Tensor Entropy for Uniform Hypergraphs

Can Chen and Indika Rajapakse

***Abstract*—In this paper, we develop the notion of entropy for uniform hypergraphs via tensor theory. We employ the probability distribution of the generalized singular values, calculated from the higher-order singular value decomposition of the Laplacian tensors, to fit into the Shannon entropy formula. We show that this tensor entropy is an extension of von Neumann entropy for graphs. In addition, we establish results on the lower and upper bounds of the entropy and demonstrate that it is a measure of regularity for uniform hypergraphs in simulated and experimental data. We exploit the tensor train decomposition in computing the proposed tensor entropy efficiently. Finally, we introduce the notion of robustness for uniform hypergraphs.**

***Index Terms*—Uniform hypergraphs, entropy, tensor decompositions, pattern recognition, random hypergraphs.**

I. Introduction

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 [1]–[3]. 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 [4], [5]. Studying such dynamic graphs often requires identifying the changes in network properties, such as degree distribution, path lengths and clustering coefficients [6]–[8].

Numerous techniques have been developed for anomaly detection based on evaluating similarities between graphs [9], [10]. 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 [11]. Two popular measures, Hamming distance [12] and Jaccard distance [13], are often problem-specific and sensitive to small perturbations or scaling, thus providing limited understanding of variations between graphs [14]. 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 [15]–[17].

The von Neumann entropy of a graph, first introduced by Braunstein *et al.* [18], 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 [19]. Additionally, the measure can be viewed as the information theoretic Shannon entropy, i.e.,

$$\mathrm{S} = -\sum_{j} \eta_j \ln \eta_j, \tag{1}$$

where, η_j are the normalized eigenvalues of the Laplacian matrix of a graph such that $\sum_j \eta_j = 1$. Passerini and Severini [20] 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 (1) is upper bounded by $\ln(n-1)$ where n is the number of vertices of a graph.

However, most data representations are multidimensional, and using graph models to describe them may lose information [21]. A hypergraph is a generalization of a graph in which a hyperedge can join any number of vertices [22], see Fig. 1. Thus, hypergraphs can represent multidimensional relationship unambiguously [21]. Examples of hypergraphs include co-authorship networks, film actor/actress networks and protein-protein interaction networks [23]. The authors in [21] also mention that hypergraphs require less storage space than graphs which may accelerate computation. Moreover, a hypergraph can be represented by a tensor if its hyperedges contain the same number of vertices (referred to as a uniform hypergraph). Tensors are multidimensional arrays generalized from vectors and matrices, preserving multidimensional patterns and capturing higher-order interactions and couplings within multiway data [24]. Tensor decompositions such as CANDECOMP/PARAFAC decomposition (CPD), higher-order singular value decomposition (HOSVD) and tensor train decomposition (TTD) can facilitate efficient computations and contain contextual interpretations regarding the target data tensors [25]–[28].

Hypergraph entropy has been recently explored by Hu *et al.* [29], Bloch *et al.* [30] and Wang *et al.* [31]. In [29], Hu

Manuscript received February 28, 2020; revised May 13, 2020; accepted June 13, 2020. Date of publication June 16, 2020; date of current version December 30, 2020. This work was supported in part by AFOSR Award No: FA9550-18-1-0028, in part by the Smale Institute, in part by the DARPA/MTO Lifelong Learning Machines program, and in part by the DARPA/ITO Guaranteeing AI Robustness Against Deception program. Recommended for acceptance by Dr. Xinbing Wang. *(Corresponding author: Can Chen.)*

Can Chen is with the Department of Mathematics and the Department of Electrical Engineering and Computer Science, University of Michigan, Ann Arbor, MI 48109 USA (e-mail: canc@umich.edu).

Indika Rajapakse is with the Department of Computational Medicine and Bioinformatics, Medical School and the Department of Mathematics, University of Michigan, Ann Arbor, MI 48109 USA (e-mail: indikar@umich.edu).

This article has supplementary downloadable material available at http://ieeexplore.ieee.org, provided by the authors.

Digital Object Identifier 10.1109/TNSE.2020.3002963

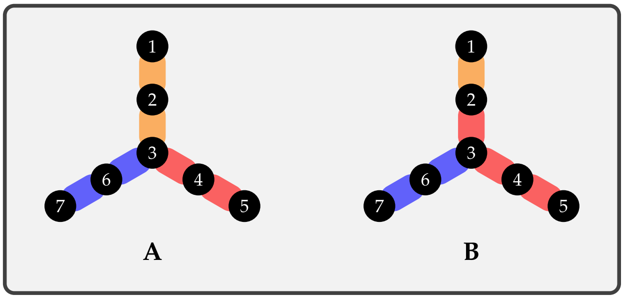

Fig. 1. Hypergraphs. (A) A 3-uniform hypergraph with hyperedges $e_1 = \{1,2,3\}$, $e_2 = \{3,4,5\}$, and $e_3 = \{3,6,7\}$. (B) A non-uniform hypergrah with hyperedges $e_1 = \{1,2\}$, $e_2 = \{2,3,4,5\}$, and $e_3 = \{3,6,7\}$.

et al. utilized the probability distribution of the vertex 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, [30] 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, [31] constructed a hypergraph entropy using vertex weighting scores, calculated from a density estimate technique, to select significant lines for model fitting. The score of a vertex (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 [19], [20], [32], we present a new spectral measure called tensor entropy, which can decipher topological attributes of uniform hypergraphs. For example, the hypergraph in Fig. 1 A with an additional hyperedge $e_4 = \{2,4,6\}$ has a higher tensor entropy than that with $e_4 = \{1,2,4\}$ because the latter destroys the hypergraph symmetrcity with longer average path length. The key contributions of this paper are as follows:

- We introduce a new notion of entropy for uniform hypergraphs based on the 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 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 vertex 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 computation and the notion of robustness for uniform hypergraphs.

The paper is organized into five sections. We first introduce the basics of tensor algebra including tensor products, tensor unfolding, higher-order singular value decomposition and tensor train decomposition in Section II-A. In Section II-B, we discuss the notion of uniform hypergraphs and extend graph-based definitions to describe uniform hypergraphs' structural properties. We then propose a new form of entropy for uniform hypergraphs with several theoretical results in Section II-C. In Section II-D, we exploit the tensor train decomposition to accelerate the tensor entropy computation. Six numerical examples are presented in Section III. Finally, we discuss directions for future research in Section IV and conclude in Section V. For ease of reading, we provide a notation table in Appendix A.

II. Method

A. Tensor Preliminaries

We take most of the concepts and notations for tensor algebra from the comprehensive works of Kolda *et al.* [33], [34]. A *tensor* is a multidimensional array. The *order* of a tensor is the number of its dimensions, also known as *modes*. A k-th order tensor usually is denoted by $\mathsf{X} \in \mathbb{R}^{n_1 \times n_2 \times \cdots \times n_k}$. 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 $\mathbf{A} \in \mathbb{R}^{m \times n}$ as second-order tensors. For a third-order tensor, *fibers* are named as *column* ($\mathsf{X}_{:j_2 j_3}$), *row* ($\mathsf{X}_{j_1 : j_3}$) and *tube* ($\mathsf{X}_{j_1 j_2 :}$), while *slices* are named as *horizontal* ($\mathsf{X}_{j_1 ::}$), *lateral* ($\mathsf{X}_{: j_2 :}$) and *frontal* ($\mathsf{X}_{:: j_3}$), see Fig. 2. 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 $\mathsf{X}_{j_1 j_2 \cdots j_k}$ is invariant under any permutation of the indices, and is called *diagonal* if $\mathsf{X}_{j_1 j_2 \cdots j_k} = 0$ except $j_1 = j_2 = \cdots = j_k$.

