

## REVIEW

# A survey of some tensor analysis techniques for biological systems

Farzane Yahyanejad<sup>1</sup>, Réka Albert<sup>2</sup>, Bhaskar DasGupta<sup>1,\*</sup>

<sup>1</sup> Department of Computer Science University of Illinois at Chicago Chicago, IL 60607, USA

<sup>2</sup> Department of Physics Pennsylvania State University University Park, PA 16802, USA

\* Correspondence: bdasgup@uic.edu

Received June 9, 2019; Revised September 13, 2019; Accepted September 16, 2019

**Background:** Since biological systems are complex and often involve multiple types of genomic relationships, tensor analysis methods can be utilized to elucidate these hidden complex relationships. There is a pressing need for this, as the interpretation of the results of high-throughput experiments has advanced at a much slower pace than the accumulation of data.

**Results:** In this review we provide an overview of some tensor analysis methods for biological systems.

**Conclusions:** Tensors are natural and powerful generalizations of vectors and matrices to higher dimensions and play a fundamental role in physics, mathematics and many other areas. Tensor analysis methods can be used to provide the foundations of systematic approaches to distinguish significant higher order correlations among the elements of a complex systems via finding ensembles of a small number of reduced systems that provide a concise and representative summary of these correlations.

**Keywords:** biological systems; tensor analysis; biological and statistical validation

**Author summary:** In this review article we provide an overview of some tensor analysis methods for biological systems that could be useful in finding significant higher-order correlations among elements of a complex biological system.

## INTRODUCTION

The biological functioning and life of a cellular system is controlled by signaling and energy transfer interactions among its numerous constituents such as proteins, RNAs, DNAs, and other small molecules, and usually involve a cascade of biochemical reactions or other physical interactions among these constituents. An investigation of such interactions is usually done by selecting, implicitly or explicitly, one or more models to characterize the interactions (physical, chemical, or statistical dependencies) between components of the cellular environment. Naturally, the selection of the model depends on several factors such as the level of details desired, the characteristics of the particular interactions studied, and the overall goal of the investigation. Often, biologists describe the model by presenting the interaction data in the form of a diagram (e.g., some type of

graph), optionally along with some mathematical formulation of its dynamics. Most mathematical formulation of the dynamics typically assumes that each node in the diagram has an associated (discrete or continuous) “state” variable (representing, for example, concentration of the corresponding protein) that is a function of the time variable  $t$ , and describes how the value of this variable at a node (“state” of the node) depends on the state of the nodes interacting with it. Examples of some common models of the above type relevant for our article include protein-protein interaction networks represented by undirected graphs without any explicit state variables for nodes and signal transduction networks represented by edge-labeled directed graphs optionally with state variables for nodes. In contrast, the more general (ordinary or partial) differential equation models omit the network diagram altogether and describe the behavior of the state variables via (ordinary or partial) differential equations.

These representations may have inter-dependencies, *e.g.*, under certain technical assumptions one can construct a signaltransduction network diagram from the differential equation model (see ([1], Chap. 5) for details).

A major drawback of using graph-theoretic tools on a single network diagram lies in ignoring the time or ignoring the higher-order correlations of the interactions which may lead to inaccurate or incomplete analysis. For example, a network diagram only encodes pairwise correlations of node state variables, and thus cannot represent a joint  $k$ -way correlation among  $k$  state variables for any  $k > 2$ . If precise equations of time evolutions of state variables are given then we could of course completely ignore the network diagrams and work with the given equations, but then we lose the advantage of employing graph-theoretic tools and instead fall back on analysis techniques which are often hard to employ effectively because of difficulties of estimating precise equations and the non-trivial non-linear natures of these equations.

In this review we provide an overview of some tensor analysis methods for biological systems. For this type of analysis, one usually models a given biological system as a  $k$ -dimensional matrix  $\mathcal{X}^{\text{def}}[x_{i_1, \dots, i_k}]$  of size  $I_1 \times \dots \times I_k$  (formally an order  $k$  tensor  $\mathcal{X}^{\text{def}} \mathcal{X}^{I_1 \times \dots \times I_k}$ , see Section of “Standard concepts and definitions related to tensor analysis”) encoding higher-order correlation of a biological system with or without time evolution. Tensor analysis methods have already been successfully used in specific contexts of pathway reconstructions in cellular systems and microarray data integration from several sources [2–6]. Outside bioinformatics, tensor analysis methods have been very successfully applied to many other application areas such as neuroscience [7–10], psychology [7, 11–13] and chemometrics [14]. Even though at first glance one is tempted to think that involving more dimensions as compared to vectors or matrices would further complicate the computational aspects of the relevant problems, this is not necessarily the case. There are many advantages of using higher mode tensors as opposed to matrices and vectors for analysis; see Section of “CANDECOMP/PARAFAC(CP) decomposition” for one such example.

## STANDARD CONCEPTS AND DEFINITIONS RELATED TO TENSOR ANALYSIS

Tensors are natural and powerful generalizations of vectors and matrices to higher dimensions and play a fundamental role in physics, mathematics and other areas. In this section we briefly review some standard concepts

and definitions associated with tensor analysis; see excellent survey articles or books such as [9, 12, 15] for further information.

### Basic definitions and notations

A tensor  $\mathcal{X}^{I_1 \times \dots \times I_k}$  (or, simply  $\mathcal{X}$ ) of *mode* (also called order)  $k$  is a  $k$ -dimensional array  $\mathcal{X}$  of size  $I_1 \times \dots \times I_k$ . Thus, a tensor of order 1 is a vector and a tensor of order 2 is a matrix; for  $k > 2$  a tensor of order  $k$  is also known as a “higher-order” tensor. Following widely used conventions, we will denote tensors, matrices and column vectors by calligraphic uppercase (*e.g.*,  $\mathcal{X}$ ), boldface uppercase (*e.g.*,  $\mathbf{X}$ ) and boldface lowercase (*e.g.*,  $\mathbf{x}$ ) letters, respectively. Individual elements will be denoted by the corresponding (non-bold) lowercase letter with appropriate indices, *e.g.*,  $x_{i,j}$  for matrix  $\mathbf{X}$ . A sequence of vectors/matrices/tensors will be indicated by using parenthesized numbers as superscripts, *e.g.*,  $\mathcal{X}^{(1)}$ ,  $\mathcal{X}^{(2)}$ , ...,  $\mathcal{X}^{(m)}$  will denote a sequence of  $m$  tensors. Table 1 succinctly summarizes some tensor operations and related notations. Many softwares such as MATLAB, Mathematica and Maple as well as packages in languages such as FORTRAN and C++ provide support for basic and advanced tensor operations. The following definitions are standard:

**Simple tensor:** A tensor  $\mathcal{T}$  of order  $k$  is a *simple* tensor if and only if there are vectors  $\mathbf{v}^{(1)}, \dots, \mathbf{v}^{(k)}$  such that  $\mathcal{T} = \mathbf{v}^{(1)} \circ \dots \circ \mathbf{v}^{(k)}$ . If the tensor  $\mathcal{T}$  represents statistical correlations among  $k$  variables then  $\mathcal{T}$  is simple exactly when the variables are mutually independent.

**Symmetric tensor:** A tensor  $\mathcal{T}^{n \times \dots \times n}$  of order  $k$  is symmetric if  $\mathcal{T}_{i_1, \dots, i_k} = \mathcal{T}_{\sigma_1, \dots, \sigma_k}$  for every permutation  $\sigma^{\text{def}} = \{\sigma_{i_1}, \dots, \sigma_{i_k}\}$  of  $\{i_1, \dots, i_k\}$ .

In the sequel, we will use  $\|\cdot\|$  to indicate a suitable tensor norm (*e.g.*, Frobenius norm, spectral norm or some other appropriate tensor norm).

### Basic tensor decomposition methods and corresponding ranks

Philosophically, this step is similar to a type of principal component analysis for matrix data. In other words, we “factor” the input tensor into a combination of simpler tensors (*e.g.*, rank one tensors, tensor of small column rank etc.). For concreteness, we mention the following two factoring methods, but other factoring methods are also often considered based on specific applications and data types, such as the multi-linear singular value decomposition (SVD) factorizations [16], higher-order eigenvalue decompositions [2, 5], and Boolean tensor factorizations [17].

