Probabilistic inference with noisy-threshold models based on a CP tensor decomposition☆

Jiří Vomlel and Petr Tichavský

Institute of Information Theory and Automation of the AS CR
Pod vodárenskou věží 4, Prague 8, 182 08, Czech Republic

Abstract

The specification of conditional probability tables (CPTs) is a difficult task in the construction of probabilistic graphical models. Several types of canonical models have been proposed to ease that difficulty. Noisy-threshold models generalize the two most popular canonical models: the noisy-or and the noisy-and. When using the standard inference techniques the inference complexity is exponential with respect to the number of parents of a variable. More efficient inference techniques can be employed for CPTs that take a special form. CPTs can be viewed as tensors. Tensors can be decomposed into linear combinations of rank-one tensors, where a rank-one tensor is an outer product of vectors. Such decomposition is referred to as Canonical Polyadic (CP) or CANDECOMP-PARAFAC (CP) decomposition. The tensor decomposition offers a compact representation of CPTs which can be efficiently utilized in probabilistic inference. In this paper we propose a CP decomposition of tensors corresponding to CPTs of threshold functions, exactly ℓ-out-of-k functions, and their noisy counterparts. We prove results about the symmetric rank of these tensors in the real and complex domains. The proofs are constructive and provide methods for CP decomposition of these tensors. An analytical and experimental comparison with the parent-divorcing method (which also has a polynomial complexity) shows superiority of the CP decomposition-based method. The experiments were performed on subnetworks of the well-known QMRT-DT network generalized by replacing noisy-or by noisy-threshold models.

Keywords: Bayesian networks, Probabilistic inference, Candecomp-Parafac tensor decomposition, Symmetric tensor rank

☆This work was supported by Czech Science Foundation through projects 13-20012S and 102/09/1278.

1. Introduction

Bayesian networks [1, 2, 3] provide a popular framework for modeling and decision-making under uncertainty. Since most of our decisions are based on information that is (at least partially) uncertain, Bayesian networks were applied in many diverse domains. Their fundamental advantage over different frameworks is their ability to divide the modeling problem into two basic stages: first, the structure of the modeled domain is described using a graph and secondly, the numerical values describing the quantitative relationship between model variables are provided. The model is either built by domain experts or automatically learned from collected data; and possibly created using a process that combines both ways. The key property of Bayesian networks that allows them to be applied in domains with up to hundreds of variables is the decomposability of the joint probability distribution they represent.

The structure of a Bayesian network is defined by an acyclic directed graph $G = (V, E)$, where E is the set of directed edges, i.e., $E \subseteq V \times V$. The joint probability of a Bayesian network is defined for all configurations $x = (x_1, \ldots, x_n)$ of discrete variables $X_1, \ldots, X_n$ as

$$P(x_1, \ldots, x_n) \quad = \quad \prod_{i=1}^{n} P(x_i | x_{pa(i)}) \ , \tag{1}$$

where $pa(i)$ denotes the set of parents of node i in the graph $G = (V, E)$, i.e., $pa(i) = \{j, (j \to i) = (j, i) \in E\}$. Using a common shorthand for a formula valid for all configurations of variables we can write formula (1) as

$$P(X_1, \ldots, X_n) \quad = \quad \prod_{i=1}^{n} P(X_i | X_{pa(i)}) \ . \tag{2}$$

Formula (1) allows efficient computations of probabilistic queries

$$P(X_i = x_i | e), \ e = \{X_j = x_j, j \in A\}, \ A \subset V$$

for all values x_i of all $X_i, i \in V \setminus A$. The computational complexity of this task when the popular junction tree method [4, 5] is used is exponential with respect to the size $|C|$ of a largest clique $C \subseteq V$ of the triangulated moralized graph G' of G, see [4, 5] for details. The value $|C| - 1$ is called the treewidth of G'. If the treewidth is not large, then computationally efficient probabilistic inference is possible. This allows the application of Bayesian networks in domains with hundreds of variables, where a naive computation with the full joint probability table would not be tractable.

Unfortunately, the treewidth is large in some applications, for example, some variables $X_i, i \in V$ may have a very large parent set ($|pa(i)| > 10^2$). In such cases the exact inference with the standard junction tree method is not tractable. One solution is to resort to approximate inference methods, e.g., to Monte-Carlo methods [6], the Pearl polytree algorithm [1] applied to Bayesian networks with loops, variational methods [7], inference using probability trees [8, 9], or binary probability trees [10].

Based on our experience with different applications of Bayesian networks we believe that the conditional probability tables (CPTs), which are the basic building blocks of Bayesian networks, may often have a simple structure, for example, they correspond to a noisy functional dependence. This property should be exploited not only when building a Bayesian network model but also during the probabilistic inference.

Various methods that can exploit the local structure of CPTs were proposed. An early example is the Quickscore algorithm [11] exploiting noisy-or relations in the Quick Medical Reference model. Olesen et al. [12] proposed the so-called parent-divorcing method. Heckerman and Breese [13] use a temporal model transformation. Zhang and Poole [14] introduced deputy variables that are used to create heterogeneous factorizations in which the factors are combined either by multiplication or by a combination operator. Takikawa and D'Ambrosio [15] used auxiliary variables, which allowed them to transform an additive factorization into a multiplicative factorization. The additions are achieved by the marginalization of the intermediate variables. Díez and Galán [16] pointed out that the transformation of noisy-max can be done using a single variable.

CP tensor decomposition, called tensor rank-one decomposition in [17], is a generalization of Díez and Galán's decomposition. It is based on the CP decomposition of tensors [18, 19]. Savicky and Vomlel [17] described CP tensor decompositions of several canonical models – noisy-max, noisy-min, noisy-add, noisy-xor.

Previous research reveals that for certain models the computational savings achieved by this transformation are very large – instead of a representation that is exponential with respect to the number of variables in a CPT we get a representation that is only quadratic [17, 20, 21]. The application of the CP tensor decomposition allows application of Bayesian networks in domains where exact probabilistic inference would not otherwise be possible. We believe this may have a substantial impact on the quality of decision-making in complex domains such as: medical decision support systems and health monitoring [12, 22, 23, 24, 25], troubleshooting complex devices [26, 27], etc.

This paper is organized as follows. Conditional probability tables with a local structure are discussed in Section 2. In Section 3 we introduce the necessary tensor notation, define tensors of the exactly ℓ-out-of-k and threshold functions, and present their basic properties. Sections 4 and 5 represent the main original contribution of this paper. We propose methods for the decomposition of tensors of the threshold and exactly ℓ-out-of-k functions in the real and complex domains and prove results about the symmetric rank of these tensors. In Section 6 we analytically compare the CP decomposition and the parent-divorcing method using an exemplary class of models. In Section 7 we present experimental comparisons of the CP decomposition method with the standard junction tree method and the parent-divorcing method. The experiments are performed on a generalized version of the QMR-DT network. In Section 8 we briefly review other methods exploiting local structure of CPTs. We outline how the CP decomposition can be combined with weighted model counting. Major proofs are moved to appendices.

2. Conditional probability tables with a local structure

Canonical models [28] represent a class of CPTs with the local structure being defined either by:

- a deterministic function of the values of the parents (they are called *deterministic models* in [28]),
- a combination of a deterministic part with independent probabilistic influence of each parent variable (called *ICI models* in [28]), or
- a combination of a deterministic part with a probabilistic relationship between the child variable of the deterministic part and its child (called *simple canonical models* in [28]).

For the graph of a deterministic model see Figure 1. The graph of an ICI model contains auxiliary variables $X'_1, \ldots, X'_k$, one for each parent, see Figure 2. The graph of a simple canonical model contains one auxiliary variable Y', see Figure 3.

The joint probability distribution of the Bayesian network in Figure 2 is

$$P(Y|X'_1, \ldots, X'_k) \prod_{i=1}^{k} P(X'_i|X_i)P(X_i) \ ,$$

where the first term $P(Y|X'_1, \ldots, X'_k)$ corresponds to a deterministic function and terms $P(X'_i|X_i)$ to the probabilistic part (often called noise). We can replace the Bayesian network of Figure 2 with a model without auxiliary variables $X'_1, \ldots, X'_k$ (see Figure 1) by marginalizing them out from the Bayesian network.

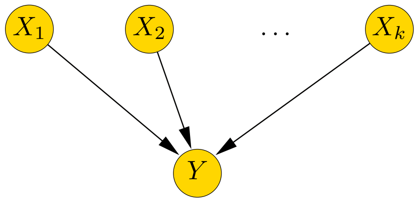

Figure 1: A Bayesian network with a canonical model $P(Y|X_1, \ldots, X_k)$ without auxiliary variables.

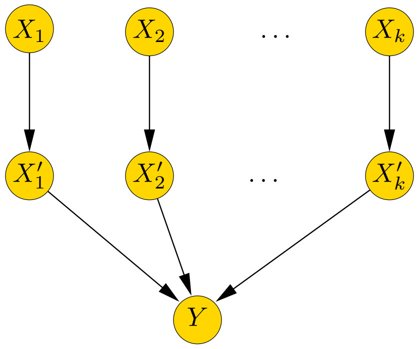

Figure 2: A Bayesian network with a canonical model with explicit deterministic part $P(Y|X'_1, \ldots, X'_k)$ and probabilistic parts $P(X'_i|X_i), i = 1, \ldots, k$.

The values of $P(Y|X_1, \ldots, X_k)$ can be computed from the original model by

$$\begin{aligned} P(Y|X_1, \ldots, X_k) &= \\ &\sum_{X'_1} \ldots \sum_{X'_k} P(Y|X'_1, \ldots, X'_k) \cdot \prod_{i=1}^{k} P(X'_i|X_i) \ . \end{aligned}$$

This model is equivalent to the original one in the sense that it can be used to compute correct marginal and conditional probabilities for any subset of its variables present in both models.

As was suggested in [17] we can rewrite each CPT as a product of two-dimensional potentials[1] $\psi_i, i = 1, \ldots, k$ and ξ:

$$P(Y|X_1, \ldots, X_k) = \sum_{B} \xi(B, Y) \cdot \prod_{i=1}^{k} \psi_i(B, X_i) \ , \tag{3}$$

[1]These potentials take values that are real (or even complex) numbers.

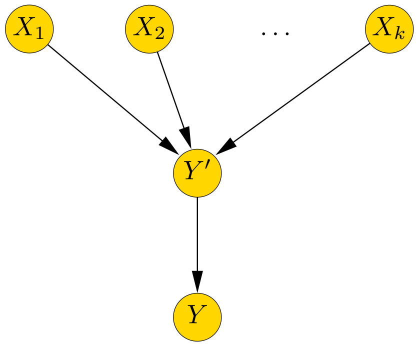

Figure 3: A Bayesian network with a canonical model with a deterministic part $P(Y'|X_1, \dots, X_k)$ and a probabilistic part $P(Y|Y')$.

where B is an auxiliary variable. This transformation can be visualized by the undirected graph given in Figure 4.

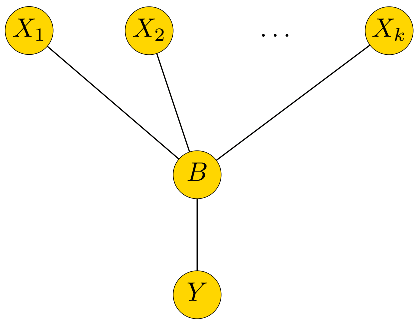

Figure 4: Model of $P(Y|X_1, \dots, X_k)$ after the transformation using auxiliary variable B.

If the state y of variable Y is observed, we can omit the variable Y and the corresponding potential and decompose the CPT $P(y|X_1, \dots, X_k)$ as:

$$P(y|X_1, \dots, X_k) \quad = \quad \sum_B \prod_{i=1}^{k} \psi_i(B, X_i) \quad . \tag{4}$$

This transformation can be visualized by the undirected graph given in Figure 5.