There are several notions of tensor products. 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

$$\langle \mathsf{X}, \mathsf{Y} \rangle = \sum_{j_1=1}^{n_1} \cdots \sum_{j_k=1}^{n_k} \mathsf{X}_{j_1 j_2 \dots j_k} \mathsf{Y}_{j_1 j_2 \dots j_k}, \tag{2}$$

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 *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

$$(\mathsf{X} \times_p \mathbf{A})_{j_1 j_2 \dots j_{p-1} i j_{p+1} \dots j_k} = \sum_{j_p=1}^{n_p} \mathsf{X}_{j_1 j_2 \dots j_p \dots j_k} \mathbf{A}_{i j_p}. \tag{3}$$

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

$$\begin{aligned} &\mathsf{X} \times_1 \mathbf{A}_1 \times_2 \mathbf{A}_2 \times_3 \cdots \times_k \mathbf{A}_k \\ &= \mathsf{X} \times \{\mathbf{A}_1, \mathbf{A}_2, \dots, \mathbf{A}_k\} \in \mathbb{R}^{m_1 \times m_2 \times \cdots \times m_k}. \end{aligned} \tag{4}$$

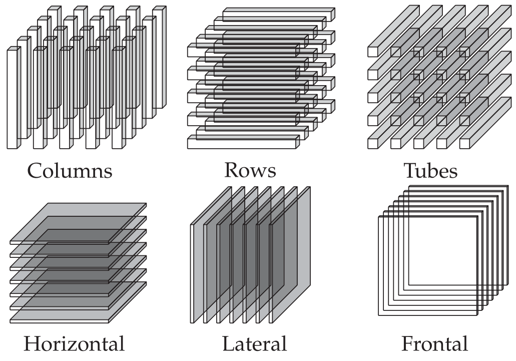

Fig. 2. Fibers and slices of a third-order tensor. The figure is adapted from [33].

Tensor unfolding is considered as a critical operation in tensor computations [35], [36]. The *p-mode unfolding* of a tensor $\mathsf{X} \in \mathbb{R}^{n_1\times n_2\times\cdots\times n_k}$, denoted by $\mathbf{X}_{(p)}$, is defined by

$$\mathsf{X}_{j_1j_2\cdots j_k} = (\mathbf{X}_{(p)})_{j_pj} \text{ for } j = 1 + \sum_{\substack{m=1\\m\neq p}}^{k}(j_m-1)\prod_{\substack{i=1\\i\neq p}}^{m-1} n_i. \tag{5}$$

The ranks of the p-mode unfoldings are called *multilinear ranks* of X, which are related to the so-called Higher-Order Singular Value Decomposition (HOSVD), a multilinear generalization of the matrix Singular Value Decomposition (SVD) [32], [37].

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

$$\mathsf{X} = \mathsf{S} \times_1 \mathbf{U}_1 \times_2 \cdots \times_k \mathbf{U}_k, \tag{6}$$

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

1) *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 $\alpha \neq \beta$;
2) *ordering*: $\|\mathsf{S}_{j_p=1}\| \geq \ldots \geq \|\mathsf{S}_{j_p=n_p}\| \geq 0$ 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.* [32] showed that the p-mode singular values from the HOSVD of X are the singular values of the p-mode unfoldings $\mathbf{X}_{(p)}$. In Section II-C, we will use the notion of p-mode singular values as the main tool to define the tensor entropy for uniform hypergraphs.

The Tensor Train Decomposition (TTD) of an k-th order tensor $\mathsf{X} \in \mathbb{R}^{n_1\times n_2\times\cdots\times n_k}$ is given by

$$\mathsf{X} = \sum_{r_k=1}^{R_k}\cdots\sum_{r_0=1}^{R_0} \mathsf{X}^{(1)}_{r_0:r_1} \circ \mathsf{X}^{(2)}_{r_1:r_2} \circ \cdots \circ \mathsf{X}^{(k)}_{r_{k-1}:r_k}, \tag{7}$$

where, $\circ$ is the *vector outer product*, $\{R_0, R_1, \ldots, R_k\}$ 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. There exist optimal TT-ranks for the TTD such that

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

for $p = 1, 2, \ldots, k-1$ [38]. 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

$$\begin{aligned}\bar{\mathbf{X}}^{(p)} &= \mathtt{reshape}(\mathsf{X}^{(p)}, R_{p-1}n_p, R_p),\\ \underline{\mathbf{X}}^{(p)} &= \mathtt{reshape}(\mathsf{X}^{(p)}, R_{p-1}, n_pR_p),\end{aligned}$$

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. 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 [38].

B. Uniform Hypergraphs

We first present some fundamental concepts of hypergraphs [39]–[41]. A *hypergraph* $\mathsf{G} = \{\mathbf{V}, \mathbf{E}\}$ where $\mathbf{V} = \{1, 2, \ldots, n\}$ is the vertex set and $\mathbf{E} = \{e_1, e_2, \ldots, e_m\}$ is the *hyperedge* set with $e_p \subseteq \mathbf{V}$ for $p = 1, 2, \ldots, m$. Two vertices are called *adjacent* if they are in the same hyperedge. A hypergraph is called *connected* if given two vertices, 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 Fig. 1 A. Significantly, every k-uniform hypergraph can be represented by a tensor.

Definition 1: Let $\mathsf{G} = \{\mathbf{V}, \mathbf{E}\}$ be a k-uniform hypergraph with n vertices. The *adjacency tensor* $\mathsf{A} \in \mathbb{R}^{n\times n\times\cdots\times n}$, which is a k-th order n-dimensional supersymmetric tensor, is defined as

$$\mathsf{A}_{j_1j_2\cdots j_k} = \begin{cases}\frac{1}{(k-1)!} \text{ if } (j_1, j_2, \ldots, j_k) \in \mathbf{E}\\ 0, \text{ otherwise}\end{cases} \tag{8}$$

The *degree tensor* D of a hypergraph G, associated with A, is a k-th order n-dimensional diagonal tensor with $\mathsf{D}_{jj\ldots j}$ equal to the number of hyperedges that consist of v_j for $j = 1, 2, \ldots, n$. If $\mathsf{D}_{jj\ldots j} = d$ for all j, then G is called *d-regular*. Given any k vertices, if they are contained in one hyperedge, then G is called *complete*.

Definition 2: Let $\mathsf{G} = \{\mathbf{V}, \mathbf{E}\}$ be a k-uniform hypergraph with n vertices. The *Laplacian tensor* $\mathsf{L} \in \mathbb{R}^{n\times n\times\cdots\times n}$ of G, which is a k-th order n-dimensional supersymmetric tensor, is defined as

$$\mathsf{L} = \mathsf{D} - \mathsf{A}, \tag{9}$$

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* [39]. Moreover, Chen *et al.* [42] 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 B. In the following, we extend several graph-based definitions to describe the structural properties of uniform hypergraphs.