**Table 1** Some basic tensor operations and related notations

Nomenclature	Notation	Mathematical details
Sum of tensors	$\mathcal{Z} = \mathcal{X} + \mathcal{Y}$	$\forall i_1, \dots, i_k : z_{i_1, \dots, i_k} = x_{i_1, \dots, i_k} + y_{i_1, \dots, i_k}$
Tensor-scalar product	$\mathcal{Y} = \alpha \mathcal{X}$	$\forall i_1, \dots, i_k : y_{i_1, \dots, i_k} = \alpha x_{i_1, \dots, i_k}$
Tensor inner product	$\langle \mathcal{X}, \mathcal{Y} \rangle$	$\langle \mathcal{X}, \mathcal{Y} \rangle = \sum_{i_1=1}^{I_1} \dots \sum_{i_k=1}^{I_k} x_{i_1, \dots, i_k} y_{i_1, \dots, i_k}$
Frobenius norm of a tensor	$\ \mathcal{X}\ _F$	$\ \mathcal{X}\ _F = \sqrt{\langle \mathcal{X}, \mathcal{X} \rangle}$
Spectral norm of a tensor	$\ \mathcal{X}^{I_1 \times \dots \times I_k}\ _s$	$\sup_{\mathbf{y}^{(1)} \in \mathbb{R}^{I_1}, \dots, \mathbf{y}^{(k)} \in \mathbb{R}^{I_k}} \sum_{i_1=1}^{I_1} \dots \sum_{i_k=1}^{I_k} x_{i_1, \dots, i_k} y_{i_1}^{(1)} \dots y_{i_k}^{(k)}$ $\ \mathbf{y}^{(1)}\  = \dots = \ \mathbf{y}^{(k)}\  = 1$
Vector outer product	$\mathcal{Y} = \mathbf{x}^{(1)} \circ \dots \circ \mathbf{x}^{(n)}$	$\forall i_1, \dots, i_n : y_{i_1, \dots, i_n} = x_{i_1}^{(1)} \dots x_{i_n}^{(n)}$
Mode- $r$ matricization of $\mathcal{X}^{I_1 \times \dots \times I_k}$	$\mathbf{X}_{(r)}^{I_r \times (\prod_{j \neq r} I_j)}$	$\mathbf{X}_{(r)}^{I_r, 1 + \sum_{\ell=1, \ell \neq r}^k (I_\ell - 1) (\prod_{p=1, p \neq r}^k I_p)} = \mathbf{X}_{i_1, \dots, i_r, \dots, i_k}$
Mode- $j$ product of $\mathcal{X}^{I_1 \times \dots \times I_k}$ and $\mathbf{Y}^{R \times I_j}$	$\mathcal{Z}^{I_1 \times \dots \times I_{j-1} \times R \times I_{j+1} \times \dots \times I_k} = \mathcal{X} \times_j \mathbf{Y}$	$\mathbf{Z}_{(j)} = \mathbf{Y} \mathbf{X}_{(j)}$
$\kappa$ -rank of matrix $\mathbf{X}$	$\kappa \mathbf{X}$	Largest $j$ such that every set of $j$ columns of $\mathbf{X}$ are linearly independent

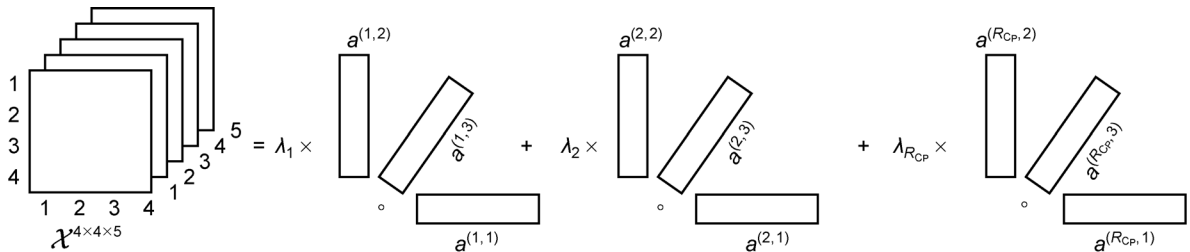
### CANDECOMP/PARAFAC (CP) decomposition [7,11,18]

Intuitively, this is a generalization to higher-order tensors of the standard SVD (singular value decomposition) of an  $m \times n$  matrix  $\mathbf{A}$  [19]:

$$\mathbf{A} = \mathbf{U} \Sigma \mathbf{V}^T = \sum_{j=1}^{\min\{m,n\}} \sigma_j \mathbf{u}_j \mathbf{v}_j^T$$

where  $\sigma_j$ 's are the diagonal entries of the diagonal matrix  $\Sigma$  and vectors  $\mathbf{u}_j$  and  $\mathbf{v}_j$  are the  $j^{\text{th}}$  columns of the matrices  $\mathbf{U}$  and  $\mathbf{V}$ , respectively. Likewise, in CP decomposition, one expresses the input tensor  $\mathcal{X}$  as a sum (with minimum number of terms) of outer products of 1-rank tensors, i.e., one finds  $\lambda_r, \mathbf{a}^{(r,1)}, \dots, \mathbf{a}^{(r,k)}$ , for  $k = 1, 2, \dots, R_{\text{CP}}$ , that solves the problem:

$$\begin{aligned} & \text{minimize } R_{\text{CP}} \text{ subject to } \mathcal{X}^{I_1 \times \dots \times I_k} \\ & = \sum_{r=1}^{R_{\text{CP}}} \lambda_r \mathbf{a}^{(r,1)} \circ \dots \circ \mathbf{a}^{(r,k)} \end{aligned} \quad (1)$$


**Figure 1.** A pictorial representation of the CP decomposition in Equation (1) for  $k = 3$ .

where  $\mathbf{a}^{(r,1)}$  is a vector of dimension  $I_j$  and  $\lambda_r$ 's are scalars to bound the norms of  $\mathbf{a}^{(r,1)}$ 's (see Figure 1 for a pictorial illustration). The matrix  $\hat{\mathbf{A}}^{(j)} = [\mathbf{a}^{(1,j)} \dots \mathbf{a}^{(R_{\text{CP}},j)}]$  formed by taking  $\mathbf{a}^{(1,j)} \dots \mathbf{a}^{(R_{\text{CP}},j)}$  as columns is called the  $j^{\text{th}}$  factor matrix of this factoring. We call this minimum possible value of  $R_{\text{CP}}$  as the CP-rank of the tensor  $\mathcal{X}$  and denote it by  $\text{rank CP}(\mathcal{X})$ .

A special case of CP decomposition of considerable interest to research communities is the orthogonal tensor decomposition. An orthogonal decomposition of a symmetric tensor  $\mathcal{X}^{n \times \dots \times n}$  of order  $k$  is a decomposition

$$\mathcal{X}^{n \times \dots \times n} = \sum_{r=1}^{R_{\text{CP}}} \lambda_r \underbrace{\mathbf{a}^{(r)} \circ \dots \circ \mathbf{a}^{(r)}}_{k \text{ times}} \quad (2)$$

such that the vectors  $\mathbf{a}^{(1)}, \dots, \mathbf{a}^{(R_{\text{CP}})} \in \mathbb{R}^n$  form an orthonormal family of vectors. A tensor is called orthogonally decomposable if it has an orthogonal decomposition. Note that for an orthogonal decomposition  $R_{\text{CP}} \leq n$ .

The following point is worth mentioning about the CP decomposition and the CP-rank. Consider the matrix factoring  $\mathbf{A} = \mathbf{X} \mathbf{Y}^T$  of a matrix  $\mathbf{A}$  that plays a crucial role

in many matrix analysis methods. Using the standard SVD of matrix  $\mathbf{A}$ , we can easily write  $\mathbf{A} = \mathbf{U}\mathbf{\Sigma}\mathbf{V}^T = \mathbf{X}\mathbf{Y}^T$  where  $\mathbf{X} = \mathbf{U}\mathbf{\Sigma}$  and  $\mathbf{Y} = \mathbf{V}$ . However, the factoring is hardly unique since we can also take  $\mathbf{X} = \mathbf{U}\mathbf{\Sigma}\mathbf{Z}$  and  $\mathbf{Y} = \mathbf{V}\mathbf{Z}$  for any orthogonal matrix  $\mathbf{Z}$ . In contrast, the CP factoring of a higher mode tensor  $\mathcal{X}$  is unique under less stringent conditions. For example, in [7] it is shown that there is any symmetric tensor of order 3 has a unique CP decomposition provided we only allows the case when all the vectors in the decomposition are mutually linearly independent, and such a decomposition can in fact be found in polynomial time. The reader is referred to [7,20,21] for further results in this direction.

