform hypergraphs' structural properties in Section 3.1. We then propose a new form of entropy for uniform hypergraphs with several theoretical results in Section 3.2. In Section 3.3, we exploit the tensor train decomposition to accelerate the tensor entropy computation. Six numerical examples are presented in Section 3.4. Finally, we introduce the notion of robustness for uniform hypergraphs in Section 3.5. All the content of this chapter has published in [23].

3.1 Hypergraph Structural Prpoperties

In the following, we extend several graph-based definitions to describe the structural properties of uniform hypergraphs.

Definition III.1. Given a hypergraph G, the *index of dispersion* of the node degree distribution of G is defined to be the ratio of its variance to its mean.

Definition III.2. Given a k-uniform hypergraph G with n nodes, the *average path length* of G is defined by

$$L_{\text{avg}} = \frac{1}{n(n-1)} \sum_{j \neq i} d(v_j, v_i), \tag{3.2}$$

where $d(v_j, v_i)$ denotes the shortest distance between v_j and v_i.

Definition III.3. Given a k-uniform hypergraph G with n nodes, the *average clustering coefficient* of G is defined by

$$\begin{aligned} C_j &= \frac{|\{e_{ilp} : v_i, v_l, v_p \in \mathbf{N}_j, e_{ilp} \in \mathbf{E}\}|}{\binom{|\mathbf{N}_j|}{k}}, \\ \Rightarrow C_{\text{avg}} &= \frac{1}{n} \sum_{j=1}^{n} C_j, \end{aligned} \tag{3.3}$$

where $\mathbf{N}_j$ is the set of nodes that are immediately connected to v_j, and $\binom{|\mathbf{N}_j|}{k} = \frac{|\mathbf{N}_j|!}{(|\mathbf{N}_j|-k)!k!}$ returns the binomial coefficients. If $|\mathbf{N}_j| < k$, we set $C_j = 0$.

Definition III.4. Given a k-uniform hypergraph G, the *small world coefficient* of G is defined by

$$\sigma = \frac{C_{\text{avg}}/C_{\text{rand}}}{L_{\text{avg}}/L_{\text{rand}}}, \tag{3.4}$$

where C_{avg} and L_{avg} are the average clustering coefficient and path length of G, respectively, and C_{rand} and L_{rand} are the same quantities of its equivalent random uniform hypergraph.

The equivalent random uniform hypergraphs of G can be constructed analogously as Erdős-Rényi graphs [60], i.e., randomly generating uniform hypergraphs that share the same numbers of nodes and hyperedges with G. All these definitions can be used for quantifying the performance of entropy measures for uniform hypergraphs.

3.2 Tensor Entropy

Similar to von Neumann entropy, we exploit the spectrum of Laplacian tensors from HOSVD to define the notion of tensor entropy for uniform hypergraphs.

Definition III.5. Let G be a k-uniform hypergraph with n nodes. The *tensor entropy* of G is defined by

$$\mathrm{S} = -\sum_{j=1}^{n} \hat{\gamma}_j \ln \hat{\gamma}_j, \tag{3.5}$$

where $\hat{\gamma}_j$ are the normalized k-mode singular values of L such that $\sum_{j=1}^{n} \hat{\gamma}_j = 1$.

The convention $0 \ln 0 = 0$ is used if $\hat{\gamma}_j = 0$. The k-mode singular values of L can be computed from the matrix SVD of the k-mode unfolding $\mathbf{L}_{(k)}$, which results in a $\mathcal{O}(n^{k+1})$ time complexity and a $\mathcal{O}(n^k)$ space complexity, see Algorithm 3. Since L

is supersymmetric, any mode unfolding of L would yield the same unfolding matrix with the same singular values. Moreover, the tensor entropy (3.5) can be viewed as a variation of von Neumann entropy defined for graphs, in which we regard $c\mathbf{L}_{(k)}\mathbf{L}_{(k)}^{\top}$ as the density matrix for some normalization constant c [15, 86, 95]. In particular, when $k = 2$, the tensor entropy is reduced to the classical von Neumann entropy for graphs. Like the eigenvalues of Laplacian matrices, the k-mode singular values play a significant role in identifying the structural patterns for uniform hypergraphs.

Algorithm 3 Computing tensor entropy from SVD

1: Given a k-uniform hypergraph G with n nodes
2: Construct the adjacency tensor $\mathsf{A} \in \mathbb{R}^{n\times n\times\cdots\times n}$ from
G and compute the Laplacian tensor $\mathsf{L} = \mathsf{D} - \mathsf{A}$
where D is the degree tensor
3: Find the k-mode unfolding of L, i.e., $\mathbf{L}_{(k)} = \texttt{reshape}(\mathsf{L}, n, n^{k-1})$
4: Compute the economy-size matrix SVD of $\mathbf{L}_{(k)}$, i.e., $\mathbf{L}_{(k)} = \mathbf{USV}^{\top}$ and let $\{\gamma_j\}_{j=1}^{n} = \texttt{diag}(\mathbf{S})$
5: Set $\hat{\gamma}_j = \frac{\gamma_j}{\sum_{i=1}^{n}\gamma_i}$ and compute $\mathrm{S} = -\sum_{j=1}^{n}\hat{\gamma}_j \ln\hat{\gamma}_j$
6: **return** The tensor entropy S of G.

Lemma III.6. *Suppose that G is a k-uniform hypergraph with $k \geq 3$. Then L has a k-mode singular value zero, with multiplicity p, if and only if G contains p number of non-connected nodes.*

Proof. The result follows immediately from the definitions of Laplacian tensor and k-mode unfolding of L. □

The multiplicity of the zero k-mode singular value can be used to determine the number of connected components of uniform hypergraphs. Moreover, one can derive the lower bound of tensor entropy based on Lemma III.6.

Proposition III.7. *Suppose that G is a k-uniform hypergraph with n nodes and nonempty hyperedge set $\boldsymbol{E}$ for $k \geq 3$. Then the minimum tensor entropy of G is given by*

$$\mathrm{S}_{\min} = \ln k. \tag{3.6}$$

Proof. Since G is a k-uniform hypergraph on n nodes, the maximum multiplicity of the zero normalized k-mode singular value of L is $n-k$ according to Lemma III.6. In addition, the other normalized k-mode singular values of are necessarily $\frac{1}{k}$. Hence, it is straightforward to show that $\mathrm{S}_{\min} = \ln k$. □

Every k-uniform hypergraph can achieve the minimum tensor entropy $\ln k$. As the number of nodes contained in hyperedges increases, the lower limit of tensor entropy also increases. In the following, we present results about the upper limit of tensor entropy and its relation to regular uniform hypergraphs.

Proposition III.8. *Suppose that G is a k-uniform hypergraph with n nodes for $k \geq 3$. Then the maximum tensor entropy of G occurs when it is a 1-regular uniform hypergraph, and is given by*

$$\mathrm{S}_{\max} = \ln n. \tag{3.7}$$

Proof. Since G is a k-uniform hypergraph on n nodes, the maximum tensor entropy occurs when the multiplicity of a normalized k-mode singular value of L is n. Based on the definitions of Laplacian tensor and k-mode unfolding, for a 1-regular uniform hypergraph, the number of nonzero elements in $\mathbf{L}_{(k)}$ are fixed for jth row with one entry

$$(\mathbf{L}_{(k)})_{j[1+\sum_{m=1}^{k-1}(j-1)n^{m-1}]} = 1$$

and $(k-1)!$ entries $-\frac{1}{2}$ for $j = 1, 2, \ldots n$. Moreover, since all the hyperedges contain distinct nodes, the column indices of the nonzero entries are unique for $\mathbf{L}_{(k)}$. Thus, $\mathbf{L}_{(k)}\mathbf{L}_{(k)}^{\top}$ is a diagonal matrix with equal diagonal elements, and the result follows immediately. □

Proposition III.9. *Suppose that G is a k-uniform hypergraph with n nodes for $k \geq 3$. If $\log_k n$ is an integer, then the maximum tensor entropy of G can be achieved when it is a d-regular uniform hypergraph for $1 \leq d \leq \log_k n$.*

Proof. Suppose that $\log_k n$ is an integer. Any two hyperedges of a d-regular uniform hypergraph contain at least $k-1$ distinct nodes for $1 \leq d \leq \log_k n$. Similar to Proposition III.8, it can be shown that $\mathbf{L}_{(k)}\mathbf{L}_{(k)}^{\top}$ is a diagonal matrix with equal diagonal elements. Therefore, the result follows immediately. □

According to Proposition III.8 and III.9, not all uniform hypergraphs with n nodes can achieve the tensor entropy $\ln n$. However, when $\log_k n$ is an integer, one can utilize tensor entropy to measure the regularity of uniform hypergraphs. Moreover, if G contains p number of non-connected nodes and $\log_k (n-p)$ is an integer, then $\mathrm{S}_{\max} = \ln (n-p)$. Therefore, larger tensor entropy can be obtained with more connected components in this case. Next, we establish results regarding complete uniform hypergraphs.

Proposition III.10. *Suppose that G is a complete k-uniform hypergraph with n nodes for $k \geq 3$. Then the tensor entropy of G is given by*

$$\begin{aligned} \mathrm{S}_c = & \frac{(1-n)\alpha}{(n-1)\alpha+\beta} \ln \frac{\alpha}{(n-1)\alpha+\beta} \\ & - \frac{\beta}{(n-1)\alpha+\beta} \ln \frac{\beta}{(n-1)\alpha+\beta} \end{aligned} \tag{3.8}$$

where,

$$\alpha = \Big(\frac{\Gamma(n)(\Gamma(n-k+1)+\Gamma(n))}{\Gamma(k)^2\Gamma(n-k+1)^2} - \frac{\Gamma(n-1)}{\Gamma(k)^2\Gamma(n-k)}\Big)^{\frac{1}{2}}, \tag{3.9}$$

$$\beta = \Big(\frac{\Gamma(n)(\Gamma(n-k+1)+\Gamma(n))}{\Gamma(k)^2\Gamma(n-k+1)^2} + \frac{(n-1)\Gamma(n-1)}{\Gamma(k)^2\Gamma(n-k)}\Big)^{\frac{1}{2}}, \tag{3.10}$$

and $\Gamma(\cdot)$ is the Gamma function.

Proof. Based on the definitions of Laplacian tensor and k-mode unfolding, the matrix $\mathbf{L}_{(k)}\mathbf{L}_{(k)}^\top \in \mathbb{R}^{n\times n}$ is given by

$$\mathbf{L}_{(k)}\mathbf{L}_{(k)}^\top = \begin{bmatrix} d & \rho & \rho & \dots & \rho \\ \rho & d & \rho & \dots & \rho \\ \rho & \rho & d & \dots & \rho \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ \rho & \rho & \rho & \dots & d \end{bmatrix},$$

where,

$$d = \binom{n-1}{k-1}^2 + \binom{n-1}{k-1}\frac{1}{(k-1)!}, \text{ and } \rho = \frac{T_{n-k}}{(k-1)!^2}.$$

Here T_m are the k-simplex numbers (e.g., when $k = 3$, T_m are the triangular numbers). Moreover, the eigenvalues of $\mathbf{L}_{(k)}\mathbf{L}_{(k)}^\top$ are $d - \rho$ with multiplicity $n - 1$ and $d + (n-1)\rho$ with multiplicity 1. Hence, the result follows immediately. We write all the expressions using the Gamma function for simplicity. □

From Proposition III.9 and III.10, for arbitrary k-uniform hypergraph with n nodes and $k \geq 3$, S_c could be smaller than the entropies of other d-regular hypergraphs, and $\mathrm{S}_c \leq \mathrm{S}_{\max} \leq \ln n$. Furthermore, it can be shown that when n becomes large, $\mathrm{S}_{\max} \approx \ln n$.

Corollary III.11. *Suppose that G is a complete k-uniform hypergraph with n nodes for $k \geq 3$. Then the tensor entropy $\mathrm{S}_c \to \ln n$ as $n \to \infty$.*

Proof. As $n \to \infty$, $\frac{\Gamma(n)(\Gamma(n-k+1)+\Gamma(n))}{\Gamma(k)^2\Gamma(n-k+1)^2} \gg \frac{\Gamma(n-1)}{\Gamma(k)^2\Gamma(n-k)}$ for fixed k. Thus, $\alpha \approx \beta$, and the result follows immediately. □

In Section 3.3, we will show evidence that the tensor entropy (3.5) is a measure of regularity for general uniform hypergraphs. Large tensor entropy is characterized

by the large number of connected nodes, high uniformity of node degrees, short path lengths, and high level of nontrivial symmetricity. The entropy is small for uniform hypergraphs with large *cliques* and long path lengths, i.e., hypergraphs in which the nodes form highly connected clusters. Tensor entropy is also related to the clustering coefficients of uniform hypergraphs in a very nuanced way.

3.3 Numerical Method via Tensor Trains

In reality, hypergraphs like co-authorship networks and protein-protein interaction networks exist in a very large scale, and computing the tensor entropy using the economy-size matrix SVD could be computationally expensive. Klus *et al.* [68] exploited TTD to efficiently calculate the Moore-Penrose (MP) inverse of the matrix obtained from any chosen unfolding of a given tensor. TTD provides a good compromise between numerical stability and level of compression, and has an associated algebra that facilitates computations. We thus adapt the framework of Klus *et al.* for the computation of the tensor entropy, see Algorithm 4. In Step 2, we assume that the construction of adjacency and degree tensors in the TT-format can be achieved due to their simple structures. In Step 4, the left- and right-orthonormalization algorithms can be found in [68]. The computation and memory complexities of Algorithm 4 are estimated as $\mathcal{O}(knr^3)$ and $\mathcal{O}(knr^2)$, respectively, where r can be viewed as the "average" rank of the TT-ranks. Both complexities are much lower than those from Algorithm 3 when r is small.

3.4 Numerical Examples

All the numerical examples presented were performed on a Linux machine with 8 GB RAM and a 2.4 GHz Intel Core i5 processor in MATLAB 2018b. The last example also used the MATLAB TT-Toolbox by Oseledets *et al.* [92].

Algorithm 4 Computing tensor entropy from TTD

1: Given a k-uniform hypergraph G with n nodes
2: Construct the adjacency and degree tensors $\mathsf{A}, \mathsf{D} \in \mathbb{R}^{n\times n\times\cdots\times n}$ in the TT-format from G
3: Compute the Laplacian tensor $\mathsf{L} = \mathsf{D} - \mathsf{A}$ with core tensors $\mathsf{X}^{(p)}$ and TT-ranks $\{R_0, R_1, \ldots, R_k\}$ based on the tensor train summation operation
4: Left-orthonormalize the first $k-2$ cores and right-orthonormalize the last core of L
5: Compute the economy-size matrix SVD of $\bar{\mathbf{X}}^{(k-1)}$, i.e., $\bar{\mathbf{X}}^{(k-1)} = \mathbf{USV}^\top$ for $s = \texttt{rank}(\mathbf{S})$ and let $\{\gamma_j\}_{j=1}^s = \texttt{diag}(\mathbf{S})$
6: Set $\hat{\gamma}_j = \frac{\gamma_j}{\sum_{i=1}^s \gamma_i}$ and compute $\mathrm{S} = -\sum_{j=1}^s \hat{\gamma}_j \ln \hat{\gamma}_j$
7: **return** The tensor entropy S of G.

3.4.1 Hyperedge Growth Model

We consider the case where the number of nodes is fixed and new hyperedges are iteratively added to the uniform hypergraph. Figure 3.1 presents the hyperedge growth evolution of a 3-uniform hypergraph with 7 nodes, and it describes the tensor entropy maximization and minimization evolutions. In addition to plotting the two entropy trajectories, we also compute some statistics of the structural properties including average shortest path length, index of dispersion of the degree distribution, and average clustering coefficient of the hypergraphs during the two evolutions, see Figure 3.2. If the two nodes are disconnected, we set the distance between them to be 4 for the purpose of visualization in Figure 3.2B.

Let's denote the hypergraphs that achieve maximum (or minimum) tensor entropy at step j as $\mathsf{G}_{\max}^{(j)}$ (or $\mathsf{G}_{\min}^{(j)}$) for $j = 1, 2, \ldots, 35$. Similar to maximizing graph entropy, maximizing the tensor entropy will first connect all the nodes and then prefer to choose lower degree nodes with larger average geodesic distances, i.e., finding the geodesic distances between each pair in the triples and taking the mean, see Figure 3.1 and 3.2B. The average geodesic distances may lose importance if one wants to predict the next step as the hypergraph becomes complex. Moreover, the nodes of $\mathsf{G}_{\max}^{(j)}$ tend to have "almost equal" or equal degree which leads to a low index of dispersion, see Figure 3.2C. In particular, $\mathsf{G}_{\max}^{(j)}$ are the $\frac{k}{n}j$-regular hypergraphs for the early stages

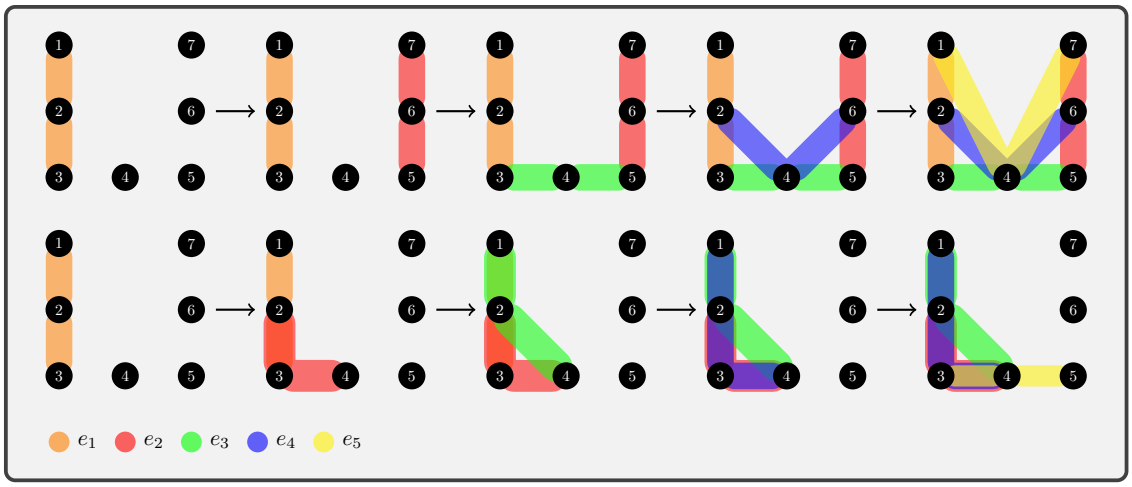


Figure 3.1: **Tensor entropy maximization/minimization.** The top row describes the first five stages of the tensor entropy maximization evolution with a growing number of hyperedges in the order of $e_1 = \{1, 2, 3\}$, $e_2 = \{5, 6, 7\}$, $e_3 = \{3, 4, 5\}$, $e_4 = \{2, 4, 6\}$, and $e_5 = \{1, 4, 7\}$. The bottom row reports the first five stages of the tensor entropy minimization process with a growing number of hyperedges in the order of $e_1 = \{1, 2, 3\}$, $e_2 = \{2, 3, 4\}$, $e_3 = \{1, 2, 4\}$, $e_4 = \{1, 3, 4\}$, and $e_5 = \{3, 4, 5\}$.

of the evolution, i.e., $j = 7, 14$. However, as the hypergraph becomes dense, it is possible that $\mathsf{G}_{\max}^{(j)}$ will miss the regularity, i.e., $j = 21$. Additionally, the average clustering coefficient, in general, grows with increase of hyperedges, but the average growth rate for $\mathsf{G}_{\max}^{(j)}$ is lower than that for $\mathsf{G}_{\min}^{(j)}$, see Figure 3.2D. Furthermore, nontrivial symmetricity plays a role in maximizing the tensor entropy. For example, in $\mathsf{G}_{\max}^{(3)}$, the nodes $\{1, 2, 6\}$ and $\{2, 4, 6\}$ have the same average geodesic distances (both are equal to $\frac{7}{3}$), and the maximized tensor entropy returns the more symmetric $\mathsf{G}_{\max}^{(4)}$. We also find that candidate hyperedges that intersect more existing hyperedges would return higher tensor entropy, which also explains the above example.