Definition 3: Given a hypergraph G, the *index of dispersion* of the vertex degree distribution of G is defined to be the ratio of its variance to its mean.

Definition 4: Given a k-uniform hypergraph G with n vertices, the *average path length* of G is defined by

$$L_{\text{avg}} = \frac{1}{n(n-1)} \sum_{j \neq i} d(v_j, v_i), \tag{10}$$

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

Definition 5: Given a k-uniform hypergraph G with n vertices, the *average clustering coefficient* of G is defined by

$$C_j = \frac{|\{e_{ilk} : v_i, v_l, v_k \in \mathbf{N}_j, e_{ilk} \in \mathbf{E}\}|}{\binom{|\mathbf{N}_j|}{k}},$$
$$\Rightarrow C_{\text{avg}} = \frac{1}{n} \sum_{j=1}^{n} C_j, \tag{11}$$

where, $\mathbf{N}_j$ is the set of vertices 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 6: Given a k-uniform hypergraph G, the *small world coefficient* of G is defined by

$$\sigma = \frac{C_{\text{avg}}/C_{\text{rand}}}{L_{\text{avg}}/L_{\text{rand}}}, \tag{12}$$

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 [43], i.e., randomly generating uniform hypergraphs that share the same numbers of vertices and hyperedges with G. All these definitions can be used for quantifying the performance of entropy measures for uniform hypergraphs.

C. Tensor Entropy

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

Definition 7: Let G be a k-uniform hypergraph with n vertices. The *tensor entropy* of G is defined by

$$\mathrm{S} = -\sum_{j=1}^{n} \hat{\gamma}_j \ln \hat{\gamma}_j, \tag{13}$$

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

Algorithm 1: Computing Tensor Entropy From SVD.

```
Given a $k$-uniform hypergraph G with $n$ vertices
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
Find the $k$-mode unfolding of L, i.e., $\mathbf{L}_{(k)} = \texttt{reshape}(\mathsf{L}, n, n^{k-1})$
Compute the economy-size matrix SVD of $\mathbf{L}_{(k)}$, i.e., $\mathbf{L}_{(k)} = \mathbf{U}\mathbf{S}\mathbf{V}^\top$ and let $\{\gamma_j\}_{j=1}^{n} = \texttt{diag}(\mathbf{S})$
Set $\hat{\gamma}_j = \frac{\gamma_j}{\sum_{i=1}^{n} \gamma_i}$ and compute $\mathrm{S} = -\sum_{j=1}^{n} \hat{\gamma}_j \ln \hat{\gamma}_j$
return The tensor entropy S of G.
```

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 1. Since L is supersymmetric, any mode unfolding of L would yield the same unfolding matrix with the same singular values. Moreover, the tensor entropy (13) 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 [19], [20], [44]. 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.

Lemma 1: 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 vertices.

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 1.

Proposition 1: Suppose that G is a k-uniform hypergraph with n vertices and nonempty hyperedge set $\mathbf{E}$ for $k \geq 3$. Then the minimum tensor entropy of G is given by

$$\mathrm{S}_{\min} = \ln k. \tag{14}$$

Proof: Since G is a k-uniform hypergraph on n vertices, the maximum multiplicity of the zero normalized k-mode singular value of L is $n - k$ according to Lemma 1. 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 vertices 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 2: Suppose that G is a k-uniform hypergraph with n vertices for $k \geq 3$. Then the maximum tensor entropy of G occurs when it is a 1-regular uniform hypergraph, and is given by

$$\mathrm{S}_{\max} = \ln n. \tag{15}$$

Proof: Since G is a k-uniform hypergraph on n vertices, 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 j-th 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 $j = 1, 2, \dots n$. Moreover, since all the hyperedges contain distinct vertices, 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 3: Suppose that G is a k-uniform hypergraph with n vertices 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 vertices for $1 \leq d \leq \log_k n$. Similar to Proposition 2, 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 2 and 3, not all uniform hypergraphs with n vertices 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 vertices 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 4: Suppose that G is a complete k-uniform hypergraph with n vertices for $k \geq 3$. Then the tensor entropy of G is given by

$$\begin{aligned}\mathrm{S}_c = &\frac{(1-n)\alpha}{(n-1)\alpha+\beta}\ln\frac{\alpha}{(n-1)\alpha+\beta}\\ &-\frac{\beta}{(n-1)\alpha+\beta}\ln\frac{\beta}{(n-1)\alpha+\beta}\end{aligned} \tag{16}$$

where,

$$\alpha = \left(\frac{\Gamma(n)(\Gamma(n-k+1)+\Gamma(n))}{\Gamma(k)^2\Gamma(n-k+1)^2} - \frac{\Gamma(n-1)}{\Gamma(k)^2\Gamma(n-k)}\right)^{\frac{1}{2}}, \tag{17}$$

$$\beta = \left(\frac{\Gamma(n)(\Gamma(n-k+1)+\Gamma(n))}{\Gamma(k)^2\Gamma(n-k+1)^2} + \frac{(n-1)\Gamma(n-1)}{\Gamma(k)^2\Gamma(n-k)}\right)^{\frac{1}{2}}, \tag{18}$$

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

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

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 3 and 4, for arbitrary k-uniform hypergraph with n vertices 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 1: Suppose that G is a complete k-uniform hypergraph with n vertices 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 III, we will show evidence that the tensor entropy (13) is a measure of regularity for general uniform hypergraphs. Large tensor entropy is characterized by the large number of connected vertices, high uniformity of vertex 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 vertices form highly connected clusters. Tensor entropy is also related to the clustering coefficients of uniform hypergraphs in a very nuanced way.

D. 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.* [45] 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 2. 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 [45]. The computation and memory complexities of Algorithm 2 are estimated as $\mathscr{O}(knr^3)$ and $\mathscr{O}(knr^2)$, respectively, where r can be viewed as the "average" rank of the TT-ranks. Both complexities are much lower than those from Algorithm 1 when r is small.