#### TUCKER decomposition [22–25]

This can be intuitively thought of as a generalization to higher-order tensors of a generic matrix factoring  $\mathbf{A} = \mathbf{X}\mathbf{Y}^T$  of a matrix  $\mathbf{A}$ . In TUCKER factoring one expresses  $\mathcal{X}$  as a core tensor  $\mathcal{K}^{J_1 \times \dots \times J_k}$  of minimal size transformed by a matrix  $\mathbf{Y}^{(l)}$  of size  $I_l \times J_l$  along each mode  $l$ , i.e., one finds  $\mathcal{K}^{J_1 \times \dots \times J_k}$ ,  $\mathbf{Y}^{(1)}, \dots, \mathbf{Y}^{(k)}$ , for arbitrary  $k$  that solves the minimization problem:

$$\begin{aligned} & \text{minimize size}(\mathcal{K}^{J_1 \times \dots \times J_k}) \text{ subject to} \\ & \mathcal{X} = \mathcal{K} \times_1 \mathbf{Y}^{(1)} \dots \times_k \mathbf{Y}^{(k)}. \end{aligned} \quad (3)$$

It is sometimes more convenient to interpret the tensor equality  $\mathcal{X} = \mathcal{K} \times_1 \mathbf{Y}^{(1)} \dots \times_k \mathbf{Y}^{(k)}$  via the following equivalent element-wise equality:

$$x_{i_1, i_2, \dots, i_n} = \sum_{j_1=1}^{J_1} \sum_{j_2=1}^{J_2} \dots \sum_{j_n=1}^{J_n} k_{j_1, j_2, \dots, j_n} y_{i_1, j_1} y_{i_2, j_2} \dots y_{i_n, j_n}.$$

In Equation (3), size  $(\mathcal{K}^{J_1 \times \dots \times J_k})$  is a suitable measure of the complexity of the tensor  $\mathcal{K}^{J_1 \times \dots \times J_k}$  (e.g., size  $(\mathcal{K}^{J_1 \times \dots \times J_k}) = (\prod_{i=1}^k J_i)^{1/k}$ ). We call the *minimum* possible value of size  $(\mathcal{K}^{J_1 \times \dots \times J_k})$  as the TUCKER-rank of  $\mathcal{X}$  and denote it by rank Tucker  $(\mathcal{X})$ .

#### Algorithmic and computational complexity aspects

A well-known algorithmic method to compute tensor decomposition while minimizing the corresponding rank is the alternating least square approach (see [7,11] for CP factoring and [26] for TUCKER factoring), but unfortunately no known non-trivial provable accuracy guarantees are known for these heuristic approaches (except worst-case NP-hardness [27,28] results which are mostly of theoretical interest only).

Very recently, there has been a surge in interest in the algorithmic community in applying the sum-of-squares

(SOS) approach for special cases of CP decomposition of symmetric tensors to obtain provable guarantees with high probability [29–34]. The SOS approach is a powerful mathematical technique that deals with determining the emptiness of a given semialgebraic set. Unfortunately, the mathematical details of full generalities of algorithmic applications of SOS approach for tensor decomposition is beyond the scope of this review paper, but we give some informal intuition about the approach; see [35,36] for excellent surveys on the SOS approach and its applications. Consider the CP decomposition framework in Equation (1) for a symmetric “square” tensor (i.e.,  $I_1 = \dots = I_k = q$  and  $\mathbf{a}^{(r,1)} = \dots = \mathbf{a}^{(r,k)} = \mathbf{a}^{(r)}$ ). Using a binary search scheme similar to that used in transforming a linear-programming optimization problem to a problem of determining the feasibility of a system of linear inequalities (see [37], we assume that we know the value of  $R_{CP}$  up to any desired accuracy, and thus for some  $p \approx R_{CP}$  we need to compute  $\lambda_r$ ’s and  $\mathbf{a}^{(r)}$ ’s such that  $\frac{1}{p} \mathcal{X} \approx \frac{1}{p} \sum_{r=1}^p \mathbf{b}^{(r) \circ k} = \mathcal{M}(k)$ , where  $\mathbf{b}^{(r) \circ k}$  denotes  $(\lambda_r^{1/k} \mathbf{a}^{(r)}) \circ \dots \circ (\lambda_r^{1/k} \mathbf{a}^{(r)})$  ( $k$  times). Note that  $\mathcal{M}(k)$  can be thought of as the  $k^{\text{th}}$ -moment of all possible  $\mathbf{b}^{(r)} \in \mathbb{R}^q$  over some unknown distribution  $\mathcal{D} : \mathbb{R}^q \rightarrow [0, 1]$ , and under this same (unknown) distribution  $\mathcal{M}(j) = \frac{1}{p} \sum_{r=1}^p \mathbf{b}^{(r) \circ j}$  can also be thought of as the  $j^{\text{th}}$ -moment of all possible  $\mathbf{b}^{(r)} \in \mathbb{R}^q$  for any  $j$ . Since finding  $\mathcal{D}$  is in general NP-hard, the SOS approach pursues the following alternate route:

**(I) (lifting moments higher)** Select a suitable  $l \geq k$  and relax the distribution  $\mathcal{D}$  appropriately to another “almost distribution”  $\tilde{\mathcal{D}} : \mathbb{R}^q \rightarrow \mathbb{R}$  such that (a)  $\tilde{\mathcal{D}}$  can be computed efficiently and (b) the “ $j^{\text{th}}$ -moment” under  $\tilde{\mathcal{D}}$  is very close to the  $j^{\text{th}}$ -moment under  $\mathcal{D}$  for all  $j = 1, \dots, l$  (in SOS terminology  $\tilde{\mathcal{D}}$  is called a degree- $l$  pseudo-distribution [29,30,35]).

**(II) (extracting factors from higher moments)** Use a “postprocessing” step to “approximately” infer the  $\mathbf{b}^{(r)}$ ’s from the pseudo-distribution  $\tilde{\mathcal{D}}$ .

## REPRESENTING BIOLOGICAL SYSTEMS AS TENSORS

An order  $k$  tensor  $\mathcal{X}^{I_1 \times \dots \times I_k}$  can easily model higher-order correlation of a biological system of  $n$  components with or without time evolution in the following manner:

**(tensors for static system)** For a static model where time evolution is ignored (such as the ones modelled by fixed interaction maps),  $I_1 = \dots = I_k$  and  $x_{i_1, \dots, i_k}$  may denote the joint  $k$ -wise correlation value of the  $i_1^{\text{th}}, \dots, i_k^{\text{th}}$  components of the system. For  $k = 2$ , this corresponds to a fixed edge-weighted graph in which an undirected edge between two nodes represents a pairwise correlation with

the edge weight representing a quantitative estimate of the correlation; for  $k > 2$  it properly generalizes such graph-theoretic models.

**(tensors for dynamic systems)** For a (discrete) time-varying dynamical model,  $I_1 = \dots = I_{k-1}$  the last dimension  $I_k$  corresponds to discrete time steps and  $x_{i_1, \dots, i_n}$  denotes the joint  $(k-1)$ -wise correlation value of the  $i_1^{\text{th}}, \dots, i_{k-1}^{\text{th}}$  components of the system at time  $t = i_n$ . For  $k = 2$ , this corresponds to time-series data models generated by experimental methods such as those using DNA microarrays. For  $k > 2$ , such a model is popular in representing dynamic social networks of various types in the context of data mining (e.g., see [38]).

Thus, we are led to the following natural questions

▷ How do we represent known real biological systems or models as order  $k$  tensors for some  $k > 2$ ?

