KYBERNETIKA — VOLUME *43* (2007), NUMBER 5, PAGES XXX – YYY

EXPLOITING TENSOR RANK-ONE DECOMPOSITION IN PROBABILISTIC INFERENCE

Petr Savicky and Jiří Vomlel

We propose a new additive decomposition of probability tables – *tensor rank-one decomposition*. The basic idea is to decompose a *probability table* into a series of tables, such that the table that is the sum of the series is equal to the original table. Each table in the series has the same domain as the original table but can be expressed as a product of one-dimensional tables. Entries in tables are allowed to be any real number, i. e. they can be also negative numbers. The possibility of having negative numbers, in contrast to a multiplicative decomposition, opens new possibilities for a compact representation of probability tables. We show that *tensor rank-one decomposition* can be used to reduce the space and time requirements in probabilistic inference. We provide a closed form solution for minimal tensor rank-one decomposition for some special tables and propose a numerical algorithm that can be used in cases when the closed form solution is not known.

Keywords: graphical probabilistic models, probabilistic inference, tensor rank

AMS Subject Classification: 68T37, 62E15, 15A69

1. INTRODUCTION

The fundamental property of probabilistic graphical models [11, 13] that allows their application in domains with hundreds to thousands variables is the multiplicative factorization of the joint probability distribution. The multiplicative factorization is exploited in inference methods, e. g., in the junction tree propagation method [12]. However, in some real applications the models may become intractable even when the junction tree propagation method (or other exact inference methods) are used because after the moralization and triangularization steps the graphical structure becomes too dense, cliques consist of too many variables, and, consequently, the probability tables corresponding to the cliques are too large to be efficiently manipulated. In such case, one usually turns to an approximative inference method. Following the ideas presented in [5, 17] we propose a new decomposition of probability tables that allows to use exact inference in some models where – without the suggested decomposition – the exact inference using the standard methods is impossible. The basic idea is to decompose a probability table into a series of tables, such that the table that is the sum of the series is equal to the original table. Each table in the series has the same domain as the original table but can be expressed as a

product of one-dimensional tables. Entries in tables are allowed to be any real number. Extending the range to negative numbers opens new possibilities for compact representation of probability tables and allows to find a shorter series.

It is convenient to formally specify the task using the tensor terminology. Assume variables X_i, $i \in N \subset \mathbb{N}$, each variable X_i taking values (a value of X_i will be denoted x_i) from a finite set $\mathcal{X}_i$. Let for any $A \subseteq N$ the symbol x_A denote a vector of the values $(x_i)_{i\in A}$ from the Cartesian product $\mathcal{X}_A = \times_{i\in A}\mathcal{X}_i$, where for all $i \in A$: x_i is a value from $\mathcal{X}_i$.

Definition. Tensor Let $A \subset N$. A *tensor* ψ over $\mathbb{R}$ indexed by $\mathcal{X}_A$ is a mapping

$$\mathcal{X}_A \quad \mapsto \quad \mathbb{R}.$$

The cardinality $|A|$ is called *tensor dimension*.

Note that every probability table can be looked upon as a tensor. Tensor ψ indexed by $\mathcal{X}_A$ is an (unconditional) probability table if for every x_A it holds that $0 \leq \psi(x_A) \leq 1$ and $\sum_{x_A} \psi(x_A) = 1$. Tensor ψ is a conditional probability table (CPT) if for every x_A it holds that $0 \leq \psi(x_A) \leq 1$ and if there exists $B \subset A$ such that for every x_B it holds $\sum_{x_{A\setminus B}} \psi(x_B, x_{A\setminus B}) = 1$.