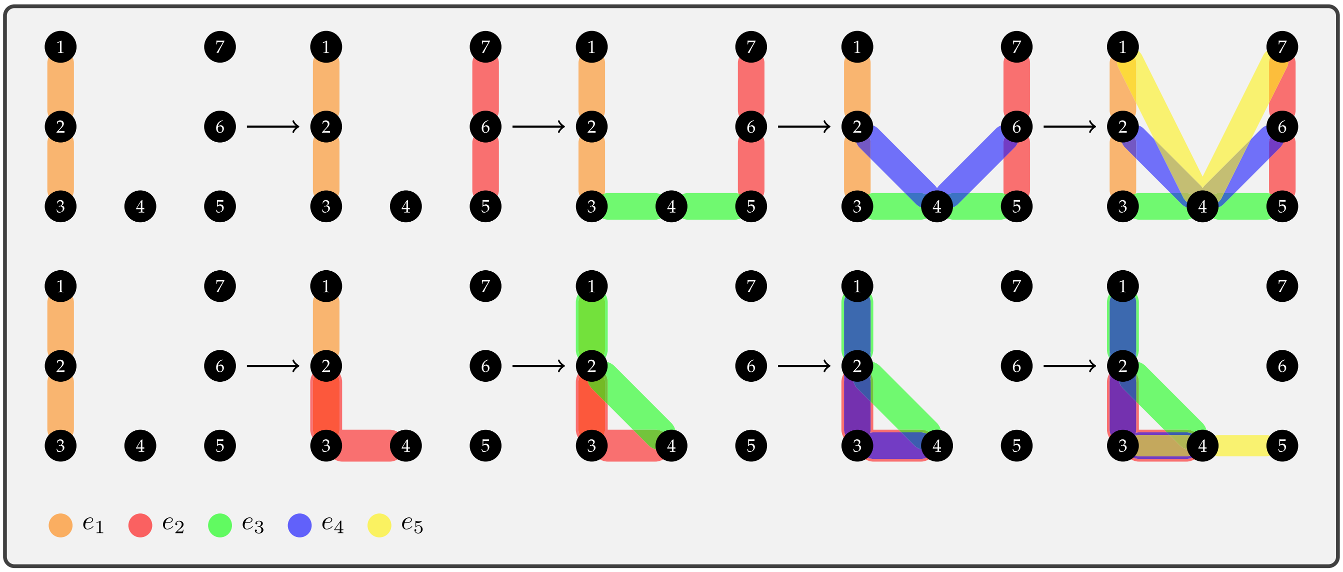

Fig. 3. Tensor entropy maximization/minimization. The top row describes the first five stages of the tensor entropy maximization evolution with a growing number of hyperedges in the order of $e_1 = \{1,2,3\}$, $e_2 = \{5,6,7\}$, $e_3 = \{3,4,5\}$, $e_4 = \{2,4,6\}$, and $e_5 = \{1,4,7\}$. The bottom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

Algorithm 2: Computing Tensor Entropy From TTD.

```
Given a $k$-uniform hypergraph $\mathsf{G}$ with $n$ vertices
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 $\mathsf{G}$
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
Left-orthonormalize the first $k-2$ cores and right-orthonormalize the last core of $\mathsf{L}$
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 $\{\gamma_j\}_{j=1}^{s} = \texttt{diag}(\mathbf{S})$
Set $\hat{\gamma}_j = \frac{\gamma_j}{\sum_{i=1}^{s}\gamma_i}$ and compute $\mathsf{S} = -\sum_{j=1}^{s}\hat{\gamma}_j \ln \hat{\gamma}_j$
**return** The tensor entropy $\mathsf{S}$ of $\mathsf{G}$.
```

III. Experiments

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 (Section III-F) also used the MATLAB TT-Toolbox by Oseledets *et al.* [46].

A. Hyperedge Growth Model

We consider the case where the number of vertices is fixed and new hyperedges are iteratively added to the uniform hypergraph. Fig. 3 presents the hyperedge growth evolution of a 3-uniform hypergraph with 7 vertices, 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 Fig. 4. If the two vertices are disconnected, we set the distance between them to be 4 for the purpose of visualization in Fig. 4 B.

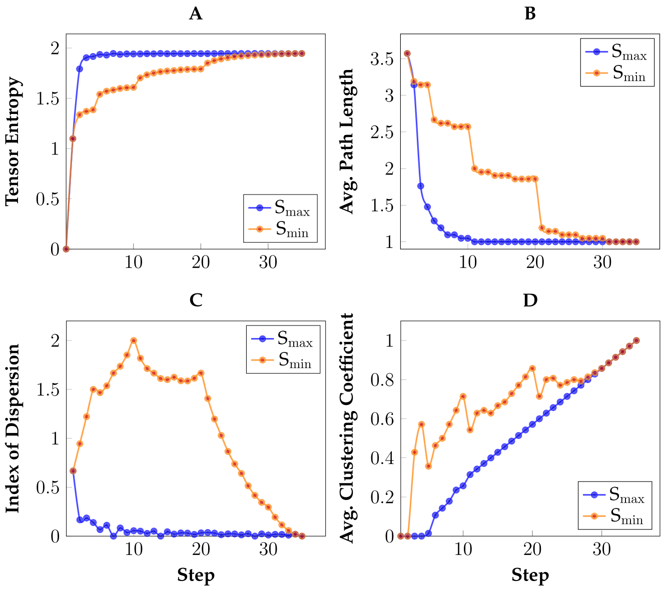

Fig. 4. Hyperedge growth model features. (A), (B), (C), and (D) Trajectories of tensor entropy, average path length, index of dispersion and average clustering coefficient with respect to the hyperedge adding steps.

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 vertices and then prefer to choose lower degree vertices with larger average geodesic distances, i.e., finding the geodesic distances between each pair in the triples and taking the mean, see Figs. 3 and 4 B. The average geodesic distances may lose importance if one wants to predict the next step as the

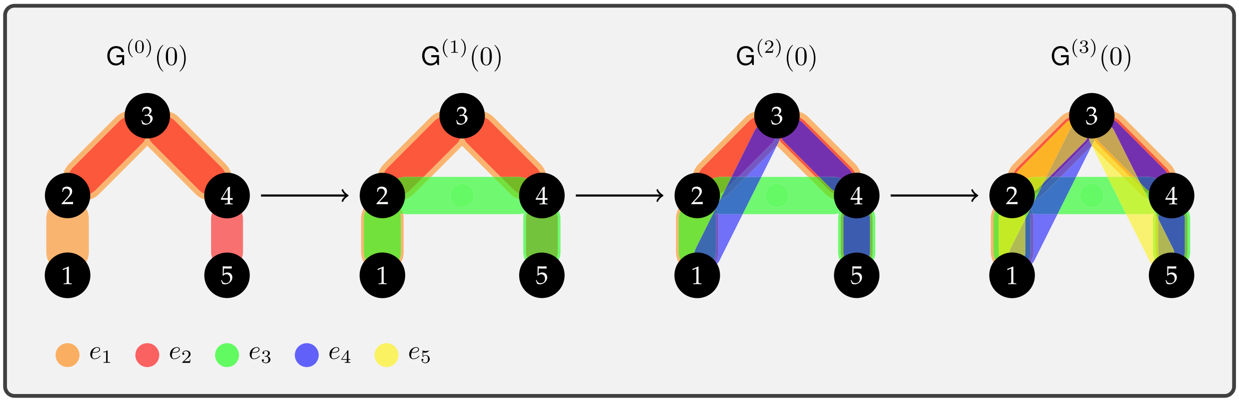

Fig. 5. Initial hypergraphs' structures for different q. The plot describes the *cliques*' formation in the first five vertices of the uniform hypergraph with the rewiring probability zero, in which $e_1 = \{1, 2, 3, 4\}$, $e_2 = \{2, 3, 4, 5\}$, $e_3 = \{1, 2, 4, 5\}$, $e_4 = \{1, 3, 4, 5\}$, and $e_5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