▷ How do we generate large number of simulated biological system tensors which are critical in providing statistical validity of tensor analysis methods?

Answers to these questions are discussed in the next few subsections.

## Raw data sources for real biological systems

**Interaction maps with node dynamics.** Curated repositories of published systems biology models include a large number of such systems that are freely accessible. For example, the BioModels Database [39] contains 1640 dynamic models (640 are manually curated and encoded in the Systems Biology Markup Language). Discrete dynamic models are available in the cell collective [40] and the GINsim model repository [41].

**Time Series Data.** Published research works and large-scale repositories such as ArrayExpress [42] report a large amount of freely-accessible data, generated by various experimental methods (e.g., using DNA microarrays or RNA-seq), in the form of a matrix  $\mathbf{Y}^{q \times T}$ , where  $y_{i,t}$  is the value of the expression level of the  $i^{\text{th}}$  component (e.g., gene) at the  $t^{\text{th}}$  time step. For example, Chou *et al.* [43] provide yeast cell cycle time-series gene expression data for a number of time points with relatively small time intervals.

## Generating data for simulated biological systems

We can generate simulated biological systems that faithfully reproduce various types of real biological systems using a variety of approaches as discussed next.

**Interaction maps with node dynamics.** To generate simulated tensors for a specific type of biological systems, we will start with the known interaction map (with node dynamics) of a system of the same type (for example, for plant signal transduction system, we may use the light and drought signal transduction system in plants from

[44,45]). For the case of order 3 dynamic tensors, we then use the method in Section of “Biological systems to tensors” to generate a tensor  $\mathcal{X}^{q \times q \times T}$ . Using the mode 3 matricization of  $\mathcal{X}$ , we can view  $\mathcal{X}^{q \times q \times T}$  as a sequence of  $q \times q$  interaction/correlation maps, say  $\mathcal{Y}_1^{q \times q}, \dots, \mathcal{Y}_T^{q \times q}$ . Independently for each map  $\mathcal{Y}_j^{q \times q}$ , we can use several known methods, such as the following, to generate a new simulated correlation map:

▷ We may generate random maps using the Markov-chain algorithm in [46] by repeatedly swapping randomly chosen compatible pairs of entries in  $\mathcal{Y}_j^{q \times q}$ . This approach is widely used in systems biology context, e.g., see [1] and the references therein.

▷ If all the entries in  $\mathcal{Y}_1^{q \times q}, \dots, \mathcal{Y}_T^{q \times q}$  are from the set  $\{-1, 0, 1\}$ , we can treat each  $\mathcal{Y}_j^{q \times q}$  as a graph (with activation and inhibition edge labels) and thus may generate random maps from the degree-distribution of nodes in  $\mathcal{Y}_j^{q \times q}$  using the method pioneered by Newman and others in several publications [47–51] that preserves in expectation the degree distribution of each node, and label the edges independently randomly with appropriate probabilities as either activation or inhibition such that their percentages match those in  $\mathcal{Y}_j^{q \times q}$  in expectation.

Simulated order  $k$  dynamic tensors for  $k > 3$  can also be generated by a straightforward recursive generalization of the above procedures, e.g., generate higher-order time-varying correlations using the matrix algebra in [52] and then use a recursive matricization.

**Time-series data.** One can make use of already existing algorithmic implementations for generating timeseries data, such as the software package in [53] that can generate gene regulatory networks with external perturbations from differential equation models.

Finally, as in many simulation methods that use random distribution generators, if necessary the bias of our random distributions can be corrected using standard statistical techniques used in statistical data mining [54]. For example, given sample values  $x_1, \dots, x_m$  with average  $\mu$  and standard deviation  $\sigma$ , one such method is to first calculate the standardized value  $s_i = (x_i - \mu) / \sigma$  of  $x_i$ , then calculate the standardized range  $\alpha = \max_{1 \leq i \leq n} \{s_i\} - \min_{1 \leq i \leq n} \{s_i\}$ , and finally replace each original  $x_i$  by  $s_i / \alpha$ .

## Biological systems to tensors

**Interaction maps with node dynamics.** For such biological systems, we can start with an initial choice of states of nodes of the system as follows. If there is a preferred choice of initial non-steady state assignments, we can start with this choice (e.g., if our goal is to study apoptosis in a disease network starting from expressions of specific genes then our initial choice of states will assign the expression levels of these gene nodes to a higher value

and the rest of the nodes to a lower value). Otherwise, we may start with a suitable random choice of initial states of nodes. We then run the system up to a suitably large time step  $T$ . Our simulation outputs can be summarized in the form of a matrix  $\mathbf{M}^{q \times T}$ , where  $q$  is the number of nodes whose expression levels are measures over times  $t = 1, 2, \dots, T$  and  $m_{i,t}$  is the value of the expression level of the  $i^{\text{th}}$  gene at the  $t^{\text{th}}$  time step. From this data, we can, for example, construct an order 3 tensor  $\mathcal{X}^{q \times q \times T}$  representing the *second-order* time-evolving correlations among the nodes in the following manner:

▷ Let  $\mathbf{M}^{(t)}$  be the sub-matrix obtained by taking the first  $t$  columns of  $\mathbf{M}$ . Ensure that the mean of the observed data for each gene in  $\mathbf{M}^{(t)}$  is zero by subtracting  $\sum_{j=1}^t m_{i,j}^{(t)} / t$  from each  $m_{i,j}^{(t)}$  for  $i = 1, 2, \dots, q$  and  $j = 1, 2, \dots, t$ .

▷ Compute  $\mathbf{Z}^{(t)} = \mathbf{M}^{(t)} \mathbf{M}^{(t)\top}$  and set  $x_{i,j,t} = z_{i,j}^{(t)}$ .

Higher-order time-varying correlations can also be constructed by generalizing the above approach; for the relevant matrix algebra, see, for example [52].

**Time series data.** These data types can be handled in the same manner as done for interaction maps with node dynamics, except that we do not need to run any model to generate the time series.

**Time snapshots of interaction maps.** For input data consisting of explicit time snapshots of a given biological system, the corresponding tensor representation is straightforward.

## SYSTEM ANALYSIS VIA TENSOR DECOMPOSITION

There are a few reasons why finding an “exact” tensor decomposition (*cf.* Equations (1)–(3)) could be unrealistic for real or simulated biological tensors. Firstly, as noted before, most tensor decompositions are NP-hard to compute in the worst case [27,28]. Secondly, the input tensor data may be slightly noisy and therefore a computationally efficient inexact but almost accurate tensor decomposition may suffice. Finally, an exact decomposition may involve numbers (such as  $\sqrt{2}$ ) that have no finite-precision representations. Thus, in practice, for tensor decompositions one usually uses an “suitably approximate” version of tensor decomposition. For example, for a (sufficiently small) real number  $\varepsilon > 0$  and a suitable tensor norm  $\|\cdot\|$ , Equations (1)–(3) can be modified to their approximate versions (1)′–(3)′ as follows:

$$\mathcal{X}^{I_1 \times \dots \times I_k} \approx_{\varepsilon} \mathcal{A}^{I_1 \times \dots \times I_k} \stackrel{\text{def}}{=} \|\mathcal{X}^{I_1 \times \dots \times I_k} - \sum_{r=1}^{R_{\text{CP}}(\varepsilon)} \lambda_r \underbrace{\mathbf{a}^{(r,1)} \circ \dots \circ \mathbf{a}^{(r,k)}}_{=\mathcal{A}^{(r)} I_1 \times \dots \times I_k}\| \leq \varepsilon \quad (1')$$

$$\mathcal{X}^{n \times \dots \times n} \approx_{\varepsilon} \mathcal{A}^{n \times \dots \times n} \stackrel{\text{def}}{=} \|\mathcal{X}^{n \times \dots \times n} - \sum_{r=1}^{R_{\text{CP}}(\varepsilon)} \lambda_r \underbrace{\mathbf{a}^{(r)} \circ \dots \circ \mathbf{a}^{(r)}}_{=\mathcal{A}^{(r)} n \times \dots \times n}\| \leq \varepsilon \quad (2')$$

$$\|\mathcal{X} - \mathcal{K}(\varepsilon) \times_1 \mathbf{Y}^{(1)} \dots \times_k \mathbf{Y}^{(k)}\| \leq \varepsilon \quad (3')$$