To guarantee either of the above equalities, variable B has to have a certain number of states. Trivially, the equality can always be satisfied if the number of states of B is the product of the number of states of variables $X_1, \dots, X_k$. However, the transformation becomes computationally advan-

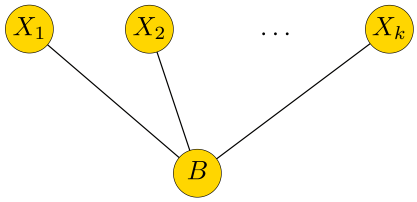

Figure 5: Model of $P(y|X_1, \dots, X_k)$ after the transformation using auxiliary variable B.

tageous if the number of states is substantially lower[2]. Again, this model is equivalent to the original one in the sense that it can be used to compute correct marginal and conditional probabilities for any subset of its variables present in both models.

The above decomposition specified by formula (3) or (4) can be integrated into any inference engine that allows us to work with real-valued tables (potentials). It can be beneficial to perform inference with complex numbers. Although the complexity of addition and multiplication of two complex numbers is higher than of those operations on real numbers, the decompositions based on complex numbers have better numerical stability.

In a junction tree method, the decomposition can be applied as a preprocessing step replacing the moralization step. Instead of connecting all parents of a node (as is done in moralization), one auxiliary node is added. This node is connected to all parents by an undirected edge and, if the child node was not observed, also to the child node. This undirected graph is then triangulated and a junction tree with cliques as its nodes is created. Each table is attached to a clique containing all variables of that table. The computations than proceed in the same way as in the standard junction tree method. Note that in [20] results were reported of numerical experiments based on an experimental R implementation[3] of the lazy propagation [29] exploiting approximate decomposition based on formula (4) in the real domain.

---

[2] As we will later see, this is the case for the CPTs of exactly ℓ-out-of-k and threshold functions

[3] R: A Language and Environment for Statistical Computing, `http://www.R-project.org`

3. Tensors of the exactly ℓ-out-of-k and threshold functions

Each probability table can be understood as a tensor $\boldsymbol{\mathcal{A}}$. A tensor is simply a mapping[4] $\boldsymbol{\mathcal{A}} : \mathbb{I} \rightarrow \mathbb{R}$ or $\mathbb{I} \rightarrow \mathbb{C}$, where $\mathbb{I} = I_1 \times \ldots \times I_k$, k is a natural number called the order of tensor $\boldsymbol{\mathcal{A}}$, and $I_j, j = 1, \ldots, k$ are index sets. Typically, I_j are sets of integers of cardinality n_j. Then we can say that tensor $\boldsymbol{\mathcal{A}}$ has dimensions $n_1, \ldots, n_k$. All index sets considered in this paper will be $I_j = \{0, 1\}, j = 1, \ldots, k$.

Example 1 (Tensor of order 4 with all dimensions being 2)**.** We can visualize tensors using nested matrices with successive dimensions alternating between rows and columns.

$$\boldsymbol{\mathcal{A}} = \left(\begin{array}{cc} \left(\begin{array}{cc} 1 & 0 \\ 0 & 0 \end{array} \right) & \left(\begin{array}{cc} 0 & 0 \\ 0 & 0 \end{array} \right) \\ \left(\begin{array}{cc} 0 & 0 \\ 0 & 0 \end{array} \right) & \left(\begin{array}{cc} 0 & 0 \\ 0 & 0 \end{array} \right) \end{array} \right)$$

Probability table $P(y|X_1, \ldots, X_k)$ defines tensor $\boldsymbol{\mathcal{A}}$ as

$$\boldsymbol{\mathcal{A}}_{i_1,\ldots,i_k} = P(y|X_1 = x_{i_1}, \ldots, X_k = x_{i_k}),$$

for all combinations of states $(x_{i_1}, \ldots, x_{i_k}), (i_1, \ldots, i_k) \in I_1 \times \ldots \times I_k$ of variables $X_1, \ldots, X_k$.

Definition 1. Tensor $\boldsymbol{\mathcal{A}}$ has rank one in real or complex domain, respectively, if it can be written as an outer product of vectors, i. e.,

$$\boldsymbol{\mathcal{A}} = \boldsymbol{a}_1 \otimes \ldots \otimes \boldsymbol{a}_k$$

with the outer product being defined for all $(i_1, \ldots, i_k) \in I_1 \times \ldots \times I_k$ as

$$\boldsymbol{\mathcal{A}}_{i_1,\ldots,i_k} = a_{1,i_1} \cdot \ldots \cdot a_{k,i_k} \ ,$$

where $\boldsymbol{a}_j = (a_{j,i})_{i \in I_j}$, $j = 1, \ldots, k$ are real or complex valued vectors, respectively.

Example 2. The tensor $\boldsymbol{\mathcal{A}}$ from Example 1 has rank one since

$$\boldsymbol{\mathcal{A}} = (1, 0) \otimes (1, 0) \otimes (1, 0) \otimes (1, 0) \ .$$

[4] In this paper we consider decompositions of real-valued tensors, but the tensors in the decomposition can actually be real-valued or complex-valued.

Definition 2. Let $\mathcal{A}, \mathcal{B}$ be two tensors of the same order k and the same dimensions. Then their sum $\mathcal{C} = \mathcal{A} + \mathcal{B}$ is defined for all $(i_1, \ldots, i_k)$ as

$$\mathcal{C}_{i_1,\ldots,i_k} = \mathcal{A}_{i_1,\ldots,i_k} + \mathcal{B}_{i_1,\ldots,i_k} .$$

Definition 3. The rank of a tensor $\mathcal{A}$, denoted $rank(\mathcal{A})$, is the minimal r such that $\mathcal{A}$ can be decomposed as a linear combination of rank-one tensors:

$$\mathcal{A} = \sum_{i=1}^{r} b_i \cdot \boldsymbol{a}_{i,1} \otimes \ldots \otimes \boldsymbol{a}_{i,k} . \tag{5}$$

The decomposition of a tensor $\mathcal{A}$ to tensors of rank one that sum up to $\mathcal{A}$ is called CP tensor decomposition.

It was observed in [17] that the minimum number of states of B in the decomposition defined by formula (4) equals the rank of tensor $\mathcal{A}$. The decomposition of tensors into the form corresponding to the right hand side of formula (4) has been studied for more than forty years [19, 18] and it is now known as Canonical Polyadic (CP) or CANDECOMP-PARAFAC (CP) decomposition. In [30] it is called an outer-product decomposition.

In this paper we deal with conditional probability tables representing two specific canonical models – deterministic threshold and exact functions and their noisy counterparts. An (ℓ, k)-threshold function is a function of k binary arguments that yields the value 1 if at least ℓ out of its k arguments take value 1 – otherwise the function value is zero, the exactly ℓ-out-of-k function takes value 1 if exactly ℓ out of its k arguments take value 1. The noisy version allows noise at the inputs of the function. In the model of Figure 2 the noise is represented by conditional probability tables $P(X'_i|X_i)$.

The noisy-threshold models represent a generalization of two popular models – noisy-or and noisy-and. They constitute an alternative to noisy-or and noisy-and in case they are too rough. The conditional probability tables of the noisy-threshold models appear, for example, in medical applications of Bayesian networks [24, 23, 31, 32]. The noise on the parent variables cannot increase the rank, see [17, Theorem 6]. Therefore all results about rank of deterministic tensors represent an upper bound on rank of their noisy counterparts. Also, it is easy to combine the noise with the CP decomposition of a deterministic tensor to get a CP decomposition of its noisy counterpart. For details, see [17]. In this paper we present results about tensors of deterministic canonical models since they are easily extendable to ICI models and simple canonical models.

Definition 4. Let p be a propositional formula taking values $true$ and $false$. Function δ is defined as

$$\delta(p) \quad = \quad \begin{cases} 1 & \text{if } p = true \\ 0 & \text{otherwise.} \end{cases}$$

Definition 5. Tensor $\mathcal{S}(\ell, k) : \{0,1\}^k \to \{0,1\}$ represents an exactly ℓ-out-of-k function if it holds for $(i_1, \ldots, i_k) \in \{0,1\}^k$:

$$\mathcal{S}_{i_1,\ldots,i_k}(\ell, k) \quad = \quad \delta(i_1 + \ldots + i_k = \ell)$$

Example 3.

$$\mathcal{S}(2,4) \quad = \quad \begin{pmatrix} \begin{pmatrix} 0 & 0 \\ 0 & 1 \end{pmatrix} & \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix} \\ \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix} & \begin{pmatrix} 1 & 0 \\ 0 & 0 \end{pmatrix} \end{pmatrix}$$

Definition 6. Tensor $\mathcal{T}(\ell, k) : \{0,1\}^k \to \{0,1\}$ represents an (ℓ, k)-threshold function if it holds for $(i_1, \ldots, i_k) \in \{0,1\}^k$:

$$\mathcal{T}_{i_1,\ldots,i_k}(\ell, k) \quad = \quad \delta(i_1 + \ldots + i_k \geq \ell)$$

Example 4.

$$\mathcal{T}(2,4) \quad = \quad \begin{pmatrix} \begin{pmatrix} 0 & 0 \\ 0 & 1 \end{pmatrix} & \begin{pmatrix} 0 & 1 \\ 1 & 1 \end{pmatrix} \\ \begin{pmatrix} 0 & 1 \\ 1 & 1 \end{pmatrix} & \begin{pmatrix} 1 & 1 \\ 1 & 1 \end{pmatrix} \end{pmatrix}$$

Remark. Note that the tensor of the logical "or" function is a special case of the tensor of the (ℓ, k)-threshold function for $\ell = 1$. Similarly, the tensor of the logical "and" function is a special case of the tensor of the (ℓ, k)-threshold function for $\ell = k$.

It is straightforward to see that the exactly ℓ-out-of-k tensors and the (ℓ, k)-threshold tensors are related as

$$\mathcal{T}(\ell, k) = \sum_{m=\ell}^{k} \mathcal{S}(m, k) \ . \tag{6}$$

The above tensors correspond to the CPTs where the value of Y was observed to be 1. Next, we shall consider extended versions of the above tensors that correspond to CPTs with Y being unobserved. Such CPTs are part of the model if Y has a successor with evidence that is not its child[5]. This applies, for example, to the case where observations of the state of Y are noisy, which can be modeled by an auxiliary variable Y' being deterministically related to $X_1, \ldots, X_k$ with a child Y of Y' probabilistically related to Y' by a CPT $P(Y|Y')$. These models are called simple canonical models in [28], see Figure 3.

Definition 7. Tensor $\overline{\boldsymbol{\mathcal{S}}}(\ell, k) : \{0,1\}^{k+1} \to \{0,1\}$ is an extended tensor of an exactly ℓ-out-of-k function if it holds for $(i_0, i_1, \ldots, i_k) \in \{0,1\}^{k+1}$:

$$\overline{\boldsymbol{\mathcal{S}}}_{i_0,i_1,\ldots,i_k}(\ell, k) = \begin{cases} \boldsymbol{\mathcal{S}}_{i_1,\ldots,i_k}(\ell, k) & \text{if } i_0 = 1 \\ 1 - \boldsymbol{\mathcal{S}}_{i_1,\ldots,i_k}(\ell, k) & \text{if } i_0 = 0 \ . \end{cases} \tag{7}$$

Example 5.

$$\overline{\boldsymbol{\mathcal{S}}}(2,4) = \begin{pmatrix} \begin{pmatrix} \begin{pmatrix} 1 & 1 \\ 1 & 0 \end{pmatrix} & \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} \\ \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} & \begin{pmatrix} 0 & 1 \\ 1 & 1 \end{pmatrix} \end{pmatrix} \\ \begin{pmatrix} \begin{pmatrix} 0 & 0 \\ 0 & 1 \end{pmatrix} & \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix} \\ \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix} & \begin{pmatrix} 1 & 0 \\ 0 & 0 \end{pmatrix} \end{pmatrix} \end{pmatrix}$$

Definition 8. Tensor $\overline{\boldsymbol{\mathcal{T}}}(\ell, k) : \{0,1\}^{k+1} \to \{0,1\}$ is an extended tensor of an (ℓ, k)-threshold function if it holds for $(i_0, i_1, \ldots, i_k) \in \{0,1\}^{k+1}$:

$$\overline{\boldsymbol{\mathcal{T}}}_{i_0,i_1,\ldots,i_k}(\ell, k) = \begin{cases} \boldsymbol{\mathcal{T}}_{i_1,\ldots,i_k}(\ell, k) & \text{if } i_0 = 1 \\ 1 - \boldsymbol{\mathcal{T}}_{i_1,\ldots,i_k}(\ell, k) & \text{if } i_0 = 0 \ . \end{cases} \tag{8}$$

Example 6.

$$\overline{\boldsymbol{\mathcal{T}}}(2,4) = \begin{pmatrix} \begin{pmatrix} \begin{pmatrix} 1 & 1 \\ 1 & 0 \end{pmatrix} & \begin{pmatrix} 1 & 0 \\ 0 & 0 \end{pmatrix} \\ \begin{pmatrix} 1 & 0 \\ 0 & 0 \end{pmatrix} & \begin{pmatrix} 0 & 0 \\ 0 & 0 \end{pmatrix} \end{pmatrix} \\ \begin{pmatrix} \begin{pmatrix} 0 & 0 \\ 0 & 1 \end{pmatrix} & \begin{pmatrix} 0 & 1 \\ 1 & 1 \end{pmatrix} \\ \begin{pmatrix} 0 & 1 \\ 1 & 1 \end{pmatrix} & \begin{pmatrix} 1 & 1 \\ 1 & 1 \end{pmatrix} \end{pmatrix} \end{pmatrix} .$$

[5] If the value of Y is unobserved and Y has no observed successor then its CPT and CPTs of its successors can be discarded from the model – they are *barren nodes* [2].