hypergraph becomes complex. Moreover, the vertices of $\mathsf{G}^{(j)}_{\max}$ tend to have "almost equal" or equal degree which leads to a low index of dispersion, see Fig. 4 C. In particular, $\mathsf{G}^{(j)}_{\max}$ are the $\frac{k}{n}j$-regular hypergraphs for the early stages of the evolution, i.e., $j = 7, 14$. However, as the hypergraph becomes dense, it is possible that $\mathsf{G}^{(j)}_{\max}$ 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 $\mathsf{G}^{(j)}_{\max}$ is lower than that for $\mathsf{G}^{(j)}_{\min}$, see Fig. 4 D. Furthermore, nontrivial symmetricity plays a role in maximizing the tensor entropy. For example, in $\mathsf{G}^{(3)}_{\max}$, the vertices $\{1, 2, 6\}$ 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 $\mathsf{G}^{(4)}_{\max}$. 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 Fig. 4 A. 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 decrease. In other words, for these highly regular or highly symmetric $\mathsf{G}^{(j)}_{\max}$, the corresponding tensor entropies $\mathsf{S}^{(j)}_{\max}$ achieve local maxima. On the other hand, $\mathsf{S}^{(j)}_{\min}$, the tensor entropies of $\mathsf{G}^{(j)}_{\min}$, are similar to the von Neumann graph entropy. Minimizing the tensor entropy would result in the formations of complete subhypergraphs (*cliques*), see Fig. 3. We can detect large jumps and drops in the next steps after completions of the subhypergraphs in Fig. 4 A and B, respectively. In order to make the discoveries more convincing, we repeated the same processes for k-uniform hypergraphs with different number of vertices and values of k, and observed similar results.

B. 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 vertices, and p be the rewiring probability of hyperedges, see Fig. 5. 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 $\mathsf{S}^{(3)}(0) = 4.5527$, 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,

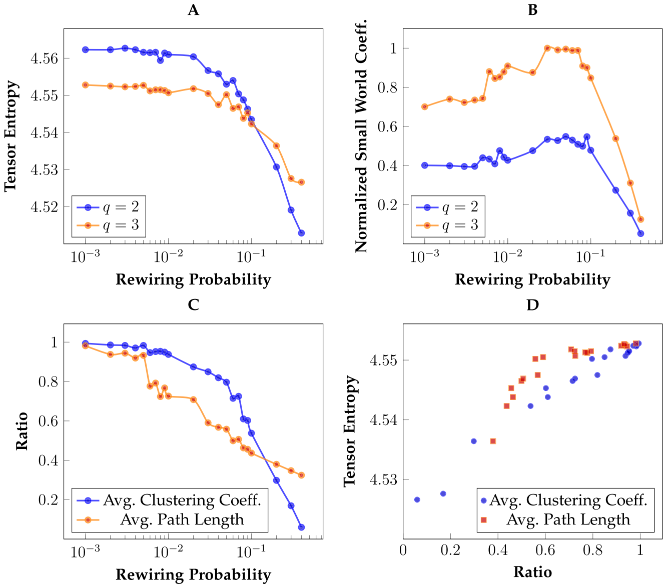

Fig. 6. The Watts-Strogatz model features. (A) Tensor entropies of random uniform hypergraphs with different rewiring probabilities for different q. (B) Normalized small world coefficients of random uniform hypergraphs with different rewiring probabilities for different q. (C) Ratios $C^{(3)}_{\text{avg}}(p)/C^{(3)}_{\text{avg}}(0)$ 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).

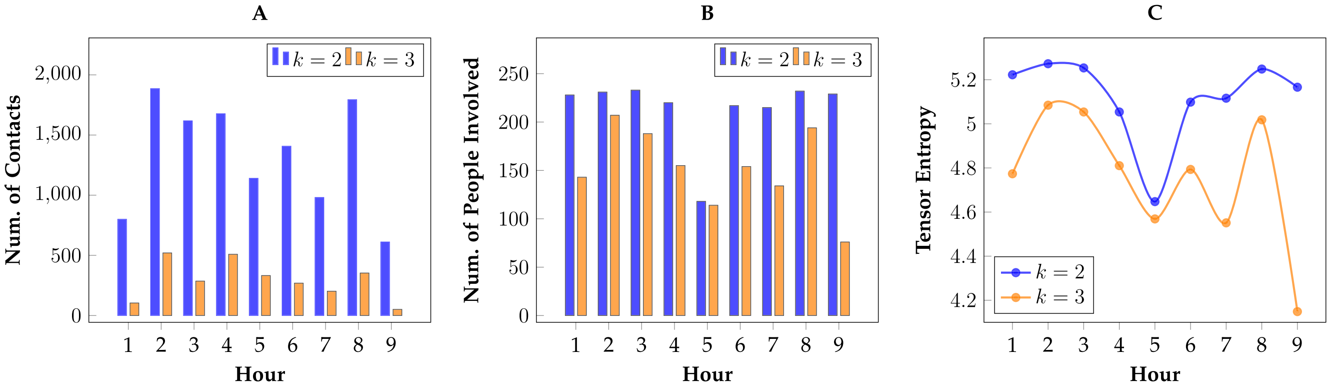

Fig. 7. Primary school contact features. (A) Number of the two-person and three-person contacts amongst the children and teachers every one hour of a day. (B) Number of children and teachers involved every one hour of a day in the two-person and three-person contacts, respectively. (C) Trajectories of the von Neumann entropy and the tensor entropy for the two-person and three-person contacts of a day.

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 Fig. 6. In general, with increasing rewiring probability p, the tensor entropy decays for both $q = 2$ and 3, see Fig. 6 A. 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 Fig. 6 D. 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 Fig. 6 B are very similar to the one in the Watts-Strogatz graph, in which it 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 vertices. 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 Fig. 6 C.

C. 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 [47], [48]. 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 Fig. 7, 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 vertices, see Fig. 7 B 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 Fig. 7 A.

D. 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

$$r^2 = c_1^2 + c_2^2 + c_3^2 - 2c_1c_2c_3, \tag{19}$$

where, c_1, c_2 and c_3 are the correlations between the three variables [49]. 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 Fig. 8, in which (A), (B) and (C) are the first 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 Fig. 8 D, 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

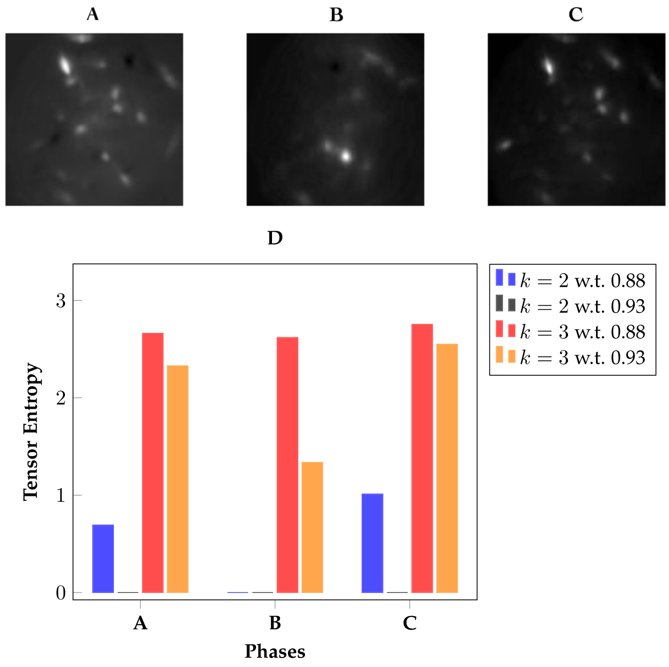

Fig. 8. Mouse neuron endomicroscopy features. (A), (B), and (C) First eigenfaces of the three phases - fed, fast and re-fed. (D) Tensor entropies of the k-uniform hypergraphs constructed from the corresponding three phases with $k = 2, 3$ (here w.t. stands for "with threshold").