However, there exists one huge disparity between the von Neumann graph entropy and the tensor entropy. The tensor entropy can temporarily decrease during the maximizing process as seen in Figure 3.2A. We observe that once the maximization evolution reaches some regularity or high level of nontrivial symmetricity, and the next step breaks such regularity or symmetricity, the tensor entropy will

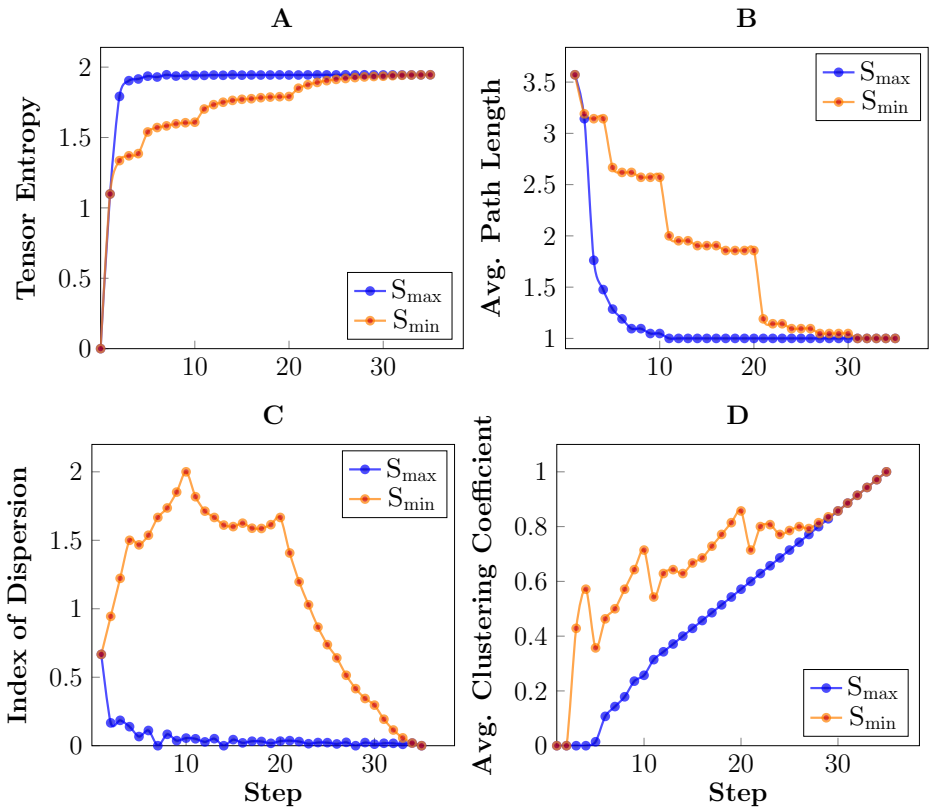


Figure 3.2: **Hyperedge growth model features.** (A), (B), (C), and (D) Trajectories of tensor entropy, average path length, index of dispersion and average clustering coefficient with respect to the hyperedge adding steps.

decrease. In other words, for these highly regular or highly symmetric $\mathsf{G}_{\text{max}}^{(j)}$, the corresponding tensor entropies $\mathrm{S}_{\text{max}}^{(j)}$ achieve local maxima. On the other hand, $\mathrm{S}_{\text{min}}^{(j)}$, the tensor entropies of $\mathsf{G}_{\text{min}}^{(j)}$, are similar to the von Neumann graph entropy. Minimizing the tensor entropy would result in the formations of complete sub-hypergraphs (*cliques*), see Figure 3.1. We can detect large jumps and drops in the next steps after completions of the sub-hypergraphs in Figure 3.2A and B, respectively. In order to make the discoveries more convincing, we repeated the same processes for k-uniform hypergraphs with different number of nodes and values of k, and observed similar results.

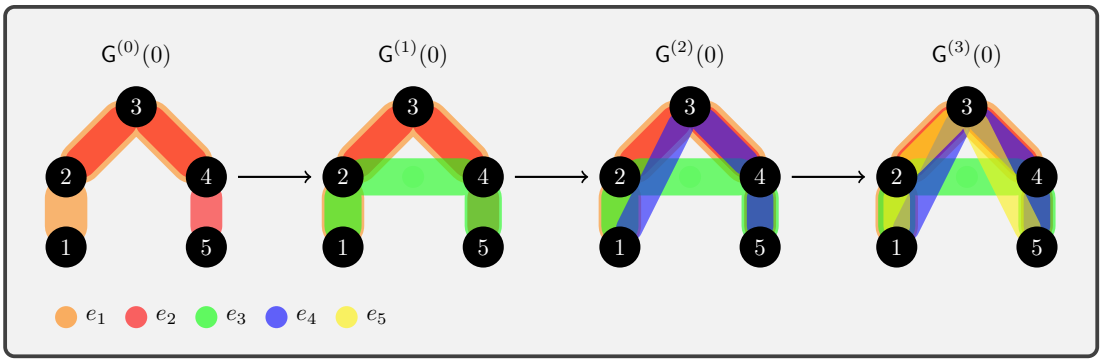


Figure 3.3: **Initial hypergraphs' structures for different q.** The plot describes the *cliques*' formation in the first five nodes of the uniform hypergraph with the rewiring probability zero, in which $e_1 = \{1, 2, 3, 4\}$, $e_2 = \{2, 3, 4, 5\}$, $e_3 = \{1, 2, 4, 5\}$, $e_4 = \{1, 3, 4, 5\}$, and $e_5 = \{1, 2, 3, 5\}$. The rest have the same patterns in every five nodes for a corresponding q.

3.4.2 The Watts-Strogatz Model

We perform an experiment on a synthetic random uniform hypergraph G with $n = 100$ and $k = 4$. Similar to the Watts-Strogatz graph, the initial hypergraph is 2-regular with lattice structure. Let q be the number of hyperedges added to the hypergraph in order to form *cliques* in every five nodes, and p be the rewiring probability of hyperedges, see Figure 3.3. Then $\mathsf{G}^{(q)}(p)$ denotes the random uniform hypergraphs generated by the rewiring probability p for different q. Particularly, when $q = 3$, the tensor entropy $\mathrm{S}^{(3)}(0) = 4.5527$, the average clustering coefficient $C^{(3)}_{\text{avg}}(0) = 0.7571$ and the average path length $L^{(3)}_{\text{avg}}(0) = 7.0606$. The goal of the experiment is to explore the relations between the tensor entropy, the average clustering coefficient, and path length with increasing the hyperedge rewiring probability p for different q. We also calculate the small world coefficient for the random hypergraphs, denoted by $\sigma^{(q)}(p)$. For each $\mathsf{G}^{(q)}(p)$, we compute its tensor entropy, average clustering coefficient, average path length, and small world coefficient 10 times and take the means for $q = 2, 3$.

The results are shown in Figure 3.4. In general, with increasing rewiring proba-

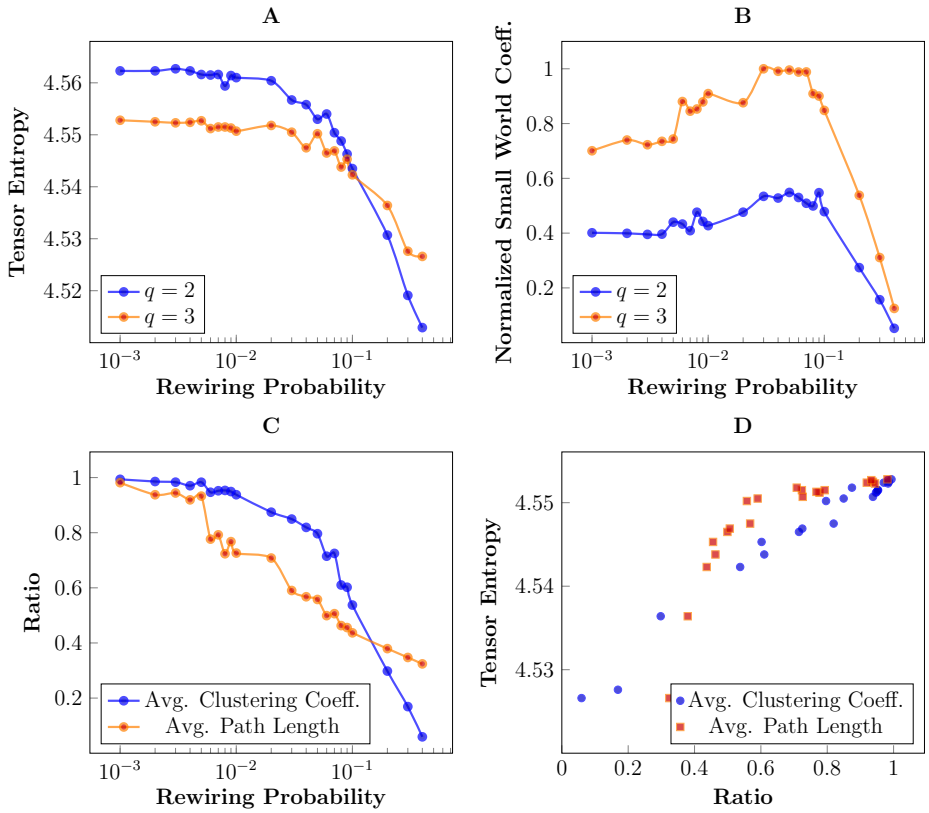


Figure 3.4: **The Watts-Strogatz model features.** (A) Tensor entropies of random uniform hypergraphs with different rewiring probabilities for different q. (B) Normalized small world coefficients of random uniform hypergraphs with different rewiring probabilities for different q. (C) Ratios $C^{(3)}_{\text{avg}}(p)/C^{(3)}_{\text{avg}}(0)$ and $L^{(3)}_{\text{avg}}(p)/L^{(3)}_{\text{avg}}(0)$ of random uniform hypergraphs with different rewiring probabilities for $q = 3$. (D) Scatter plot between the tensor entropy and the two ratios from (C).

bility p, the tensor entropy decays for both $q = 2$ and 3, see Figure 3.4A. Initially, the tensor entropies for $q = 2$ are higher than those for $q = 3$, which implies that lower average clustering coefficient yields larger tensor entropy at the same probability ($\sim$0.50 and $\sim$0.73, respectively). However, we see a strictly positive correlation between the tensor entropy and the average clustering coefficient as p increases, see Figure 3.4D. After $p = 0.1$, the tensor entropies for $q = 2$ decays faster than that for $q = 3$, indicating that higher average path length returns lower tensor entropy ($\sim$2.75 and $\sim$2.66, respectively). Moreover, the curves of the small world coefficient in Figure 3.4B are very similar to the one in the Watts-Strogatz graph, in which it

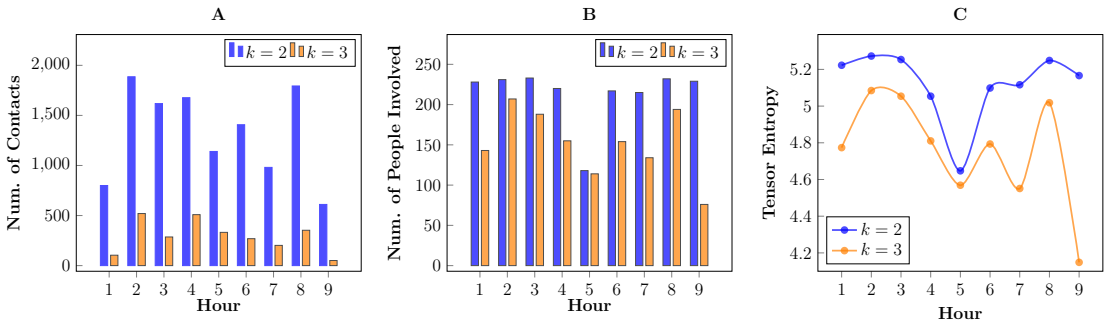


Figure 3.5: **Primary school contact features.** (A) Number of the two-person and three-person contacts amongst the children and teachers every one hour of a day. (B) Number of children and teachers involved every one hour of a day in the two-person and three-person contacts, respectively. (C) Trajectories of the von Neumann entropy and the tensor entropy for the two-person and three-person contacts of a day.

grows gradually for $p < 0.1$ and decreases quickly after $p > 0.1$. When the rewiring probability p is between 0.03 and 0.1, $\mathsf{G}^{(3)}(p)$ have apparent small world characteristics, e.g., $C^{(3)}_{\text{avg}}(0.07) = 0.5488$ and $L^{(3)}_{\text{avg}}(0.07) = 3.5748$, over the 100 nodes. In addition, we find that the average clustering coefficient pattern is similar to tensor entropy. On the contrary, the average path length has a different trend. It decreases faster at small p and more slowly than the average clustering coefficient at large p, see Figure 3.4C.

3.4.3 Primary School Contact

The primary school contact dataset contains the temporal network of face-to-face contacts amongst the children and teachers (242 people in total) at a primary school, in which an active contact can include more than two people [51, 118]. In this study, we consider the cases of two-person contacts (i.e., a normal graph) and three-person contacts (i.e., a 3-uniform hypergraph) per hour, and explore the relations of tensor entropy with contact frequencies and number of people involved over one school day. The results are shown in Figure 3.5, in which the two entropies have a similar and reasonable pattern. Both two-person and three-person contacts are more active at

the second and eighth hours, and are less active at the fifth hour. Like the von Neumann entropy ($k = 2$), the tensor entropy ($k = 3$) is expected to grow with increased number of connected nodes, see Figure 3.5B and C, which implies that more children and teachers involved will yield larger tensor entropies. On the other hand, the entropy also heavily relies on the complexity and regularity of the uniform hypergraphs as demonstrated before. For instance, the number of people involved at the seventh hour is greater than that at the fifth hour for $k = 3$, but the tensor entropies are opposite because more contacts are made at the fifth hour, increasing the complexity or regularity in the uniform hypergraph, see Figure 3.5A.

3.4.4 Mouse Neuron Endomicroscopy

The goal of the experiment is to observe mouse neuron activation patterns using fluorescence across space and time before and after food treatment in the mouse hypothalamus. Large changes in fluorescence are inferred to be active neurons that are "firing." The mouse endomicroscopy dataset is an imaging video created under the 10-minute periods of feeding, fasting and re-feeding. The imaging region contains in total 20 neuron cells, and the levels of "firing" are also recorded for each neuron. In this study, we build k-uniform hypergraphs for each 10-minute interval based on the correlations/multi-correlations of the neuron "firing" level for $k = 2, 3$. The multi-correlation among three variables is defined by

$$\rho^2 = 1 - \det(\mathbf{R}), \tag{3.11}$$

where $\mathbf{R}$ is the correlation matrices between the three variables [62]. It turns out that the multi-correlation is a generalization of Pearson correlation which can measure the strength of multivariate correlation.

The results are shown in Figure 3.6, in which (A), (B), and (C) are the first

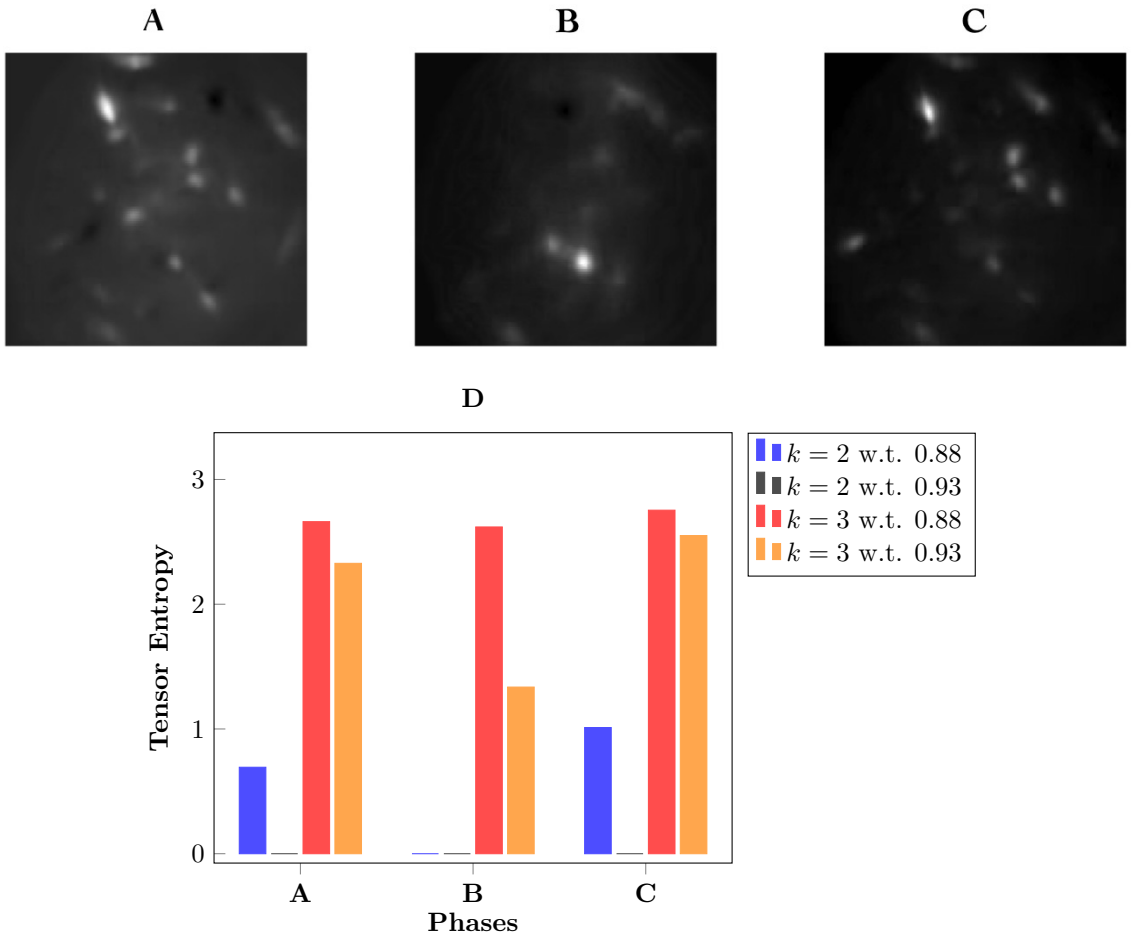


Figure 3.6: **Mouse neuron endomicroscopy features.** (A), (B), and (C) First eigenfaces of the three phases - fed, fast, and re-fed. (D) Tensor entropies of the k-uniform hypergraphs constructed from the corresponding three phases with $k = 2, 3$ (here w.t. stands for "with threshold").

eigenfaces of the corresponding three phases showing the dominant features in these phases. For computing the entropy, we choose the cutoff threshold to be 0.93 in the construction of edge/hyperedge. In Figure 3.6D, the von Neumann entropy ($k = 2$) stays constant because the threshold is too high to generate edges in the graph model. However, the tensor entropy ($k = 3$) is able to capture changes in neuronal activity, which is lower during the fast phase and higher during the fed/re-fed phase. If we lower the threshold, a similar pattern is observed for $k = 2$. To maintain the model accuracy, we want to keep the threshold as high as possible. This

supports use of tensor entropy over von Neumann entropy. As validation for using tensor entropy in biological data, we find that mouse neuron activation patterns can be more accurately captured through 3-uniform hypergraphs. Under the threshold 0.93, the two hypergraphs for the fed and re-fed phases contain a number of common hyperedges. These hyperedges are mainly composed of nodes with high degrees, representing scenarios where more than two neurons synchronize, or "co-fire", in the mouse hypothalamus. This suggests that these neurons are involved in mouse appetite regulation, which is not captured using the graph model.