Definition 9. Tensor $\boldsymbol{\mathcal{A}} : \{0,1\}^k \to \mathbb{R}$ is symmetric if for $(i_1, \ldots, i_k) \in \{0,1\}^k$ it holds that

$$\boldsymbol{\mathcal{A}}_{i_1,\ldots,i_k} = \boldsymbol{\mathcal{A}}_{i_{\sigma(1)},\ldots,i_{\sigma(k)}} ,$$

for any permutation σ of $\{1, \ldots, k\}$.

Tensors $\boldsymbol{\mathcal{S}}(\ell, k)$, $\boldsymbol{\mathcal{T}}(\ell, k)$ are symmetric, and $\overline{\boldsymbol{\mathcal{S}}}(\ell, k)$, $\overline{\boldsymbol{\mathcal{T}}}(\ell, k)$ are partially symmetric with respect to the last k coordinates. We concentrate on CP decomposition of the former symmetric tensors first. We will present results for tensors $\boldsymbol{\mathcal{S}}(\ell, k)$ and $\boldsymbol{\mathcal{T}}(\ell, k)$ that correspond to evidence $Y = 1$. The decompositions of tensors corresponding to evidence $\mathrm{Y} = 0$ can be derived similarly. CP decomposition of the extended tensors will be studied in Section 5. For symmetric tensors it is possible to define a symmetric rank as follows:

Definition 10. The symmetric rank $srank(\boldsymbol{\mathcal{A}})$ of a tensor $\boldsymbol{\mathcal{A}}$ is the minimum number of symmetric rank-one tensors such that their linear combination equals $\boldsymbol{\mathcal{A}}$

$$\begin{aligned} \boldsymbol{\mathcal{A}} &= \sum_{i=1}^{r} b_i \cdot \underbrace{\boldsymbol{a}_i \otimes \ldots \otimes \boldsymbol{a}_i}_{k \text{ copies}} \\ &= \sum_{i=1}^{r} b_i \cdot \boldsymbol{a}_i^{\otimes k} , \end{aligned} \tag{9}$$

where we adopt the notation $\boldsymbol{a}^{\otimes k} = \underbrace{\boldsymbol{a} \otimes \ldots \otimes \boldsymbol{a}}_{k \text{ copies}}$ of [30].

Remark. It is not known whether it holds for symmetric tensors $\boldsymbol{\mathcal{A}}$ that $rank(\boldsymbol{\mathcal{A}}) = srank(\boldsymbol{\mathcal{A}})$. In [30] it is conjectured that the equality holds and the equality is proved for a restricted class of tensors.

We will use the following property of symmetric tensors.

Definition 11. Let $\boldsymbol{\mathcal{A}}$ be a symmetric tensor, $\boldsymbol{\mathcal{A}} : \{0,1\}^k \to \mathbb{R}$. Then we say that tensor $\boldsymbol{\mathcal{A}}' : \{0,1\}^k \to \mathbb{R}$ is constructed from $\boldsymbol{\mathcal{A}}$ by swapping all its coordinates if

$$\boldsymbol{\mathcal{A}}'_{i_1,\ldots,i_k} = \boldsymbol{\mathcal{A}}_{\delta(i_1=0),\ldots,\delta(i_k=0)} ,$$

where $\delta(i = j)$ is defined in Definition 4.

Example 7. Consider $\mathcal{T}(\ell, k)$ from Example 4. Then $\mathcal{T}'(\ell, k)$ constructed from $\mathcal{T}(\ell, k)$ by swapping all its coordinates is

$$\mathcal{T}'(2,4) = \begin{pmatrix} \begin{pmatrix} 1 & 1 \\ 1 & 1 \end{pmatrix} & \begin{pmatrix} 1 & 1 \\ 1 & 0 \end{pmatrix} \\ \begin{pmatrix} 1 & 1 \\ 1 & 0 \end{pmatrix} & \begin{pmatrix} 1 & 0 \\ 0 & 0 \end{pmatrix} \end{pmatrix} .$$

Lemma 1. *Assume a symmetric tensor $\mathcal{A}$ of order k and having all dimensions equal to two. Let $\mathcal{A}'$ be the tensor constructed from $\mathcal{A}$ by swapping all its coordinates. Then it holds that*

$$srank(\mathcal{A}') = srank(\mathcal{A}) .$$

Proof. Note that

$$\mathcal{A} = \sum_{i=1}^{r} (u_i, v_i)^{\otimes k} \Leftrightarrow \mathcal{A}' = \sum_{i=1}^{r} (v_i, u_i)^{\otimes k}$$

for any $u_i, v_i \in \mathbb{R}$ or $u_i, v_i \in \mathbb{C}$. □

An immediate consequence of Lemma 1 is Corollary 1.

Corollary 1.

$$srank(\mathcal{S}(\ell, k)) = srank(\mathcal{S}(k - \ell, k)) .$$

Each symmetric tensor $\mathcal{A} : \{0, 1\}^k \to \mathbb{R}$ of rank-one can be written as

$$\mathcal{A} = \begin{cases} (0, a)^{\otimes k} & \text{if } \mathcal{A}_{0,\ldots,0} = 0 \\ b \cdot (1, a)^{\otimes k} & \text{otherwise,} \end{cases} \tag{10}$$

where $a, b \in \mathbb{R}$ or $a, b \in \mathbb{C}$.

In the following lemmas we treat the border cases with symmetric rank equal to one.

Lemma 2. The symmetric rank of tensors $\mathcal{S}(\ell, k)$ representing the respective exactly ℓ-out-of-k function for $\ell \in \{0, k\}$ is one.

Proof.

$$\begin{aligned} \mathcal{S}(k, k) &= (0, 1)^{\otimes k} \\ \mathcal{S}(0, k) &= (1, 0)^{\otimes k} . \end{aligned}$$

□

Lemma 3. The symmetric rank of tensors $\mathcal{T}(\ell, k)$ representing the respective (ℓ, k)-threshold function for $\ell \in \{0, k\}$ is one.

Proof.

$$\begin{aligned} \mathbf{\mathcal{T}}(k,k) &= (0,1)^{\otimes k} \\ \mathbf{\mathcal{T}}(0,k) &= (1,1)^{\otimes k} \ . \end{aligned}$$

□

In the next lemma we present a case with a low symmetric rank equal to two.

Lemma 4. The symmetric rank of tensors $\mathbf{\mathcal{T}}(1,k)$ representing the respective $(1,k)$-threshold function is two.

Proof.

$$\mathbf{\mathcal{T}}(1,k) = (1,1)^{\otimes k} - (1,0)^{\otimes k}$$

and there does not exist any vector $\boldsymbol{a}$ such that

$$\mathbf{\mathcal{T}}(1,k) = \boldsymbol{a}^{\otimes k} \ .$$

To see this, note that $\mathbf{\mathcal{T}}(1,k)_{0,\ldots,0} = 0$. This requires $\boldsymbol{a} = (0,a), a \in \mathbb{R}$. But tensor $(0,a)^{\otimes k}$ has all its values except the one at $(1,\ldots,1)$ equal to 0 and thus cannot be equal to $\mathbf{\mathcal{T}}(1,k)$. □

Remark. Contrary to $\mathbf{\mathcal{T}}(1,k)$ having symmetric rank of two in both real and complex domains, the rank of $\mathbf{\mathcal{S}}(1,k)$ is k – even in complex domain – see Proposition 3.

4. CP decompositions of tensors of the exactly $\boldsymbol{\ell}$-out-of-$\boldsymbol{k}$ and threshold functions

In this section we establish the rank of tensors $\mathbf{\mathcal{S}}(\ell,k)$ and $\mathbf{\mathcal{T}}(\ell,k)$ and present explicit formulas for the CP decomposition of these tensors in both the real and complex domains. We conclude the section by explaining the relationship between the symmetric tensor decomposition and the decomposition of homogeneous polynomials.

Let $\boldsymbol{\mathcal{A}}$ be a symmetric tensor of order k, in particular one of $\mathbf{\mathcal{S}}(\ell,k)$, $\mathbf{\mathcal{T}}(\ell,k)$ for some $\ell = 2,\ldots,k-1$. We shall seek for a decomposition of a tensor $\boldsymbol{\mathcal{A}}$ in the form of

$$\boldsymbol{\mathcal{A}} = \boldsymbol{\mathcal{A}}(\boldsymbol{a},\boldsymbol{b}) = \sum_{i=1}^{r} b_i \cdot (1,a_i)^{\otimes k} \ , \tag{11}$$

where $\boldsymbol{a} = (a_1,\ldots,a_r)$ is a vector of nonlinear parameters and $\boldsymbol{b} = (b_1,\ldots,b_r)$ is referred to as a vector of amplitudes. We will refer to this decomposition as the *symmetric CP decomposition to r terms.*

Remark. Note that a general result of [30, Section 4.1] says that a symmetric rank of a symmetric tensor of the order k and dimensions of two is always less or equal to $k+1$ in the complex domain.

Remark. Note that a minimal decomposition of this type with respect to r does not imply that the symmetric rank is r. It can be smaller than r since we excluded from this decomposition rank-one tensors of the type $(0,a)^{\otimes k}$. Actually, this happens in the case of $\mathbf{\mathcal{S}}(k,k)$ and $\mathbf{\mathcal{T}}(k,k)$.

4.1. CP tensor decomposition in the real domain

In this section we will discuss CP tensor decompositions of $\mathbf{\mathcal{S}}(\ell,k)$ and $\mathbf{\mathcal{T}}(\ell,k)$ in the real domain.

Proposition 1. *In the real domain:*

(A) the symmetric rank of $\mathbf{\mathcal{S}}(\ell,k)$, for $\ell \in \{2,\ldots,k-2\}$, is k.

(B) the symmetric rank of $\mathbf{\mathcal{T}}(\ell,k)$, for $\ell \in \{3,\ldots,k-2\}$, is at least $k-1$,

The proof of the proposition is constructive, see Appendix B and Appendix C, respectively.

In the formulas we will use polynomials $p_m, m=1,\ldots,k$ determined by their roots a_i, $i=1,\ldots,k$, $i \neq m$,

$$p_m(x) \quad = \quad \prod_{i=1,i\neq m}^{k} (x-a_i) \quad = \quad \sum_{i=1}^{k} p_m[i]\cdot x^{i-1} \quad , \tag{12}$$

where $p_m[j]$ denotes the $j-$th coefficient of polynomial p_m.

In the case of decomposition of tensor $\mathbf{\mathcal{S}}(\ell,k)$ the first $k-1$ elements of vector $\boldsymbol{a}=(a_1,\ldots,a_k)$ are taken at random, but distinct and with the restrictions[6] that $p_i[\ell+1]\neq 0, i=1,\ldots,k-1$. Then a_k is computed as a rational function of these $a_i, i=1,\ldots,k-1$:

$$a_k \quad = \quad \frac{p_k[\ell]}{p_k[\ell+1]} \quad . \tag{13}$$

The vector $\boldsymbol{b}$ of amplitudes has its elements defined for $m=1,\ldots,k$ as

$$b_m \quad = \quad \frac{p_m[\ell+1]}{p_m(a_m)} \quad . \tag{14}$$

[6]The theoretical probability that randomly generated values of $\boldsymbol{a}$ do not meet the restrictions is zero.