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 vertices 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.

E. 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 [4], [5], [51]. 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 [50], 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 (19) 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 1 MB resolution with a total of 89 genomic loci. Our goal is to quantitatively detect a bifurcation in the fibroblast proliferation and reprogramming data, and accurately identify the critical transition point between cell identities during reprogramming. The results are shown in Fig. 9. 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 Fig. 9 A is consistent with the ground-truth statistic provided in [51]. 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 Fig. 9 B.

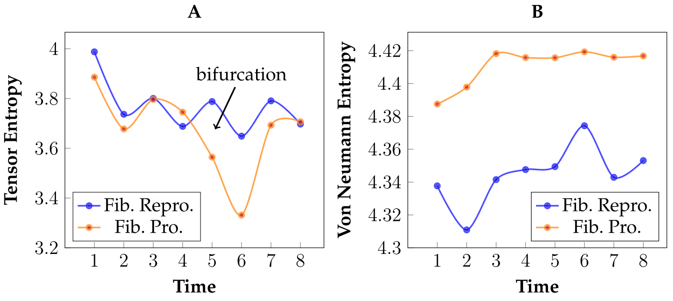

Fig. 9. Cellular reprogramming features. (A) Tensor entropies of the uniform hypergraphs recovered from Hi-C measurements with multi-correlation cutoff threshold 0.95. (B) Von Neuman entropies of the binarized Hi-C matrices with weight cutoff threshold 0.95.

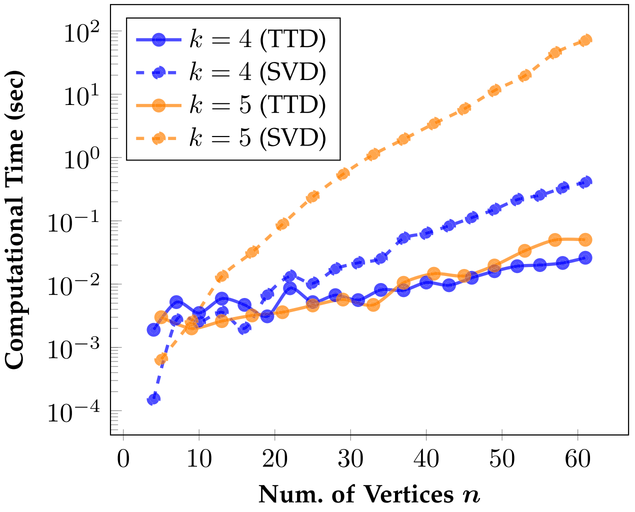

Fig. 10. Computational time comparisons between the SVD-based and TTD-based algorithms. For the TTD-based entropy computation, we reported the times of left- and right-orthonormalization and economy-size matrix SVD. For the SVD-based entropy computation, we only reported the time of economy-size matrix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

F. Algorithm Run Time Comparison

In this example, the k-uniform hypergraphs are constructed with n vertices by forming a strip structure in which every

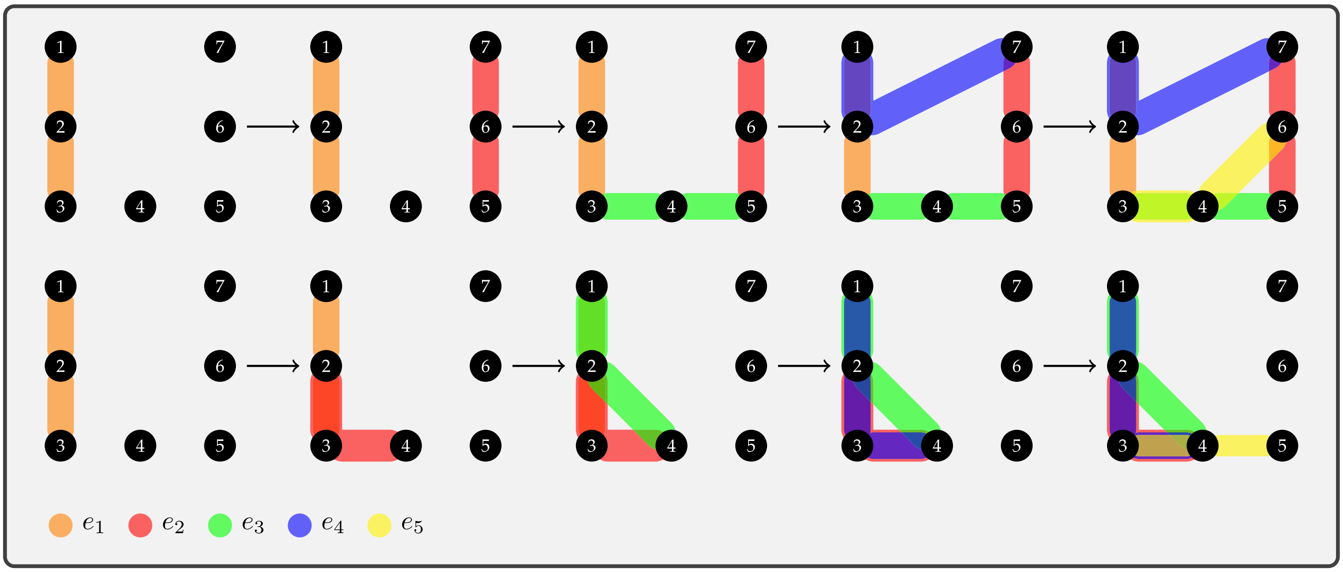

Fig. 11. Eigenvalue entropy maximization/minimization. The top row describes the first five stages of the eigenvalue entropy maximization evolution with a growing number of hyperedges in the order of $e_1 = \{1,2,3\}$, $e_2 = \{5,6,7\}$, $e_3 = \{3,4,5\}$, $e_4 = \{1,2,7\}$, and $e_5 = \{3,4,6\}$. The eigenvalue entropy $\mathsf{S}^{(j)}_{\max} =$ 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 $\mathsf{S}^{(j)}_{\min} =$ 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 [52], [53].

pair of connected hyperedges only contains one common vertex. We compare the computational efficiency of the SVD-based Algorithm 1 and the TTD-based Algorithm 2 in computing the tensor entropy. The results are shown in Fig. 10. 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 2 is more time efficient than Algorithm 1 for 4-uniform and 5-uniform hypergraphs with the strip structure as n becomes larger, see Fig. 10. 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 1 increases exponentially, while it grows at a much slower rate if using Algorithm 2. In the meantime, we compute the relative errors between the tensor entropies computed from the two algorithms, all of which are within 10^{-14}.

IV. 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 II-D, many simple structure tensors can be directly created in the TT-format without requiring construction of the full representations. For example, Oseledets *et al.* [46] 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 B for a short introduction to tensor eigenvalues for supersymmetric tensors. Based on the tensor eigenvalue formulations, we can establish the eigenvalue entropy measure for uniform hypergraphs.

