ANNALI DELL'UNIVERSITA' DI FERRARA (2024) 70:1767–1783
https://doi.org/10.1007/s11565-024-00541-5

Boolean operators and neural networks

Sara Marziali[1]

Received: 19 December 2023 / Accepted: 17 July 2024 / Published online: 22 August 2024

Abstract
We compute the homogeneous ideals of varieties, in a projective space of tensors, associated to different choices of the Boolean operators that describe the decision process in small neural networks. We prove that, starting with networks with three nodes, the varieties associated to different Boolean operators are all distinct.

Keywords Tensors · Networks · Boolean operators

Mathematics Subject Classification 14N07

1 Introduction

The note is devoted to determine the behaviour of some algebraic (projective) varieties which are naturally associated to the study of neural tensor networks (we refer to [1] and [14] for an introduction to the algebraic geometry aspects of the theory of neural networks).

In the tensor networks that we consider a digital signal passes through the nodes, and every node decides its status (and its output) by combining the inputs following some rules. Our task is to detect the rules acting on the nodes (that we suppose uniform in the network) by observing the total tensor that describes the status of the network. More precisely, we are interested in detecting how nodes react after receiving different signals from their input connections.

If a node C receives just one signal then its status is determined. When two signals arrive together in C, then C adopts a rule that establishes its activation as a function Δ of the two inputs it receives. Since there are only four possible combinations of two digital inputs, and the output is 0 or 1, there are 16 possibilities for the function that determines the final status. These functions Δ, that we call *logical operators*, are listed below (Table 1). They correspond to non-linear activating functions in the language of signal theory.

✉ Sara Marziali
sara.marziali@student.unisi.it

1 Dipartimento di Ingegneria dell'Informazione e Scienze Matematiche, Universitá degli Studi di Siena, 53100 Siena, Italy

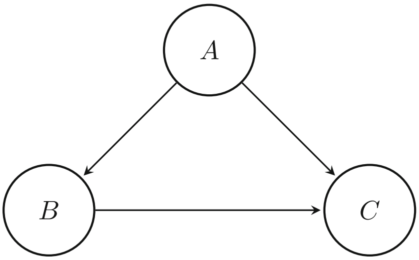


Fig. 1 Neural tensor network with three nodes A, B and C

In our networks, *we suppose that each signal is perturbed* with a probability that depends on a 2×2 perturbation matrix M. We assume M to be a Jukes–Cantor matrix, constant along the edges of the network. The total status of the network thus determines a subset (i.e. a *model*) in a tensor space of type $2 \times 2 \times \cdots \times 2$ (see [13] and [3]) where each factor corresponds to a node. The subset is described by parametric equations in the parameters of the perturbation matrix and in the parameters describing the distribution of the source signal. The model will depend on the topology of the network, as well as the choice of the logical operator Δ.

In the paper, we consider very simple networks in which every node has at most two inputs. The simplest network, formed by two nodes connected by two edges, turns out to be too simple for our purposes (see Example 4.10). Thus we will start with a network with three nodes A, B, C, in which A, the source, sends the signal both to B and C, and B also sends a signal to C (Fig. 1). All signal are perturbed by a perturbation matrix acting on the edges.

The final goal is to determine the logical operator Δ adopted by C from a series of experiments consisting in sending a signal through the network and computing the relative status of the nodes. In practice, we will determine, for each Δ, the parametric equations of the associated model M_Δ. Indeed, we will characterise the model in the complex extension of the space of tensors. The main result we found is that whenever $\Delta_1 \neq \Delta_2$, then also $M_{\Delta_1} \neq M_{\Delta_2}$ (see Theorem 4.7).

The result follows from tools of Algebraic geometry. Each model is represented by parametric equations which turn out to be homogeneous polynomials of the same degree. Thus, we can consider each M_Δ as a variety in the projective tensor space $\mathbb{P} = \mathbb{P}(\mathbb{C}^2 \otimes \mathbb{C}^2 \otimes \mathbb{C}^2)$ over the complex field. Working in projective spaces will simplify the analysis, since we will need not to restrict the attention to stochastic vectors, matrices, or tensors (see [3]).

We will determine the parametric equations of each M_Δ, then, following [7], by means of the SINGULAR computer algebra package [9] we will find the radical, saturated, homogeneous ideal of each M_Δ in $\mathbb{P}$, which turn to be all of dimension 4 (see also [2]). By comparing the ideals, again with the aid of computer algebra algorithms, we will prove our main result.

We stress that the comparison among ideals of the models M_Δ can only yield that they are different in their extensions over the complex field, because it implicitly uses the Nullstellensatz. However, we feel that the observation contained in our main result can be useful to understand the behaviour of concrete networks. We also note, in Example 4.9, that the procedure applies also to larger networks, provided that they can be reduced (by taking subnetworks, or by grouping nodes) to the simple network in Fig. 3.

2 Related works

The study of neural networks associated with graphs by exploiting matrices and tensors finds its initial motivation in quantum physics and, more specifically, in the theory of quantum many-body systems [18]. In quantum mechanics, a vector in Hilbert space describes the state of an isolated system, while a tensor product can be used to represent the state space of a compound system when it is the product of state spaces [17]. The phenomenon that occurs when the tensor is not of rank 1 is called entanglement [6].

Other applications of tensor networks include numerical solutions of partial differential equations, chemistry and numerous additional areas. In phylogenetics, among other fields, the analysis of networks has become increasingly popular because of their ability to describe a wider variety of evolutionary events than their tree-like counterparts. In fact, studies of Markovian models of phylogenetic networks are crucial for reconstructing events such as hybridization and horizontal gene transport [1, 4, 10].

In general, tensor networks are associated to simple graphs, that is, undirected graphs without loops or multi-arcs, with one vertex for each electron of the system [11]. In addition, we can consider a set of positive integer weights on the edges, called bond dimensions, and a set of natural numbers associated with the nodes, referred to as the local dimensions of the graph [2]. For each node of the graph, in fact, a vector space of dimension determined by the local dimension assigned to the node is constructed. Then, the tensor network is a triplet that consists of the simple graph, the set of bond dimensions, and the collection of local dimensions. In [5], the tensor network analysis is extended to the case of hypergraphs and, in [8], the separations of tensor network decompositions are studied, showing that many of them vanish in the approximate case. Finally, [12] reports a more detailed description of tensor networks from a geometric point of view.