Here $p_m(a_m)$ is the value of the polynomial $p_m(x)$ at point a_m. The derivation of (13) and (14) is given in Appendix B. Note that the right-hand side of (13) is a function of $a_1, \ldots, a_{k-1}$ and is distinct from these a_i's provided that they are mutually distinct - otherwise a decomposition with $k-1$ terms would exist, which is not possible.

Remark. In [20] it was shown that for $\boldsymbol{\mathcal{S}}(\ell, k)$ there exist tensors that have relatively low rank (at most $\min\{\ell+1, k-\ell+1\}$) and approximate tensor $\boldsymbol{\mathcal{S}}(\ell, k)$ with an arbitrarily small error. However, the computations with these low rank approximations are not guaranteed to be numerically stable.

For tensor $\boldsymbol{\mathcal{T}}(\ell, k)$, a similar decomposition as in the case (A) was proposed in [21], with the difference that (13) and (14) are replaced by

$$a_k = \frac{\sum_{i=\ell}^{k} p_k[i]}{\sum_{i=\ell+1}^{k} p_k[i]} = 1 + \frac{p_k[\ell]}{\sum_{i=\ell+1}^{k} p_k[i]} \quad \text{and} \tag{15}$$

$$b_m = \frac{\sum_{i=\ell+1}^{k} p_m[i]}{p_m(a_m)}, \quad \text{for } m = 1, \ldots, k. \tag{16}$$

In Appendix C we present a novel decomposition of the tensor $\boldsymbol{\mathcal{T}}(\ell, k)$ to $k-1$ factors for $2 < \ell < k-1$, to prove part (B) of Proposition 1. Here, the first $k-3$ elements a_i are taken at random, but distinct, and a_{k-2} and a_{k-1} are computed so that the decomposition holds. These a_{k-2} and a_{k-1} are given as solutions of a quadratic equation whose coefficients are functions of a_i, $i = 1, \ldots, k-3$. It appears that the roots of the equation are not always real-valued. For now, for all tested values of ℓ and k ($k \leq 100$) we were able to find (by maximizing a discriminant of the equation) a choice of a_i, $i = 1, \ldots, k-3$ such that the discriminant was positive and all a_is were real-valued. However, we were not able to prove that such a choice always exists – therefore we state it only as a conjecture.

Conjecture 1. *In the real domain the symmetric rank of $\boldsymbol{\mathcal{T}}(\ell, k)$, for $\ell \in \{3, \ldots, k-2\}$, is $k-1$.*

The decomposition is highly ambiguous, in general. For the numerical stability it is better to have the elements a_i as separate as possible but at the same time their absolute values should not be too large. In other words, Vandermonde $(k-1) \times (k-1)$ matrix $\boldsymbol{V}(\boldsymbol{a})$ defined as

$$\boldsymbol{V}(\boldsymbol{a}) = \begin{pmatrix} 1 & \ldots & 1 \\ a_1 & \ldots & a_{k-1} \\ \vdots & & \vdots \\ a_1^{k-2} & \ldots & a_{k-1}^{k-2} \end{pmatrix}. \tag{17}$$

should not be badly conditioned[7]. Therefore we propose optimizing the choice of the initial values of $\boldsymbol{a}$ by jointly minimizing the condition number of $\boldsymbol{V}(\boldsymbol{a})$. The minimization can be done by the Nelder-Mead simplex algorithm with multiple random initializations, to avoid local minima.

Example 8. For the tensor from Example 3 the minimization results in $\boldsymbol{\mathcal{S}}(2,4) = \boldsymbol{\mathcal{A}}(\boldsymbol{a}, \boldsymbol{b})$ with

$$\begin{aligned}\boldsymbol{a} &= (-1.4588, -1.0151, -0.0685, 0.6491) \\ \boldsymbol{b} &= (-0.3341, 1.2563, -1.9324, 1.0101)\end{aligned}$$

Example 9. For the tensor from Example 4 the minimization results in $\boldsymbol{\mathcal{T}}(2,4) = \boldsymbol{\mathcal{A}}(\boldsymbol{a}, \boldsymbol{b})$ with

$$\begin{aligned}\boldsymbol{a} &= (-1.1132, -0.6173, 0.3972, 0.9999) \\ \boldsymbol{b} &= (-0.1391, 0.8801, -1.8737, 1.1327)\ .\end{aligned}$$

Note that Proposition 1 cannot be applied in this case since $\ell = 2$ and Proposition 1 requires $\ell \geq 3$. Instead we used formulas (15) and (16).

4.2. CP tensor decomposition in the complex domain

For large k's, such as $k = 25$ and higher, the condition number of the Vandermonde matrix becomes high for any choice of initial values of $a_i, i = 1, \ldots, k-3$. Therefore, we suggest considering a complex-valued decomposition of the tensors, which can be expressed in closed forms, and might be numerically more suitable. In particular, a_i are uniformly distributed on the unit circle and depend only on the tensor order k, but not on parameter ℓ.

The following proposition gives a method for how to construct a decomposition of tensors $\boldsymbol{\mathcal{S}}(\ell, k)$ into k terms for $0 \leq \ell \leq k-1$ with the same set of the nonlinear parameters a_i, and consequently decompose any symmetric tensor (and $\boldsymbol{\mathcal{T}}(\ell, k)$ in particular) into $k+1$ terms. For example, a "soft threshold" tensor (see Example 10) can also be decomposed in this way.

Example 10. An example of a "soft threshold" tensor. The tensor is symmetric. Contrary to the "sharp" threshold, its values are not 0 and 1 only, but changing from 0 to 1 with an increasing total sum of coordinates.

$$\boldsymbol{\mathcal{U}} = \begin{pmatrix} \begin{pmatrix} 0 & 0.2 \\ 0.2 & 0.5 \end{pmatrix} & \begin{pmatrix} 0.2 & 0.5 \\ 0.5 & 0.8 \end{pmatrix} \\ \begin{pmatrix} 0.2 & 0.5 \\ 0.5 & 0.8 \end{pmatrix} & \begin{pmatrix} 0.5 & 0.8 \\ 0.8 & 1 \end{pmatrix} \end{pmatrix} .$$

[7] Note that for the computations of values of vector $\boldsymbol{b}$ we need to invert the Vandermonde matrix.

Proposition 2. *Let $j = \sqrt{-1}$, $z = e^{2\pi j/k}$, and $\ell \in \{1, \ldots, k-1\}$ be given. In the complex domain, $\boldsymbol{\mathcal{S}}(\ell, k)$ and $\boldsymbol{\mathcal{T}}(\ell, k)$ can be decomposed into k terms and $k+1$ terms, respectively, as*

$$
\begin{aligned}
\boldsymbol{\mathcal{S}}(\ell, k) &= \frac{1}{k} \sum_{m=1}^{k} z^{(k-\ell)(m-1)} (1, z^{m-1})^{\otimes k} && (18) \\
\boldsymbol{\mathcal{T}}(\ell, k) &= (0,1)^{\otimes k} + \frac{k-\ell}{k} (1,1)^{\otimes k} \\
&\quad + \frac{1}{k} \sum_{m=1}^{k-1} \frac{z^m - z^{m(k-\ell+1)}}{1 - z^m} (1, z^m)^{\otimes k} \ . && (19)
\end{aligned}
$$

Proof. The decomposition of $\boldsymbol{\mathcal{S}}(\ell, k)$ can easily be proven by verifying validity of the linear system (B.1) with

$$
\begin{aligned}
\boldsymbol{a} &= (1, z, z^2, \ldots, z^{k-1}) \\
\boldsymbol{b} &= \frac{1}{k} (1, z^{k-\ell}, z^{2(k-\ell)}, \ldots, z^{(k-1)(k-\ell)})
\end{aligned}
$$

for the particular right-hand side. The decomposition of $\boldsymbol{\mathcal{T}}(\ell, k)$ is computed using (18), Lemma 2, and (6). □

Example 11. Tensor $\boldsymbol{\mathcal{S}}(2, 4)$ from Example 3 can be decomposed in the complex domain as $\boldsymbol{\mathcal{A}}(\boldsymbol{a}, \boldsymbol{b})$ defined by (11) and taking values

$$
\begin{aligned}
\boldsymbol{a} &= (1, j, -1, -j) \\
\boldsymbol{b} &= (\frac{1}{4}, -\frac{1}{4}, \frac{1}{4}, -\frac{1}{4}) \ ,
\end{aligned}
$$

which means that

$$
\boldsymbol{\mathcal{S}}(2, 4) = \frac{1}{4}(1,1)^{\otimes 4} - \frac{1}{4}(1,j)^{\otimes 4} + \frac{1}{4}(1,-1)^{\otimes 4} - \frac{1}{4}(1,-j)^{\otimes 4} \ .
$$

Example 12. Tensor $\boldsymbol{\mathcal{T}}(2, 4)$ from Example 4 can be decomposed in complex domain as

$$
\boldsymbol{\mathcal{T}}(2, 4) = (0,1)^{\otimes 4} + \frac{1}{2}(1,1)^{\otimes 4} + \frac{-1+j}{4}(1,j)^{\otimes 4} + \frac{-1-j}{4}(1,-j)^{\otimes 4} \ ,
$$

where the term corresponding to vector $(1, -1)$ vanishes since its corresponding coefficient is 0.

Remark. In the same way, any symmetric tensor can be decomposed into $k+1$ terms, because any symmetric tensor of the dimension $2 \times \ldots \times 2$ and order k can be written as a linear combination of $\boldsymbol{\mathcal{S}}(\ell, k)$, $\ell = 1, \ldots, k$.

The decompositions in Proposition 2 are not guaranteed to be minimal. We note from Proposition 1 that $\mathcal{T}(\ell, k)$ has rank $k-1$ in the real domain; hence its rank cannot be higher in the complex domain. The importance of Proposition 2 lies in the decomposition method of an arbitrary symmetric tensor of the order k and dimensions 2.

The next proposition determines the true symmetric rank of the tensors $\mathcal{S}(\ell, k)$ and $\mathcal{T}(\ell, k)$ in the complex domain, and presents an example of the decomposition.

Proposition 3. *Let $j = \sqrt{-1}$. In the complex domain:*

(A) for $\ell \in \{1, \ldots, k-1\}$

$$r \;=\; \mathit{srank}(\mathcal{S}(\ell, k)) \;=\; \max\{\ell + 1, k - \ell + 1\}$$

A decomposition is $\mathcal{S}(\ell, k) = \boldsymbol{\mathcal{A}}(\boldsymbol{a}, \boldsymbol{b})$ with

$$\begin{aligned} \boldsymbol{a} &= (1, z^1, \ldots, z^{r-1}) \\ \boldsymbol{b} &= \frac{1}{r}\,(1, z^{r-\ell}, z^{2(r-\ell)}, \ldots, z^{(r-1)(r-\ell)}) \end{aligned}$$

where $z = e^{2\pi j/r}$,

(B) for $\ell \in \{2, \ldots, k-1\}$

$$r \;=\; \mathit{srank}(\mathcal{T}(\ell, k)) \;=\; \max\{\ell + 1, k - \ell + 1\}$$

A decomposition is $\mathcal{T}(\ell, k) = \boldsymbol{\mathcal{A}}(\boldsymbol{a}, \boldsymbol{b})$ with $a_i, i = 1, \ldots, r$ defined as the roots of the polynomial $p(x) = x^r - x^{r-1} + 1$, and

$$\boldsymbol{b} \;=\; [\boldsymbol{V}(\boldsymbol{a})]^{-1}(\mathbf{0}_{1\times\ell}, \mathbf{1}_{1\times(r-\ell)})^T \qquad (20)$$

where $\boldsymbol{V}(\boldsymbol{a})$ is the Vandermonde $r \times r$ matrix and $\boldsymbol{v}^T$ denotes the transposition of $\boldsymbol{v}$.

For the proof see Appendix D.

The following two examples illustrate that by using Proposition 3 we get CP decompositions that require fewer terms than those based on Proposition 2.