Definition 8: Let G be a k-uniform hypergraph with n vertices. The *eigenvalue entropy* of G is defined by

$$\mathsf{S} = -\sum_{j=1}^{d} |\hat{\lambda}_j| \ln |\hat{\lambda}_j|, \tag{20}$$

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 Fig. 11. The entropy minimization evolution trajectory 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

TABLE I
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

of a tensor is an NP-hard problem [54]. 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 [55]. The authors in [56] 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 the k-mode singular values from the Laplacian tensors to describe the robustness of uniform hypergraphs.

Definition 9: Let G be a connected k-uniform hypergraph with n vertices. The *effective resistance* of G is defined by

$$\mathrm{R} = n \sum_{j=1}^{n} \frac{1}{\gamma_j}, \tag{21}$$

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 (21), 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 I. We can also expect that the smaller the effective resistance is, the more robust the uniform hypergraph. We believe that the effective resistance (21) is a good measure for uniform hypergraph robustness, but more theoretical and numerical support is needed to verify this hypothesis.

V. CONCLUSION

In this paper, 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 vertex 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 was proposed for computing the tensor entropy efficiently. We also applied this spectral measure to real biological networks for anomaly detection, and achieved better performances compared to the von Neumann graph entropy. As discussed in Section IV, the detailed relations between tensor eigenvalues and entropy, and the theoretical investigations of hypergraph robustness require further exploration. Controllability and influenceability of uniform hypergraphs are also important for future research.

ACKNOWLEDGMENT

We thank Dr. Frederick Leve at the Air Force Office of Scientific Research (AFOSR) for support and encouragement.

REFERENCES

[1] L. A. N. Amaral, A. Scala, M. Barthélémy, and H. E. Stanley, “Classes of small-world networks,” *Proc. Nat. Academy Sci.*, vol. 97, no. 21, pp. 11 149–11 152, 2000. [Online]. Available: https://www.pnas.org/content/97/21/11149

[2] A. Barabasi and E. Bonabeau, “Scale-free networks,” *Sci. Amer.*, vol. 288, pp. 60–69, 2003. [Online]. Available: http://dx.doi.org/10.1038/scientificamerican0503-60

[3] D. Watts and S. Strogatz, “Collective dynamics of “small-world” networks,” *Nature*, vol. 393, pp. 440–442, 1998.

[4] I. Rajapakse and M. Groudine, “On emerging nuclear order,” *J. Cell Biol.*, vol. 192, no. 5, pp. 711–721, 2011. [Online]. Available: http://jcb.rupress.org/content/192/5/711

[5] T. Ried and I. Rajapakse, “The 4D nucleome,” *Methods*, vol. 123, pp. 1–2, 2017, the 4D Nucleome. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S1046202317302220

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

[7] N. D. Monnig and F. G. Meyer, “The resistance perturbation distance: A metric for the analysis of dynamic networks,” *Discrete Appl. Math.*, vol. 236, pp. 347–386, 2018. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0166218X17304626

[8] S. Ranshous, S. Shen, D. Koutra, S. Harenberg, C. Faloutsos, and N. F. Samatova, “Anomaly detection in dynamic networks: A survey,” *WIREs Comput. Statist.*, vol. 7, no. 3, pp. 223–247, May 2015. [Online]. Available: https://doi.org/10.1002/wics.1347

[9] L. Akoglu, H. Tong, and D. Koutra, “Graph based anomaly detection and description: A survey,” *Data Mining Knowl. Discovery*, vol. 29, no. 3, pp. 626–688, 2015.

[10] P. Papadimitriou, A. Dasdan, and H. Garcia-Monlina, “Web graph similarity for anomaly detection,” *J. Internet Service Appl.*, vol. 1, pp. 19–30, 2010.

[11] S.-H. Cha, “Comprehensive survey on distance/similarity measures between probability density functions,” *Int. J. Math. Models Methods Appl. Sci.*, vol. 1, no. 4, pp. 300–307, 2007. [Online]. Available: http://www.gly.fsu.edu/ parker/geostats/Cha.pdf

[12] R. W. Hamming, “Error detecting and error correcting codes,” *Bell Syst. Tech. J.*, vol. 29, no. 2, pp. 147–160, Apr. 1950.

[13] M. Levandowsky and D. Winter, “Distance between sets,” *Nature*, vol. 234, pp. 34–35, 1971.

[14] C. Donnat and S. Holmes, “Tracking network dynamics: A survey using graph distances,” *Ann. Appl. Statist.*, vol. 12, pp. 971–1012, 2018.

[15] G. Simonyi, “Graph entropy: A survey,” in *Proc. Combinatorial Optim.*, vol. 20, pp. 399–441, 1995.

[16] C. Donnat and S. Holmes, “Discovering important nodes through graph entropy the case of enron email database,” in *Proc. 3rd Int. Workshop Link Discovery*, 2005, pp. 74–81.

[17] A. Li and Y. Pan, “Structural information and dynamical complexity of networks,” *IEEE Trans. Inf. Theory*, vol. 62, no. 6, pp. 3290–3339, Jun. 2016.

[18] S. Braunstein, S. Ghosh, T. Mansour, S. Severini, and R. C. Wilson, “Some families of density matrices for which separability is easily tested,” *Phys. Rev. A*, vol. 73, p. 012320, 2006.

[19] G. Minello, L. Rossi, and A. Torsello, “On the von Neumann entropy of graphs,” *J. Complex Netw.*, vol. 7, no. 4, pp. 491–514, 11 2018. [Online]. Available: https://doi.org/10.1093/comnet/cny028

[20] F. Passerini and S. Severini, “The von neumann entropy of networks,” 2008. [Online]. Available: https://arxiv.org/abs/0812.2597

[21] M. M. Wolf, A. M. Klinvex, and D. M. Dunlavy, “Advantages to modeling relational data using hypergraphs versus graphs,” in *Proc. IEEE High Perform. Extreme Comput. Conf.*, Sep. 2016, pp. 1–7.

[22] C. Berge, *Hypergraphs, Combinatorics of Finite Sets*. 3rd Ed. Amsterdam, The Netherlands: North-Holland, 1989.

[23] M. Newman, *Networks: An Introduction*. Oxford, United Kingdom, Oxford University Press, OUP Oxford, 2010. [Online]. Available: https://books.google.com/books?id=q7HVtpYVfC0C

[24] C. Chen, A. Surana, A. Bloch, and I. Rajapakse, “Data-driven model reduction for multilinear control systems via tensor trains,” *ArXiv*, 1912.03569.

[25] A. Cichocki, N. Lee, I. Oseledets, A. Phan, Q. Zhao, and D. P. Mandic, *Tensor Networks for Dimensionality Reduction and Large-scale Optimization: Part 2 Applications and Future Perspectives*. 2017. [Online]. Available: https://ieeexplore.ieee.org/document/8187112

[26] H. Lu, K. N. Plataniotis, and A. N. Venetsanopoulos, “MPCA: Multilinear principal component analysis of tensor objects,” *IEEE Trans. Neural Netw.*, vol. 19, no. 1, pp. 18–39, Jan. 2008.