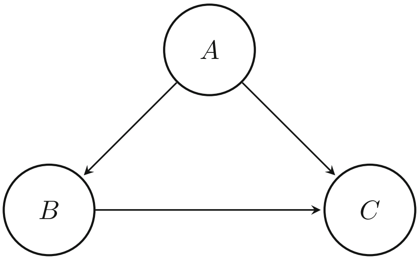
Following a different, simplified approach, tensor networks are associated to probabilistic graphical models (see [16]). The example in Fig. 2 reports a probabilistic graphical model, namely a graph in which the nodes represent random variables and the arcs determine conditional independence assumptions. In the particular case, the nodes are binary random variables. Each node can be associated with a matrix that describes its final state based on the state assumed by its parent nodes. Thus, the joint probability of all nodes in the Bayesian network can be defined with the chain rule as follows:

$$P(A, B, C) = P(A) \times P(B \mid A) \times P(C \mid A, B).$$

In our approach, we will take a point of view very close to this second description of tensor networks. Indeed, we presents a novel analysis of the network, although strictly dependent on the underlying graph, with the goal of determining the final tensor of states obtained from the Boolean activation functions.

In the tensor network analysis we propose, the decision matrix associated with each node corresponds to a logical connective with a matrix of only zeros and ones

$P(A=F)$	$P(A=T)$
0.5	0.5



A	$P(B=F)$	$P(B=F)$
F	0.5	0.5
T	0.9	0.1

A B	$P(B=F)$	$P(B=F)$
F F	1.0	0.0
T F	0.1	0.9
F T	0.1	0.9
T T	0.01	0.99

Fig. 2 Probabilistic graphical model with three nodes A, B and C

equal for all nodes. In addition, we introduce a nondeterministic factor represented by a perturbation matrix that we associate with each arc. The unknowns of the system are, then, the perturbation matrices on the arcs and the Boolean logical operators that determine the output of each node.

Our model allows us to construct the total tensor that describes the final status of the network in Fig. 1 and our analysis shows that the varieties associated with the tensors constructed with different logical connectives are distinct. Hence, given the topology of the graph with three nodes, it is possible to distinguish the logical decision operator of the nodes.

3 Notation

We work with matrices and tensors defined over the complex field.

Definition 3.1 A 2×2 Jukes–Cantor matrix is a matrix of type:

$$M = \begin{pmatrix} \alpha & \beta \\ \beta & \alpha \end{pmatrix},$$

with $\alpha, \beta \in \mathbb{C}$.

Remark 3.2 Jukes–Cantor (JC for short) matrices are closed under the product. Invertible JC matrices form a subgroup of $GL_{\mathbb{C}}(2)$.

JC matrices are often associated with a perturbation of a binary signal. From this point of view, sometimes it is required that the JC matrix satisfies $\alpha + \beta = 1$, and that

Table 1 The sixteen logical operators

	Matrix	Symbol	Nickname	Notes
(1)	$\begin{pmatrix} 0 & 0 \\ 0 & 0 \end{pmatrix}$	$\perp$	Null	Always returns 0
(2)	$\begin{pmatrix} 0 & 0 \\ 0 & 1 \end{pmatrix}$	$\wedge, \forall$	and, forall	Returns 1 when both entries are 1
(3)	$\begin{pmatrix} 0 & 0 \\ 1 & 0 \end{pmatrix}$	$>$	More	Returns 1 when $A > B$
(4)	$\begin{pmatrix} 0 & 0 \\ 1 & 1 \end{pmatrix}$	A	A	Returns the value of A
(5)	$\begin{pmatrix} 0 & 1 \\ 0 & 0 \end{pmatrix}$	$<$	Less	Returns 1 when $A < B$
(6)	$\begin{pmatrix} 0 & 1 \\ 0 & 1 \end{pmatrix}$	B	B	Returns the value of B
(7)	$\begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}$	$\neq$	Not equal	Returns 1 when A, B are different
(8)	$\begin{pmatrix} 0 & 1 \\ 1 & 1 \end{pmatrix}$	$\vee, \exists$	or, exists	Returns 1 when at least one entry is 1
(9)	$\begin{pmatrix} 1 & 0 \\ 0 & 0 \end{pmatrix}$	$\nexists$	nor	Returns 1 when both A, B are 0
(10)	$\begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}$	$=$	Equal	Returns 1 when A, B are equal
(11)	$\begin{pmatrix} 1 & 0 \\ 1 & 0 \end{pmatrix}$	$\neg B$	Not B	Returns 1 when $B = 0$
(12)	$\begin{pmatrix} 1 & 0 \\ 1 & 1 \end{pmatrix}$	$\Leftarrow$	Is implied	Returns 1 when $A \geq B$
(13)	$\begin{pmatrix} 1 & 1 \\ 0 & 0 \end{pmatrix}$	$\neg A$	Not A	Returns 1 when $A = 0$
(14)	$\begin{pmatrix} 1 & 1 \\ 0 & 1 \end{pmatrix}$	$\Rightarrow$	Implies	Returns 1 when $A \leq B$
(15)	$\begin{pmatrix} 1 & 1 \\ 1 & 0 \end{pmatrix}$	$\mid$	nand	Returns 1 unless both A, B are 0
(16)	$\begin{pmatrix} 1 & 1 \\ 1 & 1 \end{pmatrix}$	$\top$	Yes	Always returns 1

α, β are both real and positive. We obtain stochastic JC matrices, in which α is the probability that the signal is correctly transmitted between the nodes, while β is the probability that the signal is changed during the transmission.

We will not use the previous restrictions for JC matrices, since they are not influential in our geometrical analysis.

Definition 3.3 A binary logical operator, or connective, is a function $\Delta : \mathbb{Z}_2^2 \to \mathbb{Z}_2$.

There are $2^4 = 16$ total binary logical operators, corresponding to the number of possible arrangements with repetition of the set of two elements $\{0, 1\}$. According to

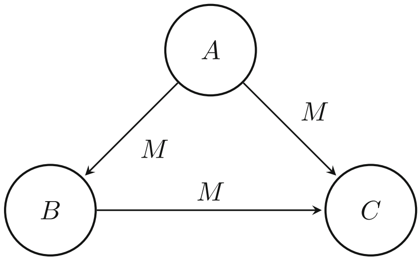


Fig. 3 Neural network with three nodes A, B and C with the associated Jukes–Cantor perturbation matrices

[15], each logical operator is usually represented by a 2×2 logical matrix:

$$\Delta = \begin{pmatrix} \Delta(0,0) & \Delta(0,1) \\ \Delta(1,0) & \Delta(1,1) \end{pmatrix} = \begin{pmatrix} q_{0,0} & q_{0,1} \\ q_{1,0} & q_{1,1} \end{pmatrix},$$

with $q_{i,j} = 0, 1, \forall i, j = 0, 1$.