3.4.5 Cellular Reprogramming

Cellular reprogramming is a process that introduces proteins called transcription factors as a control mechanism for transforming one cell type into another. The unbiased genome-wide technology of chromosome conformation capture (Hi-C) has been used to capture the dynamics of reprogramming [79, 103, 106]. However, the pairwise contacts from Hi-C data fail to include the multiway interactions of chromatin. Furthermore, the notion of transcription factories supports the existence of simultaneous interactions involving mutiple genomic loci [35], implying that the human genome configuration can be represented by a hypergraph. Therefore, in this example, we use 3-uniform hypergraphs to partially recover the 3D configuration of the genome based on the multi-correlation (3.11) from Hi-C matrices. We believe that such reconstruction can provide more information about genome structure and patterns, compared to the pairwise Hi-C contacts. We use a cellular reprogramming dataset, containing normalized Hi-C data from fibroblast proliferation and MyoD-mediated fibroblast reprogramming (MyoD is the transcription factor used for control) for Chromosome 14 at 1MB resolution with a total of 89 genomic loci. Our goal is to quantitatively detect a bifurcation in the fibroblast proliferation and reprogramming

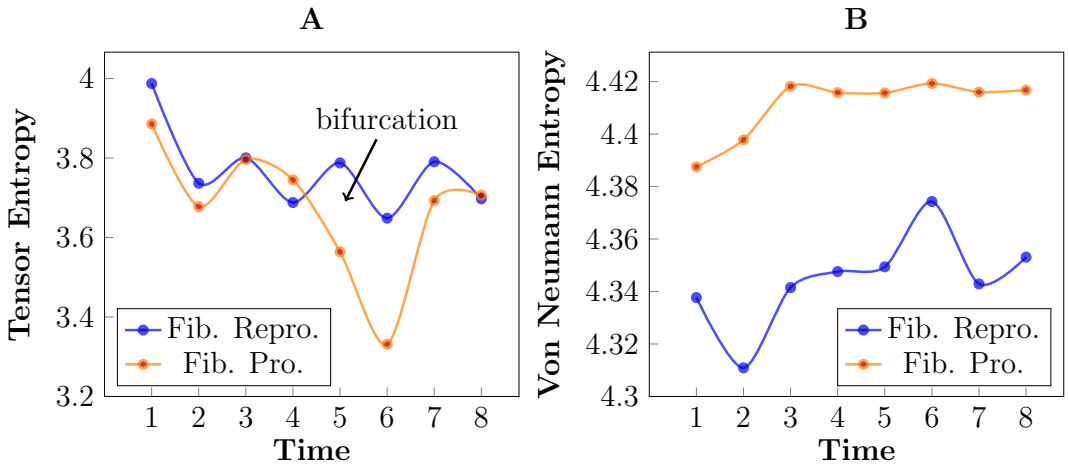


Figure 3.7: **Cellular reprogramming features.** (A) Tensor entropies of the uniform hypergraphs recovered from Hi-C measurements with multi-correlation cutoff threshold 0.95. (B) Von Neuman entropies of the binarized Hi-C matrices with weight cutoff threshold 0.95.

data, and accurately identify the critical transition point between cell identities during reprogramming. The results are shown in Figure 3.7. We can clearly observe a bifurcation between the two trajectories using the tensor entropy of the 3-uniform hypergraphs recovered from the Hi-C measurements. Crucially, the critical transition point marked in Figure 3.7A is consistent with the ground-truth statistic provided in [79]. In contrast, the von Neumann entropy cannot provide adequate information about the bifurcation and critical transition point, if one analyzes the Hi-C measurements as adjacency matrices. The two trajectories are separate from the beginning, see Figure 3.7B.

3.4.6 Algorithm Run Time Comparison

In this example, the k-uniform hypergraphs are constructed with n nodes by forming a strip structure in which every pair of connected hyperedges only contains one common node. We compare the computational efficiency of the SVD-based Algorithm 3 and the TTD-based Algorithm 4 in computing the tensor entropy. The results are shown in Figure 3.8. For the TTD-based entropy computations, we assume that

all the adjacency and degree tensors of the uniform hypergraphs are already provided in the TT-format. Evidently, Algorithm 4 is more time efficient than Algorithm 3 for 4-uniform and 5-uniform hypergraphs with the strip structure as n becomes larger, see Figure 3.8. Particularly, when $k = 5$, the TTD-based algorithm exhibits a huge time advantage as predicted in the computation complexity. The time curve from the SVD-based Algorithm 3 increases exponentially, while it grows at a much slower rate if using Algorithm 4. In the meantime, we compute the relative errors between the tensor entropies computed from the two algorithms, all of which are within 10^{-14}.

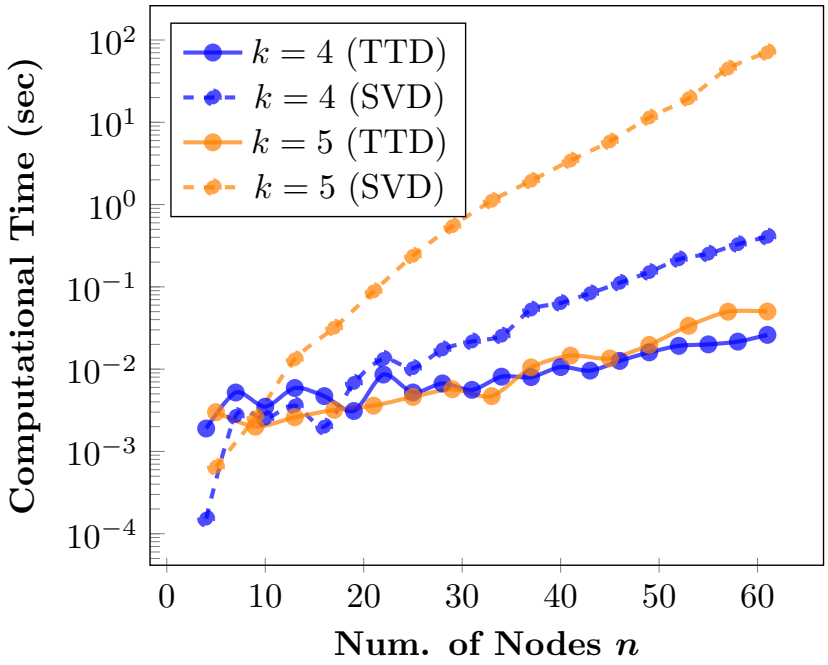


Figure 3.8: **Computational time comparisons between the SVD-based and TTD-based algorithms.** For the TTD-based entropy computation, we reported the times of left- and right-orthonormalization and economy-size matrix SVD. For the SVD-based entropy computation, we only reported the time of economy-size matrix SVD. For the purpose of accuracy, we ran each algorithm 10 times and took the average of the computational times.

3.5 Discussion

The first five numerical examples reported here highlight that the k-mode singular values computed from the HOSVD of the Laplacian tensors can provide nice

predictions of structural properties for uniform hypergraphs. This method can also be used for anomaly detection in the context of dynamics as we demonstrated in the mouse neuron endomicroscopy and cellular reprogramming datasets. However, more theoretical and numerical investigations are required to assess the real advantages of hypergraphs versus normal graphs, and is an important avenue of future research. Moreover, as we pointed out in Section 3.3, many simple structure tensors can be directly created in the TT-format without requiring construction of the full representations. For example, Oseledets *et al.* [92] built the Laplacian operator in the TT-format for the discretized heat equations. We believe that similar results can happen to the adjacency, degree, and Laplacian tensors for uniform hypergraphs.

Instead of looking at the k-mode singular values, we can consider the tensor eigenvalues in defining the tensor entropy. We will refer to it as the eigenvalue entropy later. See Appendix .1 for a short introduction to tensor eigenvalues. Based on the tensor eigenvalue formulations, we can establish the eigenvalue entropy measure for uniform hypergraphs.

Definition III.12. Let G be a k-uniform hypergraph with n nodes. The *eigenvalue entropy* of G is defined by

$$\mathrm{S} = -\sum_{j=1}^{d} |\hat{\lambda}_j| \ln |\hat{\lambda}_j|, \tag{3.12}$$

where $|\hat{\lambda}_j|$ are the normalized modulus of the eigenvalues of the Laplacian tensor L such that $\sum_{j=1}^{d} |\hat{\lambda}_j| = 1$, and $d = n(k-1)^{n-1}$ is the total number of the eigenvalues.

The convention $0 \ln 0 = 0$ is used if $|\hat{\lambda}_j| = 0$. We can use other tensor eigenvalue notions including H-eigenvalue, E-eigenvalue, and Z-eigenvalue with a corresponding d to fit into the formula. For curiosity, we repeat the hyperedge growth model using the eigenvalue entropy, see Figure 3.9. The entropy minimization evolution trajectory

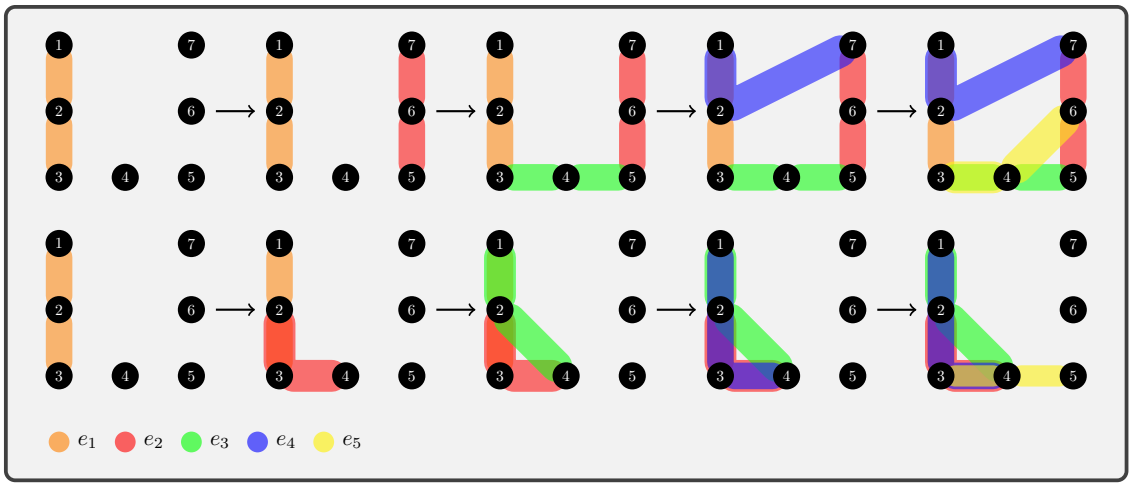


Figure 3.9: **Eigenvalue entropy maximization/minimization.** The top row describes the first five stages of the eigenvalue entropy maximization evolution with a growing number of hyperedges in the order of $e_1 = \{1, 2, 3\}$, $e_2 = \{5, 6, 7\}$, $e_3 = \{3, 4, 5\}$, $e_4 = \{1, 2, 7\}$, and $e_5 = \{3, 4, 6\}$. The eigenvalue entropy $\mathrm{S}_{\max}^{(j)} = 4.4910$, 5.6342, 5.8608, 5.9490 and 6.0091 for $j = 1, 2, 3, 4, 5$. The bottom row reports the first five stages of the eigenvalue entropy minimization process with a growing number of hyperedges in the order of $e_1 = \{1, 2, 3\}$, $e_2 = \{2, 3, 4\}$, $e_3 = \{1, 2, 4\}$, $e_4 = \{1, 3, 4\}$, and $e_5 = \{3, 4, 5\}$. The eigenvalue entropy $\mathrm{S}_{\min}^{(j)} = 4.4910$, 5.3604, 5.4434, 5.4715, and 5.6334 for $j = 1, 2, 3, 4, 5$. All the tensor eigenvalues of the Laplacian tensors in this experiment are computed from the MATLAB Toolbox TenEig [29, 30].

is the same for the first five stages in which *cliques* are formed. The maximization evolution trajectory becomes different from the fourth stage after the hypergraph is connected, in which short path lengths and high level of nontrivial symmetricity are no longer the factors that maximize the entropy. In addition, computing the eigenvalues of a tensor is an NP-hard problem [55]. Hence, the eigenvalue entropy may not be used to predict the structural properties for uniform hypergraphs, but it might contain other unknown features that are required to explore in the future.

Furthermore, the notion of robustness for uniform hypergraphs is an important topic. In graph theory, one of the popular measures is called effective resistance [45]. The authors in [67] show that the effective graph resistance can be written in terms of the reciprocals of graph Laplacian eigenvalues, and robust networks have small effective graph resistance. Hence, we attempt to establish similar relationship using

Table 3.1: **The effective resistance and tensor entropy.**

Step	$j = 3$	$j = 5$	$j = 15$	$j = 25$	$j = 35$
$\mathrm{R}_{\max}^{(j)}$	34.8384	20.9432	7.3863	4.4819	3.2166
$\mathrm{S}_{\max}^{(j)}$	1.9037	1.9359	1.9430	1.9448	1.9456

the k-mode singular values from the Laplacian tensors to describe the robustness of uniform hypergraphs.

Definition III.13. Let G be a connected k-uniform hypergraph with n nodes. The *effective resistance* of G is defined by

$$\mathrm{R} = n \sum_{j=1}^{n} \frac{1}{\gamma_j}, \tag{3.13}$$

where γ_j are the k-mode singular values of L.

If a uniform hypergraph is non-connected, then the effective resistance $\mathrm{R} = \infty$. Based on (3.13), we compute the effective resistance of the uniform hypergraphs $\mathsf{G}_{\max}^{(j)}$ in the hyperedge growth model example, denoted by $\mathrm{R}_{\max}^{(j)}$. Similar to the effective graph resistance, $\mathrm{R}_{\max}^{(j)}$ strictly decreases when hyperedges are added and achieves the minimum at the final step when the hypergraph is complete, see Table 3.1. We can also expect that the smaller the effective resistance is, the more robust the uniform hypergraph. We believe that the effective resistance (3.13) is a good measure for uniform hypergraph robustness, but more theoretical and numerical support is needed to verify this hypothesis.

CHAPTER IV

Controllability of Hypergraphs

Controlling complex networks is one of the most challenging problems in modern network science [34, 49, 81, 127, 126, 130, 132]. Lin [32] first proposed the concept of structural controllability of directed graphs in 1970s. Later on, Tanner [122] and Rahmani *et al.* [101, 102] applied the idea of structural controllability for multi-agent systems with the aim of selecting a subset of agents (called leaders) which are able to control the whole system by exploiting the graph Laplacian and linear control theory. In particular, Rahmani *et al.* [101] also showed how the symmetry structure of a graph directly relates to the controllability of the corresponding multi-agent system.

In 2011, Liu *et al.* [81, 82] explored the (structural) controllability of complex graphs with n nodes by using the canonical linear time-invariant dynamics

$$\dot{\mathbf{x}} = \mathbf{A}\mathbf{x} + \mathbf{B}\mathbf{u}, \tag{4.1}$$

where $\mathbf{A} \in \mathbb{R}^{n\times n}$ is the adjacency matrix of a graph, and $\mathbf{B} \in \mathbb{R}^{n\times m}$ is the control matrix. The time-dependent vector $\mathbf{x} \in \mathbb{R}^n$ captures the states of the nodes, and $\mathbf{u} \in \mathbb{R}^m$ is the time-dependent control vector. The authors exploited the Kalman rank condition, i.e., the linear system (4.1) is controllable if and only if the controllability matrix

$$\mathbf{C} = \begin{bmatrix} \mathbf{B} & \mathbf{AB} & \dots & \mathbf{A}^{n-1}\mathbf{B} \end{bmatrix} \tag{4.2}$$

has full rank, to determine the minimum number of control nodes (MCN) in order to achieve full control of the graph (similar to the role of leaders discussed previously). They also identified the MCN of a graph using the idea of "maximum matching" [81]. In addition, Yuan *et al.* [130] developed a notion of exact controllability of graphs. They took advantage of the Popov-Belevitch-Hautus rank condition (i.e., the linear system (4.1) is controllable if and only if $\text{rank}(\begin{bmatrix} s\mathbf{I} - \mathbf{A} & \mathbf{B} \end{bmatrix}) = n$ for any complex number s) to prove that for an arbitrary graph, the MCN is determined by the maximum geometric multiplicity of the eigenvalues of the corresponding adjacency matrix $\mathbf{A}$. Furthermore, Nacher *et al.* [88] analyzed MCN required to fully control multilayer graphs, and a similar notion of exact controllability for multilayer graphs is defined in [131].