Informally, system analysis via tensor decomposition aims to address research questions on extracting an ensemble of a small number of reduced subsystems (*i.e.*, subsystems that have less complexity, such as fewer correlations, than the original one) out of a given system to provide a concise and representative summary of the important correlations between components of the system such that non-trivial system analysis methods can operate on these reduced subsystems. For concreteness, our remaining discussion in this section assumes the tensor decomposition as found in Equation (1)′ with  $I_1 = \dots = I_k = n$ , but similar discussions hold for other tensor decompositions as well. Assume, without loss of generality, that we have re-scaled and re-indexed the values of  $\lambda_r$ ’s such that  $\lambda_1 \geq \lambda_2 \geq \dots \geq \lambda_{R_{\text{CP}}(\varepsilon)}$ , and  $\|\mathbf{a}^{r,j}\|_F = 1$  for all  $r = 1, 2, \dots, R_{\text{CP}}(\varepsilon)$  and  $j = 1, 2, \dots, k$ . Let

$\lambda'_j = \lambda_j / (\lambda_1 + \lambda_2 + \dots + \lambda_{R_{\text{CP}}(\varepsilon)})$ . Note that the value of  $\mathcal{A}_{i_1, \dots, i_k}^{(r)} \in [-1, 1]$  for any  $r$  and any  $i_1, \dots, i_k \in \{1, \dots, n\}$  can be interpreted as the value of a  $k$ -way correlation between the  $k$  variables, say  $x_{i_1}, \dots, x_{i_k}$ , corresponding to the  $k$  indices in the  $k$  dimensions, and each  $\lambda'_r$  can be thought of the significance probability of the corresponding  $r^{\text{th}}$  factor  $\mathcal{A}^{(r)}$ . Based on such interpretations, one can retrieve significant correlations from the factors in various ways. For example, one such method could be the following.

► First, select the significant factors in an appropriate way. Some possibilities could be as follows:

▷ For a suitable threshold (real number)  $0 < \Theta < 1$ , select all factors  $\mathcal{A}^{(j)}$  satisfying  $\lambda_j \geq \Theta$ .

▷ Select factors probabilistically where the  $j^{\text{th}}$  factor  $\mathcal{A}^{(j)}$  is selected with probability  $\lambda'_j$ . The randomized

strategy may be more suitable when the  $\lambda'_j$  values are not sufficiently spread out (e.g., their standard deviation is small).

▷ In some applications that require strong statistical decoupling, it may be desirable to select factors such that vectors corresponding to the same variable in different factors are mutually linearly independent (or close to being mutually linearly independent). For such situations, strategies such as the following could be used. Recall that  $\hat{\mathbf{A}}^{(j)}$  denotes the  $j^{\text{th}}$  factor matrix of the tensor  $\mathcal{A}$ . For a suitable  $\tau \in \left[ \min_{j=1}^k \left\{ k_{\hat{\mathbf{A}}^{(j)}} \right\} \leq \max_{j=1}^k \left\{ k_{\hat{\mathbf{A}}^{(j)}} \right\} \right]$ , select  $\tau$  indices, say  $l_1, \dots, l_\tau$ , such that for each  $j$  the vectors  $\mathbf{a}^{(l_1, j)}, \dots, \mathbf{a}^{(l_\tau, j)}$  form a mutually orthogonal family of vectors. We can then select the  $\tau$  factors  $\mathcal{A}^{(l_1)}, \dots, \mathcal{A}^{(l_\tau)}$ . The selection can be further refined using one or more rules discussed before.

► Once a factor is selected, we can retrieve significant correlations encoded by it by using an appropriate threshold, i.e., for a suitable threshold (real number)  $\Phi \in [-1, 1]$ , for a selected factor  $\mathcal{A}^{(j)}$  consider the correlation among the variables  $x_{i_1}, \dots, x_{i_k}$  as statistically significant if  $\mathcal{A}_{i_1, \dots, i_k}^{(r)} \geq \Phi$ . Other more complex application-specific strategies can also be designed.

One could interpret the value of the rank  $R_{\text{CP}}(\mathcal{E})$  itself as some measure of “complexity” of the input tensor  $\mathcal{X}$ . For other applications, it is possible to interpret the factors and the components of the factors in different ways (i.e., not necessarily as correlations between variables); for example see [2, 3, 55, 6, 55].

## STATISTICAL AND BIOLOGICAL VALIDATIONS OF TENSOR ANALYSIS

Validations of our tensor analysis can be classified into three categories:

**Methodological validation:** How do we estimate the “quality” of our tensor decomposition methods?

**Statistical validation:** How do we compute the “statistical significance” for our tensor analysis results?

**Biological validation:** Do our tensor decomposition methods recover reported correlations or pathways for known (published) biological systems?

For subsequent discussion purposes, assume that our tensor decomposition at the completion of final steps in Section of “System analysis via tensor decomposition” generated a collection of  $m$  significant tensor factors  $\mathcal{A}^{(1)}, \dots, \mathcal{A}^{(m)}$  and the combined single tensor  $\mathcal{A}' = \sum_{r=1}^m \mathcal{A}^{(r)}$  for the input tensor  $\mathcal{X}$ .

### Methodological validation

The parameter  $\varepsilon$  in the optimization framework for tensor decomposition (cf. Equations (1)–(3)) controls the initial accuracy of the decomposition and varying  $\varepsilon$  we can obtain decompositions of different accuracies. However, we need also to have control on the accuracy after the reduction steps in Section of “System analysis via tensor decomposition”. One possible way to do this is via calculation of the relative error  $\xi = \|\mathcal{X} - \mathcal{A}'\| / \|\mathcal{X}\|$ .

### Statistical validation

**Noise sensitivity and robustness of the decompositions in Equations (1)–(3).** We can use the theoretical framework in [56, 57] inspired by the famous smoothed-analysis results in [58, 59]. We illustrate the framework for the rank measure for CP decomposition (Equation (1); adoptions for other decompositions are very similar. We perturb each vector  $\mathbf{a}^{(r, j)}$  suitably<sup>①</sup> to obtain a new vector  $\hat{\mathbf{a}}^{(r, j)}$ , build the new tensor  $\hat{\mathcal{A}} = \sum_{r=1}^{R_{\text{CP}}(\mathcal{E})} \lambda_r \hat{\mathbf{a}}^{(r, 1)} \circ \dots \circ \hat{\mathbf{a}}^{(r, k)}$ , use the same CP decomposition algorithm on  $\hat{\mathcal{A}}$  to compute the new rank  $\widehat{R_{\text{CP}}}(\mathcal{E})$ , and finally measure the sensitivity by the relative error  $(R_{\text{CP}}(\mathcal{E}) - \widehat{R_{\text{CP}}}(\mathcal{E})) / R_{\text{CP}}(\mathcal{E})$ .

**p-value calculation.** There are many ways to do this; we illustrate one such approach. We can use the method used in [60, 61] to calculate the  $p$ -values for the rank or individual significant correlations that we found at the completion of final steps in Section of “System analysis via tensor decomposition”. We illustrate the method for the rank. Suppose that we wish to calculate the  $p$ -value of a particular evaluation of the rank  $R(\mathcal{X})$  of a biological system (tensor)  $\mathcal{X}$ . We will generate a large number  $q$  of simulated systems  $\mathcal{X}^{(1)}, \dots, \mathcal{X}^{(q)}$  of the same type as  $\mathcal{X}$  using the Markov-chain algorithm of Section of “Generating data for simulated biological systems”, compute the corresponding ranks  $R(\mathcal{X}^{(1)}), \dots, R(\mathcal{X}^{(q)})$  of these simulated systems, and then use an appropriate statistical test, such as a (unpaired) one-sample student’s  $t$ -test, to determine the probability that  $R(\mathcal{X})$  can be generated by a distribution that fits the data points  $R(\mathcal{X}^{(1)}), \dots, R(\mathcal{X}^{(q)})$ .