Example 13. It follows from the first part of Proposition 3 that tensor $\mathcal{S}(2, 4)$ has a rank of $r = \max\{\ell + 1, k - \ell + 1\} = 3$ in the complex domain. Therefore it can be decomposed using only three terms:

$$\mathcal{S}(2, 4) \;=\; \frac{1}{3}\,(1, 1)^{\otimes 4} \;+ \frac{e^{\frac{2\pi j}{3}}}{3}\left(1, e^{\frac{2\pi j}{3}}\right)^{\otimes 4} + \frac{e^{-\frac{2\pi j}{3}}}{3}\left(1, e^{-\frac{2\pi j}{3}}\right)^{\otimes 4} \;.$$

Example 14. It follows from the second part of Proposition 3 that tensor $\boldsymbol{\mathcal{T}}(2,4)$ has a rank of $r = 3$ in the complex domain and it can be decomposed using only three terms (numbers are rounded to six decimal points):

$$\begin{aligned}\boldsymbol{\mathcal{T}}(2,4) \;=\; & 0.310629\,(1,-0.754878)^{\otimes 4} \\ & +(-0.155314+0.340362j)\,(1,0.877439-0.744862j)^{\otimes 4} \\ & +(-0.155314-0.340362j)\,(1,0.877439+0.744862j)^{\otimes 4} \quad .\end{aligned}$$

4.3. Relationship to the decomposition of homogeneous polynomials

Finding a symmetric CP decomposition of a symmetric tensor of order d and dimension n is equivalent with (can be formulated as) a decomposition of a homogeneous polynomial in n variables of total degree d as a sum of dth powers of linear forms – see [33].

All symmetric tensors studied in this paper have a dimension of $n = 2$. Therefore, the corresponding polynomials are homogeneous polynomials in two variables, which are also called binary forms. As long ago as in 1886 J. J. Sylvester [34] proved a theorem whose proof is constructive and yields an algorithm that, for a given binary form, constructs its minimal decomposition as a sum of powers of linear forms. Brachat et al. [33] generalized Sylvester's algorithm for dimensions $n > 2$. However, Sylvester's algorithm offers solutions to our problem in the complex domain only. The rank of the decomposed tensors is not known until the algorithm is completed. On the other hand, the CP decompositions described in the previous subsections explicitly present the rank of tensors of our interest and their decompositions into factors.

Also, note that, generally, tensors $\boldsymbol{\mathcal{S}}(\ell,k)$ and $\boldsymbol{\mathcal{T}}(\ell,k)$ are not generic tensors in the sense of Definition 1 in [35], i.e., the sets of tensors having ranks the same as those of individual tensors $\boldsymbol{\mathcal{S}}(\ell,k)$ and $\boldsymbol{\mathcal{T}}(\ell,k)$ are not open dense subsets of the set of all tensors that have the given order and given dimensions. Therefore, Sylvester's theorem [35, Theorem 3], which implies that the rank of generic symmetric tensors of order k and dimensions 2 is at most $\lfloor \frac{k}{2} \rfloor + 1$, does not apply to them.

5. A CP decomposition of extended tensors

Let $\boldsymbol{\mathcal{U}}(\ell,k)$ be one of $\boldsymbol{\mathcal{S}}(\ell,k)$ and $\boldsymbol{\mathcal{T}}(\ell,k)$ and let it be decomposed as $\boldsymbol{\mathcal{A}}$ in (4) with a certain r,

$$\boldsymbol{\mathcal{U}}(\ell,k) \;=\; \sum_{i=1}^{r} b_i \cdot (1,\, a_i)^{\otimes k} \tag{21}$$

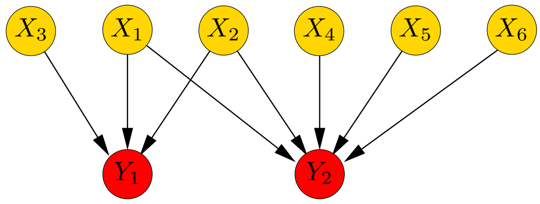

Figure 6: A subnetwork of the QMR-DT network.

Then, the corresponding extended tensor $\overline{\mathfrak{U}}(\ell,k)$ in (7) and in (8), respectively, can be written as

$$\begin{aligned}\overline{\mathfrak{U}}(\ell,k) &= (1,0)\otimes\big((1,1)^{\otimes k}-\mathfrak{U}(\ell,k)\big)+(0,1)\otimes\mathfrak{U}(\ell,k)\\ &= (1,0)\otimes(1,1)^{\otimes k}+(-1,1)\otimes\mathfrak{U}(\ell,k)\\ &= (1,0)\otimes(1,1)^{\otimes k}+(-1,1)\otimes\sum_{i=1}^{r} b_i\cdot(1,\,a_i)^{\otimes k}\,. \end{aligned} \tag{22}$$

It follows that the extended tensor has a decomposition to $r+1$ rank-one terms, and hence its rank is at most $r+1$. If, however, one of a_i's can be set to 1, say $a_1=1$, then the decomposition can be written in only r terms,

$$\overline{\mathfrak{U}}(\ell,k) \;=\; (1-b_1,b_1)\otimes(1,1)^{\otimes k}+(-1,1)\otimes\sum_{i=2}^{r} b_i\cdot(1,\,a_i)^{\otimes k}\,. \tag{23}$$

Note that the condition $a_1 = 1$ can be satisfied in constructions treated in Appendix B and in Proposition 2, but not in those considered in Appendix C, Proposition 1(B), and in Appendix D, Proposition 3(B).

6. Comparison with Parent-Divorcing

In this section we compare complexity of statistical inference using the proposed CP decomposition method versus the parent divorcing. Consider a Bayesian network like the one in Figure 6, which has two layers. The lower layer has two nodes, and the upper layer has $r+s+t$ nodes: the first r nodes are connected only to the first node in the lower layer, the middle s nodes are connected to both nodes in the lower layer, and the remaining t nodes are connected to the second node in the lower layer. In Figure 6, $r=1$, $s=2$, and $t=3$.

In both techniques, auxiliary nodes are added to the network, together with some additional edges due to moralization (only in case of parent-divorcing) and triangularization (in both methods). The resultant graphs

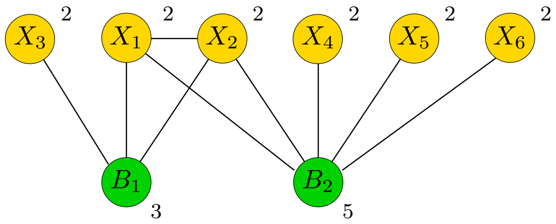

Figure 7: The resulting graph after the CP decomposition and consequent triangulation of the QMR-DT subnetwork from Figure 6. The small numbers attached to nodes represent number of states of corresponding variables. The total table size is 68.

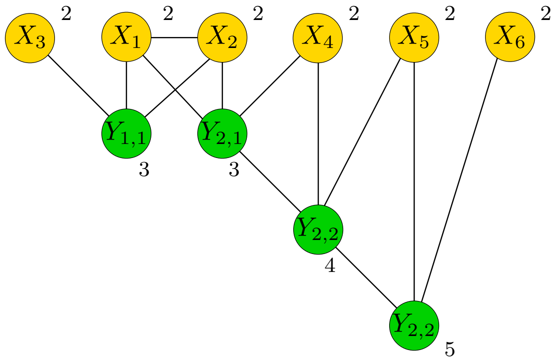

Figure 8: The resulting graph after parent divorcing and consequent moralization and triangulation of the QMR-DT subnetwork from Figure 6. The small numbers attached to nodes represent number of states of corresponding variables. The total table size is 104.

are shown in Figure 7 and Figure 8, together with the number of states at each node. The optimal triangulation was obtained by the procedure implemented in the software package Hugin[8]. Complexity of each method, measured by the total table size, is 68 and 104, respectively.

In the case of general r, s and t and larger s, a different triangulation is optimal, namely the one where the nodes B_1 and B_2 are connected by an edge. The number of the states of the auxiliary variables B_1 and B_2 depends on parameter ℓ. For $\ell = 2$, for example, the number of the states (ranks of the tensors) is maximal and it is equal to the number of parents, i.e., $r + s$ and $s + t$, respectively. We will use this worst case for our comparisons. The corresponding graph will contain s cliques consisting of three nodes having $r + s$, $s + t$, and 2 states plus r cliques consisting of two nodes with $r + s$ and 2 states and t cliques with $s + t$ states. Together, the total table size for the

[8]Hugin Expert A/S, `http://www.hugin.com`

CP decomposition equals

$$\begin{aligned} tts_{CP}(r,s,t) &= 2\left(s^3 + (r+t)s^2 + (r+t+rt)s + 2r^2 + 2t^2\right) \\ &= \mathcal{O}(s^3) \ . \end{aligned}$$

In the parent-divorcing method, the graph structure is more complex, see Figure 8. Assuming $s \geq 4$ the optimal triangulation contains, at the part common to both observations, two cliques of the table sizes $2 \cdot s \cdot (s+1)$, $2 \cdot 2 \cdot s \cdot (s-1)$, and $3 \cdot (s-1) \cdot 3 \cdot 2 \cdot 2$ and for each $i = 1, \ldots, s-4$ one clique with the table size $(s-i) \cdot (s-i-1) \cdot 2 \cdot (i+2)$ and one with the table size $(s-i-1) \cdot 2 \cdot (i+2) \cdot (i+3)$. At the part exclusive for the first observation there are cliques with $(s+i) \cdot (s+i+1) \cdot 2$ table size for $i = 1, \ldots r-1$ and one clique of the table size $(s+r) \cdot 2$. At the part exclusive for the second observation there are cliques with $(s+i) \cdot (s+i+1) \cdot 2$ table size for $i = 1, \ldots t-1$ and one clique of the table size $(s+t) \cdot 2$. The total table size is

$$\begin{aligned} tts_{PD}(r,s,t) &= s^4/3 - 2s^3 + (2r + 2t + 11/3)s^2 \\ &\quad +2(t^2 + r^2 + 21)s + 2(r^3 + t^3 + 2r + 2t)/3 \\ &= \mathcal{O}(s^4) \ . \end{aligned}$$

We can see that the total table size of the PD method grows as a biquadratic function of s, while that of the former, CP method grows as a cubic function of s. The former method is clearly superior.

Remark. In the standard junction tree method, which does not exploit the local structure of CPTs, the moralization has to be performed. This implies that all parents of Y_1 and Y_2 are pairwise interconnected by edges. Two cliques with $r+s$ binary nodes and $s+t$ binary nodes, respectively, are created. Consequently, the total table size is $2^{r+s} + 2^{s+t}$, which is exponential with respect to s.

7. Experiments

We performed experiments with the Quick Medical Reference - Decision Theoretic version (QMR-DT) derived from the original QMR [36] as it is described in [22]. The Bayesian network of QMR-DT contains 570 diseases (variables X_i) and 4075 observations (variables Y_j). The conditional probability tables for observations given related diseases are noisy-or models. We generalized the QMR-DT by replacing noisy-or with noisy-threshold models. These experiments were performed with subnetworks of QMR-DT.

In the first test, we randomly selected 14 observations. We included all their parents in the generated subnetwork. In Figure 6 we give an example of a subnetwork of the QMR-DT network generated by two observations and their parents. We generated 100 different networks this way. For each network we compared computational complexity of the junction tree method [5] applied to models after two different transformations:

- moralization and triangulation - *the standard method*
- *the parent-divorcing method* [12] applied to CPTs with the number of parents greater than eight[9] followed by moralization and triangulation
- *the tensor CP decomposition* applied to CPTs with the number of parents more than six[10] followed by triangulation. In the experiments we assumed the CP decomposition consists of k real-valued terms.

In the second test, we repeated the same process with 28 observations instead of 14. In both tests we utilized 200 bipartite graphs with their sizes ranging from 38 to 582 nodes.

We measured the computational complexity of probabilistic inference by the total table size of models computed by the optimal triangulation procedure implemented in the software package Hugin. If the total table size is larger than 2^{64} the models are intractable in Hugin. Also, for some models Hugin was not able to find an optimal triangulation within reasonable time[11]. In our experiments we limited the time allowed for the triangulation algorithm to a maximum of 300 seconds.