Multilinear dynamical systems were first proposed by Rogers *et al.* [107] and Surana *et al.* [120] where the system evolution is generated by the action of multilinear operators which are formed using Tucker products of matrices. Chen *et al.* [24, 27] developed the tensor algebraic conditions for stability, reachability, and observability for input/output multilinear time-invariant systems. By using tensor unfolding, an operation that transforms a tensor into a matrix, a multilinear system can be unfolded to a corresponding linear system. However, the tensor-based multilinear systems, proposed in this paper, are different from the ones defined in [24, 107, 120], and in fact they belong to the family of nonlinear polynomial systems. Hence, they can capture network dynamics more precisely than systems based on standard graphs which use linear dynamics assumption. Basic knowledge of nonlinear control concepts such as Lie algebra and Lie brackets is required in order to better understand the controllability of such systems. The key contributions of this chapter are as follows:

- We propose a new tensor-based multilinear system representation inspired by uniform hypergraphs, and study the controllability of such systems by exploiting tensor algebra and polynomial control theory. We establish a Kalman-rank-like condition to determine the controllability of even uniform hypergraphs.
- We establish theoretical results on the MCN of even uniform hyperchains, hyperrings, and hyperstars as well as complete even uniform hypergraphs. We also observe that the MCN of odd uniform hypergraphs, identified by the Kalman-rank-like condition, behaves similarly to that of even uniform hypergraphs in simulated examples (although the condition is not applicable in terms of controllability). We discover that MCN is related to the hypergraph degree distribution, and high degree nodes are preferred to be controlled in these configurations and their variants.
- We propose MCN as a measure of robustness for uniform hypergraphs, and use it to quantify structural differences. We present applications to two real world biological examples: a mouse neuron endomicroscopy dataset and an allele-specific Hi-C dataset.
- We present a fast and memory-efficient computational framework for determining the rank of a matrix related to the Kalman-rank-like condition for nonlinear controllability. In addition, we propose a heuristic approach to identify the MCN of uniform hypergraphs efficiently.
- We perform preliminary explorations of the controllability of general non-uniform hypergraphs.

This chapter is organized into seven sections. We start with some definitions of special hypergraphs in Section 4.1. We propose a new tensor-based multilinear

system to capture the dynamics of uniform hypergraphs in Section 4.2. We then formulate a Kalman-rank-like condition to determine the controllability of even uniform hypergraphs in Section 4.3. We establish theoretical results on the MCN of even uniform hyperchains, hyperrings, and hyperstars as well as complete even uniform hypergraphs in Section 4.4. In Section 4.5, we argue that MCN can be used to measure hypergraph robustness, and provide a heuristic approach to find the MCN efficiently. Five numerical examples are presented in Section 4.6. Finally, we discuss the controllability of general hypergraphs in Section 4.7. All the content of this chapter has published in [26].

4.1 Special Hypergraphs

In this section, we extend the definitions of chains, rings, and stars from graph theory to uniform hypergraphs.

Definition IV.1. A k-uniform *hyperchain* is a sequence of n nodes such that every k consecutive nodes are adjacent, i.e., nodes $j, j+1, \ldots, j+k-1$ are contained in one hyperedge for $j = 1, 2, \ldots, n-k+1$.

Definition IV.2. A k-uniform *hyperring* is a sequence of n nodes such that every k consecutive nodes are adjacent, i.e., nodes $\sigma_n(j), \sigma_n(j+1), \ldots, \sigma_n(j+k-1)$ are contained in one hyperedge for $j = 1, 2, \ldots, n$, where $\sigma_n(j) = j$ for $j \leq n$ and $\sigma_n(j) = j - n$ for $j > n$.

Definition IV.3. A k-uniform *hyperstar* is a collection of $k-1$ internal nodes that are contained in all the hyperedges, and $n-k+1$ leaf nodes such that every leaf node is contained in one hyperedge with the internal nodes.

In k-uniform hyperchains, hyperrings and hyperstars, every two hyperedges have exactly $k-1$ overlapping nodes, see Figure 4.1. When $k = 2$, they are reduced to

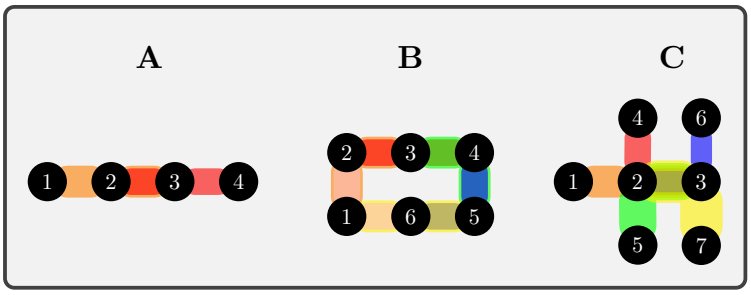


Figure 4.1: **Examples of hyperchains, hyperrings, and hyperstars.** (A) 3-uniform hyperchain with $e_1 = \{1, 2, 3\}$ and $e_2 = \{2, 3, 4\}$. (B) 3-uniform hyperring with $e_1 = \{1, 2, 3\}$, $e_2 = \{2, 3, 4\}$, $e_3 = \{3, 4, 5\}$, $e_4 = \{4, 5, 6\}$, $e_5 = \{5, 6, 1\}$, and $e_6 = \{6, 1, 2\}$. (C) 3-uniform hyperstar with $e_1 = \{1, 2, 3\}$, $e_2 = \{2, 3, 4\}$, $e_3 = \{2, 3, 5\}$, $e_4 = \{2, 3, 6\}$, and $e_5 = \{2, 3, 7\}$.

standard chains, rings, and stars. We will determine the minimum number of control nodes (MCN) of uniform hyperchains, hyperrings, and hyperstars in Section 4.4.

4.2 Uniform Hypergraph Dynamics

We represent the dynamics of a k-uniform hypergraph G with n nodes by multilinear time-invariant differential equations using the adjacency tensor of G.

Definition IV.4. Given a k-uniform hypergraph G with n nodes, the dynamics of G with control inputs can be represented by

$$\dot{\mathbf{x}} = \mathsf{A}\mathbf{x}^{k-1} + \sum_{j=1}^{m} \mathbf{b}_j u_j, \tag{4.3}$$

where $\mathsf{A} \in \mathbb{R}^{n \times n \times \cdots \times n}$ is the adjacency tensor of G, and $\mathbf{B} = \begin{bmatrix} \mathbf{b}_1 & \mathbf{b}_2 & \dots & \mathbf{b}_m \end{bmatrix} \in \mathbb{R}^{n \times m}$ is the control matrix.

In this paper, we consider the case in which each input can only be imposed at one node, i.e., $\mathbf{b}_j$ are the scaled standard basis vectors, similar to the treatments in [81, 130]. The time-dependent vector $\mathbf{x}$ captures the state of the n nodes, and the system is controlled using the time-dependent input vector $\mathbf{u} = \begin{bmatrix} u_1 & u_2 & \dots & u_m \end{bmatrix}^\top \in \mathbb{R}^m$. The multilinear system (4.3) formulated by the tensor vector multiplications is

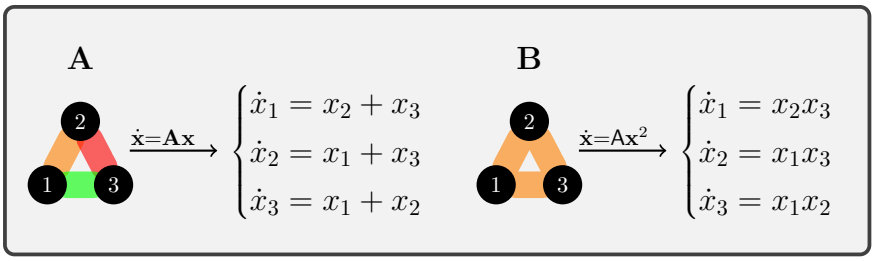


Figure 4.2: **Graphs versus uniform hypergraphs.** (A) Standard graph with three nodes and edges $e_1 = \{1, 2\}$, $e_2 = \{2, 3\}$, and $e_3 = \{1, 3\}$, and its corresponding linear dynamics. (B) 3-uniform hypergraph with three nodes and a hyperedge $e_1 = \{1, 2, 3\}$, and its corresponding nonlinear dynamics.

indeed able to capture the simultaneous interactions among nodes for uniform hypergraphs as illustrated in Figure 4.2. All the interactions are characterized using multiplications instead of the additions that are typically used in a standard graph. It is known that multiplication often stands for simultaneity and addition for sequentiality in many mathematical fields. For example, the probability of two independent events to happen at the same time is equal to the product of their individual probabilities. A similar form of nonlinear dynamical system representation has been used to model protein-protein interaction [48], which as mentioned can be represented by hypergraphs. Compared to the dynamical systems of hypergraphs defined in [21, 38], our model (4.3) is simpler and retains the higher-order coupling information. More significantly, we can discuss the controllability of such systems. In the next section, we will establish a Kalman-rank-like condition by exploiting nonlinear control theory.

4.3 Controllability of Uniform Hypergraphs

If one rewrites the tensor vector multiplications in the multilinear system (4.3) explicitly as in Figure 4.2B, the drift term $\mathsf{A}\mathbf{x}^{k-1}$ is in fact a homogeneous polynomial system of degree $k-1$. The controllability of polynomial systems was studied extensively back in 1970s and 80s [6, 19, 63, 111]. In particular, Jurdjevic and Kupka [63]

obtained strong results in terms of the controllability of homogeneous polynomial systems with constant input multipliers (i.e., $\mathbf{b}_j$ are constant vectors).

Definition IV.5. A dynamical system is called *strongly controllable* if it can be driven from any initial state to any target state in any instant of positive time given a suitable choice of control inputs.

Theorem IV.6. *Consider the following system*

$$\dot{\boldsymbol{x}} = \boldsymbol{f}(\boldsymbol{x}) + \sum_{j=1}^{m} \boldsymbol{b}_j u_j. \tag{4.4}$$

Suppose that $\boldsymbol{f}$ is a homogeneous polynomial system of odd degree. Then the system (4.4) is strongly controllable if and only if the rank of the Lie algebra spanned by the set of vector fields $\{\boldsymbol{f}, \boldsymbol{b}_1, \boldsymbol{b}_2, \ldots, \boldsymbol{b}_m\}$ is n at all points of $\mathbb{R}^n$. Moreover, the Lie algebra is of full rank at all points of $\mathbb{R}^n$ if and only if it is of full rank at the origin.

The rank of the Lie algebra can be found by evaluating the recursive Lie brackets of $\{\mathbf{f}, \mathbf{b}_1, \mathbf{b}_2, \ldots, \mathbf{b}_m\}$ at the origin. The Lie bracket of two vector fields $\mathbf{f}$ and $\mathbf{g}$ at a point $\mathbf{x}$ is defined as

$$[\mathbf{f}, \mathbf{g}]_{\mathbf{x}} = \nabla \mathbf{g}(\mathbf{x})\mathbf{f}(\mathbf{x}) - \nabla \mathbf{f}(\mathbf{x})\mathbf{g}(\mathbf{x}), \tag{4.5}$$

where ∇ is the gradient operation. Detailed definitions of Lie algebra and Lie brackets can be found in any differential geometry textbook. Based on Theorem IV.6, we can derive a Kalman-rank-like condition for the tensor-based multilinear system (4.3).

Definition IV.7. Let $\mathcal{C}_0$ be the linear span of $\{\mathbf{b}_1, \mathbf{b}_2, \ldots, \mathbf{b}_m\}$ and $\mathsf{A} \in \mathbb{R}^{n \times n \times \cdots \times n}$ be a supersymmetric tensor. For each integer $q \geq 1$, define $\mathcal{C}_q$ inductively as the linear span of

$$\mathcal{C}_{q-1} \cup \{\mathsf{A}\mathbf{v}_1\mathbf{v}_2 \ldots \mathbf{v}_{k-1} | \mathbf{v}_l \in \mathcal{C}_{q-1}\}. \tag{4.6}$$

Denote the subspace $\mathcal{C}(\mathsf{A}, \mathbf{B}) = \cup_{q \geq 0} \mathcal{C}_q$ where $\mathbf{B} = \begin{bmatrix} \mathbf{b}_1 & \mathbf{b}_2 & \ldots & \mathbf{b}_m \end{bmatrix} \in \mathbb{R}^{n \times m}$.

Corollary IV.8. *Suppose that k is even. The multilinear system (4.3) is strongly controllable if and only if the subspace $\mathcal{C}(\mathsf{A}, \boldsymbol{B})$ spans $\mathbb{R}^n$, or equivalently, the matrix $\boldsymbol{C}$, including all the column vectors from $\mathcal{C}(\mathsf{A}, \boldsymbol{B})$, has rank n.*

Proof. We show that $\mathcal{C}(\mathsf{A}, \mathbf{B})$ consists of all the recursive Lie brackets of $\{\mathsf{A}\mathbf{x}^{k-1}, \mathbf{b}_1, \mathbf{b}_2, \ldots, \mathbf{b}_m\}$ at the origin. Without loss of generality, assume that $m = 1$. Since A is supersymmetric, the recursive Lie brackets are given by (we omit all the scalars in the calculation)

$$
\begin{aligned}
[\mathbf{b}, \mathsf{A}\mathbf{x}^{k-1}]_{\mathbf{0}} &= (\frac{d}{d\mathbf{x}}\Big|_{\mathbf{x}=\mathbf{0}} \mathsf{A}\mathbf{x}^{k-1})\mathbf{b} = \mathbf{0}, \\
[\mathbf{b}, [\mathbf{b}, \mathsf{A}\mathbf{x}^{k-1}]]_{\mathbf{0}} &= (\frac{d}{d\mathbf{x}}\Big|_{\mathbf{x}=\mathbf{0}} \mathsf{A}\mathbf{x}^{k-2}\mathbf{b})\mathbf{b} = \mathbf{0}, \\
&\vdots \\
[\mathbf{b}, [\ldots, [[\mathbf{b}, \mathsf{A}\mathbf{x}^{k-1}]]]]_{\mathbf{0}} &= (\frac{d}{d\mathbf{x}}\Big|_{\mathbf{x}=\mathbf{0}} \mathsf{A}\mathbf{x}\mathbf{b}^{k-2})\mathbf{b} = \mathsf{A}\mathbf{b}^{k-1}.
\end{aligned}
$$

We then repeat the recursive process for the brackets $[\mathsf{A}\mathbf{b}^{k-1}, \mathsf{A}\mathbf{x}^{k-1}]$, $[\mathsf{A}\mathbf{b}^{k-1}, \mathsf{A}\mathbf{x}^{k-2}\mathbf{b}]$, $\ldots$, $[\mathsf{A}\mathbf{b}^{k-1}, \mathsf{A}\mathbf{x}\mathbf{b}^{k-2}]$ in the second iteration. After the q-th iteration for some q, the subspace $\mathcal{C}(\mathsf{A}, \mathbf{B})$ contains all the Lie brackets of the vector fields $\{\mathsf{A}\mathbf{x}^{k-1}, \mathbf{b}\}$ at the origin. Lastly, when k is even, the drift term $\mathsf{A}\mathbf{x}^{k-1}$ is a family of homogeneous polynomial fields of odd degree. Based on Theorem IV.6, the result follows immediately. □

Corollary IV.9. *Given the subspace $\mathcal{C}(\mathsf{A}, \boldsymbol{B}) = \cup_{q\geq 0}\mathcal{C}_q$, there exists an integer $q \leq n$ such that $\mathcal{C}(\mathsf{A}, \boldsymbol{B}) = \mathcal{C}_q$.*

Proof. The proof follows immediately from the fact that $\mathcal{C}(\mathsf{A}, \mathbf{B})$ is a finite-dimensional vector space [63]. □

We can treat the matrix $\mathbf{C}$ as the controllability matrix of the multilinear system (4.3). When $k = 2$ and $q = n - 1$, Corollary 1 is reduced to the famous Kalman

Algorithm 5 Computing the reduced controllability matrix.

1: Given a supersymmetric tensor $\mathsf{A} \in \mathbb{R}^{n\times n\times\cdots\times n}$ and a control matrix $\mathbf{B} \in \mathbb{R}^{n\times m}$
2: Unfold A into a matrix $\mathbf{A}$ by stacking the last $k-1$ modes, i.e., $\mathbf{A} \in \mathbb{R}^{n\times n^{k-1}}$
3: Set $\mathbf{C}_r = \mathbf{B}$ and $j = 0$
4: **while** $j < n$ **do**
5: Compute $\mathbf{L} = \mathbf{A}(\mathbf{C}_r \otimes \mathbf{C}_r \otimes^{\cdots}{}^{k-1} \otimes \mathbf{C}_r)$
6: Set $\mathbf{C}_r = \begin{bmatrix}\mathbf{C}_r & \mathbf{L}\end{bmatrix}$
7: Compute the economy-size SVD of $\mathbf{C}_r$, and remove the zero singular values, i.e., $\mathbf{C}_r = \mathbf{U}\mathbf{S}\mathbf{V}^\top$ where $\mathbf{S} \in \mathbb{R}^{s\times s}$, and s is the rank of $\mathbf{C}_r$
8: Set $\mathbf{C}_r = \mathbf{U}$, and $j = j + 1$
9: **end while**
10: **return** The reduced controllability matrix $\mathbf{C}_r$.

rank condition for linear systems. However, computing the controllability matrix can be computationally demanding as the number of columns of $\mathbf{C}$ grows exponentially with k. In Algorithm 5, we offer a memory-efficient approach to obtain a reduced form of the controllability matrix, denoted by $\mathbf{C}_r$, by exploiting the economy-size matrix SVD. Step 2 is referred to as the 1-mode tensor unfolding (matrization). Here $\otimes$ denotes the Kronecker product, and Step 5 can be computed fast without evaluating the actual Kronecker products of $\mathbf{C}_r$ [71]. One may also exploit the sparse tensor/matrix structure to further save the computation and memory.

Lemma IV.10. *Suppose that $\boldsymbol{A} \in \mathbb{R}^{n\times n^{k-1}}$ is defined as in Algorithm 5, and $\boldsymbol{X} \in \mathbb{R}^{n\times m}$ is an arbitrary matrix with rank s. Then the following two matrices*

$$\boldsymbol{P} = \boldsymbol{A}(\boldsymbol{X} \otimes \boldsymbol{X} \otimes^{\cdots}{}^{k-1} \otimes \boldsymbol{X}) \in \mathbb{R}^{n\times m^{k-1}},$$

$$\boldsymbol{Q} = \boldsymbol{A}(\boldsymbol{U} \otimes \boldsymbol{U} \otimes^{\cdots}{}^{k-1} \otimes \boldsymbol{U}) \in \mathbb{R}^{n\times s^{k-1}},$$

share the same column space, where $\boldsymbol{U} \in \mathbb{R}^{n\times s}$ is the matrix including the first s left singular vectors of $\boldsymbol{X}$.