### Biological validation

Here we determine how close our tensor analysis is in preserving important properties of a known system. These validations are conceptually straightforward and can

<sup>①</sup> The authors in [55, 56] add independent Gaussian noise to each coordinate of  $\mathbf{a}^{(r, j)}$ , but other distributions may be used.

check a variety of properties of a known biological system in its corresponding reduced system. For example:

►We can check the known presence or absence of a significant correlation of a known (published) system in most significant tensors factors produced after Section of “System analysis via tensor decomposition”. Based on the number of true positives, false positives, true negatives and false negatives, we may compute the four standard metrics true positive rate, false positive rate, accuracy rate and precision to assess the validity of our methods. For fine tuning a specific parameter of our method, such as the  $\varepsilon$  parameter in Equation (1), we may use the ROC (receiver operating characteristic) plot over the ranges of  $\varepsilon$ .

►For dynamical tensors, we may check if our significant tensors produce a system that preserve known significant dynamical properties (such as limit cycles, attractors, monotonicity, controllability and observability) of a known (published) system.

As a concrete illustration, let  $\mathcal{X}$  be an order 3 tensor generated from the Boolean dynamic model for the *guard cell ABA signaling* as obtained in [44] using the method described in Section of “Biological systems to tensors”, and suppose that our CP decomposition together with the optimizations in Section of “System analysis via tensor decomposition” produces  $q$  most significant tensors  $\mathcal{A}^{(1)} = \mathbf{a}^{(1,1)} \circ \mathbf{a}^{(1,2)} \circ \mathbf{a}^{(1,3)}, \dots, \mathcal{A}^{(q)} = \mathbf{a}^{(q,1)} \circ \mathbf{a}^{(q,2)} \circ \mathbf{a}^{(q,3)}$  for  $\mathcal{X}$ . The double and triple knockout experiments in [44] suggest that the states of CalM ( $\text{Ca}^{2+}$  influx through the plasma membrane) and AnionEM (anion efflux at the plasma membrane) are jointly correlated to the state of Closure (ABA signalling). Assuming this to be the

ground truth, we may check if this correlation also exists in at least one of  $\mathcal{A}^{(1)}, \dots, \mathcal{A}^{(q)}$ .

## SPECIFIC ILLUSTRATIONS OF TENSOR ANALYSIS

In this section we provide two specific illustrations of tensor analysis for biological systems. The first one is an artificial toy example. The second illustration summarizes tensor analysis results for a specific research result.

### A toy example of tensor analysis insights of an artificial biological system

Consider the following mode 3 tensor  $\mathcal{X} \stackrel{\text{def}}{=} \mathcal{X}^{t \times b \times c}$  that describes time-evolving cross-correlations between two sets of four components  $\mathbf{b} = (b_1, b_2, b_3, b_4)$  and  $\mathbf{c} = (c_1, c_2, c_3, c_4)$  of a biological system over discrete time  $\mathbf{t} = (1, 2, 3, 4)$  (Figure 2) (modified from an example in [62,63]):

Consider CP decomposition as shown in Figure 2 with appropriate normalization of the inter-modal factors (c.f. Section of “CANDECOMP/PARAFAC (CP) decomposition”):

Assuming that this tensor decomposition has been statistically validated, a simplest predictive algorithm will identify

$$\begin{pmatrix} -1 \\ 0 \\ 0 \\ 0 \end{pmatrix} \circ \begin{pmatrix} -1 \\ 0 \\ 0 \\ 0 \end{pmatrix} \circ \begin{pmatrix} -1 \\ 0 \\ 0 \\ 0 \end{pmatrix},$$

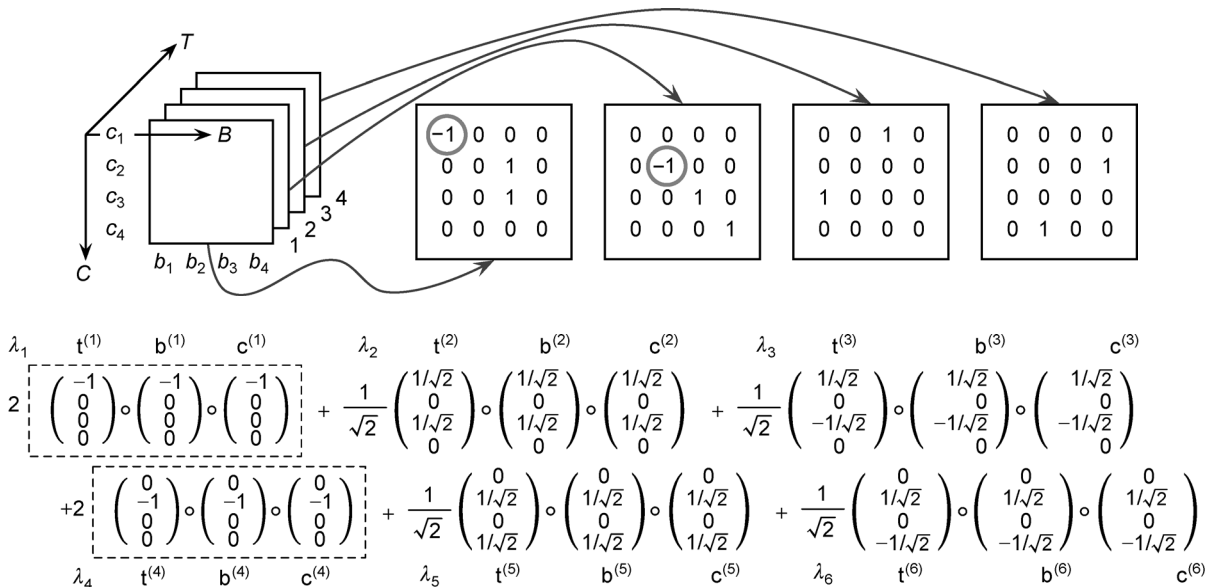


Figure 2. An example of an artificial biological system.

and

$$\begin{pmatrix} 0 \\ -1 \\ 0 \\ 0 \end{pmatrix} \circ \begin{pmatrix} 0 \\ -1 \\ 0 \\ 0 \end{pmatrix} \circ \begin{pmatrix} 0 \\ -1 \\ 0 \\ 0 \end{pmatrix}$$

(shown by dotted boxes) as the two most significant factors, and they can be traced back to the correlations between  $b_1$  and  $c_1$  at  $t = 1$  and between  $b_2$  and  $c_2$  at  $t = 2$  (circled in gray color).

### A review of specific tensor analysis in two research papers

In this section, we briefly review the tensor analysis methods and the biological conclusions drawn therefrom in the two research papers [2,5]. The application of tensor decomposition methods in these two papers to yeast (*S. cerevisiae*) time course expression data illustrates the biological insights that can be gained from this type of analysis. All the measurements were made in *S. cerevisiae* cell cultures under the influence of the pheromone  $\alpha$ -factor (to synchronize their cell cycles).

Alter and Golub [2] used matrix eigenvalue decomposition to decompose a matrix of pairwise gene correlations into rank-1 sub-matrices. They then constructed a tensor that integrates information about gene expression and the binding of select transcription factors to the promoter regions to each gene. By applying tensor higher order eigenvalue decomposition, they identified decorrelated rank-1 sub-networks that can be associated with independent biological pathways. They applied this methodology to a time-course of mRNA expression data for more than 4,000 *S. cerevisiae* genes, integrated with binding data on 12 cell cycle related transcription factors and 12 developmental transcription factors. The analysis uncovers three significant sub-networks that capture 40%, 15%, and 9%, respectively, of the expression correlation among genes. The first subnetwork is associated with the  $\alpha$ -factor signal transduction pathway, and expresses the correlations among genes that are up-regulated (or down-regulated, respectively) in response to pheromone. The second and third sub-networks are associated with the two known pathways of antipodal (opposite) cell cycle expression oscillations. The coupling between the first sub-network and the second sub-network expresses the exit from pheromone-induced cell cycle arrest and entry into cell cycle progression.