Next, we will recall the basic tensor notion. If, for a tensor indexed by $\mathcal{X}_A$, $|A| = 1$, then the tensor is a vector. If $|A| = 2$ then the tensor is a matrix. The *outer product* $\psi \otimes \varphi$ of two tensors $\psi : \times_{i\in A}\mathcal{X}_i \mapsto \mathbb{R}$ and $\varphi : \times_{i\in B}\mathcal{X}_i \mapsto \mathbb{R}$, $A \cap B = \emptyset$ is a tensor $\xi : \times_{i\in A\cup B}\mathcal{X}_i \mapsto \mathbb{R}$ defined for all $x_{A\cup B}$ as

$$\xi(x_{A\cup B}) \quad = \quad \psi(x_A) \cdot \varphi(x_B).$$

Now, let ψ and φ are defined on the same domain $\times_{i\in A}\mathcal{X}_i$. The sum $\psi + \varphi$ of two tensors is tensor $\xi : \times_{i\in A}\mathcal{X}_i \mapsto \mathbb{R}$ such that for all x_A

$$\xi(x_A) \quad = \quad \psi(x_A) + \varphi(x_A).$$

Definition 2. Tensor rank (Håstad [10]) Tensor of dimension $|A|$ has *rank* one if it is an outer product of $|A|$ vectors. *Rank* of tensor ψ is the minimal number of tensors of rank one that sum to ψ. *Rank* of tensor ψ will be denoted as $\mathrm{rank}(\psi)$.

Remark. Note that the standard matrix rank is a special case of the tensor rank for $|A| = 2$. An alternative definition of the matrix rank is that the rank of an $m \times n$ matrix $\boldsymbol{M}$ is the smallest r for which there exist a $m \times r$ matrix $\boldsymbol{F}$ and a $r \times n$ matrix $\boldsymbol{G}$ such that

$$\boldsymbol{M} \quad = \quad \boldsymbol{F}\boldsymbol{G}. \tag{1}$$

Let $\boldsymbol{f}^i$ be the ith column of $\boldsymbol{F}$ and $\boldsymbol{g}^i$ be the ith row of $\boldsymbol{G}$. One can equivalently write equation (1) as

$$\boldsymbol{M} \quad = \quad \sum_{i=1}^{r} \boldsymbol{f}^i \otimes \boldsymbol{g}^i. \tag{2}$$

Since $\mathrm{rank}(\boldsymbol{f}^i \otimes \boldsymbol{g}^i) = 1$ one can see that the tensor rank of $\boldsymbol{M}$ is equal to the matrix rank of $\boldsymbol{M}$.

Definition 3. Tensor rank-one decomposition Assume a tensor ψ indexed by $\mathcal{X}_A$ and an integer r. A series of tensors $\{\varrho_b\}_{b=1}^r$ indexed by $\mathcal{X}_A$ such that

- for $b = 1, \ldots, r$: $\text{rank}(\varrho_b) = 1$, i. e., $\varrho_b = \otimes_{i \in A} \varphi_{i,b}$, where $\varphi_{i,b}, i \in A$ are vectors and
- $\psi = \sum_{b=1}^r \varrho_b$

is called *tensor rank-one decomposition* of ψ of length r.

Note that from the definition of tensor rank it follows that such a series exists iff $r \geq \text{rank}(\psi)$. The decomposition is minimal if $r = \text{rank}(\psi)$.

Example 1. Let $\psi : \{0,1\} \times \{0,1\} \times \{0,1\} \mapsto \mathbb{R}$ be[1]

$$\begin{pmatrix} \begin{pmatrix} 1 \\ 2 \end{pmatrix} & \begin{pmatrix} 2 \\ 4 \end{pmatrix} \\ \begin{pmatrix} 2 \\ 4 \end{pmatrix} & \begin{pmatrix} 4 \\ 9 \end{pmatrix} \end{pmatrix}.$$

This tensor has rank two since[2]

$$\psi \quad = \quad (1,2) \otimes (1,2) \otimes (1,2) \; + \; (0,1) \otimes (0,1) \otimes (0,1)$$

and there are no three vectors whose outer product is equal to ψ.

It was proved in [10] that the computation of tensor rank is an NP-hard problem, therefore determining the minimal rank-one decomposition is also an NP-hard problem.