Table 1 reports the sixteen logical operators. Every logical operator Δ has a dual $\Delta^\vee$ defined by $\Delta^\vee(i, j) \neq \Delta(i, j)$. For instance, 'more' is the dual of 'implies'. In Table 1, the dual of the operator (n) is the operator (17-n), e.g. the dual of (11) is (6).

4 A simple neural network

We will consider a tensor-theoretical description of the neural network presented in Fig. 3. The neural network consists of three nodes A, B and C, referred to as neurons, which are connected as in the figure. Specifically, node A transmits a binary signal, 0 or 1, to nodes B and C. In turn, node B sends the signal it has received from node A to node C. Signals transmitted between nodes are exposed to interference described by perturbation matrices, that we consider all equal to a fixed Jukes–Cantor matrix as follows:

$$M = \begin{pmatrix} \alpha & \beta \\ \beta & \alpha \end{pmatrix}.$$

In the network, node C receives two signals, sent by both nodes A, B. Then, C determines its final state by applying one logical operator Δ to the signals it receives.

The analysis of the neural network is carried out using the Algebraic geometry approach introduced in [3]: the states of the nodes in the network are reported in a tensor of type $2 \times 2 \times \cdots \times 2$ (n times, where n is the number of nodes in the network, in our case equal to 3). The tensor is thus seen as a distribution over the variables of the network.

The tensor T obtained is a function of:

- the initial distribution on node A, which is a vector $D = (a, b)$ that we can freely choose in $\mathbb{C}^2$;
- the entries of the fixed perturbation JC matrix M;
- the logical operator Δ.

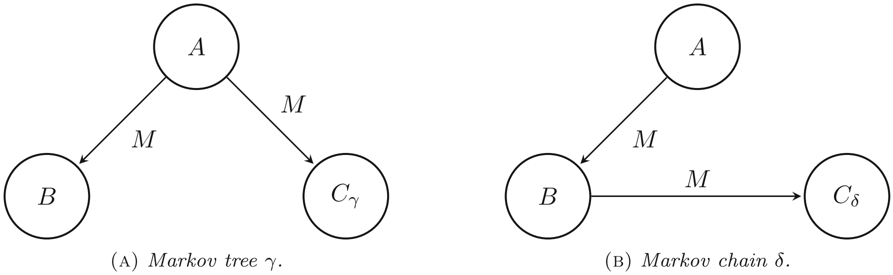


Fig. 4 Trees in which the neural network can be splitted in order to study the output on node C. In **A**, node A sends the signal to node B and node C. In **B**, node A sends the signal to node B which, in turn, transmits it to node C

We will denote with $T^D_\Delta(\alpha, \beta)$ the resulting tensors.

For each choice of logical operator Δ (and for each choice of the initial distribution D) a model M^D_Δ (i.e. a subset of the tensor space) is thus obtained:

$$M^D_\Delta = \{T^D_\Delta(\alpha, \beta) : \alpha, \beta \in \mathbb{C}\}.$$

The resulting models are parametric models, in which the parameters correspond to the coefficients of the Jukes–Cantor matrices. In order to determine the tensors associated to a fixed operator Δ, we will divide the network in Fig. 3 in two simpler networks (Fig. 4).

The first subnetwork, that we call γ, is described in Fig. 4A: node C_γ and node B interpret a signal received directly from node A, and B, C_γ are independent of each other. In other words, γ is a Markov tree with root A and leaves B, C_γ. The distribution on Markov trees is well known. Depending on the initial distribution $D = (a, b)$ on A and the matrix M, we obtain $2 \times 2 \times 2$ tensors of type T_γ as in the System of Eqs. 1. Tensors of type T_γ are represented in Fig. 5.

$$\begin{cases} T_\gamma(0,0,0) = a\alpha^2 \\ T_\gamma(0,0,1) = a\alpha\beta \\ T_\gamma(0,1,0) = a\alpha\beta \\ T_\gamma(0,1,1) = a\beta^2 \\ T_\gamma(1,0,0) = b\beta^2 \\ T_\gamma(1,0,1) = b\alpha\beta \\ T_\gamma(1,1,0) = b\alpha\beta \\ T_\gamma(1,1,1) = b\alpha^2. \end{cases} \tag{1}$$

The subnetwork δ in Fig. 4B is equivalent to a Markov chain in which the signal is transmitted from node A to the final node C_δ passing through node B. Also the distribution on Markov chains is well known. Depending on the initial distribution

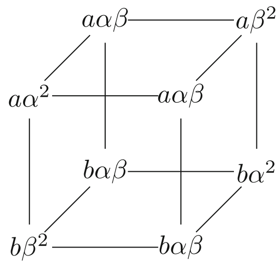


Fig. 5 Tensor of type T_γ, given the initial distribution $D = (a, b)$ on A and the perturbation matrix M

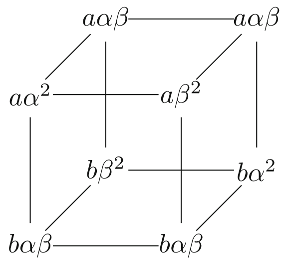


Fig. 6 Tensor of type T_δ, given the initial distribution $D = (a, b)$ on A and the perturbation matrix M

$D = (a, b)$ on A and the matrix M, we obtain $2 \times 2 \times 2$ tensors of type T_δ as in the System of Eq. 2. Tensors of type T_δ are represented in Fig. 6.

$$\begin{cases} T_\delta(0,0,0) = a\alpha^2 \\ T_\delta(0,0,1) = a\alpha\beta \\ T_\delta(0,1,0) = a\beta^2 \\ T_\delta(0,1,1) = a\alpha\beta \\ T_\delta(1,0,0) = b\alpha\beta \\ T_\delta(1,0,1) = b\beta^2 \\ T_\delta(1,1,0) = b\alpha\beta \\ T_\delta(1,1,1) = b\alpha^2. \end{cases} \tag{2}$$

Remark 4.1 The two tensors T_γ, T_δ, together, determine the tensor of distribution T' on the network defined by A, B, C_γ, C_δ, in which $T'(i, j, k, l)$ corresponds to the cases in which, in the original network, A emits i, B emits j, C receives k from A and l from B.