Omberg *et al.* [5] used higher-order singular value decomposition to decompose a data tensor of genes versus experimental settings into rank-1 sub-tensors. They applied this methodology to mRNA expression data for more than 4,000 *S. cerevisiae* genes, measured over 13 time points, for three conditions, namely

oxidative stress due to exposure to hydrogen peroxide or menadione, respectively, and oxidative-stress-free control. They found that the significant sub-tensors represent independent biological programs. The first and most significant sub-tensor, which captures 70% of the expression information, represents the steady state of mRNA expression in response to hydrogen peroxide, menadione, or  $\alpha$ -factor, averaged over time and conditions. Three sub-tensors that follow in significance (explaining 1% to 6% of the information) represent change in expression in response to oxidative stress. The three following sub-tensors, each explaining around 1% of the expression information, represent pheromone responses and pheromone-induced oxidative stress responses. The three following sub-tensors (explaining 0.6% to 0.9% of the information) reflect the differences in the responses to hydrogen peroxide and to menadione.

### CONCLUSION

In this article we have reviewed some basic aspects of powerful tensor analysis methods that provide the foundations of systematic approaches to determine significant higher order correlations among elements of biological systems by finding ensembles of small number of reduced systems that provide a concise and representative summary of these correlations.

Admittedly, a short review article such as this one can cover only some aspects of tensor analysis of biological systems, leaving other aspects in the references. For example, following are some of the topics that are not covered in this article but may be of significance to some researchers:

**Learning models via tensor decompositions:** References such as [55,64] discuss algorithms for learning hidden Markov models or other types of models using tensor decompositions.

**Eigenvalues and eigenvectors of tensors:** Eigenvalues and eigenvectors of matrices (*i.e.*, tensors of order 2) play a crucial role in spectral analysis of network algorithms and processes, and in principal component analysis for matrix data. One can extend these definitions to higher-order tensors (*e.g.*, see [65]) but the full potential of these generalizations to higher-order tensor analysis is still not clear.

Although tensor analysis methods have been used in specific contexts of computational biology before, their usage in bioinformatics is not as widespread as matrix-based or linear algebraic methods. In our opinion, there is a pressing need for more research on applying tensor analysis methods for biological systems, as interpretations of results of high-throughput experiments have advanced at a much slower pace than the data accumulation. The state of the art in gene expression data analysis is still to

focus on a small group of key genes (*e.g.*, those that are most highly correlated, or most differentially expressed when comparing two contexts) and discard the rest of the information. This is partly because of the computational complexity of all types of follow-up analyses, and partly because of the noise and uncertainty affecting all biological measurements. Tensor-based analysis provides a principled way of using all available information to achieve a clearer understanding. In addition, by identifying reduced systems, a smaller set of most-supported correlations naturally emerges that are optimal for all follow-up analyses without making potentially limiting assumptions. The tensor analysis framework does not have the limitations and specific requirements of methods, such as the bi-clustering approach [66], that are used to determine clusters of genes correlated over a set of conditions but not in conditions outside of this set. The tensor framework is also naturally suited to incorporate and study the effect of additional variables (*e.g.*, variable environmental influences), thus allowing integrated studies of genetic and environmental factors. It is our hope that this article will catalyze and motivate further research in the fascinating inter-disciplinary interplay between biology and tensor analysis methods.

## ACKNOWLEDGEMENTS

B.D. and F.Y. was supported by NSF grant IIS-1814931. R.A. was supported by NSF grant IIS-1814405.

## COMPLIANCE WITH ETHICS GUIDELINES

The authors Farzane Yahyanejad, Réka Albert and Bhaskar DasGupta declare that they have no conflict of interests.

This article is a review article and does not contain any studies with human or animal subjects performed by any of the authors.

## REFERENCES

1. DasGupta, B. and Liang, J. (2016) Models and Algorithms for Biomolecules and Molecular Networks. John Wiley and Sons, Inc.
2. Alter, O. and Golub, G. H. (2005) Reconstructing the pathways of a cellular system from genome-scale signals by using matrix and tensor computations. *Proc. Natl. Acad. Sci. USA*, 102, 17559–17564
3. Dyrby, M., Baunsgaard, D., Bro, R. and Engelsen, S. B. (2005) Multiway chemometric analysis of the metabolic response to toxins monitored by NMR. *Chemom. Intell. Lab. Syst.*, 76, 79–89
4. Hore, V., Viñuela, A., Buil, A., Knight, J., McCarthy, M. I., Small, K. and Marchini, J. (2016) Tensor decomposition for multiple-tissue gene expression experiments. *Nat. Genet.*, 48, 1094–1100
5. Omberg, L., Golub, G. H. and Alter, O. (2007) A tensor higher-order singular value decomposition for integrative analysis of DNA microarray data from different studies. *Proc. Natl. Acad. Sci. USA*, 104, 18371–18376
6. Yener, B., Acar, E., Aguis, P., Bennett, K., Vandenberg, S. L. and Plopper, G. E. (2008) Multiway modeling and analysis in stem cell systems biology. *BMC Syst. Biol.*, 2, 63
7. Harshman, R. A. (1970) Foundations of the PARAFAC procedure: Models and conditions for an “explanatory” multi-modal factor analysis. *UCLA Working Papers in Phonetics*, 16, 1–84
8. Miwakeichi, F., Martínez-Montes, E., Valdés-Sosa, P. A., Nishiyama, N., Mizuhara, H. and Yamaguchi, Y. (2004) Decomposing EEG data into space-time-frequency components using Parallel Factor Analysis. *Neuroimage*, 22, 1035–1045
9. Mørup, M. (2011) Applications of tensor (multiway array) factorizations and decompositions in data mining. *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery*, 1, 24–40
10. Mørup, M., Hansen, L. K. and Arnfred, S. M. (2007) ERPWAVELAB a toolbox for multi-channel analysis of time-frequency transformed event related potentials. *J. Neurosci. Methods*, 161, 361–368
11. Carroll, J. D. and Chang, J. J. (1970) Analysis of individual differences in multidimensional scaling via an N-way generalization of “Eckart-Young” decomposition. *Psychometrika*, 35, 283–319
12. Kroonenberg, P. M. (2008) Applied Multiway Data Analysis. John Wiley & Sons
13. Murakami, T. and Kroonenberg, P. M. (2003) Three-mode models and individual differences in semantic differential data. *Multivariate Behav. Res.*, 38, 247–283
14. Appellof, C. J. and Davidson, E. R. (1981) Strategies for analyzing data from video fluorometric monitoring of liquid chromatographic effluents. *Anal. Chem.*, 53, 2053–2056
15. Kolda, T. G. and Bader, B. W. (2009) Tensor decompositions and applications. *SIAM Rev.*, 51, 455–500
16. De Lathauwer, L., De Moor, B. and Vandewalle, J. (2000) A multilinear singular value decomposition. *SIAM J. Matrix Anal. Appl.*, 21, 1253–1278
17. Miettinen, P. (2011) Boolean tensor factorizations. In: 11th IEEE International Conference on Data Mining, 447–456
18. Kiers, H. A. L. (2000) Towards a standardized notation and terminology in multiway analysis. *J. Chemometr.*, 14, 105–122
19. Golub, G. H. and Van Loan, C. F. (1996) Matrix Computations, 3rd edition. The Johns Hopkins University Press
20. Kruskal, J. B. (1977) Three-way arrays: Rank and uniqueness of trilinear decompositions, with application to arithmetic complexity and statistics. *Linear Algebra Appl.*, 18, 95–138
21. Kruskal, J. B. (1989) Rank, decomposition, and uniqueness for 3-way and N-way arrays. In: Multiway Data Analysis, Coppi, R. and Bolasco, S. (Eds.), pp. 7–18. North-Holland, Amsterdam
22. Levin, J. (1963) Three-Mode Factor Analysis. Dissertation for the Doctoral Degree, University of Illinois, Urbana
23. Tucker, L. R. (1963) Implications of factor analysis of three-way matrices for measurement of change. In: Problems in Measuring Change, Harris, C. W. (ed.), University of Wisconsin Press, 122–137
24. Tucker, L. R. (1964) The extension of factor analysis to three-dimensional matrices. In: Contributions to Mathematical Psychol-