Proof. Without loss of generality, assume that $k = 3$. Suppose that $\mathbf{X} = \mathbf{U}\mathbf{S}\mathbf{V}^\top$

with $\mathbf{U} \in \mathbb{R}^{n\times s}$. Based on the properties of the Kronecker product, one can write

$$\begin{aligned}\mathbf{P} &= \mathbf{A}[(\mathbf{U}\mathbf{S}\mathbf{V}^\top) \otimes (\mathbf{U}\mathbf{S}\mathbf{V}^\top)] \\ &= \mathbf{A}[(\mathbf{U}\otimes\mathbf{U})(\mathbf{S}\otimes\mathbf{S})(\mathbf{V}\otimes\mathbf{V})^\top] \\ &= \mathbf{Q}[(\mathbf{S}\otimes\mathbf{S})(\mathbf{V}\otimes\mathbf{V})^\top] = \mathbf{Q}\tilde{\mathbf{S}}\tilde{\mathbf{V}}^\top,\end{aligned}$$

where $\tilde{\mathbf{S}} = \mathbf{S}\otimes\mathbf{S} \in \mathbb{R}^{s^2\times s^2}$ is a diagonal matrix and $\tilde{\mathbf{V}} = \mathbf{V}\otimes\mathbf{V} \in \mathbb{R}^{m^2\times s^2}$ is a semi-orthogonal matrix. Thus, it follows immediately that $\mathbf{P}$ and $\mathbf{Q}$ share the same column space. □

Proposition IV.11. *The column space of the reduced controllability matrix $\boldsymbol{C}_r$ is $\mathcal{C}(\boldsymbol{A}, \boldsymbol{B})$.*

Proof. The result follows immediately from Lemma IV.10 and Corollary IV.9. □

Remark: The controllability of homogeneous polynomial systems of even degree is still an open problem to best of authors knowledge [2, 84]. The reason is intimately related to the fact that the roots of polynomial systems of even degree might all be complex [2]. Therefore, we cannot guarantee the condition for controllability of odd uniform hypergraphs using Corollary IV.8. Nevertheless, a weaker form of controllability, called *(local) accessibility*, can be obtained for the multilinear system (4.3) with odd k based on the Kalman-rank-like condition.

Given $\mathbf{x}_0 \in \mathbb{R}^n$ and control inputs, define $\mathcal{R}(\mathbf{x}_0, t)$ to be the set of all $\mathbf{x} \in \mathbb{R}^n$ for which the system can be driven from $\mathbf{x}_0$ to $\mathbf{x}$ at time t.

Definition IV.12. A dynamical system is called *accessible* if for any initial state $\mathbf{x}_0 \in \mathbb{R}^n$ and $T > 0$, the reachable set $\mathcal{R}_T(\mathbf{x}_0) = \cup_{0\le t\le T}\mathcal{R}(\mathbf{x}_0, t)$ contains a nonempty open set.

The accessibility of a control system requires only that the reachable set from a given point contains a nonempty open set, rather than being equal to the whole space $\mathbb{R}^n$ (required for strong controllability). Accessibility holds at a point if the span of the smallest Lie algebra of vector fields containing the drift and input vector fields is $\mathbb{R}^n$ at that point [12].

Corollary IV.13. *The multilinear system (4.3) is accessible if the subspace* $\mathcal{C}(A, \boldsymbol{B})$ *spans* $\mathbb{R}^n$*, or equivalently, the matrix* $\boldsymbol{C}$*, including all the column vectors from* $\mathcal{C}(A, \boldsymbol{B})$*, has rank* n*.*

Proof. The smallest Lie algebra of vector fields containing $\mathsf{A}\mathbf{x}^{k-1}$ and $\mathbf{b}_1, \ldots, \mathbf{b}_m$ at the origin is $\mathcal{C}(\mathsf{A}, \mathbf{B})$ by Corollary IV.8. Based on the second part of Theorem IV.6 (i.e., the Lie algebra is of full rank at all points of $\mathbb{R}^n$ if and only if it is of full rank at the origin), the result follows immediately. □

4.4 MCN of Special Hypergraphs

According to Corollary IV.8 and IV.9, we can discuss the controllability of even uniform hypergraphs. Similarly to [81, 130], we want to determine the MCN, denoted by n^*, whose control is sufficient for achieving controllability of the hypergraph. For example, let's consider the simplest even uniform hypergraph, i.e., the 4-uniform hypergraph with four nodes. We find that the MCN of this hypergraph is three based on the Kalman-rank-like condition, see Figure 4.3. More significantly, we discover that the MCN of even uniform hyperchains, hyperrings, and hyperstars as well as complete even uniform hypergraphs behaves similarly to those of chains, rings, stars, and complete graphs.

Proposition IV.14. *Suppose that* k *is even. If* G *is a* k*-uniform hyperchain with* n *nodes, then the MCN of* G *is given by* $n^* = k - 1$*.*

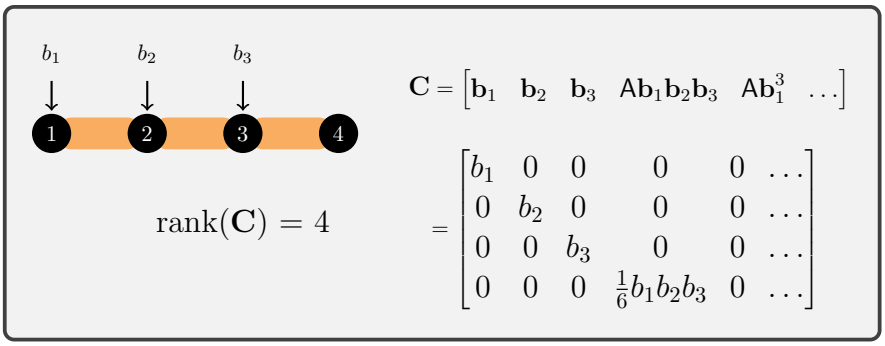


Figure 4.3: **Controllability matrix.** 4-uniform hypergraph with four nodes and a hyperedge $\{1, 2, 3, 4\}$, and its controllability matrix $\mathbf{C}$.

Proof. We first show that when the number of control inputs $m = k - 2$, G is never controllable. Based on the definition of tensor vector multiplication and the special structure of the adjacency tensor, it can be shown that for any choice of the control matrix $\mathbf{B}$ with $m = k - 2$,

$$\mathsf{A}\mathbf{b}_{j_1}\mathbf{b}_{j_2}\dots\mathbf{b}_{j_{k-1}} = \mathbf{0},$$

where A is the adjacency tensor of G. This is because the set of vectors $\mathbf{b}_{j_l}$ must contain a duplicate for one $l = 1, 2, \dots, k-1$. Therefore, the controllability matrix $\mathbf{C}$ has rank $k - 2$, and G is not controllable. Next, we present one control strategy with $m = k - 1$.

Assume that the first $k - 1$ nodes are controlled, and $\mathbf{b}_j$ is associated with the node j for $j = 1, 2, \dots, k-1$. Let

$$\mathbf{b}_j = \mathsf{A}\mathbf{b}_{j-k+1}\mathbf{b}_{j-k+2}\dots\mathbf{b}_{j-1}$$

for $j = k, k+1, \dots, n$. Similarly, it can be shown that the following matrix

$$\begin{bmatrix} \mathbf{b}_1 & \mathbf{b}_2 & \dots & \mathbf{b}_n \end{bmatrix} \in \mathbb{R}^{n\times n}$$

is upper triangular with nonzero entries along the diagonal. Hence, the controllability matrix $\mathbf{C}$ has rank n, and the MCN of G is $k - 1$. □

Proposition IV.15. *Suppose that k is even and $k \geq 4$. If G is a k-uniform hyperring with n nodes and $n > k+1$, then the MCN of G is given by $n^* = k-1$.*

Proof. We first note that when $n = k+1$, G is complete, see Proposition 5 below. For $n > k+1$, the first part of the proof follows in exactly the same fashion as in Proposition 2. Moreover, due to the special structure of the adjacency tensor and the definition of tensor vector multiplication, it is straightforward to show that for $k \geq 4$, controlling the first $k-1$ nodes will be enough to make the rank of the controllability matrix $\mathbf{C}$ equal to n. Therefore, the MCN of G is $k-1$. □

Proposition IV.16. *Suppose that k is even. If G is a k-uniform hyperstar with n nodes and $n > k$, then the MCN of G is given by $n^* = n-2$.*

Proof. We first note that when $n = k$, G is a uniform hyperchain. For $n > k$, we show that when the number of control inputs $m = n-3$, G is never controllable. Let m_{int} denote the number of control inputs selected from the set of the internal nodes. Then $m - m_{\text{int}}$ is the number of control inputs selected from the set of the leaf nodes. According to the definition of tensor vector multiplication and the special structure of the adjacency tensor, it can be shown that

Case 1: When $m_{\text{int}} = k-1$, $\text{rank}(\mathbf{C}) = n-2$;

Case 2: When $m_{\text{int}} = k-2$, $\text{rank}(\mathbf{C}) = n-1$;

Case 3: When $0 \leq m_{\text{int}} < k-2$, $\text{rank}(\mathbf{C}) = n-3$.

Therefore, G is not controllable. However, if one adds one more input from the set of the leaf nodes in Case 2, the rank of the controllability matrix $\mathbf{C}$ will reach n. Hence, the MCN of G is $n-2$. □

Proposition IV.17. *Suppose that k is even. If G is a complete k-uniform hypergraph with n nodes, then the MCN of G is given by $n^* = n-1$.*

Proof. We first show that when the number of control inputs $m = n - 2$, G is never controllable. Without loss of generality, assume that the first $n - 2$ nodes are controlled, and $\mathbf{b}_j$ is associated with the node j for $j = 1, 2, \ldots, n-2$. According to the definition of tensor vector multiplication and the fact that G is complete, it can be shown that all the column vectors in the controllability matrix $\mathbf{C}$ have the last two entries equal. Thus, the rank of $\mathbf{C}$ is equal to $n - 1$, and G is not controllable. However, when the first $n-1$ nodes are controlled, the last node can be easily reached by any combination of $(k-1)$ $\mathbf{b}_j$ for $j = 1, 2, \ldots, n-1$ since G is complete, and the controllability matrix $\mathbf{C}$ has rank n. Therefore, the MCN of G is $n - 1$. □

Proposition IV.14, IV.16, and IV.17 are valid when $k = 2$ where the MCN of chains, stars and complete graphs are equal to 1, $n - 2$ and $n - 1$, respectively. Similarly to rings, the MCN of even uniform hyperrings does not depend on n. However, Proposition IV.15 does not hold for $k = 2$ because the pairwise transitions between nodes would produce linearly dependent relations in the standard rings. Furthermore, we discover that the MCN of odd uniform hyperchains, hyperrings, and hyperstars as well as complete odd uniform hypergraphs, identified by the Kalman-rank-like condition, follows exactly the same patterns as stated in Proposition IV.14 to IV.17 even though the condition only offers accessibility. Nonetheless, controllability or accessibility of a uniform hypergragh is closely associated to its underlying architecture, and the MCN can be used to quantify some topological attributes of a uniform hypergraph.

4.5 Hypergraph Robustness

Network robustness is the ability of a network to survive from random failures or deliberate attacks (e.g., removal of nodes or edges) [1, 9, 20]. It is intimately related

to the underlying network structure/topology. Many measures have been proposed to quantify the robustness of a graph, and one of the popular measures is called effective resistance [116, 45]. We propose MCN as a measure of hypergraph robustness since it can provide insights into the topology and connectivity of uniform hypergraphs according to their controllability or accessibility. Intuitively, if the MCN of a uniform hypergraph is high, it will take more effort/energy to control the hypergraph or steer the underlying system.

In Section 4.6.1 and 4.6.2, we will determine the MCN of even and odd uniform hyperchains, hyperrings, and hyperstars, and their different variants in simulated datasets. As expected, the rules for selecting the minimum subset of "control nodes" of odd uniform hypergraphs associated with the MCN follow the same patterns as these of even uniform hypergraphs. More interestingly, we find that the MCN of these configurations are related to their degree distributions, and the high degree nodes are often preferred as control nodes.

4.5.1 Control Nodes Selection with MCN

In order to measure the robustness of uniform hypergraphs, we need to efficiently determine their MCN. However, finding the MCN of uniform hypergraphs using a brute-force search will be computationally demanding. We provide a heuristic approach for estimating the minimum subset of control nodes of a uniform hypergraph in which nodes are chosen based on the maximum change in the rank of the reduced controllability matrix, see Algorithm 6. Here $\mathbf{C}_D$ denotes the reduced controllability matrix formed from the inputs in the index set D, and can be computed using Algorithm 5. If a uniform hypergraph is non-connected, we can first identify the connected components (which can be defined similarly as in graphs), and then apply the algorithm to each component. In Step 7, if multiple s^* are obtained, we can pick

one randomly, or use some other conditions to break the tie, e.g., by selecting the node with the highest degree. It turns out that Algorithm 6 with high likelihood can find the MCN of a medium-sized uniform hypergraph, and it is much faster than a brute-force search, see Section 4.6.5.

Algorithm 6 Greedy Control Nodes Selection with MCN.

Given a supersymmetric tensor $\mathsf{A} \in \mathbb{R}^{n\times n\times\cdots\times n}$
Let $S = \{1, 2, \ldots, n\}$ and $D = \emptyset$
while $\operatorname{rank}(\mathbf{C}_D) < n$ **do**
  for $s \in S \setminus D$ **do**
    Compute $\Delta(s) = \operatorname{rank}(\mathbf{C}_{D\cup\{s\}}) - \operatorname{rank}(\mathbf{C}_D)$ using Algorithm 5
  end for
  Set $s^* = \operatorname{argmax}_{s\in S\setminus D}\Delta(s)$
  Set $D = D \cup \{s^*\}$
end while
return A subset of control nodes D.

4.6 Numerical Examples

All the numerical examples presented were performed on a Linux machine with 16 GB RAM and a 2.0 GHz Quad-Core Intel Core i5 processor in MATLAB R2020a.

4.6.1 Even Uniform Hypergraphs

Recall that in k-uniform hyperchains, hyperrings, and hyperstars, every hyperegde has exactly $k-1$ overlapping nodes. However, for $k \geq 3$, one can relax this requirement allowing number of overlapping nodes between hyperedges to vary, and obtain variants of the three configurations. We consider the case where every intersection between hyperedges contains r nodes for $0 < r < k-1$, and denote these variants by r-hyperchains, r-hyperrings, and r-hyperstars. Note that k-uniform hyperchains, hyperrings, and hyperstars are the cases where $r = k-1$. Uniform hyperchains, hyperrings, and hyperstars, and their different variants have many applications in real life. For example, we can use them to model complex networks such as computer networks, supply chains and organizational hierarchy. Understanding the control

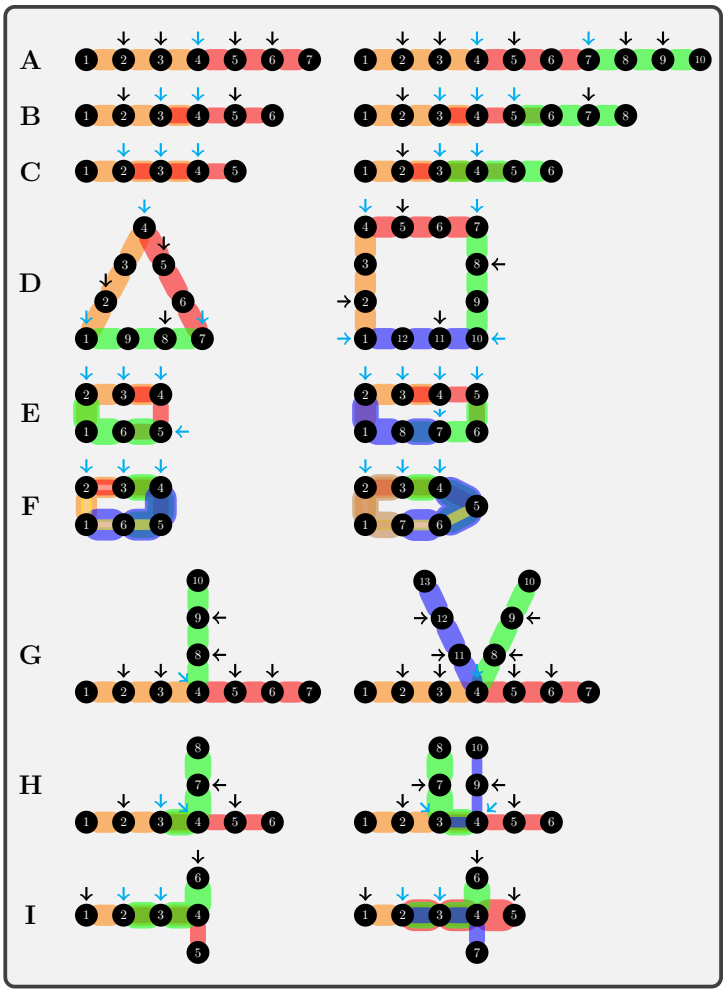


Figure 4.4: **MCN of 4-uniform hyperchains, hyperrings, and hyperstars, and their variants.** (A), (B), and (C) 4-uniform 1-hyperchains, 2-hyperchains, and hyperchains. (D), (E), and (F) 4-uniform 1-hyperrings, 2-hyperrings, and hyperrings. (G), (H), and (I) 4-uniform 1-hyperstars, 2-hyperstars, and hyperstars. The nodes with arrows are denoted as the control nodes, and the cyan arrows indicate the control nodes with highest degrees in the configurations.

mechanisms of these configurations will be greatly beneficial for achieving network security, efficient communications and energy savings.

In this example, we determine the MCN of 4-uniform hyperchains, hyperrings, and hyperstars, and their variants. The results are shown in Figure 4.4. We only present the most representative minimum subset of control nodes for each configuration. First, the MCN of 4-uniform hyperchains, hyperrings, and hyperstars is consistent with the results stated in Proposition 2 to 4. Controlling 4-uniform hyperchains, and

hyperrings only requires control of three nodes, and controlling 4-uniform hyperstars requires control of $n-2$ nodes, see Figure 4.4C, F, and I. Moreover, we discover that the MCN of these configurations is related to their degree distributions. Intuitively, controlling the high degree nodes is the easiest and most natural way for achieving hypergraph control. In particular, all the hypergraph configurations in Figure 4.4 contain at least one control node with the highest degree in the corresponding degree distributions. For 4-uniform 1-hyperchains, 1-hyperrings, and 1-hyperstars, in which there is one common node between hyperedges, the MCN can be achieved when all the 2-degree nodes are controlled with each hyperedge having three control nodes, see Figure 4.4A, D, and G. Furthermore, the control strategies for 4-uniform 2-hyperchains, 2-hyperrings, and 2-hyperstars, in which there are two common nodes between hyperedges, are more like some combinations of the previous two, which also require controlling nodes with the highest degree, see Figure 4.4B, E, and H. However, it is possible that low degree nodes can accomplish the same goal. For example, the minimum subset of control nodes $\{1, 2, 3, 5, 6, 8, 9\}$ can also achieve the full control of the 1-hyperchain with ten nodes.