Once the two distributions have been established, in order to find the distribution of the original network, we must combine them using the choice of a logical operator Δ, which determines the final state of C.

Example 4.2 Assume for instance that C uses the logical operator 'and', and let us determine the final tensor T of the network.

Consider $T(0, 1, 0)$ and $T(0, 1, 1)$. They concern the case in which A sends 0 and B sends 1. From the two tensors T_γ and T_δ we see that C receives from A directly

Table 2 Distribution of the signals received by node C in the case of the logical operator 'and': $T(0, 1, 0)$ and $T(0, 1, 1)$

$\gamma\backslash\delta$	0	1
0	$a^2\alpha\beta^3$	$a^2\alpha^2\beta^2$
1	$a^2\beta^4$	$a^2\alpha\beta^3$

(subnetwork γ) $a\alpha\beta$ times 0 and $a\beta^2$ times 1, while C receives from B (subnetwork δ) $a\beta^2$ times 0 and $a\alpha\beta$ times 1. Since we are assuming that the two paths are independent, once A, B are fixed, then the distribution of what C receives is described in Table 2.

Thus the reaction of C, using the operator 'and', is to assume $a^2\alpha\beta^3 + a^2\alpha^2\beta^2 + a^2\alpha^2\beta^2$ times the state 0 and $a^2\beta^4$ times the state 1.

$$T(0, 1, 0) = a^2\alpha\beta^3 + a^2\alpha^2\beta^2 + a^2\beta^4;$$
$$T(0, 1, 1) = a^2\alpha\beta^3.$$

Remark 4.3 The calculation introduced in Example 4.2 can be generalised. If Δ is a logical operator, the associated tensor $T = T^D_\Delta(\alpha, \beta)$ can be computed as follows. First, we construct the table $E_{ij}(s, t)$ that determines the distribution of the two inputs of C when $A = i$ and $B = j$. The table is obtained by the following formula:

$$E_{ij}(s, t) = T_\gamma(i, j, s)T_\delta(i, j, t).$$

Then, the tensor T of the distribution is given by:

$$T(i, j, 0) = \sum_{s,t=0,1} E_{ij}(s, t)(1 - \Delta(s, t));$$
$$T(i, j, 1) = \sum_{s,t=0,1} E_{ij}(s, t)\Delta(s, t).$$

Note that $T(i, j, 0)$ is the sum of the values of $E_{i,j}$ corresponding to the positions where Δ has 0, while $T(i, j, 1)$ is the sum of the values of $E_{i,j}$ corresponding to the positions where Δ has 1.

Example 4.4 Continuing Example 4.2, we can complete the description of the tensors that can arise from the choice of the logical operator 'and'.

In other words, the model is represented by the variety described in space of $2\times2\times2$ tensors T (Fig. 7) by the following parametric equations:

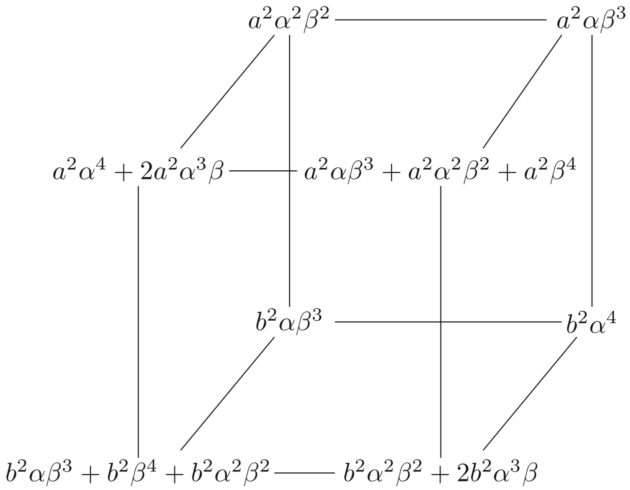


Fig. 7 Variety described in space of $2 \times 2 \times 2$ tensors T

$$\begin{cases} T(0,0,0) = a^2\alpha^4 + 2a^2\alpha^3\beta \\ T(0,0,1) = a^2\alpha^2\beta^2 \\ T(0,1,0) = a^2\alpha\beta^3 + a^2\alpha^2\beta^2 + a^2\beta^4 \\ T(0,1,1) = a^2\alpha\beta^3 \\ T(1,0,0) = b^2\alpha\beta^3 + b^2\beta^4 + b^2\alpha^2\beta^2 \\ T(1,0,1) = b^2\alpha\beta^3 \\ T(1,1,0) = b^2\alpha^2\beta^2 + 2b^2\alpha^3\beta \\ T(1,1,1) = b^2\alpha^4. \end{cases} \tag{3}$$

With a similar procedure, we can write parametric equations for the models associated to the choice of any of the 16 logical operators Δ. The parametric equations are listed in the Table 3.

Remark 4.5 We notice that, for any logical operator Δ, the parametric equations of M_Δ are algebraic, homogeneous, of the same degree. It follows from the Chow's Theorem (see [3] Theorem 10.6.3) that all the models M_Δ represent projective varieties in the projective space $\mathbb{P} = \mathbb{P}(\mathbb{C}^2 \otimes \mathbb{C}^2 \otimes \mathbb{C}^2)$ of tensors $2 \times 2 \times 2$.

4.1 Comparing models

We can now proceed to compare the projective varieties corresponding to the models M_Δ^D that we obtain from different choices of the operator Δ.

(*) *In the computations of this section we always use* $D = (1, 1)$.

Then, in this section, we will often drop the superscript D in the symbols.

We explain the procedure by analysing in detail one example: the operators Δ_2= 'and', Δ_8 = 'or'. The parametric expressions that define the two models M_{Δ_2} and M_{Δ_8} can be found in Table 1, but they are not immediately suitable for the comparison.