Definition 4. Tensor rank-one approximation Assume a tensor ψ and an integer $s \geq 1$. A *tensor rank-one approximation* of tensor ψ of length s is a series $\{\varrho_b\}_{b=1}^s$ of rank-one tensors ϱ_b that is a tensor rank-one decomposition of a tensor $\hat{\psi}$ with $\text{rank}(\hat{\psi}) = s$. If $\hat{\psi}$ minimizes $\sum_x (\psi(x) - \hat{\psi}(x))^2$ we say that it is a *best tensor rank-one approximation of length s*.

Higher-dimensional tensors are studied in multilinear algebra [3]. The problem of tensor rank-one decomposition is also known as *canonical decomposition* (CANDECOMP) or *parallel factors* (PARAFAC). A typical task is to find a tensor of rank one that is a best approximation of a tensor ψ. This task is usually solved using an

[1] We visualize tensors using a matrix convention in the same way they are diplayed by the commnad `MatrixForm` in the computational tool Mathematica. The first dimension corresponds to the raw of the most outer matrix, the second dimension to the column of the most outer matrix, the third dimension to the raw of the second most outer matrix, etc. This means that a tensor of dimension one will be displayed as a column vector, a tensor of dimension two as a matrix and a tensor of dimension three as a matrix of vectors.

[2] To simplify notation we write all vectors in an outer product without the transposition signs, i. e., as raw vectors.

alternating least square algorithm (ALS) that is a higher-order generalization of the power method for matrices [4].

The rest of the paper is organized as follows. In Section 2 we show using a simple example how tensor rank-one decomposition can be used to reduce the space and time requirements for the probability inference using the junction tree method. We compare sizes of the junction tree for the standard approach, the parent divorcing method, and the junction tree after tensor rank-one decomposition[3]. In Section 3 the main theoretical results are presented: the lower bound on the tensor rank for a class of tensors and minimal tensor rank-one decompositions for some special tensors – max, add, xor, and their noisy counterparts. In Section 4 we propose a numerical method that can be used to find a tensor rank-one decomposition. We also present results of experiments with the numerical method.

2. TENSOR RANK–ONE DECOMPOSITION AND PROBABILISTIC INFERENCE

We will use an example of a simple Bayesian network to show computational savings of the proposed decomposition. Assume a Bayesian network having the structure given on the left hand side of Figure 1. Variables $X_1, \ldots, X_m$ are binary taking values 0 and 1. For simplicity, assume that $m = 2^d, d \in \mathbb{N}, 2 \leq d$. Further assume a variable $Y \stackrel{\text{df}}{=} X_{m+1}$, whose values $y = \sum_{i=1}^{m} x_i$. This means that Y takes $m + 1$ values.

If one uses the standard junction tree construction [12], all parents of Y get connected by edges (this step is often called moralization). One need not perform triangulation since the graph is already triangulated. The resulting junction tree consists of one clique containing all variables, i. e., $C_1 = \{X_1, \ldots, X_m, Y\}$. The resulting junction tree consists of one clique $C_1 = \{X_1, \ldots, X_m, Y\}$. See the right hand side of Figure 1.

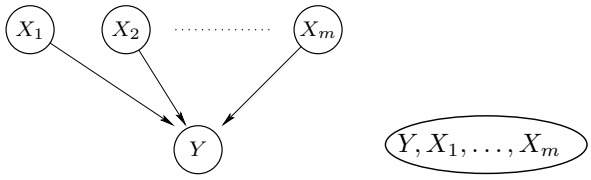

Fig. 1. Bayesian network structure and its junction tree.

Since the CPT $P(Y \mid X_1, \ldots, X_m)$ has a special form, one can use the parent divorcing method [14] and introduce a number of auxiliary variables, one auxiliary variable for a pair of parent variables. This is used hierarchically, i. e. one gets a tree of auxiliary variables with node Y being the root of the tree. The resulting

[3]Several other methods were proposed to exploit a special structure of CPTs. For a review of these methods see, for example, [17]. In this paper we do comparisons with the parent divorcing method only.