We summarize the MCN of 4-uniform 1-, 2-hyperchains, 1-,2-hyperrings, and 1-,2-hyperstars with n nodes in Table 4.1 (column $n^*_{\text{4-unif}}$) based on our observations (i.e., they are not proved). Interestingly, for cases of 2-hyperchains, 2-hyperings and 2-hyperstars, we find that the MCN is $\frac{n+2}{2}$. Whether this results hold in general needs to be further investigated. Moreover, one may easily obtain the MCN of some hybrids of uniform hyperchians, hyperrings, and hyperstars, and their variants according to the control strategies discussed above.

Table 4.1: **MCN of the variants of 4- and 3-uniform hyperchains, hyperrings, and hyperstars based on our observations.** Note that 3-uniform 2-hyperchains, 2-hyperrings, and 2-hyperstars are the 3-uniform hyperchains, hyperrings, and hyperstars.

Configuration	$\boldsymbol{n}^*_{\text{4-unif}}$	$\boldsymbol{n}^*_{\text{3-unif}}$
1-Hyperchain	$\frac{2n+1}{3}$	$\frac{n+1}{2}$
1-Hyperring	$\frac{2n}{3}$	$\frac{n}{2}$
1-Hyperstar	$\frac{2n+1}{3}$	$\frac{n+1}{2}$
2-Hyperchain	$\frac{n+2}{2}$	2
2-Hyperring	$\frac{n+2}{2}$	2
2-Hyperstar	$\frac{n+2}{2}$	$n-2$

4.6.2 Odd Uniform Hypergraphs

The goal of this example is to determine the MCN of 3-uniform hyperchains, hyperrings, and hyperstars, and their variants using the Kalman-rank-like condition even though the condition only offers accessibility for these configurations. The results are shown in Figure 4.5. The rules for selecting the minimum subset of "control nodes" of 3-uniform hyperchains, hyperrings and hyperstars, and their variants follow the same patterns as these discussed in Section 4.4. The MCN of 3-uniform hyperchains, hyperrings, and hyperstars are matched with the results stated in Proposition 2 to 4 despite k being odd, see Figure 4.5B, D, and F. Moreover, for 3-uniform 1-hyperchains, 1-hyperrings, and 1-hyperstars, in which there is one common node between hyperedges, the MCN can be achieved when all the 2-degree nodes are "controlled" with each hyperedge having two "control nodes", see Figure 4.5A, C, and E. Similarly, we can conclude that the MCN of these configurations are related to their degree distributions. We summarize the MCN of 3-uniform hyperchains, hyperrings, and hyperstars, and their variants with n nodes in Table 4.1 (column $n^*_{\text{3-unif}}$) based on our observations. Again, we want to remark that although the "control nodes" of 3-uniform hypergraphs may not have physical interpretations in terms of controllability, the MCN can be used to measure hypergraph robustness and detect structural changes, as shown in the following example.

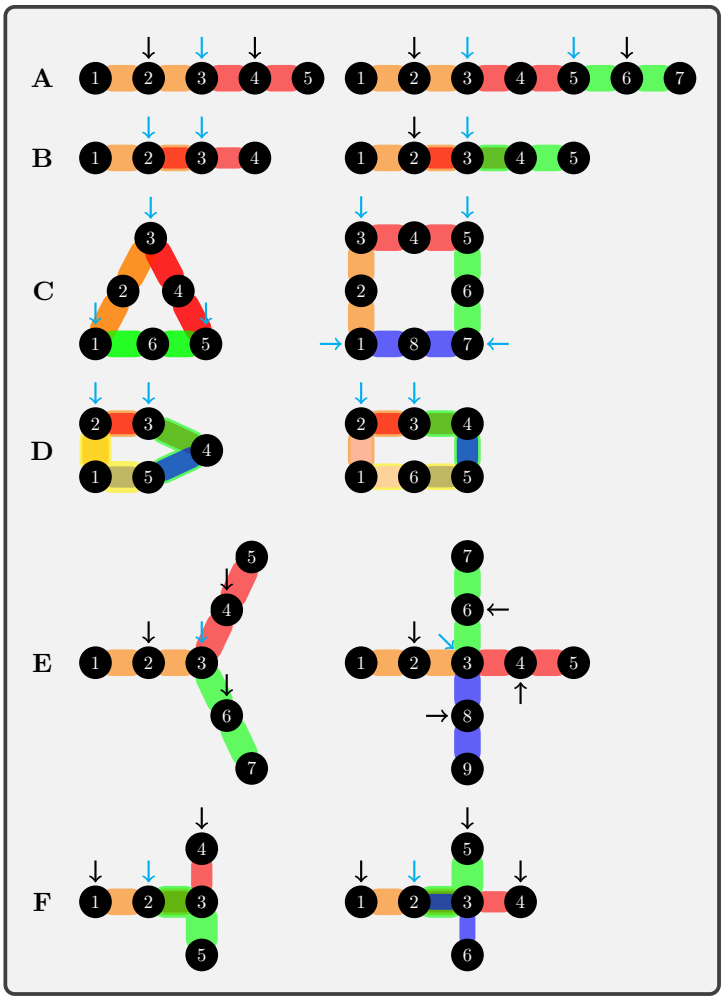


Figure 4.5: **MCN of 3-uniform hyperchains, hyperrings, and hyperstars, and their variants.** (A) and (B) 3-uniform 1-hyperchains and hyperchains. (C) and (D) 3-uniform 1-hyperrings and hyperrings. (E) and (F) 3-uniform 1-hyperstars and hyperstars. The nodes with arrows are denoted as the control nodes, and the cyan arrows indicate the control nodes with the highest degrees in the configurations.

4.6.3 Mouse Neuron Endomicroscopy

The mouse endomicroscopy dataset is an imaging video created under 10-minute periods of feeding, fasting and re-feeding using fluorescence across space and time in a mouse hypothalamus [23, 121]. Twenty neurons are recorded with individual levels of "firing". Similar to [23], we want to quantitatively differentiate the three phases using 3-uniform hypergraphs with MCN. First, we compute the multi-correlation among every three neurons, which is defined in (3.11) When the multi-correlation ρ

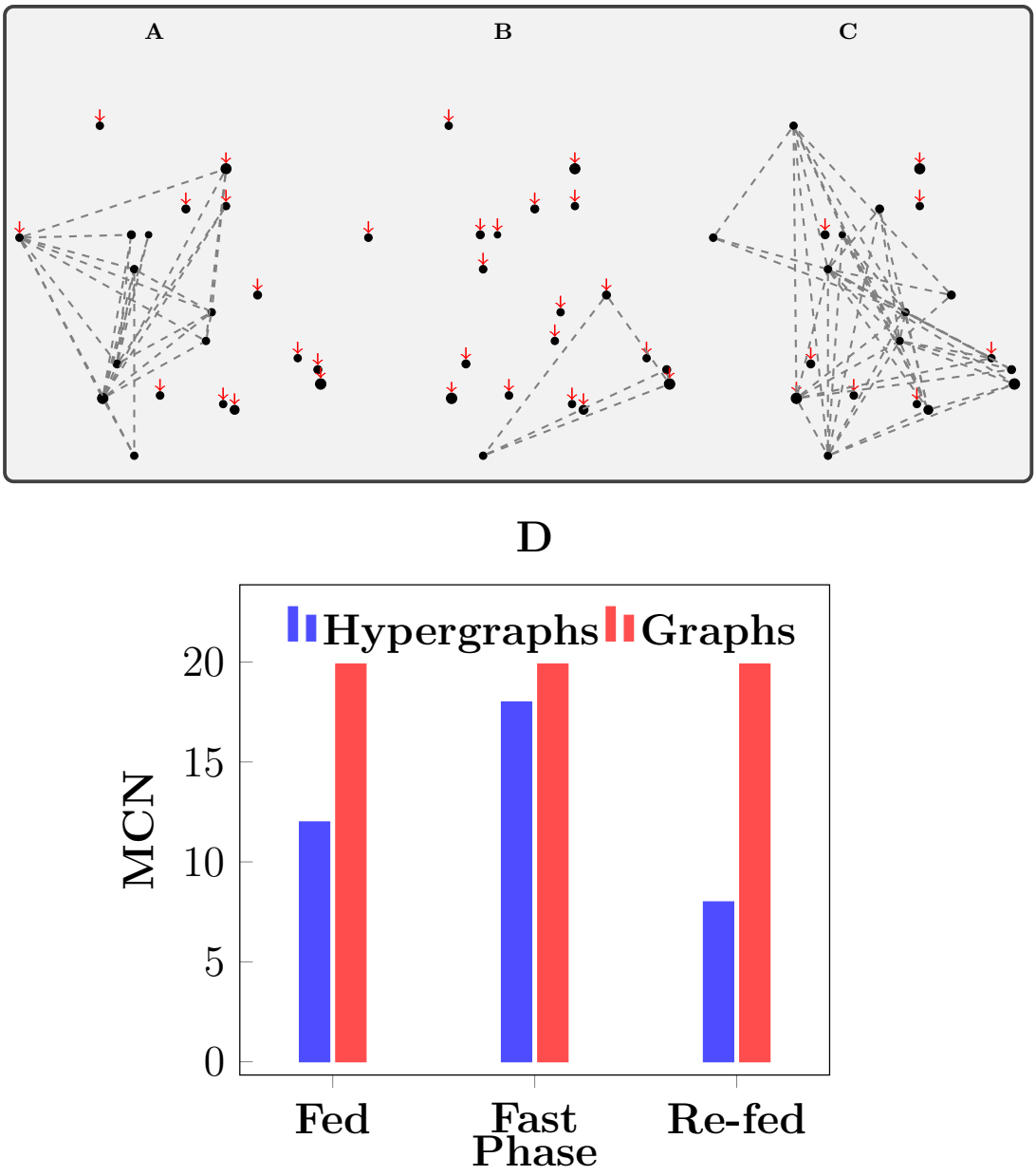


Figure 4.6: **Mouse neuron endomicroscopy features.** (A), (B), and (C) Neuronal activity networks of the three phases - fed, fast, and re-fed, which depicts the spatial location and size of individual cells. Each 2-simplex (i.e., a triangle) represents a hyperedge, and red arrows indicate those control nodes. (D) MCN for the neuronal activity networks modelled by 3-uniform hypergraphs and standard graphs. The cutoff threshold is 0.95 for both the hypergraph and graph models.

is greater than a prescribed threshold, we build hyperedges among the three neurons.

The results are shown in Figure 4.6, in which (A), (B), and (C) are network diagrams modelled by 3-uniform hypergraphs for a representative mouse depicting the

spatial location and size of individual cells (every 2-simplex is a hyperedge). It is evident from Figure 4.6D that the hypergraph MCN (blue) can successfully differentiate the three phases with different food treatments under the cutoff threshold 0.95. In particular, the fasting phase requires more neurons to be "controlled" due to fewer connections, while in the re-fed phase, the MCN is significantly reduced because of an outburst of neuron interactions. On the other hand, the MCN (red in Figure 4.6D) computed from the graph model fails to capture the changes in neuronal activity. The threshold 0.95 in the graph model is too high to produce any connection. This supports the fact that more than two neurons synchronize, or "co-fire", in the mouse hypothalamus, and the interactions can be more accurately captured by hypergraphs. Note that our choice of the prescribed threshold is arbitrary, though higher values are desirable as they capture stronger neuronal interactions as the relevant edges/hyeredges. To assess the sensitivity, we also performed our MCN analysis for values of threshold in the range from 0.90 to 0.95 and found similar qualitative results.

4.6.4 Allele-Specific Chromosomal Conformation Capture

Studies have revealed that there is significant coordination between allelic gene expression biases and local genome architectural changes [78]. The unbiased genome-wide technology of chromosome conformation capture (Hi-C) has been used to capture the architecture of the genome through the cell cycle [53, 103, 106]. The notion of transcription factories supports the existence of simultaneous interactions involving multiple genomic loci [35], implying that the human genome configuration can be represented by a hypergraph [23]. In the example, we are given Hi-C data for a small region of chromosome 15 (100 kb bin resolution) which contains two imprinted genes (*SNRPN* and *SNURF*). Imprinted genes are known to only express from one allele,

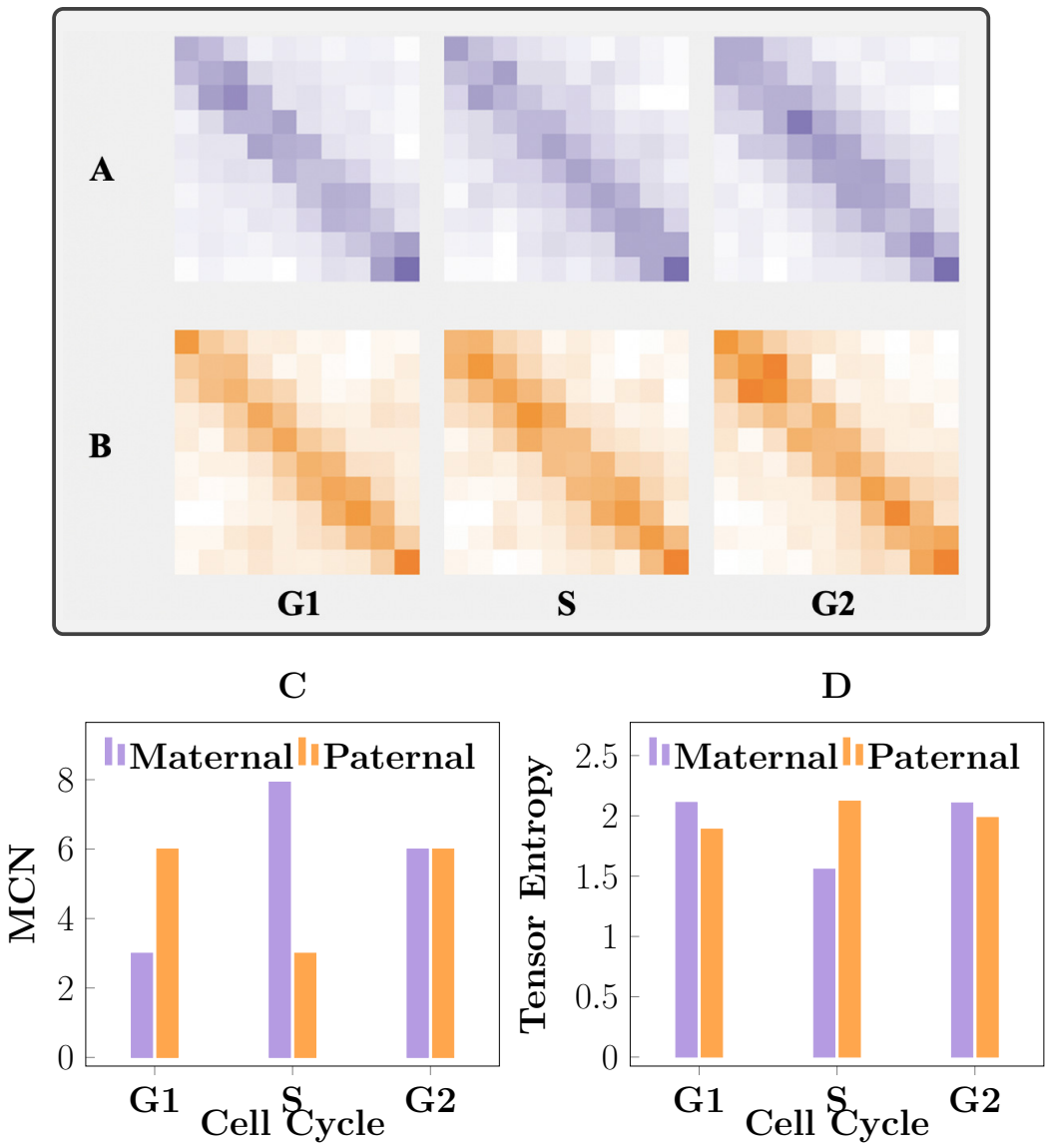


Figure 4.7: **Allele-specific Hi-C features.** (A) and (B) Hi-C maps of a local region surrounding the imprinted genes *SNRPN* and *SNURF* from the maternal and paternal Chromosome 15, respectively, through the cell cycle phases G1, S, and G2. The darker the color, the more interactions between two loci. (C) MCN of the 4-uniform hypergraphs, recovered from Hi-C measurements with multi-correlation cutoff threshold 0.99, through the cell cycle phases G1, S, and G2. (D) Tensor entropies of the 4-uniform hypergraphs described in (C).

so we want to explore any corresponding differences in local genome architecture [105]. Here we use 4-uniform hypergraphs to partially recover the 3D configuration of the genome according to the multi-correlation (3.11) from the Hi-C matrices.

The results are shown in Figure 4.7. Clearly, it is hard to tell the difference

between the maternal and paternal genome architectures directly from the Hi-C maps, see Figure 4.7A and B. However, after converting to hypergraphs, we can easily detect the structural discrepancy using the notion of MCN in the cell cycle phases G1 and S, see Figure 4.7C. Although the MCN are equal between the maternal and paternal networks in the cell cycle phase G2, the maximum possible choices of minimum subsets of control nodes are different (one is twenty four, and one is thirty three). This indicates that there are some architectural similarities between the maternal and paternal architectures in G2 compared to the previous two phases. Furthermore, we corroborate our results by using the notion of tensor entropy. Tensor entropy is a spectral measure, which can decipher topological attributes of uniform hypergraphs [23]. In particular, the two results of tensor entropy and MCN are consistent, in the sense that the largest gap of tensor entropy between the maternal and paternal architectures occurs in S, and the smallest gap occurs in G2, see Figure 4.7D. Biologically, in S phase, DNA replication of the genomes may lead to a large structural dissimilarity between the maternal and paternal architectures, while in the G2 phase, both genomes prepare for mitosis which may result in a small structural discrepancy. Moreover, we believe that the control loci (nodes) can play a significant role in cellular reprogramming, a process that introduces proteins called transcription factors as a control mechanism for transforming one cell type into another.