Table 3 The parametric equations of the models M_Δ, for each of the 16 logical operators

	Δ	Parametric equations of the model M_Δ^D
(1)	null	$T(0,0,0)=a^2(\alpha^4+2\alpha^3\beta+\alpha^2\beta^2)$ $T(0,0,1)=0$ $T(0,1,0)=a^2(\alpha^2\beta^2+2\alpha\beta^3+\beta^4)$ $T(0,1,1)=0$ $T(1,0,0)=b^2(\alpha^2\beta^2+2\alpha\beta^3+\beta^4)$ $T(1,0,1)=0$ $T(1,1,0)=b^2(\alpha^4+2\alpha^3\beta+\alpha^2\beta^2)$ $T(1,1,1)=0$
(2)	and	$T(0,0,0)=a^2(\alpha^4+2\alpha^3\beta)$ $T(0,0,1)=a^2\alpha^2\beta^2$ $T(0,1,0)=a^2(\alpha^2\beta^2+\alpha\beta^3+\beta^4)$ $T(0,1,1)=a^2\alpha\beta^3$ $T(1,0,0)=b^2(\alpha^2\beta^2+\alpha\beta^3+\beta^4)$ $T(1,0,1)=b^2\alpha\beta^3$ $T(1,1,0)=b^2(2\alpha^3\beta+\alpha^2\beta^2)$ $T(1,1,1)=b^2\alpha^4$
(3)	more	$T(0,0,0)=a^2(\alpha^4+\alpha^3\beta+\alpha^2\beta^2)$ $T(0,0,1)=a^2\alpha^3\beta$ $T(0,1,0)=a^2(\alpha^2\beta^2+2\alpha\beta^3)$ $T(0,1,1)=a^2\beta^4$ $T(1,0,0)=b^2(2\alpha\beta^3+\beta^4)$ $T(1,0,1)=b^2\alpha^2\beta^2$ $T(1,1,0)=b^2(\alpha^4+\alpha^3\beta\alpha^2\beta^2)$ $T(1,1,1)=b^2\alpha^3\beta$
(4)	A	$T(0,0,0)=a^2(\alpha^4+\alpha^3\beta)$ $T(0,0,1)=a^2(\alpha^3\beta+\alpha^2\beta^2)$ $T(0,1,0)=a^2(\alpha^2\beta^2+\alpha\beta^3)$ $T(0,1,1)=a^2(\alpha\beta^3+\beta^4)$ $T(1,0,0)=b^2(\alpha\beta^3+\beta^4)$ $T(1,0,1)=b^2(\alpha^2\beta^2+\alpha\beta^3)$ $T(1,1,0)=b^2(\alpha^3\beta+\alpha^2\beta^2)$ $T(1,1,1)=b^2(\alpha^4+\alpha^3\beta)$
(5)	less	$T(0,0,0)=a^2(\alpha^4+\alpha^3\beta+\alpha^2\beta^2)$ $T(0,0,1)=a^2\alpha^3\beta$ $T(0,1,0)=a^2(2\alpha\beta^3+\beta^4)$ $T(0,1,1)=a^2\alpha^2\beta^2$ $T(1,0,0)=b^2(\alpha^2\beta^2+2\alpha\beta^3)$ $T(1,0,1)=b^2\beta^4$ $T(1,1,0)=b^2(\alpha^4+\alpha^3\beta+\alpha^2\beta^2)$ $T(1,1,1)=b^2\alpha^3\beta$
(6)	B	$T(0,0,0)=a^2(\alpha^4+\alpha^3\beta)$ $T(0,0,1)=a^2(\alpha^3\beta+\alpha^2\beta^2)$ $T(0,1,0)=a^2(\alpha\beta^3+\beta^4)$ $T(0,1,1)=a^2(\alpha^2\beta^2+\alpha\beta^3)$ $T(1,0,0)=b^2(\alpha^2\beta^2+\alpha\beta^3)$ $T(1,0,1)=b^2(\alpha\beta^3+\beta^4)$ $T(1,1,0)=b^2(\alpha^3\beta+\alpha^2\beta^2)$ $T(1,1,1)=b^2(\alpha^4+\alpha^3\beta)$
(7)	not equal	$T(0,0,0)=a^2(\alpha^4+\alpha^2\beta^2)$ $T(0,0,1)=2a^2\alpha^3\beta$ $T(0,1,0)=2a^2\alpha\beta^3$ $T(0,1,1)=a^2(\alpha^2\beta^2+\beta^4)$ $T(1,0,0)=2b^2\alpha\beta^3$ $T(1,0,1)=b^2(\alpha^2\beta^2+\beta^4)$ $T(1,1,0)=+b^2(\alpha^4+\alpha^2\beta^2)$ $T(1,1,1)=2b^2\alpha^3\beta$
(8)	or	$T(0,0,0)=a^2\alpha^4$ $T(0,0,1)=a^2(2\alpha^3\beta+\alpha^2\beta^2)$ $T(0,1,0)=a^2\alpha\beta^3$ $T(0,1,1)=a^2(\alpha^2\beta^2+\alpha\beta^3+\beta^4)$ $T(1,0,0)=b^2\alpha\beta^3$ $T(1,0,1)=b^2(\alpha^2\beta^2+\alpha\beta^3+\beta^4)$ $T(1,1,0)=b^2\alpha^2\beta^2$ $T(1,1,1)=b^2(2\alpha^3\beta+\alpha^4)$
(9)	nor	$T(0,0,0)=a^2(2\alpha^3\beta+\alpha^2\beta^2)$ $T(0,0,1)=a^2\alpha^4$ $T(0,1,0)=a^2(\alpha^2\beta^2+\alpha\beta^3+\beta^4)$ $T(0,1,1)=a^2\alpha\beta^3$ $T(1,0,0)=b^2(\alpha^2\beta^2+\alpha\beta^3+\beta^4)$ $T(1,0,1)=b^2\alpha\beta^3$ $T(1,1,0)=b^2(\alpha^4+2\alpha^3\beta)$ $T(1,1,1)=b^2\alpha^2\beta^2$
(10)	equal	$T(0,0,0)=2a^2\alpha^3\beta$ $T(0,0,1)=a^2(\alpha^4+\alpha^2\beta^2)$ $T(0,1,0)=a^2(\alpha^2\beta^2+\beta^4)$ $T(0,1,1)=2a^2\alpha\beta^3$ $T(1,0,0)=b^2(\alpha^2\beta^2+\beta^4)$ $T(1,0,1)=2b^2\alpha\beta^3$ $T(1,1,0)=2b^2\alpha^3\beta$ $T(1,1,1)=b^2(\alpha^4+\alpha^2\beta^2)$
(11)	not B	$T(0,0,0)=a^2(\alpha^3\beta+\alpha^2\beta^2)$ $T(0,0,1)=a^2(\alpha^4+\alpha^3\beta)$ $T(0,1,0)=a^2(\alpha^2\beta^2+\alpha\beta^3)$ $T(0,1,1)=a^2(\alpha\beta^3+\beta^4)$ $T(1,0,0)=b^2(\alpha\beta^3+\beta^4)$ $T(1,0,1)=b^2(\alpha^2\beta^2+\alpha\beta^3)$ $T(1,1,0)=b^2(\alpha^4+\alpha^3\beta)$ $T(1,1,1)=b^2(\alpha^3\beta+\alpha^2\beta^2)$
(12)	is implied	$T(0,0,0)=a^2\alpha^3\beta$ $T(0,0,1)=a^2(\alpha^4+\alpha^3\beta+\alpha^2\beta^2)$ $T(0,1,0)=a^2\alpha^2\beta^2$ $T(0,1,1)=a^2(2\alpha\beta^3+\beta^4)$ $T(1,0,0)=b^2\beta^4$ $T(1,0,1)=b^2(\alpha^2\beta^2+2\alpha\beta^3)$ $T(1,1,0)=b^2\alpha^3\beta$ $T(1,1,1)=b^2(\alpha^4+\alpha^3\beta+\alpha^2\beta^2)$
(13)	not A	$T(0,0,0)=a^2(\alpha^3\beta+\alpha^2\beta^2)$ $T(0,0,1)=a^2(\alpha^4+\alpha^3\beta)$ $T(0,1,0)=a^2(\alpha\beta^3+\beta^4)$ $T(0,1,1)=a^2(\alpha^2\beta^2+\alpha\beta^3)$ $T(1,0,0)=b^2(\alpha^2\beta^2+\alpha\beta^3)$ $T(1,0,1)=b^2(\alpha\beta^3+\beta^4)$ $T(1,1,0)=b^2(\alpha^4+\alpha^3\beta)$ $T(1,1,1)=b^2(\alpha^3\beta+\alpha^2\beta^2)$
(14)	implies	$T(0,0,0)=a^2\alpha^3\beta$ $T(0,0,1)=a^2(\alpha^4+\alpha^3\beta+\alpha^2\beta^2)$ $T(0,1,0)=a^2\beta^4$ $T(0,1,1)=a^2(\alpha^2\beta^2+2\alpha\beta^3)$ $T(1,0,0)=b^2\alpha^2\beta^2$ $T(1,0,1)=b^2(2\alpha\beta^3+\beta^4)$ $T(1,1,0)=b^2\alpha^3\beta$ $T(1,1,1)=b^2(\alpha^4+\alpha^3\beta+\alpha^2\beta^2)$
(15)	nand	$T(0,0,0)=a^2\alpha^2\beta^2$ $T(0,0,1)=a^2(\alpha^4+2\alpha^3\beta)$ $T(0,1,0)=a^2\alpha\beta^3$ $T(0,1,1)=a^2(\alpha^2\beta^2+\alpha\beta^3+\beta^4)$ $T(1,0,0)=b^2\alpha\beta^3$ $T(1,0,1)=b^2(\alpha^2\beta^2+\alpha\beta^3+\beta^4)$ $T(1,1,0)=b^2\alpha^4$ $T(1,1,1)=b^2(2\alpha^3\beta+\alpha^2\beta^2)$
(16)	yes	$T(0,0,0)=0$ $T(0,0,1)=a^2(\alpha^4+2\alpha^3\beta+\alpha^2\beta^2)$ $T(0,1,0)=0$ $T(0,1,1)=a^2(\alpha^2\beta^2+2\alpha\beta^3+\beta^4)$ $T(1,0,0)=0$ $T(1,0,1)=b^2(\alpha^2\beta^2+2\alpha\beta^3+\beta^4)$ $T(1,1,0)=0$ $T(1,1,1)=b^2(\alpha^4+2\alpha^3\beta+\alpha^2\beta^2)$