The results of our experiments are summarized in Figure 9. Note the logarithmic scales. In plots of the top row in this figure we present results of experiments with graphs for which Hugin found optimal triangulation for both methods under comparison. It is important to note that using the standard method we were not able to get optimal triangulations for 89 out of 200 graphs (i.e., for 44.5% graphs) and using the parent-divorcing for 109 out of 200 graphs (i.e., for 54.5% graphs). For graphs that Hugin was not able to triangulate optimally we used a triangulation heuristics implemented

[9] For CPTs with number of parents smaller than or equal to eight we used moralization instead. The reason is that, for a small number of parents, the full table has a lower number of entries than the total number of entries in the tables after parent divorcing.

[10] For CPTs with number of parents less or equal to six we used moralization instead. Again, the reason is that for small number of parents the full table has a lower number of entries than the the total number of entries in the tables after CP decomposition.

[11] In our experiments this happened for the standard and parent-divorcing methods only, it never happened for the tensor CP decomposition.

in Hugin. This heuristics is based on restricting the maximum number of minimal separators (we used the value 10^6). To make fair comparisons we used the same triangulation heuristics also for the graphs obtained after tensor CP decomposition (despite the fact that we were able to find optimal triangulations of these graphs).

Numerical experiments reveal that by using tensor CP decomposition on the QMR subnetworks we can get a gain in the order of several magnitudes over the standard method and a gain of one up to four magnitudes over the parent-divorcing methods. More importantly, many intractable models become tractable when the tensor CP decomposition is used.

8. Relationship to arithmetic circuits

A different approach that exploits local structures of CPTs makes use of arithmetic circuits [37]. An arithmetic circuit is a rooted, acyclic directed graph. The leaf nodes are labeled with numeric constants or variables and all other nodes correspond to summation or multiplication. Arithmetic circuits are usually constructed by a conversion to multilinear function, which is then converted to an algebraic circuit through a logical formula of propositional logic. Details can be found in [37, 38]. In [39] the CP tensor decomposition was used to preprocess Bayesian networks containing noisy-or models. The ACs of the preprocessed networks were compared with ACs created by Ace[12] from networks after parent-divorcing. The CP tensor decomposition decreased the size of ACs for a majority of tested networks (about 88%). We conjecture we would get similar results for experiments reported in this section. However, we did not perform these experiments – since Ace does not have any direct support for noisy-threshold models. Performing experiments with Ace without this support (i.e., treating them as general CPTs) would most probably lead to results similar to the standard junction tree method.

For the construction of arithmetic circuits, methods used for logical reasoning can be utilized. It was shown that when Bayesian networks exhibit a lot of determinism or context-specific independence, the weighted model counting (WMC) can be an efficient method for probabilistic inference [40]. The basic idea is to encode the Bayesian network in a conjunctive normal form (CNF), associate weights to literals according to the CPTs of the Bayesian network, and then compute the probability of given evidence as a sum of weights of all logical models consistent with that evidence. The weight of a logical model is the product of weights of all literals. A naive computation of WMC by listing all models and summing their weights is intractable

[12] Ace, A Bayesian Network Compiler, 2008, `http://reasoning.cs.ucla.edu/ace/`

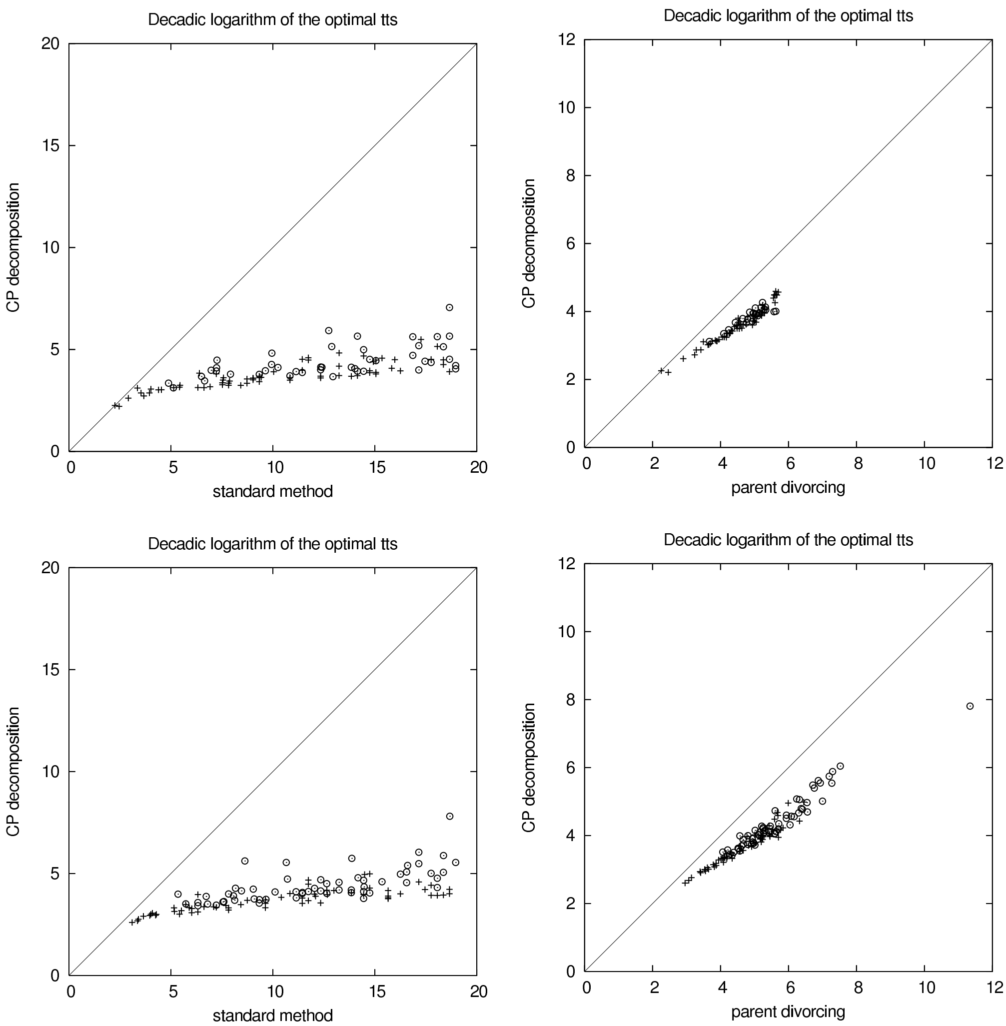

Figure 9: Comparison of total table sizes for optimally triangulated QMR-DT subnetworks (top row) and QMR-DT subnetworks triangulated by a heuristics (bottom row) for the standard method versus the CP tensor decomposition (the left hand side) and the parent-divorcing versus the CP tensor decomposition (the right-hand side). Crosses correspond to subnetworks with 14 observations, circles to subnetworks with 28 observations.

for large problems. Fortunately, efficient WMC solvers exist, that compute WMC using several advanced techniques such as clause learning, component caching, etc. An example of a successful WMC solver is Cachet[13].

The CP tensor decomposition can be combined with weighted model counting (WMC). This was done by Wei Li et al. [41] for noisy-or and noisy-max models. They have shown that for BNs with CPTs representing noisy-or and noisy-max models the Díez and Galán's transformation [16], which is a special case of the CP decomposition, often improves the efficiency of inference by several orders of magnitude. The BNs with CPTs representing noisy-threshold models (or exactly ℓ-out-of-k models) can also be transformed by the CP tensor decomposition and then encoded as a CNF. Either Darwiche's encoding [42] or Sang, Beame, and Kautz's encoding [43] can be used. Then WMC solvers as Cachet can be utilized to compute the probability of evidence as the WMC of logical formula encoded by the CNF.

As it was noted in [40], it should be possible to use a WMC solver for the construction of an arithmetic circuit by modifying it to keep a trace of its operations. As a criteria for fair comparisons of different inference methods we suggest using the number of multiplication and addition operations in the computations, e.g., in the constructed arithmetic circuit. This would allow us to make fair comparisons of different inference methods with inference by arithmetic circuits or by WMC methods based on different CNF encodings and using different WMC solvers such as Cachet and Ace. The mentioned experiments represent an interesting topic for future research.

9. Conclusions

We proposed a CP decomposition of tensors corresponding to conditional probability tables of the threshold and exactly ℓ-out-of-k functions and their noisy counterparts. We applied this decomposition to probabilistic inference in Bayesian networks containing conditional probability tables representing noisy-threshold functions. We performed computational experiments with a generalized version of QMR-DT where the noisy-or models were replaced by noisy-threshold models. The CP tensor decomposition when compared to the standard junction tree method led to a computational gain in the order of several magnitudes and made many intractable models manageable. Theoretical analysis and numerical experiments reveal that tensor CP decomposition is clearly superior to the parent-divorcing method. Our CP decomposition approach can be used as a preprocessing step for weighted model counting.

[13]Cachet, Model Counting using Component Caching and Clause Learning, 2005, `http://www.cs.rochester.edu/u/kautz/Cachet/index.htm`

Acknowledgments

We would like to thank Frank Jensen for the access to the triangulation method implemented in Hugin and Gregory F. Cooper from University of Pittsburgh for the structural part of QMR-DT model. We also thank Petr Savicky for his hints towards the CP decomposition in the complex domain.

[1] J. Pearl, Probabilistic reasoning in intelligent systems: networks of plausible inference, Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, 1988.

[2] F. V. Jensen, Bayesian networks and decision graphs, Springer-Verlag, New York, 2001.

[3] F. V. Jensen, T. D. Nielsen, Bayesian Networks and Decision Graphs, 2nd ed., Springer, 2007.

[4] S. L. Lauritzen, D. J. Spiegelhalter, Local computations with probabilities on graphical structures and their application to expert systems (with discussion), Journal of the Royal Statistical Society, Series B 50 (1988) 157–224.

[5] F. V. Jensen, S. L. Lauritzen, K. G. Olesen, Bayesian updating in recursive graphical models by local computation, Computational Statistics Quarterly 4 (1990) 269–282.

[6] A. E. Gelfand, A. F. M. Smith, Sampling-based approaches to calculating marginal densities, Journal of the American Statistical Association 85 (410) (1990) 398–409.

[7] M. I. Jordan, Z. Ghahramani, T. S. Jaakkola, L. K. Saul, An introduction to variational methods for graphical models, in: M. I. Jordan (Ed.), Learning in Graphical Models, Vol. 89, Kluver Academic Publishers, 1998, pp. 105–161.

[8] A. Cano, S. Moral, A. Salmerón, Penniless propagation in join trees, International Journal of Intelligent Systems 15 (2000) 1027–1059.

[9] C. Boutilier, N. Friedman, M. Goldszmidt, D. Koller, Context-specific independence in Bayesian networks, in: E. Horvitz, F. V. Jensen (Eds.), Proceedings of the Twelfth Annual Conference on Uncertainty in Artificial Intelligence (UAI-96), Morgan Kaufmann, 1996, pp. 115–123.

[10] A. Cano, M. Gómez-Olmedo, S. Moral, Approximate inference in Bayesian networks using binary probability trees, International Journal of Approximate Reasoning 52 (2011) 49–62.

[11] D. Heckerman, A tractable inference algorithm for diagnosing multiple diseases, in: M. Henrion, R. Shachter, L. Kanal, J. Lemmer (Eds.), Proc. of the Fifth Annual Conf. on Uncertainty in AI, 1989, pp. 163–171.

[12] K. G. Olesen, U. Kjærulff, F. Jensen, F. V. Jensen, B. Falck, S. Andreassen, S. K. Andersen, A MUNIN network for the median nerve — a case study on loops, Applied Artificial Intelligence 3 (1989) 384–403, special issue: Towards Causal AI Models in Practice.

[13] D. Heckerman, J. S. Breese, A new look at causal independence, in: R. L. de Mantaras, D. Poole (Eds.), Proc. of the Tenth Conf. on Uncertainty in AI, 1994, pp. 286–292.

[14] N. L. Zhang, D. Poole, Exploiting causal independence in Bayesian network inference, Journal of Artificial Intelligence Research 5 (1996) 301–328.

[15] M. Takikawa, B. D'Ambrosio, Multiplicative factorization of noisy-max, in: K. Laskey, H. Prade (Eds.), Proc. of the Fifteenth Conf. on Uncertainty in AI, 1999, pp. 622–630.

[16] F. J. Díez, S. F. Galán, An efficient factorization for the noisy MAX, International Journal of Intelligent Systems 18 (2003) 165–177.