- ogy, Gulliksen, H. and Frederiksen, N. (eds.), Holt, Rinehardt & Winston, New York, 110–127
25. Tucker, L. R. (1966) Some mathematical notes on three-mode factor analysis. *Psychometrika*, 31, 279–311
26. Kroonenberg, P. M. and De Leeuw, J. (1980) Principal component analysis of three-mode data by means of alternating least squares algorithms. *Psychometrika*, 45, 69–97
27. Håstad, H. (1990) Tensor rank is NP-complete. *J. Algorithms*, 11, 644–654
28. Hillar, C. J. and Lim, L.-H. (2013) Most tensor problems are NP-hard. *J. ACM*, 60, 45:1–45:38
29. Barak, B., Kelner, J. A. and Steurer, D. (2015) Dictionary learning and tensor decomposition via the sum-of-squares method. In: *Proceedings of the 47th ACM Symposium on Theory of Computing*, pp. 143–151
30. Barak, B. and Moitra, A. (2016) Noisy tensor completion via the sum-of-squares hierarchy. In: *Proceedings of Machine Learning Research*, 49, 417–445
31. Ge, R. and Ma, T. (2015) Decomposing overcomplete 3rd order tensors using sum-of-squares algorithms. In: *Proceedings of the 18th International Workshop on Approximation Algorithms for Combinatorial Optimization Problems*, pp. 829–849
32. Hopkins, S. B., Shi, J. and Steurer, D. (2015) Tensor principal component analysis via sum-of-square proofs. *Proceedings of Machine Learning Research*, 40, 956–1006
33. Ma, T., Shi, J. and Steurer, D. (2016) Polynomial-time tensor decompositions with sum-of-squares. In: *Proceedings of the 57th Annual IEEE Symposium on Foundations of Computer Science*, pp. 438–446
34. Potechin, A. and Steurer, D. (2017) Exact tensor completion with sum-of-squares. In: *Proceedings of Machine Learning Research*, 65, 1–54
35. Barak, B. and Steurer, D. (2014) Sum-of-squares proofs and the quest toward optimal algorithms. In: *Proceedings of International Congress of Mathematicians*
36. Laurent, M. (2009) Sums of Squares, Moment Matrices and Optimization Over Polynomials. In: *Emerging Applications of Algebraic Geometry, The IMA Volumes in Mathematics and its Applications*, Putinar M. and Sullivant S., (eds.), 149, 157–270
37. Papadimitriou, C. H. and Steiglitz, K. (1988) *Combinatorial Optimization: Algorithms and Complexity*. Dover Publications
38. Sun, J., Tao, D. and Faloutsos, C. (2006) Beyond Streams and Graphs: Dynamic Tensor Analysis. In: *Proceedings of the 12th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 374–383
39. Chelliah, V., Juty, N., Ajmera, I., Ali, R., Dumousseau, M., Glont, M., Hucka, M., Jalowicki, G., Keating, S., Knight-Schrijver, V., *et al.* (2015) BioModels: ten-year anniversary. *Nucleic Acids Res.*, 43, D542–D548
40. Helikar, T., Kowal, B., McClenathan, S., Bruckner, M., Rowley, T., Madrahimov, A., Wicks, B., Shrestha, M., Limbu, K. and Rogers, J. A. (2012) The Cell Collective: toward an open and collaborative approach to systems biology. *BMC Syst. Biol.*, 6, 96
41. Chaouiya, C., Naldi, A. and Thieffry, D. (2012) Logical modelling of gene regulatory networks with GINsim. *Methods Mol. Biol.*, 804, 463–479
42. Kolesnikov, N., Hastings, E., Keays, M., Melnichuk, O., Tang, Y. A., Williams, E., Dylag, M., Kurbatova, N., Brandizi, M., Burdett, T., *et al.* (2015) ArrayExpress update—simplifying data submissions. *Nucleic Acids Res.*, 43, D1113–D1116
43. Cho, R. J., Campbell, M. J., Winzeler, E. A., Steinmetz, L., Conway, A., Wodicka, L., Wolfsberg, T. G., Gabrielian, A. E., Landsman, D., Lockhart, D. J., *et al.* (1998) A genome-wide transcriptional analysis of the mitotic cell cycle. *Mol. Cell*, 2, 65–73
44. Li, S., Assmann, S. M. and Albert, R. (2006) Predicting essential components of signal transduction networks: a dynamic model of guard cell abscisic acid signaling. *PLoS Biol.*, 4, e312
45. Sun, Z., Jin, X., Albert, R. and Assmann, S. M. (2014) Multi-level modeling of light-induced stomatal opening offers new insights into its regulation by drought. *PLOS Comput. Biol.*, 10, e1003930
46. Kannan, R., Tetali, P. and Vempala, S. (1999) Markov-chain algorithms for generating bipartite graphs and tournaments. *Random Struct. Algor.*, 14, 293–308
47. Girvan, M. and Newman, M. E. J. (2002) Community structure in social and biological networks. *Proc. Natl. Acad. Sci. USA*, 99, 7821–7826
48. Leicht, E. A. and Newman, M. E. J. (2008) Community structure in directed networks. *Phys. Rev. Lett.*, 100, 118703
49. Newman, M. E. J. (2004) Fast algorithm for detecting community structure in networks. *Phys. Rev. E Stat. Nonlin. Soft Matter Phys.*, 69, 066133
50. Newman, M. E. J. (2003) The structure and function of complex networks. *SIAM Rev.*, 45, 167–256
51. Newman, M. E. J. and Girvan, M. (2004) Finding and evaluating community structure in networks. *Phys. Rev. E Stat. Nonlin. Soft Matter Phys.*, 69, 026113
52. Meijer, E. (2005) Matrix algebra for higher order moments. *Linear Algebra Appl.*, 410, 112–134
53. Mendes, P. (1997) Biochemistry by numbers: simulation of biochemical pathways with Gepasi 3. *Trends Biochem. Sci.*, 22, 361–363
54. Larsen, R. J. and Marx, M. L. (2000) *An Introduction to Mathematical Statistics and Its Applications*. Third Edition, Pearson Education
55. Anandkumar, A., Ge, R., Hsu, D., Kakade, S. M. and Telgarsky, M. (2014) Tensor decompositions for learning latent variable models. *J. Mach. Learn. Res.*, 15, 2773–2832
56. Bhaskara, A., Charikar, M., Moitra, A. and Vijayaraghavan, A. (2014) Smoothed analysis of tensor decompositions. In: *Proceedings of the 46th Annual ACM Symposium on Theory of Computing*, 594–603
57. Goyal, N., Vempala, S. and Xiao, Y. (2014) Fourier PCA and robust tensor decomposition. In: *Proceedings of the 46th Annual ACM Symposium on Theory of Computing*, 584–593
58. Spielman, D. A. and Teng, S. H. (2004) Smoothed analysis of algorithms: Why the simplex algorithm usually takes polynomial

- time. *J. Assoc. Comput. Mach.*, 51, 385–463
59. Spielman, D. A. and Teng, S. H. (2009) Smoothed Analysis: An Attempt to Explain the Behavior of Algorithms in Practice. *Commun. ACM*, 52, 76–84
  60. Albert, R., DasGupta, B., Hegde, R., Sivanathan, G. S., Gitter, A., Grsoy, G., Paul, P. and Sontag, E. (2011) Computationally efficient measure of topological redundancy of biological and social networks. *Phys. Rev. E Stat. Nonlin. Soft Matter Phys.*, 84, 036117
  61. Albert, R., DasGupta, B. and Mobasheri, N. (2014) Topological implications of negative curvature for biological and social networks. *Phys. Rev. E Stat. Nonlin. Soft Matter Phys.*, 89, 032811
  62. Comon, P. (2004) Canonical Tensor Decompositions. Research report ISRN I3S/RR-2004-17-FR
  63. Comon, P. and Mourrain, B. (1996) Decomposition of quantics in sums of powers of linear forms. *Signal Processing*, 53, 93–107
  64. Anandkumar, A., Hsu, D. and Kakade, S. M. (2012) A method of moments for mixture models and hidden markov models. In: *Proceedings of the Conference on Learning Theory*, 33.1–33.34
  65. Qi, L. (2007) Eigenvalues and invariants of tensors. *J. Math. Anal. Appl.*, 325, 1363–1377
  66. Reiss, D. J., Baliga, N. S. and Bonneau, R. (2006) Integrated biclustering of heterogeneous genome-wide datasets for the inference of global regulatory networks. *BMC Bioinformatics*, 7, 280