Instead, we will use the computer algebra package Singular [9] to transform the parametric equations of M_{Δ_2} (resp. M_{Δ_8}) into equations in the coordinates $x(1), x(2), x(3), x(4), x(5), x(6), x(7)$ and $x(8)$ of the space of tensors $\mathbb{P} = \mathbb{P}(\mathbb{C}^2 \otimes \mathbb{C}^2 \otimes \mathbb{C}^2) = \mathbb{P}^7$.

The transformation is obtained by the function `eliminate` of the package, via the general command:

$$ideal\ I(n) = elim(L(n), 9, 10),$$

where $n = 1, \ldots, 16$ relate to the operator Δ_n, $L(n)$ provides the parametric equations for the model M_{Δ_n} and 9, 10 indicate the parameters α, β to be eliminated. In our case, the commands used are therefore:

$$\begin{gathered} ideal\ I(2) = elim(L(2), 9, 10); \\ ideal\ I(8) = elim(L(8), 9, 10). \end{gathered}$$

The output provides the saturated homogeneous ideal $I(2)$ (resp. $I(8)$), in the coordinate ring $R[x(1), ..., x(8)]$, of M_{Δ_2} (resp. M_{Δ_8}). The package also determines the dimensions of the two models.

$$\begin{cases} I(2)[1] = x(4) - x(6) \\ I(2)[2] = x(3) - x(5) \\ I(2)[3] = x(1) + x(2) - x(7) - x(8) \\ I(2)[4] = x(7)^2 - 3 * x(2) * x(8) - x(5) * x(8) - 3 * x(6) * x(8) \\ I(2)[5] = x(2) * x(7) + x(2) * x(8) - x(5) * x(8) - x(6) * x(8) \\ I(2)[6] = x(6)^2 + x(2) * x(7) - (5) * x(7) + 3 * x(6) * x(7) + 4 * x(2) * x(8) - 4 * x(5) * x(8) + 4* \\ \qquad\qquad * x(6) * x(8) \\ I(2)[7] = x(2) * x(6) + 2 * x(2) * x(7) - x(6) * x(7) - 4 * x(6) * x(8) \\ I(2)[8] = x(2) * x(5) + 16001 * x(2) * x(6) - x(6)^2 + 16001 * x(6) * x(7) \\ I(2)[9] = x(2)^2 - x(2) * x(5) + x(2) * x(6) + x(6)^2 \end{cases}$$