[17] P. Savicky, J. Vomlel, Exploiting tensor rank-one decomposition in probabilistic inference, Kybernetika 43 (5) (2007) 747–764.

[18] R. A. Harshman, Foundations of the PARAFAC procedure: Models and conditions for an "explanatory" multi-mode factor analysis, UCLA Working Papers in Phonetics 16 (1970) 1–84.

[19] J. D. Carroll, J. J. Chang, Analysis of individual differences in multidimensional scaling via an n-way generalization of Eckart-Young decomposition, Psychometrika 35 (1970) 283–319.

[20] J. Vomlel, Rank of tensors of l-out-of-k functions: an application in probabilistic inference, Kybernetika 47 (3) (2011) 317–336.

[21] J. Vomlel, P. Tichavský, Computationally efficient probabilistic inference with noisy threshold models based on a CP tensor decomposition, in: Proceedings of the Sixth European Workshop on Probabilistic

Graphical Models (PGM 2012), Granada, Spain, September 19-21, 2012, pp. 355–362.

[22] M. Shwe, B. Middleton, D. Heckerman, M. Henrion, E. Horvitz, H. Lehmann, G. Cooper, Probabilistic diagnosis using a reformulation of the INTERNIST-1/QMR knowledge base. I. The probabilistic model and inference algorithms, Methods of Information in Medicine 30 (1991) 241–255.

[23] M. A. J. van Gerven, R. Jurgelenaite, B. G. Taal, T. Heskes, P. J. F. Lucas, Predicting carcinoid heart disease with the noisy-threshold classifier, Artificial Intelligence in Medicine 40 (1) (2007) 45–55.

[24] S. Visscher, P. J. F. Lucas, C. A. M. Schurink, M. J. M. Bonten, Modelling treatment effects in a clinical Bayesian network using Boolean threshold functions, Artificial Intelligence in Medicine 46 (3) (2009) 251–266.

[25] W. A. J. J. Wiegerinck, H. J. Kappen, W. G. Burgers, Bayesian networks for expert systems, theory and practical applications, Interactive Collaborative Information Systems SCI 281 (2010) 547–578.

[26] D. Heckerman, J. Breese, K. Rommelse, Decision-theoretic troubleshooting, Communications of the ACM 38 (1995) 49–57.

[27] F. V. Jensen, U. Kjaerulff, B. Kristiansen, H. Langseth, C. Skaanning, J. Vomlel, M. Vomlelov, The sacso methodology for troubleshooting complex systems, Artificial Intelligence for Engineering Design, Analysis and Manufacturing 15 (2001) 321–333.

[28] F. J. Díez, M. J. Druzdzel, Canonical probabilistic models for knowledge engineering, Tech. Rep. CISIAD-06-01, UNED, Madrid, Spain (2006).

[29] A. L. Madsen, F. V. Jensen, Lazy propagation: A junction tree inference algorithm based on lazy evaluation, Artificial Intelligence 113 (1–2) (1999) 203–245. `doi:DOI:10.1016/S0004-3702(99)00062-4`.

[30] P. Comon, G. Golub, L.-H. Lim, B. Mourrain, Symmetric tensors and symmetric tensor rank, SIAM Journal on Matrix Analysis and Applications 30 (3) (2008) 1254–1279. `doi:10.1137/060661569`.

[31] R. Jurgelenaite, P. Lucas, T. Heskes, Exploring the noisy threshold function in designing Bayesian networks, in: M. Bramer, F. Coenen, T. Allen (Eds.), Research and Development in Intelligent Systems XXII, Springer, 2006, pp. 133–146.

[32] R. Jurgelenaite, T. Heskes, Learning symmetric causal independence models, Machine Learning 71 (2008) 133–153.

[33] J. Brachat, P. Comon, B. Mourrain, E. P. Tsigaridas, Symmetric tensor decomposition, Linear Algebra and its Applications 433 (11–12) (2010) 1851–187.

[34] J. J. Sylvester, Sur une extension dun théorème de Clebsch relatif aux courbes du quatrième degré, Comptes Rendus, Math. Acad. Sci. Paris 102 (1886) 1532–1534.

[35] P. Comon, B. Mourrain, Decomposition of quantics in sums of powers of linear forms, Signal Processing 53 (1996) 93–107.

[36] R. A. Miller, F. E. Fasarie, J. D. Myers, Quick medical reference (QMR) for diagnostic assistance, Medical Computing 3 (1986) 34–48.

[37] A. Darwiche, A differential approach to inference in Bayesian networks, Journal of the ACM 50 (3) (2003) 280–305.

[38] M. Chavira, A. Darwiche, Compiling Bayesian networks with local structure, in: Proc. of the 19th Int. Joint Conference on Artificial Intelligence (IJCAI), Edinburgh, UK, 2005, pp. 1306–1312.

[39] J. Vomlel, P. Savicky, Arithmetic circuits of the noisy-or models, in: M. Jaeger, T. D. Nielsen (Eds.), Proceedings of PGM'08, 2008, pp. 297–304.

[40] M. Chavira, A. Darwiche, On probabilistic inference by weighted model counting, Artificial Intelligence 172 (6) (2008) 772–799.

[41] W. Li, P. Poupart, P. van Beek, Exploiting structure in weighted model counting approaches to probabilistic inference, Journal of Artificial Intelligence Research 40 (2011) 729.

[42] A. Darwiche, A logical approach to factoring belief networks, in: Proceedings of International Conference on Knowledge Representation and Reasoning, 2002, pp. 409–420.

[43] T. Sang, P. Beame, H. Kautz, Solving Bayesian networks by weighted model counting, in: Proceedings of the Twentieth National Conference on Artificial Intelligence (AAAI-05), Vol. 1, AAAI Press, 2005, pp. 475–482.

[44] N. Macon, A. Spitzbart, Inverses of Vandermonde matrices, The American Mathematical Monthly 65 (2) (1958) 95–100.

Appendix A.

In the proof of Proposition 1 we will use the following lemma.

Lemma 5. *Let $p(x)$ be a polynomial whose roots are all distinct and real-valued. Then, it is not possible for two consecutive coefficients of the polynomial to be zeros.*

Proof. First, let us prove by mathematical induction that all derivatives of the polynomial $p(x)$ are polynomials in variable x sharing with $p(x)$ the property that all their roots are real-valued and distinct. Note that if $p(x)$ has roots $a_1 < a_2 < \ldots < a_k$ then its first derivative $p'(x)$ has roots $a'_1 < \ldots < a'_{k-1}$ at stationary points of $p(x)$, i.e., $a_1 < a'_1 < a_2 < a'_2 < \ldots < a'_{k-1} < a_k$. Assume by contradiction that $p[i] = p[i+1] = 0$ are two consecutive coefficients of $p(x)$. Then the i-th derivative of $p(x)$ is a polynomial which has both its constant term and the linear term equal to 0. It means that 0 is a double root of the polynomial, which contradicts the assumption that all roots of $p(x)$ were distinct. □

Appendix B. CP decomposition of $\mathcal{S}(\ell, k)$ in k terms

In this appendix we present the proof of part (A) of Proposition 1.

Proof. The condition for a $\overbrace{2 \times \ldots \times 2}^{k \text{ times}}$ symmetric tensor $\boldsymbol{\mathcal{A}}$ to be decomposed by the symmetric CP decomposition to r terms as (11), is

$$\begin{aligned} a_1^0 \cdot b_1 + \ldots + a_r^0 \cdot b_r &= \boldsymbol{\mathcal{A}}_{0\ldots 0} \\ a_1^1 \cdot b_1 + \ldots + a_r^1 \cdot b_r &= \boldsymbol{\mathcal{A}}_{0\ldots 0,1} \\ &\vdots \\ a_1^k \cdot b_1 + \ldots + a_r^k \cdot b_r &= \boldsymbol{\mathcal{A}}_{1\ldots 1,1} \ , \end{aligned} \tag{B.1}$$

which is a system of $k+1$ equations with $2r$ variables.

In a decomposition of a symmetric tensor there might be one more term that is not considered in formula (11) and, consequently, in (B.1). It takes the form $b_0 \cdot (0,1)^{\otimes k}$. We do not use it in the suggested decompositions. As we prove later, even without considering this term we get a minimal CP decomposition of $\mathcal{S}(\ell, k)$.

Let $m \in \mathbb{N}^+$, $\boldsymbol{a} = (a_1, \ldots, a_m), a_j \in \mathbb{R}, j \in \{1, \ldots, m\}$, and $\boldsymbol{V}(\boldsymbol{a})$ be a Vandermonde $m \times m$ matrix

$$\boldsymbol{V}(\boldsymbol{a}) = \begin{pmatrix} 1 & \ldots & 1 \\ a_1 & \ldots & a_m \\ \vdots & & \vdots \\ a_1^{m-1} & \ldots & a_m^{m-1} \end{pmatrix} .$$

Further let $\boldsymbol{e}(\ell)$ be the right-hand side of the system (B.1) without the last element. It is the $(\ell+1)$-th column of the $k \times k$ identity matrix, or

$$\boldsymbol{e}(\ell) = \left(\mathbf{0}_{1\times\ell}, 1, \mathbf{0}_{1\times(k-\ell-1)}\right)^T .$$

A sufficient condition for the solution of the system (B.1) for $r = k$ is to solve the following system of $3k$ equations. Let $\boldsymbol{a} = (a_1, \ldots, a_k)$, $\boldsymbol{b} = (b_1, \ldots, b_k)$, $\boldsymbol{c} = (c_1, \ldots, c_k)$, and $\boldsymbol{a} * \boldsymbol{b}$ denote elementwise multiplication (Hadamard product) of vectors $\boldsymbol{a}$ and $\boldsymbol{b}$. All but the last equation of system (B.1) correspond to system (B.2), all but the first equation of system (B.1) correspond to systems (B.3) and (B.4):

$$\boldsymbol{V}(\boldsymbol{a}) \cdot \boldsymbol{b} = \boldsymbol{e}(\ell) \tag{B.2}$$

$$\boldsymbol{V}(\boldsymbol{a}) \cdot \boldsymbol{c} = \boldsymbol{e}(\ell-1) \tag{B.3}$$

$$\boldsymbol{c} = \boldsymbol{a} * \boldsymbol{b} . \tag{B.4}$$

If values of $a_i, i = 1, \ldots, k$ are distinct then

$$\begin{aligned} \boldsymbol{b} &= \boldsymbol{V}(\boldsymbol{a})^{-1} \cdot \boldsymbol{e}(\ell) \\ \boldsymbol{c} &= \boldsymbol{V}(\boldsymbol{a})^{-1} \cdot \boldsymbol{e}(\ell-1) . \end{aligned}$$

Note that the explicit formula for the inverse of the Vandermonde matrix is known [44]. It implies that for $i = 1, \ldots, k$

$$b_i = \frac{p_i[\ell+1]}{p_i(a_i)} \tag{B.5}$$

$$c_i = \frac{p_i[\ell]}{p_i(a_i)} . \tag{B.6}$$

Substituting (B.5) and (B.6) into (B.4) and assuming that $p_i[\ell+1] \neq 0$ we get for $i = 1, \ldots, k$:

$$a_i = \frac{c_i}{b_i} = \frac{p_i[\ell]}{p_i[\ell+1]} , \tag{B.7}$$

where the right-hand side depends on $a_j, j \neq i$ only. Due to symmetry, if at least one equation of (B.7) holds then all these equations hold, and the system can be reduced to just one equation, e.g., to the desired equation:

$$a_k \quad = \quad \frac{p_k[\ell]}{p_k[\ell+1]} \quad .$$

Now, we will show that in the real domain no decomposition of the tensors $\mathcal{S}(\ell, k)$ with $1 < \ell < k-1$ into $r < k$ terms exists. First, assume tensor $b_0 \cdot (0,1)^{\otimes k}$ does not participate in the decomposition. We prove the desired claim by contradiction. Assume existence of distinct real-valued a_i, $i = 1, \dots, k-1$, and corresponding amplitudes b_i, such that (B.1) holds with $r = k-1$. This is true if and only if (B.1) holds with $r = k$ (i.e., a decomposition with k terms) with $b_k = 0$ and an arbitrary a_k. From $b_k = 0$ and (B.5) we have that $p_k[\ell+1] = 0$. This together with (B.4) and (B.6) implies that $p_k[\ell] = 0$ as well. This is, however, not possible due to Lemma 5.