Bayesian network is given in Figure 2. The resulting junction tree consists of $m-1$ cliques. See Figure 3.

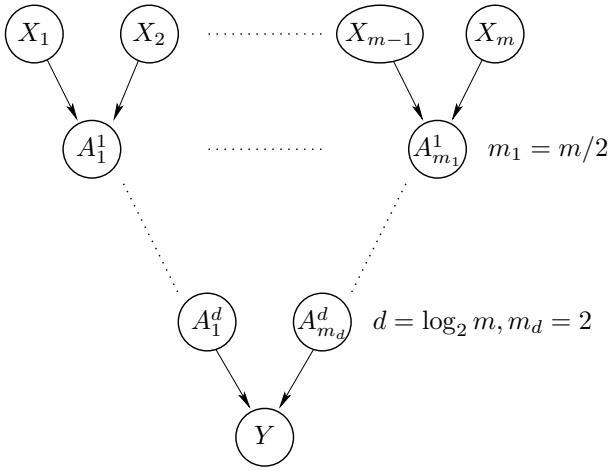

Fig. 2. Bayesian network after parent divorcing.

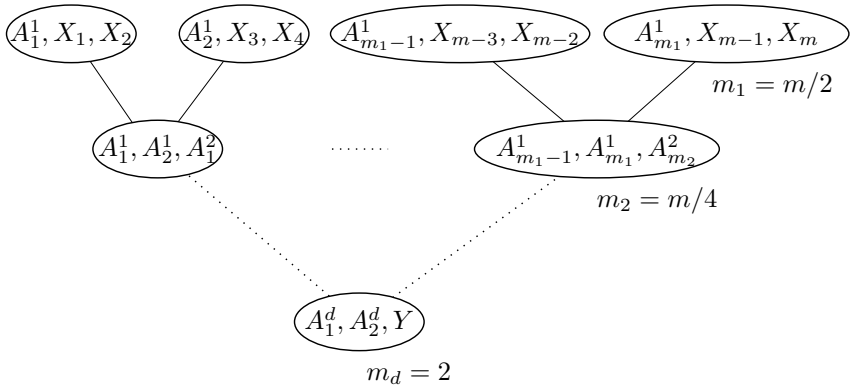

Fig. 3. Junction tree for the parent divorcing method.

In Section 3.2 we will show that if the CPT corresponds to addition of m binary variables then we can decompose this CPT to a series of $m+1$ tensors that are products of vectors

$$P(Y \mid X_1, \ldots, X_m) = \sum_{b=1}^{m+1} \xi_b \otimes (\otimes_{i=1}^{m} \varphi_{i,b}),$$

which is *a tensor rank-one decomposition* of this CPT. Since tensor rank-one decomposition is an additive decomposition, we can visualize it using one additional

variable (which we will denote B) as suggested in [5]. In case of addition of m binary variables, variable B will have $m+1$ states. Instead of moralization, we add variable B into the model and connect it with nodes corresponding to variables $Y, X_1, \ldots, X_m$. We get the structure given in Figure 4. Note that all edges are undirected, which means that we do not perform any moralization. It is not difficult to show (see [17]) that this model can be used to compute marginal probability distributions as in the original model. The resulting junction tree of this model is given in Figure 5.

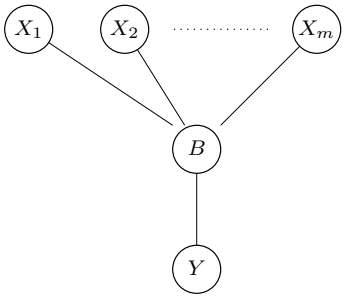

Fig. 4. Bayesian network after the decomposition.