4.6.5 MCN Computation Comparison

In this example, we compare the computational efficiency of the heuristic approach described in Algorithm 6 and brute-force search in finding the MCN of random 4-uniform hypergraphs (with hyperedge density 50%) and complete 4-uniform hypergraphs. The results are shown in Figure 4.8. Evidently, Algorithm 6 is more time-efficient than a brute-force search as the number of nodes becomes larger in

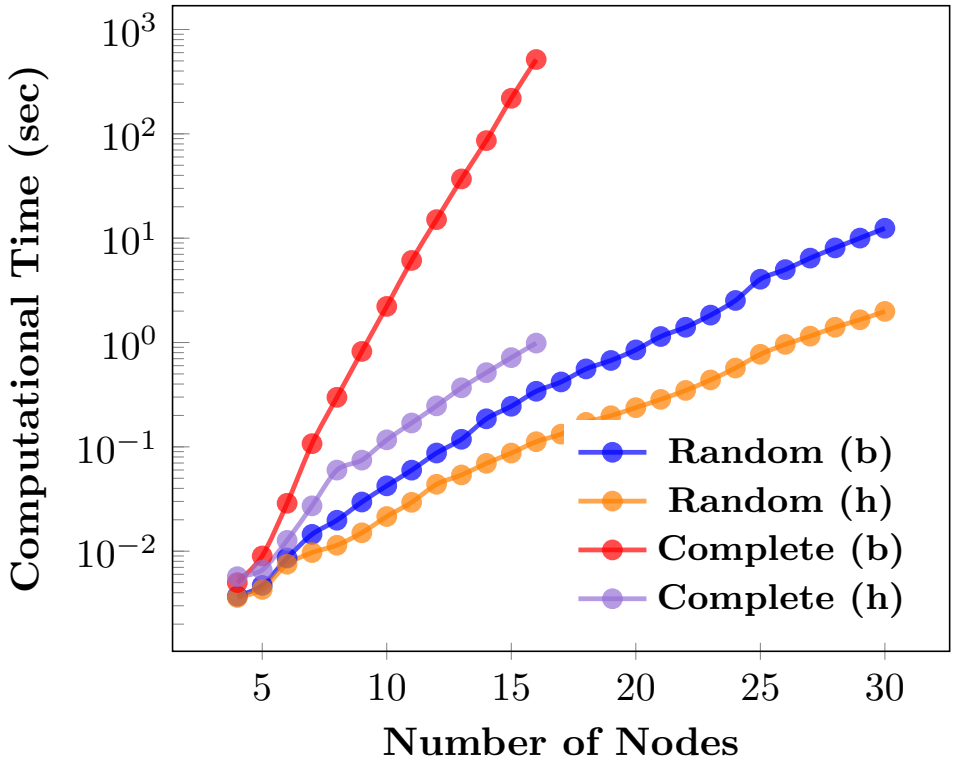


Figure 4.8: **Computational time comparisons in determining MCN.** In the legend, the letter b stands for the brute-force search, while the letter h stands for the heuristic approach. Since the computational time using a brute-force search in determining the MCN of the complete uniform hypergraphs grows very fast, we only compute them up to sixteen nodes for comparison. For the purpose of accuracy, we ran each algorithm five times and took the average of the computational times.

both the configurations. In particular, when a uniform hypergraph is complete (or nearly complete), the heuristic exhibits a huge time advantage. More significantly, it produces exactly the same MCN as a brute-force search in these simulations.

4.7 Discussion

The first four numerical examples reported here highlight that the tensor-based multilinear system (4.3) can characterize the multidimensional interactions in uniform hypergraphs. The MCN of uniform hyperchains, hyperrings, and hyperstars, and their variants are related to their degree distributions. It is also a good indicator of uniform hypergraph robustness. However, more theoretical and numerical

investigations are required to evaluate the controllability of more general uniform hypergraphs, and its relation to the hypergraph topology. Moreover, in practice, hypergraphs like co-authorship networks and protein-protein interaction networks are very large, so computing the reduced controllability matrix and its corresponding MCN is still challenging. Tensor decomposition and a "maximum matching" type approach need to be considered in order to facilitate efficient computations [5, 56, 69].

Instead of looking at uniform hypergraphs, can we think of controllability of more general hypergraphs where each hyperedge contains an arbitrary amount of nodes? The main idea is to make non-uniform hypergraphs uniform, which can then be represented by tensors. In the following, we adopt the definition of generalized adjacency tensors of non-uniform hypergraphs from [7].

Definition IV.18. Let $\mathsf{G} = \{\mathbf{V}, \mathbf{E}\}$ be a hypergraph with n nodes, and k be the maximum cardinality of the hyperedges. The adjacency tensor $\mathsf{A} \in \mathbb{R}^{n\times n\times\cdots\times n}$ of G, which is a kth order n-dimensional supersymmetric tensor, is defined as

$$\mathsf{A}_{j_1 j_2 \ldots j_k} = \begin{cases} \frac{s}{\alpha} \text{ if } (i_1, i_2, \ldots, i_s) \in \mathbf{E} \\ \\ 0, \text{ otherwise} \end{cases}, \tag{4.7}$$

where, $j_l \in \{i_1, i_2, \ldots, i_s\}$ with at least once for each element of the set for $l = 1, 2, \ldots, k$, and

$$\alpha = \sum_{\substack{k_1+k_2+\cdots+k_s=k \\ k_1,k_2,\ldots,k_s\geq 1}} \frac{k!}{\prod_{l=1}^{s} k_i!}.$$

The choice of the nonzero coefficients $\frac{s}{\alpha}$ preserves the degree of each node, i.e., the degree of node j computed using (1.10) with weights as defined above is equal to number of hyperedges containing the node in the original non-uniform hypergraph.

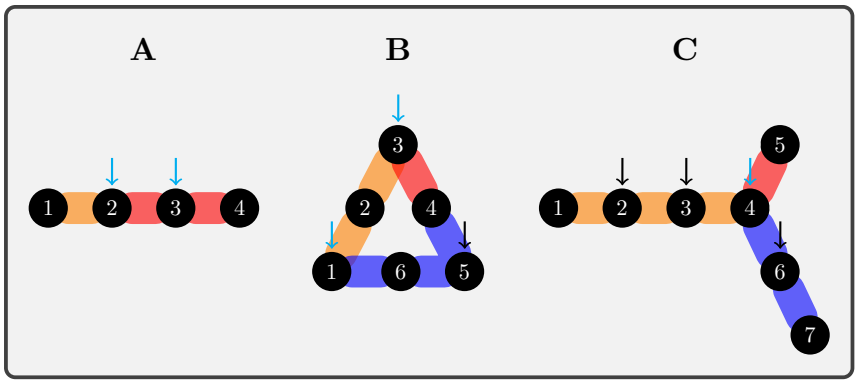


Figure 4.9: **Controllability of non-uniform hypergraphs with MCN.** (A) Non-uniform hyperchain with $e_1 = \{1, 2\}$ and $e_2 = \{2, 3, 4\}$. (B) Non-uniform hyperring with $e_1 = \{1, 2, 3\}$, $e_2 = \{3, 4\}$, and $e_3 = \{4, 5, 6, 1\}$. (C) Non-uniform hyperstar with $e_1 = \{1, 2, 3, 4\}$, $e_2 = \{4, 5\}$, and $e_3 = \{4, 6, 7\}$. Nodes with arrows from the top are the control nodes, and the cyan arrows indicate the control nodes with the highest degrees in the configurations.

When G is uniform, the above definition reduces to Definition I.2. See [7] for examples. Once we have the adjacency tensor of a hypergraph, we can discuss the controllability of the hypergraph when k is even using the techniques developed in Section 4.3. For curiosity, we build several non-uniform hypergraphs and determine their MCN. The results are shown in Figure 4.9. Intriguingly, the control strategies for non-uniform hyperchains, hyperrings and hyperstars with one overlapping nodes between hyperedges are similar to those discussed in Section 4.4. High degree nodes are preferred to be controlled with each hyperedge containing $s - 1$ control nodes where s is the cardinality of the hyperedge (there is one exception in the hyperring configuration).

CHAPTER V

Conclusion

In Chapter 2, we provided a comprehensive treatment of a newly introduced MLTI system representation using even-order paired tensors and the Einstein product. We established new results which enable one to express tensor unfolding-based stability, reachability, and observability criteria in terms of more standard notions of tensor ranks/decompositions. We introduced a generalized CPD/TTD-based model reduction framework which can significantly reduce the number of MLTI system parameters and realize the tensor decomposition-based methods. We also presented computational complexity analysis of our proposed framework, and illustrated the benefits through numerical examples. In particular, TTD offers several computational advantages over CPD and HOSVD, and provides a good representational choice for facilitating numerical computations associated with MLTI systems. As mentioned in Section 2.8, more work is required to fully realize the potential of tensor algebra-based computations for MLTI systems. It will also be worthwhile to develop theoretical and computational framework for observer and feedback control design for MLTI systems, and apply these techniques in real world complex systems. One particular application we plan to investigate is that of cellular reprogramming which involves introducing transcription factors as a control mechanism to trans-

form one cell type to another. These systems naturally have matrix or tensor state spaces describing their genome-wide structure and gene expression [80, 108]. Such applications would also need to account for nonlinearity and stochasticity in tensor based dynamical system representation and analysis framework, and is an important direction for future research.

In Chapter 3, we proposed a new notion of entropy for uniform hypergraphs based on the tensor higher-order singular value decomposition. The k-mode singular values of Laplacian tensors provide nice interpretations regarding the structural properties of uniform hypergraphs. The tensor entropy heavily depends on the node degrees, path lengths, clustering coefficients, and nontrivial symmetricity. We investigated the lower and upper bounds of the entropy, and provided the entropy formula for complete uniform hypergraphs. A TTD-based computational framework is proposed for computing the tensor entropy efficiently. We also applied this spectral measure to real biological networks for anomaly detection, and achieve better performances compared to the von Neumann graph entropy. As discussed in Section 3.5, the detailed relations between tensor eigenvalues and entropy, and the theoretical investigations of hypergraph robustness require further exploration.

In Chapter 4, we proposed a new notion of controllability for uniform hypergraphs based on tensor algebra and polynomial control theory. We represented the dynamics of uniform hypergraphs by a tensor product-based multilinear system, and derived a Kalman-rank-like condition to determine the controllability of even uniform hypergraphs. We established theoretical results on the MCN of even uniform hyperchains, hyperrings, and hyperstars as well as complete even uniform hypergraphs. We proposed MCN as a measure of hypergraph robustness, and found that is it related to the hypergraph degree distribution. We also presented a heuristic to

obtain the MCN efficiently. Additionally, we applied the notion of MCN to the real world biological networks to quantify structural differences, and achieved outstanding performances. Finally, we discussed the controllability of general hypergraphs. As mentioned in Section 4.7, a "maximum matching" type approach is needed in order to find the MCN of large undirected/directed hypergraphs in a scalable fashion. On the other hand, more work is required to fully understand the control properties of the tensor-based multilinear system (4.3). For example, it will be useful to realize the potential of tensor algebra-based computations for controllability Gramians and Lyapunov equations. In addition, it will be worthwhile to develop theoretical and computational frameworks for observer and feedback control design, and apply them to the dynamics of hypergraphs.

APPENDICES

.1 Tensor Eigenvalues

The tensor eigenvalue problems of real supersymmetric tensors were first explored by Qi [96] and Lim [74] independently. There are different notions of tensor eigenvalues; eigenvalues and E-eigenvalues are most frequently used. The *eigenvalues* $\lambda \in \mathbb{R}$ and *eigenvectors* $\mathbf{v} \in \mathbb{R}^n$ of a kth order supersymmetric tensor $\mathsf{X} \in \mathbb{R}^{n\times n\times\cdots\times n}$ are defined as follows:

$$\mathsf{X}\mathbf{v}^{k-1} = \lambda \mathbf{v}^{[k-1]},$$

where $\mathbf{v}^{[k-1]}$ stands for the element-wise $(k-1)$th power of $\mathbf{v}$. The eigenvalues λ could be complex, and if λ are real, we call them *H-eigenvalues*. The behaviors of tensor eigenvalues are close to eigenvalues of matrices. There is a total of $n(k-1)^{n-1}$ eigenpairs. The sum of all eigenvalues are equal to $(k-1)^{n-1}\text{trace}(\mathsf{X})$ where $\text{trace}(\mathsf{X}) = \sum_{j=1}^{n} \mathsf{X}_{jj\ldots j}$, and the product of all eigenvalues are equal to the hyperdeterminant $\det(\mathsf{X})$, the resultant of $\mathsf{X}\mathbf{v}^{k-1} = \mathbf{0}$.

Second, the *E-eigenvalues* $\lambda \in \mathbb{R}$ and *E-eigenvectors* $\mathbf{v} \in \mathbb{R}^n$ of a kth order supersymmetric tensor $\mathsf{X} \in \mathbb{R}^{n\times n\times\cdots\times n}$ are defined as follows:

$$\begin{cases} \mathsf{X}\mathbf{v}^{k-1} = \lambda \mathbf{v} \\ \mathbf{v}^\top \mathbf{v} = 1 \end{cases}.$$

The E-eigenvalues λ could be complex, and if λ are real, we call them *Z-eigenvalues*.

.2 MATLAB Functions

.2.1 The colon operator

The colon : is one of the most useful operators in MATLAB, which can create vectors, subscript arrays and specify for iterations. For our purpose, it acts as shorthand to include all subscripts in a particular array dimension [83]. For example, $\mathbf{A}_{:i}$ is equivalent to $\mathbf{A}_{ji}$ for all j.

.2.2 The `reshape` operator

The command $\mathsf{B} = \texttt{reshape}(\mathsf{A}, n_1, n_2, \ldots, n_k)$ reshapes a tensor A into a $n_1 \times n_2 \times \cdots \times n_k$ order tensor such that the number of elements in B matches the number of elements in A [83].

.2.3 The `rank` operator

The command $r = \texttt{rank}(\mathbf{A})$ computes the rank of a matrix $\mathbf{A}$ [83].

.2.4 The `diag` operator

The command $\mathbf{d} = \texttt{diag}(\mathbf{A})$ extracts the diagonal elements of a matrix $\mathbf{A}$ as a vector [83].

BIBLIOGRAPHY

BIBLIOGRAPHY

[1] W. Abbas and M. Egerstedt. Robust graph topologies for networked systems. *IFAC Proceedings Volumes*, 45(26):85–90, 2012.

[2] D. Aeyels. Local and global controllability for nonlinear systems. *Systems & control letters*, 5(1):19–26, 1984.

[3] L. Akoglu, H. Tong, and D. Koutra. Graph based anomaly detection and description: A survey. *Data Mining and Knowledge Discovery*, 29(3):626–688, 4 2015.

[4] L. A. N. Amaral, A. Scala, M. Barthélémy, and H. E. Stanley. Classes of small-world networks. *Proceedings of the National Academy of Sciences*, 97(21):11149–11152, 2000.

[5] B. W. Bader and T. G. Kolda. Algorithm 862: MATLAB tensor classes for fast algorithm prototyping. *ACM Transactions on Mathematical Software*, 32(4):635–653, December 2006.

[6] J. Baillieul. Controllability and observability of polynomial dynamical systems. *Nonlinear Analysis: Theory, Methods & Applications*, 5(5):543–552, 1981.

[7] A. Banerjee, A. Char, and B. Mondal. Spectra of general hypergraphs. *Linear Algebra and its Applications*, 518:14–30, 2017.

[8] A. Barabasi and E. Bonabeau. Scale-free networks. *Scientific American*, 288, 2003.

[9] A.-L. Barabási and J. Frangos. *Linked: The new science of networks*. American Association of Physics Teachers, 2003.

[10] C. Berge. *Hypergraphs, Combinatorics of Finite Sets, Third Edition*. North-Holland, Amsterdam, 1989.

[11] G. Bergqvist and E. G. Larsson. The higher-order singular value decomposition: Theory and an application [lecture notes]. *IEEE Signal Processing Magazine*, 27(3):151–154, May 2010.

[12] A. M. Bloch. An introduction to aspects of geometric control theory. In *Nonholonomic mechanics and control*, pages 199–224. Springer, 2003.

[13] I. Bloch and A. Bretto. A new entropy for hypergraphs. In M. Couprie, J. Cousty, Y. Kenmochi, and N. Mustafa, editors, *Discrete Geometry for Computer Imagery*, pages 143–154, Cham, 2019. Springer International Publishing.

[14] S. Braunstein, S. Ghosh, T. Mansour, S. Severini, and R. C. Wilson. Some families of density matrices for which separability is easily tested. *Physical Review A*, 73, 2006.

[15] S. Braunstein, S. Ghosh, and S. Severini. The laplacian of a graph as a density matrix: A basic combinatorial approach to separability of mixed states. *Annals of Combinatorics*, 10, 07 2004.

[16] M. J. Brazell, N. Li, C. Navasca, and C. Tamon. Solving multilinear systems via tensor inversion. *SIAM Journal on Matrix Analysis Applications*, 34:542–570, 2013.

[17] R. Brockett. *Finite Dimensional Linear Systems*. Society for Industrial and Applied Mathematics, Philadelphia, PA, 2015.

[18] B. J. Broxson. The kronecker product. *UNF Theses and Dissertations*, 2006.

[19] P. Brunovsky. Local controllability of odd systems. *Banach Center Publications*, 1:39–458, 1976.

[20] D. S. Callaway, M. E. Newman, S. H. Strogatz, and D. J. Watts. Network robustness and fragility: Percolation on random graphs. *Physical review letters*, 85(25):5468–5471, 2000.

[21] T. Carletti, D. Fanelli, and S. Nicoletti. Dynamical systems on hypergraphs. *Journal of Physics: Complexity*, 1(3), 2020.

[22] S.-H. Cha. Comprehensive survey on distance/similarity measures between probability density functions. *International Journal of Mathematical Models and Methods in Applied Sciences*, 1(4):300–307, 2007.

[23] C. Chen and I. Rajapakse. Tensor entropy for uniform hypergraphs. *IEEE Transactions on Network Science and Engineering*, 7(4):2889–2900, 2020.

[24] C. Chen, A. Surana, A. Bloch, and I. Rajapakse. Multilinear time invariant system theory. In *2019 Proceedings of the Conference on Control and its Applications*, pages 118–125.

[25] C. Chen, A. Surana, A. Bloch, and I. Rajapakse. Data-driven model reduction for multilinear control systems via tensor trains, 2019.

[26] C. Chen, A. Surana, A. Bloch, and I. Rajapakse. Controllability of hypergraphs. *IEEE Transactions on Network Science and Engineering*, pages 1–1, 2021.

[27] C. Chen, A. Surana, A. M. Bloch, and I. Rajapakse. Multilinear control systems theory. *SIAM Journal on Control and Optimization*, 59(1):749–776, 2021.