$dim(I(2)) = 4$,

$$\begin{cases} I(8)[1] = x(4) - x(6) \\ I(8)[2] = x(3) - x(5) \\ I(8)[3] = x(1) + x(2) - x(7) - x(8) \\ I(8)[4] = x(5) * x(7) - 16001 * x(7)^2 - 16001 * x(5) * x(8) + 16001 * x(6) * x(8) - \\ \qquad\qquad - 16001 * x(7) * x(8) \\ I(8)[5] = x(2) * x(7) + 3 * x(7)^2 - 2 * x(5) * x(8) \\ I(8)[6] = x(2) * x(6) - x(2) * x(7) - 2 * x(5) * x(7) - x(6) * x(7) - x(7)^2 \\ I(8)[7] = x(5)^2 + x(5) * x(7) - x(6) * x(7) + x(7)^2 \\ I(8)[8] = x(2) * x(5) - x(2) * x(6) + x(2) * x(7) + x(5) * x(7) + x(6) * x(7) - x(7)^2 \\ I(8)[9] = x(2)^2 + 2 * x(2) * x(7) - 3 * x(7)^2 - 4 * x(7) * x(8) \\ I(8)[10] = x(6) * x(7)^2 + 8000 * x(7)^3 - 16001 * x(5)^2 * x(8) + 16001 * x(5) * x(6) * x(8) - 8000* \\ \qquad\qquad * x(5) * x(7) * x(8) - 8001 * x(6) * x(7) * x(8) + 8001 * x(7)^2 * x(8) \end{cases}$$

$dim(I(8)) = 4$.

In order to compare the projective varieties M_{Δ_2} and M_{Δ_8}, i.e. in order to compare the homogeneous ideals, we use the operation `quotient`, which provides an answer. In fact, given two ideals I, J, we get that:

$$I \subset J \qquad \Leftrightarrow \qquad J : I = R = [1].$$

In the specific case of M_{Δ_2} and M_{Δ_8}, Singular computes the following:
$R_{28} = quotient(I(2), I(8))$:

$$\begin{cases} R_{28}[1] = x(4) - x(6) \\ R_{28}[2] = x(3) - x(5) \\ R_{28}[3] = x(1) + x(2) - x(7) - x(8) \\ R_{28}[4] = x(7)^2 - 3 * x(2) * x(8) - x(5) * x(8) - 3 * x(6) * x(8) \\ R_{28}[5] = x(2) * x(7) - x(7)^2 + 4 * x(2) * x(8) + 2 * x(6) * x(8) \\ R_{28}[6] = x(6)^2 - x(5) * x(7) + 3 * x(6) * x(7) + 3 * x(2) * x(8) - 3 * x(5) * x(8) + 5 * x(6) * x(8) \\ R_{28}[7] = x(2) * x(6) - x(6) * x(7) - 2 * x(2) * x(8) + 2 * x(5) * x(8) - 2 * x(6) * x(8) \\ R_{28}[8] = x(2) * x(5) - x(5) * x(7) + 2 * x(6) * x(7) + 2 * x(2) * x(8) - 2 * x(5) * x(8) + +4 * x(6) * \\ \qquad * x(8) \\ R_{28}[9] = x(2)^2 - x(7)^2 + 4 * x(2) * x(8) + 4 * x(6) * x(8) \end{cases}$$

$R_{82} = quotient(I(8), I(2))$:

$$\begin{cases} R_{82}[1] = x(4) - x(6) \\ R_{82}[2] = x(3) - x(5) \\ R_{82}[3] = x(1) + x(2) - x(7) - x(8) \\ R_{82}[4] = x(5) * x(7) - 16001 * x(7)^2 - 16001 * x(5) * x(8) + 16001 * x(6) * x(8) - 16001 * x(7) * \\ \qquad * x(8) \\ R_{82}[5] = x(2) * x(7) + 3 * x(7)^2 - 2 * x(5) * x(8) \\ R_{82}[6] = x(2) * x(6) + 8000 * x(2) * x(7) - x(6) * x(7) - 8000 * x(7)^2 - 16001 * x(5) * x(8) - x(6) * \\ \qquad * x(8) + x(7) * x(8) \\ R_{82}[7] = x(5)^2 - x(6) * x(7) - 16001 * x(7)^2 + 16001 * x(5) * x(8) - 16001 * x(6) * x(8) + 16001 * \\ \qquad * x(7) * x(8) \\ R_{82}[8] = x(2) * x(5) - 16001 * x(2) * x(7) - x(5) * x(7) + 16001 * x(7)^2 - x(5) * x(8) \\ R_{82}[9] = x(2)^2 + 2 * x(2) * x(7) - 3 * x(7)^2 - 4 * x(7) * x(8) \\ R_{82}[10] = x(6) * x(7)^2 + 8000 * x(7)^3 + 16001 * x(5) * x(6) * x(8) + 8001 * x(6) * x(7) * x(8) + 4000 * \\ \qquad * x(7)^2 * x(8) + 12001 * x(5) * x(8)^2 - 12001 * x(6) * x(8)^2 + 12001 * x(7) * x(8)^2 \end{cases}$$

Thus, we can conclude:

Proposition 4.6 *The two models M_{Δ_2} and M_{Δ_8} are projective varieties of dimension 4 in $\mathbb{P}$, and none of the two is contained in the other one.*

We applied the procedure outlined above to compare the models associated to all the operators Δ_i's, $i = 1, \ldots, 16$. The dimensions of the models are listed in the following table:

	Operator Δ	Dimension of M_Δ
(1)	null	4
(2)	and, forall	4
(3)	more	4
(4)	A	4
(5)	less	4
(6)	B	4
(7)	not equal	4
(8)	or, exists	4
(9)	nor	4
(10)	equal	4
(11)	not B	4
(12)	is implied	4
(13)	not A	4
(14)	implies	4
(15)	nand	4
(16)	yes	4

Comparing the saturated ideals of the models, we get that for $i \neq j$ no inclusions $\Delta_i \subset \Delta_j$ occur.

By continuity on the choice of the initial distribution, we obtain the following result.

Theorem 4.7 *For a general choice of the initial distribution D, and for all choices of $i, j = 1, \ldots, 16$, $i \neq j$, the models $M^D_{\Delta_i}, M^D_{\Delta_j}$ corresponding to the logical operators Δ_i, Δ_j, are different, and no inclusions $\Delta_i \subset \Delta_j$ occur.*

Strictly speaking Theorem 4.7 works for the network of Fig. 3, assuming that the perturbation JC matrices of all the edges of the network are the same.

We discuss below what happens if one relaxes the assumptions.

Remark 4.8 *I*f we drop the assumption that the JC matrices of the perturbations in the three edges of the network are the same, then the parametric description of the network changes and the number of parameters can increase. On the other hand, if the JC matrices are continuous functions of two parameters α, β, the previous analysis proves that for a generic choice of the matrices the models associated to the new networks are different when one changes the logical operator used by node C.