Finally, consider tensor $b_0 \cdot (0,1)^{\otimes k}$ taking part in the decomposition. By subtracting this tensor from $\mathcal{S}(\ell, k)$ we get a tensor $\mathcal{S}'(\ell, k)$ that differs from $\mathcal{S}(\ell, k)$ only in the value $\mathcal{S}'_{1,\dots,1}(\ell, k)$. Now it suffices to show that system (B.1) with $\boldsymbol{\mathcal{A}} = \mathcal{S}'(\ell, k)$ and $r = k-2$ has no solution. This system without the last equation corresponds to the condition for a decomposition of $\mathcal{S}(\ell, k')$, $k' = k-1$ into $k'-1$ terms with $b_0 \cdot (0,1)^{\otimes k}$ excluded from the decomposition. First, assume $\ell < k'-1$. For this case we have already shown (just above) that the decomposition is not possible. For the remaining border case with $\ell = k'-1$ the system without the last equation corresponds to the conditions for a decomposition of $\mathcal{S}(k'-1, k')$ into $k'-1$ terms. By Proposition 3 tensor $\mathcal{S}(k'-1, k')$ has symmetric rank k' in the complex domain and therefore no decomposition into $k'-1$ terms is possible. This concludes the proof of part (A). □

Appendix C. CP decomposition of $\mathcal{T}(\ell, k)$ in $k-1$ terms

In this appendix we present a constructive proof of part (B) of Proposition 1.

Proof. Assume that $2 < \ell < k-1$. Define

$$\begin{aligned}
\boldsymbol{h}_\ell &= (\boldsymbol{0}_{1\times\ell}, \boldsymbol{1}_{1\times(k-1-\ell)})^T \\
\boldsymbol{b} &= \boldsymbol{V}(\boldsymbol{a})^{-1}\boldsymbol{h}_\ell \\
\boldsymbol{c} &= \boldsymbol{V}(\boldsymbol{a})^{-1}\boldsymbol{h}_{\ell-1} \\
\boldsymbol{d} &= \boldsymbol{V}(\boldsymbol{a})^{-1}\boldsymbol{h}_{\ell-2} \\
p(x) &= \sum_{i=1}^{k-1} p[i]x^{i-1} = \prod_{i=1}^{k-2}(x-a_i) \\
q(x) &= \sum_{i=1}^{k-2} q[i]x^{i-1} = \prod_{i=1}^{k-3}(x-a_i)\ .
\end{aligned}$$

Since $p(x) = q(x)(x - a_{k-2})$, it holds for $i = 2, \ldots, k-2$

$$p[i] = q[i-1] - a_{k-2}\ q[i] \tag{C.1}$$

$$p[1] = -a_{k-2}\ q[1]\ . \tag{C.2}$$

The vectors $\boldsymbol{b}, \boldsymbol{c}, \boldsymbol{d}$ should fulfill

$$\boldsymbol{b} * \boldsymbol{a} = \boldsymbol{c} \tag{C.3}$$

$$\boldsymbol{c} * \boldsymbol{a} = \boldsymbol{d}\ . \tag{C.4}$$

The last elements of the vectors $\boldsymbol{b}, \boldsymbol{c}, \boldsymbol{d}$ are

$$b_{k-1} = \frac{\sum_{j=\ell+1}^{k-1} p[j]}{p(a_{k-1})} \tag{C.5}$$

$$c_{k-1} = \frac{\sum_{j=\ell}^{k-1} p[j]}{p(a_{k-1})} \tag{C.6}$$

$$d_{k-1} = \frac{\sum_{j=\ell-1}^{k-1} p[j]}{p(a_{k-1})}\ . \tag{C.7}$$

Therefore

$$\begin{aligned}
a_{k-1} &= \frac{c_{k-1}}{b_{k-1}} = \frac{\sum_{j=\ell}^{k-1} p[j]}{\sum_{j=\ell+1}^{k-1} p[j]} \\
&= \frac{d_{k-1}}{c_{k-1}} = \frac{\sum_{j=\ell-1}^{k-1} p[j]}{\sum_{j=\ell}^{k-1} p[j]}
\end{aligned}$$

The equation to be fulfilled is

$$\left(\sum_{j=\ell}^{k-1} p[j]\right)^2 = \left(\sum_{j=\ell+1}^{k-1} p[j]\right)\left(\sum_{j=\ell-1}^{k-1} p[j]\right) \tag{C.8}$$

Using substitution $y = \sum_{j=\ell}^{k-1} p[j]$ this can be rewritten as

$$
\begin{aligned}
y^2 &= (y - p[\ell])(y + p[\ell - 1]) \\
p[\ell]p[\ell - 1] &= y(p[\ell - 1] - p[\ell]) \\
&= (p[\ell - 1] - p[\ell]) \left(1 + \sum_{j=\ell}^{k-2} p[j]\right) \; .
\end{aligned}
\tag{C.9}
$$

By substituting (C.1) in (C.9) we get, after some algebra, a quadratic equation for a_{k-2},

$$
Aa_{k-2}^2 + Ba_{k-2} + C = 0
$$

where

$$
\begin{aligned}
A &= q[\ell]q[\ell - 1] + (q[\ell] - q[\ell - 1])A_1 \\
B &= (q[\ell - 2] - q[\ell])A_1 - q[\ell](q[\ell - 2] + q[\ell - 1]) \\
C &= (q[\ell - 1])^2 - (q[\ell - 2] - q[\ell - 1])A_1 \\
A_1 &= 1 + \sum_{j=\ell}^{k-3} q[j] \; .
\end{aligned}
$$

Thanks to the symmetry of the problem, the second root of the equation is a_{k-1}. Note that the decomposition to $k - 1$ terms is possible only for $2 < \ell < k - 1$.

Now, we will show that in the real domain no decomposition of tensors $\mathfrak{T}(\ell, k)$ with $2 < \ell < k-1$ into $r < k-1$ terms exists. We prove this claim by contradiction. Assume existence of distinct real-valued a_i, $i = 1, \ldots, k - 2$, and the corresponding amplitudes b_i, such that (B.1) holds with $r = k - 2$. This is true if and only if (B.1) holds with $r = k - 1$ (i.e., a decomposition with k terms) with $b_{k-1} = 0$ and an arbitrary a_{k-1}. From $b_{k-1} = 0$, (C.3), and (C.4) we get that $c_{k-1} = 0$ and $d_{k-1} = 0$. This implies that all numerators in the fractions on the right-hand sides of equations (C.5), (C.6), and (C.7) are 0. The numerators in (C.5), (C.6) and (C.6), (C.7) differ in $p[\ell]$ and $p[\ell - 1]$, respectively. Therefore $p[\ell]$ and $p[\ell - 1]$ must be 0. This is, however, not possible due to Lemma 5, because two consecutive coefficients of a polynomial with distinct roots cannot be zeros.

The treatment of the case when tensor $b_0 \cdot (0, 1)^{\otimes k}$ is allowed to take part in the decomposition is identical as in the proof of part (A) of Proposition 1 presented in Appendix B.

Thus, no decomposition to less than $k - 1$ factors in the real domain is possible, which completes the proof of part (B) of Proposition 1. □

Appendix D. Proof of Proposition 3

Proof. Consider the linear system (B.1) and assume that a_j's are distinct. The system cannot have a solution if the right-hand side contains r consecutive zeros, because the Vandermonde system $\boldsymbol{V}(\boldsymbol{a})\ \boldsymbol{b} = \boldsymbol{0}$ can only have trivial solution then, $\boldsymbol{b} = 0$.

Note that for the tensor $\boldsymbol{\mathfrak{S}}(\ell, k)$ the right-hand side contains ℓ and $k - \ell$ consecutive zeros. Therefore the symmetric rank of $\boldsymbol{\mathfrak{S}}(\ell, k)$ is greater than or equal to $r = \max(\ell + 1, k - \ell + 1)$ unless there is a decomposition of $\boldsymbol{\mathfrak{S}}(\ell, k)$ containing tensor $b_0 \cdot (0, 1)^{\otimes k}$ among its $r - 1$ terms. In the case of $\ell \geq k - \ell$ the tensor $b_0 \cdot (0, 1)^{\otimes k}$ has no influence on the part of the right-hand side containing r consecutive zeros and thus it cannot lower the symmetric rank. Also in the case of $\ell < k - \ell$ the presence of the tensor $b_0 \cdot (0, 1)^{\otimes k}$ cannot lower the symmetric rank since such lowering would contradict Corollary 1.

The validity of the decomposition can be verified using the system (B.1). The existence of the decomposition with r terms proves that the rank is equal to r.

A similar situation exists with $\boldsymbol{\mathfrak{T}}(\ell, k)$ with $\ell > k/2$. Again, the system (B.1) contains ℓ consecutive zeros and tensor $b_0 \cdot (0, 1)^{\otimes k}$ cannot influence the part of the right-hand side containing r consecutive zeros. Therefore the rank is at least $\ell + 1$. The situation with $\ell \leq k/2$ is slightly more complex. The rank of such $\boldsymbol{\mathfrak{T}}(\ell, k)$ can be deduced from the fact that $\boldsymbol{\mathfrak{H}}(\ell, k) = (1, 1)^{\otimes k} - \boldsymbol{\mathfrak{T}}(\ell, k)$ is a symmetric tensor, which has the representation (B.1) with $k-\ell+1$ consecutive zeros. Its rank is at least $k - \ell + 2$ since the inclusion of tensor $b_0 \cdot (0, 1)^{\otimes k}$ cannot lower its rank – such a lowering would contradict Lemma 1. It follows that the rank of $\boldsymbol{\mathfrak{T}}(\ell, k)$ is at least $k - \ell + 1$, since a lower rank of $\boldsymbol{\mathfrak{T}}(\ell, k)$ would be in contradiction with rank of $\boldsymbol{\mathfrak{H}}(\ell, k)$ being at least $k-\ell+2$.

Proof of the decomposition in part (B). Let $a_1, \dots, a_r$ be roots of the polynomial $p(x) = x^r - x^{r-1} + 1$. It follows that a_i obey for $i = 1, \dots, r$

$$a_i^r = a_i^{r-1} - 1\ . \tag{D.1}$$

Let b_i, $i = 1, \dots, r$ are elements of $\boldsymbol{b}$ in (20). Put

$$\varphi_m = \sum_{i=1}^{r} a_i^m b_i \qquad \text{for} \quad m = 0, 1, \dots \tag{D.2}$$

Since b_i, $i = 1, \dots, r$ are defined in (20) so that

$$\varphi_0 \ = \ \dots \ = \ \varphi_{\ell-1} \ = \ 0 \ \text{ and} \tag{D.3}$$

$$\varphi_\ell \ = \ \dots \ = \ \varphi_{r-1} \ = \ 1\ . \tag{D.4}$$

For $m \in \{r, r+1, \ldots, k\}$, the value of φ_m can be computed recursively.

$$\begin{aligned}
\varphi_m &= \sum_{i=1}^{r} a_i^m b_i \\
&= \sum_{i=1}^{r} a_i^r a_i^{m-r} b_i \\
&= \sum_{i=1}^{r} (a_i^{r-1} - 1) a_i^{m-r} b_i \\
&= \varphi_{m-1} - \varphi_{m-r} \ ,
\end{aligned}$$

where (D.1) was used. Thus it can be proved by mathematical induction that $\varphi_m = 1$ for $m = r, r+1, \ldots, k$. To see this, note that

$$\begin{aligned}
m - r &= m - \max\{\ell + 1, k - \ell + 1\} \ = \ \min\{m - \ell - 1, m - k + \ell - 1\} \\
&\leq \min\{k - \ell - 1, \ell - 1\} \ \leq \ \ell - 1 \ ,
\end{aligned}$$

which, together with (D.3), implies that $\varphi_{m-r} = 0$, for $m = r, r+1, \ldots, k$.

It follows that the system (B.1) holds true for these a_i's and b_i's and this fact proves the decomposition. Note that the choice of the coefficient $p[1]$ in the polynomial $p(x)$ can be arbitrary – it is only constrained to be nonzero in order to avoid multiple roots of the polynomial. We have chosen $p[1] = 1$ for convenience. $\square$