[28] H. Chen, J. Chen, L. A. Muir, S. Ronquist, W. Meixner, M. Ljungman, T. Ried, S. Smale, and I. Rajapakse. Functional organization of the human 4d nucleome. *Proceedings of the National Academy of Sciences*, 112(26):8002–8007, 2015.

[29] L. Chen, L. Han, T.-Y. Li, and L. Zhou. Teneig, 2015. version 2.0.

[30] L. Chen, L. Han, and L. Zhou. Computing tensor eigenvalues via homotopy methods. *SIAM Journal on Matrix Analysis and Applications*, 37(1):290–319, 2016.

[31] Y. Chen, L. Qi, and X. Zhang. The fiedler vector of a laplacian tensor for hypergraph partitioning. *SIAM Journal on Scientific Computing*, 39(6):A2508–A2537, 2017.

[32] Ching-Tai Lin. Structural controllability. *IEEE Transactions on Automatic Control*, 19(3):201–208, 1974.

[33] F. R. K. Chung, P. Erdös, and R. L. Graham. Minimal decompositions of hypergraphs into mutually isomorphic subhypergraphs. *J. Comb. Theory, Ser. A*, 32(2):241–251, 1982.

[34] C. Commault. Structural controllability of networks with dynamical structured nodes. *IEEE Transactions on Automatic Control*, 65(6):2736–2742, 2020.

[35] P. R. Cook and D. Marenduzzo. Transcription-driven genome organization: a model for chromosome structure and the regulation of gene expression tested through simulations. *Nucleic acids research*, 46(19):9895–9906, 2018.

[36] J. Cooper and A. Dutle. Spectra of uniform hypergraphs. *Linear Algebra and its Applications*, 436(9):3268–3292, 2012.

[37] L.-B. Cui, C. Chen, W. Li, and M. K. Ng. An eigenvalue problem for even order tensors with its applications. *Linear and Multilinear Algebra*, 64(4):602–621, 2016.

[38] G. F. de Arruda, G. Petri, and Y. Moreno. Social contagion models on hypergraphs. *Phys. Rev. Research*, 2:023032, Apr 2020.

[39] L. De Lathauwer, B. De Moor, and J. Vandewalle. A multilinear singular value decomposition. *SIAM Journal on Matrix Analysis and Applications*, 21(4):1253–1278, 2000.

[40] V. de Silva and L. Lim. Tensor rank and the ill-posedness of the best low-rank approximation problem. *SIAM Journal on Matrix Analysis and Applications*, 30(3):1084–1127, 2008.

[41] W. Ding, K. Liu, E. Belyaev, and F. Cheng. Tensor-based linear dynamical systems for action recognition from 3d skeletons. *Pattern Recognition*, 77:75–86, 2018.

[42] Y. Dolgin and E. Zeheb. Model reduction of uncertain systems retaining the uncertainty structure. *Systems & Control Letters*, 54(8):771–779, 2005.

[43] C. Donnat and S. Holmes. Discovering important nodes through graph entropy the case of enron email database. In *LinkKDD '05: Proceedings of the 3rd international workshop on Link discovery*, pages 74–81, 2005.

[44] C. Donnat and S. Holmes. Tracking network dynamics: A survey using graph distances. *The annals of applied statistics*, 12, 2018.

[45] W. Ellens, F. Spieksma, P. V. Mieghem, A. Jamakovic, and R. Kooij. Effective graph resistance. *Linear Algebra and its Applications*, 435(10):2491–2506, 2011. Special Issue in Honor of Dragos Cvetkovic.

[46] M. T. Fischer, D. Arya, D. Streeb, D. Seebacher, D. A. Keim, and M. Worring. Visual analytics for temporal hypergraph model exploration. *IEEE Transactions on Visualization and Computer Graphics*, 2020.

[47] L. Fortuna, G. Nunnari, and A. Gallo. *Model Order Reduction Techniques with Applications in Electrical Engineering*. Springer London, 2012.

[48] K. Fujarewicz, M. Kimmel, and A. Swierniak. On fitting of mathematical models of cell signaling pathways using adjoint systems. *Mathematical Biosciences & Engineering*, 2(3):527–534, 2005.

[49] A. J. Gates and L. M. Rocha. Control of complex networks requires both structure and dynamics. *Scientific reports*, 6(1):1–11, 2016.

[50] P. Gelß. *The Tensor-train Format and Its Applications: Modeling and Analysis of Chemical Reaction Networks, Catalytic Processes, Fluid Flows, and Brownian Dynamics*. Freie Universität Berlin, 2017.

[51] B. A. Gemmetto, V. and C. Cattuto. Mitigation of infectious disease at school: targeted class closure vs school closure. *BMC Infect Dis*, 14:695, 2014.

[52] R. W. Hamming. Error detecting and error correcting codes. *The Bell System Technical Journal*, 29(2):147–160, April 1950.

[53] L. H. Hartwell and T. A. Weinert. Checkpoints: controls that ensure the order of cell cycle events. *Science*, 246(4930):629–634, 1989.

[54] Z.-H. He, C. Navasca, and Q.-W. Wang. Tensor decompositions and tensor equations over quaternion algebra. 10 2017.

[55] C. J. Hillar and L.-H. Lim. Most tensor problems are np-hard. *J. ACM*, 60(6):45:1–45:39, Nov. 2013.

[56] J. E. Hopcroft and R. M. Karp. An n^5/2 algorithm for maximum matchings in bipartite graphs. *SIAM Journal on computing*, 2(4):225–231, 1973.

[57] D. Hu, X. L. Li, X. G. Liu, and S. G. Zhang. Extremality of graph entropy based on degrees of uniform hypergraphs with few edges. *Acta Mathematica Sinica, English Series*, 35(7):1238–1250, Jul 2019.

[58] S. Hu and L. Qi. The eigenvectors associated with the zero eigenvalues of the laplacian and signless laplacian tensors of a uniform hypergraph. *Discrete Applied Mathematics*, 169:140–151, 2014.

[59] Z. Huang and L. Qi. Positive definiteness of paired symmetric tensors and elasticity tensors. *Journal of Computational and Applied Mathematics*, 338, 05 2017.

[60] M. D. Humphries and K. Gurney. Network 'small-world-ness': A quantitative method for determining canonical network equivalence. *PLOS ONE*, 3(4):1–10, 04 2008.

[61] B. Jiang, F. Yang, and S. Zhang. Tensor and its tucker core: The invariance relationships. *Numerical Linear Algebra with Applications*, 24(3):e2086, 2017.

[62] N. Z. Jianji Wang. Measures of correlation for multiple variables. 2019.

[63] V. Jurdjevic and I. Kupka. Polynomial control systems. *Math. Ann.*, 272:361–368, 1985.

[64] T. Kailath. *Linear Systems*. Information and System Sciences Series. Prentice-Hall, 1980.

[65] M. Karsai, N. Perra, and A. Vespignani. Time varying networks and the weakness of strong ties. *Sci Rep*, 4, 2015.

[66] B. Khoromskij. *Tensor Numerical Methods in Scientific Computing*. 06 2018.

[67] D. Klein and M. Randic. Resistance distance. *Journal of Mathematical Chemistry*, 12:81–95, 1993.

[68] S. Klus, P. Gelß, S. Peitz, and C. Schütte. Tensor-based dynamic mode decomposition. *Nonlinearity*, 31(7):3359–3380, Jun 2018.

[69] T. Kolda and B. Bader. Tensor decompositions and applications. *SIAM Review*, 51(3):455–500, 2009.

[70] T. G. Kolda. Multilinear operators for higher-order decompositions. 2006.

[71] M. Kredler. Fast kronecker matrix multiplication, 2015.

[72] K. Kruppa. Comparison of tensor decomposition methods for simulation of multilinear time-invariant systems with the mti toolbox. *IFAC-PapersOnLine*, 50(1):5610–5615, 2017. 20th IFAC World Congress.

[73] J. B. Kruskal. Three-way arrays: rank and uniqueness of trilinear decompositions, with application to arithmetic complexity and statistics. *Linear Algebra and its Applications*, 18(2):95–138, 1977.

[74] Lek-Heng Lim. Singular values and eigenvalues of tensors: a variational approach. In *1st IEEE International Workshop on Computational Advances in Multi-Sensor Adaptive Processing*, pages 129–132, 2005.

[75] M. Levandowsky and D. Winter. Distance between sets. *Nature*, 234:34–35, 1971.

[76] A. Li and Y. Pan. Structural information and dynamical complexity of networks. *IEEE Transactions on Information Theory*, 62(6):3290–3339, June 2016.

[77] M. Liang, B. Zheng, and R. Zhao. Tensor inversion and its application to the tensor equations with einstein product. *Linear and Multilinear Algebra*, 67(4):843–870, 2019.

[78] S. Lindsly, W. Jia, H. Chen, S. Liu, S. Ronquist, C. Chen, X. Wen, G. Dotson, G. Omenn, S. C. Li, et al. Functional organization of the maternal and paternal human 4d nucleome. *bioRxiv*, 2020.

[79] S. Liu, H. Chen, S. Ronquist, L. Seaman, N. Ceglia, W. Meixner, P.-Y. Chen, G. Higgins, P. Baldi, S. Smale, et al. Genome architecture mediates transcriptional control of human myogenic reprogramming. *Iscience*, 6:232–246, 2018.

[80] S. Liu, H. Chen, S. Ronquist, L. Seaman, N. Ceglia, W. Meixner, P.-Y. Chen, G. Higgins, P. Baldi, S. Smale, A. Hero, L. A. Muir, and I. Rajapakse. Genome architecture mediates transcriptional control of human myogenic reprogramming. *iScience*, 6:232–246, 2018.

[81] Y. Liu, J. Slotine, and A. Barabasi. Controllability of complex networks. *Nature*, 473, 2011.

[82] Y.-Y. Liu and A.-L. Barabási. Control principles of complex systems. *Rev. Mod. Phys.*, 88:035006, Sep 2016.

[83] MATLAB. *9.7.0.1190202 (R2019b)*. The MathWorks Inc., Natick, Massachusetts, 2018.

[84] J. Melody, T. Basar, and F. Bullo. On nonlinear controllability of homogeneous systems linear in control. *IEEE transactions on automatic control*, 48(1):139–143, 2003.

[85] I. Mezic. Spectral properties of dynamical systems, model reduction and decompositions. *Nonlinear Dynamics*, 41:309–325, 08 2005.

[86] G. Minello, L. Rossi, and A. Torsello. On the von Neumann entropy of graphs. *Journal of Complex Networks*, 7(4):491–514, 11 2018.

[87] N. D. Monnig and F. G. Meyer. The resistance perturbation distance: A metric for the analysis of dynamic networks. *Discrete Applied Mathematics*, 236:347–386, 2018.

[88] J. C. Nacher, M. Ishitsuka, S. Miyazaki, and T. Akutsu. Finding and analysing the minimum set of driver nodes required to control multilayer networks. *Scientific reports*, 9(1):1–12, 2019.

[89] M. Newman. *Networks: An Introduction*. OUP Oxford, 2010.

[90] G. Obinata and B. Anderson. *Model Reduction for Control System Design*. Communications and Control Engineering. Springer London, 2012.

[91] I. Oseledets. Tensor-train decomposition. *SIAM Journal on Scientific Computing*, 33(5):2295–2317, 2011.

[92] I. Oseledets, S. Dolgov, V. Kazeev, O. Lebedeva, and T. Mach. Tt-toolbox, 2014. Version 2.2.2.

[93] I. Oseledets and E. Tyrtyshnikov. Breaking the curse of dimensionality, or how to use svd in many dimensions. *SIAM Journal on Scientific Computing*, 31(5):3744–3759, 2009.

[94] D. A. Papadimitriou, P. and Garcia-Molina. Mitigation of infectious disease at school: targeted class closure vs school closure. *J Internet Serv Appl*, 1:19–30, 2010.

[95] F. Passerini and S. Severini. The von neumann entropy of networks. 2008.

[96] L. Qi. Eigenvalues of a real supersymmetric tensor. *Journal of Symbolic Computation*, 40(6):1302–1324, 2005.

[97] L. Qi. H^+-eigenvalues of laplacian and signless laplacian tensors. *Communications in Mathematical Sciences*, 12, 03 2013.

[98] L. Qi, H.-H. Dai, and D. Han. Conditions for strong ellipticity and m-eigenvalues. 2009.

[99] S. Ragnarsson and C. Van Loan. Block tensor unfoldings. *SIAM J. Matrix Analysis Applications*, 33:149–169, 2012.

[100] S. Ragnarsson and C. Van Loan. Block tensors and symmetric embeddings. *Linear Algebra and its Applications*, 438(2):853–874, 2013. Tensors and Multilinear Algebra.

[101] A. Rahmani, M. Ji, M. Mesbahi, and M. Egerstedt. Controllability of multi-agent systems from a graph-theoretic perspective. *SIAM Journal on Control and Optimization*, 48(1):162–186, 2009.

[102] A. Rahmani and M. Mesbahi. On the controlled agreement problem. In *2006 American Control Conference*. IEEE, 2006.

[103] I. Rajapakse and M. Groudine. On emerging nuclear order. *The Journal of Cell Biology*, 192(5):711–721, 2011.

[104] S. Ranshous, S. Shen, D. Koutra, S. Harenberg, C. Faloutsos, and N. F. Samatova. Anomaly detection in dynamic networks: A survey. *WIREs Comput. Stat.*, 7(3):223–247, May 2015.

[105] W. Reik and J. Walter. Genomic imprinting: parental influence on the genome. *Nature Reviews Genetics*, 2(1):21, 2001.

[106] T. Ried and I. Rajapakse. The 4d nucleome. *Methods*, 123:1–2, 2017. The 4D Nucleome.

[107] M. Rogers, L. Li, and S. J. Russell. Multilinear dynamical systems for tensor time series. In *Advances in Neural Information Processing Systems 26*, pages 2634–2642. Curran Associates, Inc., 2013.

[108] S. Ronquist, G. Patterson, L. A. Muir, S. Lindsly, H. Chen, M. Brown, M. S. Wicha, A. Bloch, R. Brockett, and I. Rajapakse. Algorithm for cellular reprogramming. *Proceedings of the National Academy of Sciences*, 114(45):11832–11837, 2017.

[109] C. W. Rowley. Model reduction for fluids, using balanced proper orthogonal decomposition. *International Journal of Bifurcation and Chaos*, 15(03):997–1013, 2005.

[110] W. J. Rugh. *Linear System Theory (2Nd Ed.)*. Prentice-Hall, Inc., Upper Saddle River, NJ, USA, 1996.

[111] N. Samardzija. Controllability, pole placement, and stabilizability for homogenous polynomial systems. *IEEE Transactions on Automatic Control*, 29(11):1042–1045, 1984.

[112] Sandia National Laboratories. Matlab tensor toolbox, 2015. Version 2.6.

[113] N. D. Sidiropoulos and R. Bro. On the uniqueness of multilinear decomposition of n-way arrays. *Journal of Chemometrics*, 14(3):229–239, 2000.

[114] N. D. Sidiropoulos, R. Bro, and G. B. Giannakis. Parallel factor analysis in sensor array processing. *IEEE Transactions on Signal Processing*, 48(8):2377–2388, Aug 2000.

[115] G. Simonyi. Graph entropy: A survey. In *Combinatorial Optimization*, 1993.

[116] D. A. Spielman and N. Srivastava. Graph sparsification by effective resistances. *SIAM Journal on Computing*, 40(6):1913–1926, 2011.

[117] A. Stegeman and N. D. Sidiropoulos. On kruskal's uniqueness condition for the candecomp/parafac decomposition. *Linear Algebra and its Applications*, 420(2):540–552, 2007.

[118] J. Stehla, N. Voirin, A. Barrat, C. Cattuto, L. Isella, J.-F. Pinton, M. Quaggiotto, W. Van den Broeck, C. Ragis, B. Lina, and P. Vanhems. High-resolution measurements of face-to-face contact patterns in a primary school. *PLOS ONE*, 6(8):1–13, 08 2011.

[119] L. Sun, B. Zheng, C. Bu, and Y. Wei. Moore-penrose inverse of tensors via einstein product. *Linear and Multilinear Algebra*, 64(4):686–698, 2016.

[120] A. Surana, G. Patterson, and I. Rajapakse. Dynamic tensor time series modeling and analysis. In *2016 IEEE 55th Conference on Decision and Control (CDC)*, pages 1637–1642, Dec 2016.

[121] P. Sweeney, C. Chen, R. D. Cone, and I. Rajapakse. Network dynamics of hypothalamic feeding neurons. *Proceedings of the National Academy Sciences*, 2021.

[122] H. G. Tanner. On the controllability of nearest neighbor interconnections. In *2004 43rd IEEE Conference on Decision and Control (CDC)(IEEE Cat. No. 04CH37601)*, volume 3, pages 2467–2472. IEEE, 2004.

[123] P. Valdivia, P. Buono, C. Plaisant, N. Dufournaud, and J. D. Fekete. Analyzing dynamic hypergraphs with parallel aggregated ordered hypergraph visualization. *IEEE Transactions on Visualization and Computer Graphics*, 27(1):1–13, 2021.

[124] C. F. Van Loan. *Structured Matrix Problems from Tensors*, volume 2173, pages 1–63. Springer, Cham, 01 2016.

[125] H. Wang, G. Xiao, Y. Yan, and D. Suter. Searching for representative modes on hypergraphs for robust geometric model fitting. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 41(3):697–711, March 2019.

[126] L. Wang, G. Zhao, Z. Kong, and Y. Zhao. Controllability and optimization of complex networks based on bridges. *Complexity*, 2020.

[127] L.-Z. Wang, Y.-Z. Chen, W.-X. Wang, and Y.-C. Lai. Physical controllability of complex networks. *Scientific reports*, 7:40198, 2017.

[128] D. Watts and S. Strogatz. Collective dynamics of "small-world" networks. *Nature*, 393, 1998.

[129] M. M. Wolf, A. M. Klinvex, and D. M. Dunlavy. Advantages to modeling relational data using hypergraphs versus graphs. In *2016 IEEE High Performance Extreme Computing Conference (HPEC)*, pages 1–7, Sep. 2016.

[130] Z. Yuan, C. Zhao, and Z. Di. Exact controllability of complex networks. *Nature Communication*, 4, 2013.

[131] Z. Yuan, C. Zhao, W.-X. Wang, Z. Di, and Y.-C. Lai. Exact controllability of multiplex networks. *New Journal of Physics*, 16(10):103036, 2014.

[132] J. G. T. Zañudo, G. Yang, and R. Albert. Structure-based control of complex networks with nonlinear dynamics. *Proceedings of the National Academy of Sciences*, 114(28):7234–7239, 2017.