Finally, by semi-continuity, the calculation we have performed also proves that given a generic initial distribution, and not necessarily $D = (1, 1)$, all models turn out to be different.

Example 4.9 Consider a wider network, associated with an oriented connected graph in which A is the unique source, B has only one entry and C has exactly two entries. Assume that the perturbation matrices of the edges of the graph all depend on the same two parameters. Then we can extrapolate a subnetwork by considering only the nodes

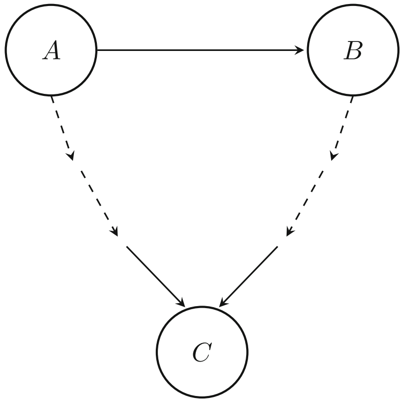


Fig. 8 Network in which A is the unique source, B has only one entry and C has exactly two entries

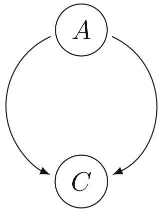


Fig. 9 Network with only two nodes, A and C. A sends a signal to C through two different routes

A, B, C. Theorem 4.7 and Remark 4.8 suggest that, in general, the models associated to different choices of the logical operator used by C to decide its state are different, and no one is contained in any other.

In a concrete example, consider a network as in Fig. 8 and assume that the perturbation matrix of each edge is M. If we consider only the vertices A, B, C, then the subnetwork corresponds to the network discussed in Remark 4.8, in which the perturbation matrices are powers of M, hence their entries are functions of the parameters α, β. The conclusion of Theorem 4.7 implies that, for general networks of this type, the associate models corresponding to different choices of the operator Δ used by C are different.

Example 4.10 The difference between models corresponding to the choice of different logical operators does not hold any more if we simplify the network to the extreme case where there are only two nodes A, C, and A sends a signal to C through two different routes, Fig. 9.

In this case the tensor associated to the total network is a 2×2 matrix, that can be computed with a procedure similar (but simpler) to the one described above for the $2 \times 2 \times 2$ case.

One computes that the operators $\Delta =$ 'more' and the operator $\Delta' =$ 'less', for instance, both determine the model in $\mathbb{C}^2 \otimes \mathbb{C}^2$ corresponding to matrices of type:

$$M = \{\begin{pmatrix} (\alpha^2 + \alpha\beta + \beta^2)a^2 & \alpha\beta a^2 \\ (\alpha^2 + \alpha\beta + \beta^2)b^2 & \alpha\beta b^2 \end{pmatrix}) : \alpha, \beta \in \mathbb{C}\}.$$

Thus they cannot be distinguished by the analysis of the state of the network.

Acknowledgements The author want to warmly thank Luca Chiantini for useful discussions on the subject of the paper.

Funding Open access funding provided by Università degli Studi di Siena within the CRUI-CARE Agreement.

Declarations

Conflict of interest The author declares that she has no Conflict of interest.

References

1. Allman, E.S., Rhodes, J.A.: Phylogenetic ideals and varieties for the general Markov model. Adv. Appl. Math. **40**, 127–148 (2008)
2. Bernardi, A., De Lazzeri, C., Gesmundo, F.: Dimension of tensor network varieties. Commun. Contemp. Math. **25**, 10 (2022)
3. Bocci, C., Chiantini, L.: An Introduction to Algebraic Statistics with Tensors. Springer, New York (2019)
4. Casanellas, M., Fernández-Sánchez, J.: Rank conditions on phylogenetic networks. Ext. Abstr. GEOMVAP **2019**, 65–69 (2021)
5. Christandl, M., Gesmundo, F., Michałek, M., Zuiddam, J.: Border rank nonadditivity for higher order tensors. SIAM J. Matrix Anal. Appl. **42**(2), 503–527 (2021). https://doi.org/10.1137/20M1357366
6. Christandl, M., Lucia, A., Vrana, P., Werner, A.H.: Tensor network representations from the geometry of entangled states. SciPost. Phys. (2020). https://doi.org/10.21468/SciPostPhys.9.3.042
7. Cox, D., Little, J., O'Shea, D.: Ideals, Varieties, and Algorithms: An Introduction to Computational Algebraic Geometry and Commutative Algebra. Springer Graduate Texts in Mathematics (2007)
8. De las Cuevas, G., Klingler, A., Netzer, T.: Approximate tensor decompositions: disappearance of many separations. J. Math. Phys. **62**, 9 (2021). https://doi.org/10.1063/5.0033876
9. Decker, W., Greuel, G.M., Pfister, G., Schönemann, H.: Singular 4-1-2. A computer algebra system for polynomial computations (2018). http://www.singular.uni-kl.de
10. Gross, E., Long, C.: Distinguishing phylogenetic networks. SIAM J. Appl. Algebra Geom. **2**(1), 72–93 (2018). https://doi.org/10.1137/17M1134238
11. Klümper, A., Schadschneider, A., Zittartz, J.: Matrix product ground states for one-dimensional spin-1 quantum antiferromagnets. Europhys. Lett. **24**, 4 (1993)
12. Landsberg, J. M.: Tensors: Asymptotic Geometry and Developments 2016–2018. In: Conference Board of the Mathematical Sciences, Regional Conference Series in Mathematics, vol. 132 (2019)
13. Landsberg, J.M.: Tensors: Geometry and Applications. Graduate Studies in Mathematics, American Mathematical Society, New York (2012)
14. Landsberg, J.M., Qi, Y., Ye, K.: On the geometry of tensor network states. Quantum Inf. Comput. **12**, 254–346 (2012)
15. Li, H., Wang, Y.: Logical matrix factorization with application to topological structure analysis of Boolean network. IEEE Trans. Autom. Control **60**(5), 1380–1385 (2015). https://doi.org/10.1109/TAC.2014.2348216
16. Murphy, M.: An introduction to graphical models. Rap. Tech. **96**, 1–19 (2001)

17. Orús, R.: Tensor networks for complex quantum systems. Nat. Rev. Phys. **1**, 538–550 (2019). https://doi.org/10.1038/s42254-019-0086-7
18. Perez-Garcia, D., Verstraete, F., Wolf, M.M., Cirac, J.I.: Matrix product state representations (2007). Preprint arXiv:quant-ph/0608197