[27] W. Wang, V. Aggarwal, and S. Aeron, “Principal component analysis with tensor train subspace,” *Pattern Recognit. Lett.*, vol. 122, pp. 86–91, 2019.

[28] A. H. Williams *et al.*, “Unsupervised discovery of demixed, low-dimensional neural dynamics across multiple timescales through tensor component analysis,” *Neuron*, vol. 98, no. 6, pp. 1099–1115.e8, 2018.

[29] 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*, vol. 35, no. 7, pp. 1238–1250, Jul. 2019. [Online]. Available: https://doi.org/10.1007/s10114-019-8093-2

[30] I. Bloch and A. Bretto, “A new entropy for hypergraphs,” in *Discrete Geometry for Computer Imagery*, M. Couprie, J. Cousty, Y. Kenmochi, and N. Mustafa, Eds., Berlin, Germany: Springer, 2019, pp. 143–154.

[31] H. Wang, G. Xiao, Y. Yan, and D. Suter, “Searching for representative modes on hypergraphs for robust geometric model fitting,” *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 41, no. 3, pp. 697–711, Mar. 2019.

[32] L. De Lathauwer, B. De Moor, and J. Vandewalle, “A multilinear singular value decomposition,” *SIAM J. Matrix Anal. Appl.*, vol. 21, no. 4, pp. 1253–1278, 2000.

[33] T. G. Kolda and B. Bader, “Tensor decompositions and applications,” *SIAM Rev.*, vol. 51, no. 3, pp. 455–500, 2009. [Online]. Available: https://doi.org/10.1137/07070111X

[34] T. G. Kolda, “Multilinear operators for higher-order decompositions,” Tech. Rep. No. SAND2006-2081, Sandia National Laboratories, 2006.

[35] S. Ragnarsson and C. Van Loan, “Block tensor unfoldings,” *SIAM J. Matrix Anal. Appl.*, vol. 33, pp. 149–169, 2012.

[36] C. Chen, A. Surana, A. Bloch, and I. Rajapakse, “Multilinear time invariant system theory,” in *Proc. Conf. Control Appl.*, 2019, pp. 118–125. [Online]. Available: https://epubs.siam.org/doi/abs/10.1137/1.9781611975758.18

[37] G. Bergqvist and E. G. Larsson, “The higher-order singular value decomposition: Theory and an application,” *IEEE Signal Process. Mag.*, vol. 27, no. 3, pp. 151–154, May 2010.

[38] I. Oseledets, “Tensor-train decomposition,” *SIAM J. Sci. Comput.*, vol. 33, no. 5, pp. 2295–2317, 2011. [Online]. Available: https://doi.org/10.1137/090752286

[39] L. Qi, “H$^+$-eigenvalues of laplacian and signless laplacian tensors,” *Commun. Math. Sci.*, vol. 12, no. 6, pp. 1045–1064, 2013.

[40] S. Hu, L. Qi, and J. Xie, “The largest laplacian and signless laplacian H-eigenvalues of a uniform hypergraph,” *Linear Algebra Appl.*, vol. 469, pp. 1–27, 2015. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S002437951400754X

[41] J. Xie and A. Chang, “On the Z-eigenvalues of the adjacency tensors for uniform hypergraphs,” *Linear Algebra Appl.*, vol. 439, no. 8, pp. 2195–2204, 2013. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0024379513004679

[42] Y. Chen, L. Qi, and X. Zhang, “The fiedler vector of a laplacian tensor for hypergraph partitioning,” *SIAM J. Scientific Comput.*, vol. 39, no. 6, pp. A2508–A2537, 2017. [Online]. Available: https://doi.org/10.1137/16M1094828

[43] M. D. Humphries and K. Gurney, “Network ‘small-world-ness’: A quantitative method for determining canonical network equivalence,” *Plos One*, vol. 3, no. 4, pp. 1–10, 2008. [Online]. Available: https://doi.org/10.1371/journal.pone.0002051

[44] 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,” *Ann. Combinatorics*, vol. 10, pp. 291–317, 2004.

[45] S. Klus, P. Gelß, S. Peitz, and C. Schutte, “Tensor-based dynamic mode decomposition,” *Nonlinearity*, vol. 31, no. 7, pp. 3359–3380, Jun. 2018. [Online]. Available: https://doi.org/10.1088%2F1361-6544%2Faabc8f

[46] I. Oseledets, S. Dolgov, V. Kazeev, O. Lebedeva, and T. Mach, “TT-toolbox,” 2014, version 2.2.2. [Online]. Available: https://github.com/oseledets/TT-Toolbox

[47] V. Gemmetto, A. Barrat, and C. Cattuto, “Mitigation of infectious disease at school: Targeted class closure vs school closure,” *BMC Infectious Diseases*, vol. 14, p. 695, 2014.

[48] J. Stehla *et al.*, “High-resolution measurements of face-to-face contact patterns in a primary school,” *Plos One*, vol. 6, no. 8, pp. 1–13, 2011. [Online]. Available: https://doi.org/10.1371/journal.pone.0023176

[49] N. Zheng and J. Wang, “Measures of correlation for multiple variables,” 2019. [Online]. Available: https://arxiv.org/abs/1401.4827

[50] 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 Res.*, vol. 46, no. 19, pp. 9895–9906, 2018. [Online]. Available: https://academic.oup.com/nar/article/46/19/9895/5099447

[51] S. Liu *et al.*, “Genome architecture mediates transcriptional control of human myogenic reprogramming,” *iScience*, vol. 6, pp. 232–246, 2018a. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S2589004218301147

[52] L. Chen, L. Han, T.-Y. Li, and L. Zhou, “Teneig,” 2015, version 2.0. [Online]. Available: https://users.math.msu.edu/users/chenlipi/TenEig.html

[53] L. Chen, L. Han, and L. Zhou, “Computing tensor eigenvalues via homotopy methods,” *SIAM J. Matrix Anal. Appl.*, vol. 37, no. 1, pp. 290–319, 2016. [Online]. Available: https://doi.org/10.1137/15M1010725

[54] C. J. Hillar and L.-H. Lim, “Most tensor problems are np-hard,” *J. ACM*, vol. 60, no. 6, pp. 45:1–45:39, Nov. 2013. [Online]. Available: http://doi.acm.org/10.1145/2512329

[55] W. Ellens, F. Spieksma, P. V. Mieghem, A. Jamakovic, and R. Kooij, “Effective graph resistance,” *Linear Algebra Appl.*, vol. 435, no. 10, pp. 2491–2506, 2011. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0024379511001443

[56] D. Klein and M. Randic, “Resistance distance,” *J. Math. Chemistry*, vol. 12, pp. 81–95, 1993.

Can Chen is currently a Ph.D. candidate in applied and interdisciplinary mathematics and a master’s student in electrical and computer engineering at the University of Michigan, Ann Arbor. He received B.S. degree in mathematics at the University of California, Irvine in 2016. His research is focused on data-guided control of multiway dynamical systems.

Indika Rajapakse is currently an Associate Professor of computational medicine and bioinformatics, in the Medical School, and an Associate Professor of mathematics at the University of Michigan, Ann Arbor. He is also a Member of the Smale Institute. His research is at the interface of biology, engineering, and mathematics. His areas include dynamical systems, networks, mathematics of data, and cellular reprogramming.