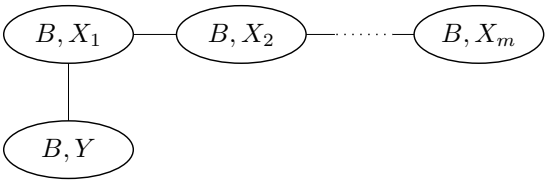

Fig. 5. Junction tree for the model after the rank-one decomposition.

After a little algebra we get that the total clique size in the standard case is $(m+1)\cdot 2^m$, after parent divorcing it is $\frac{1}{3}m^3 + \frac{5}{2}m^2 + 2m\log m - \frac{11}{6}m - 1$, and after the tensor rank-one decomposition (described later in this paper) it is only $3m^2+4m+1$. In Figure 6, we compare dependence of the total size of junction trees on the number of parent nodes[4] m of node Y. Note that we use the logarithmic scale for the vertical axis.

It should be noted that the tensor rank-one decomposition can be applied to any probabilistic model. The savings depend on the graphical structure of the probabilistic model. In fact, by avoiding the moralization of the parents, we give the triangulation algorithm more freedom for the construction of a triangulated graph so that the resulting tables corresponding to the cliques of the triangulated graph can be smaller.

[4]It may seem unrealistic to have a node with more than ten parents in an real world application, but it can easily happen, especially, when one needs to introduce logical constraints into the model.

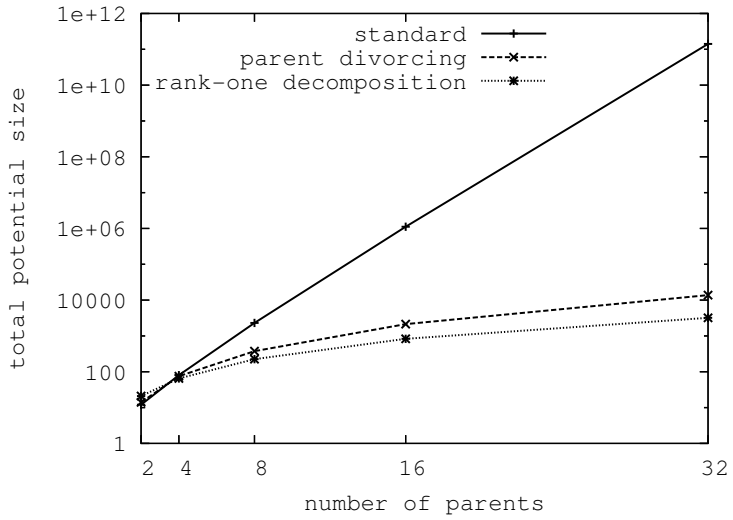

Fig. 6. Comparison of the total size of junction tree.

We would like to stress that the above mentioned methods are not the only methods that exploit local structure of conditional probability tables in probabilistic inference. Since an overview of these methods would substiantially go beyond scope of this paper we only provide few references to those we find most important.

One of the earliest examples is the Quickscore algorithm of Heckerman [7] which exploits noisy-or relations in the Quick Medical Reference model. Most of other approaches are based on a transformation of the network structure. Olesen et al. [14] proposed the parent divorcing method. Heckerman and Breese [9] used a temporal transformation. Zhang and Poole [18] introduced deputy variables that are used to create a heterogeneous factorization in which the factors can be combined either by multiplication or by a combination operator. Takikawa and D'Ambrosio [16] used intermediate (hidden) variables, which allowed them to transform an additive factorization into a multiplicative factorization. The additions are then achieved by standard marginalization of the intermediate variables. Díez [5] pointed out that the transformation of noisy-max can be done using a single variable.

A different approach that exploits local structures of CPTs are arithmetic circuits [2]. An arithmetic circuit is a rooted, directed acyclic graph. The leaf nodes are labelled with numeric constants or variables and all other nodes correspond to summation or multiplication. They are usually constructed from a Bayesian network via the conversion of Bayesian network to a multilinear function, which is then converted to a an algebraic circuit (usually via a logical formula of propositional logic) – see [1, 2] for details.

3. MINIMAL TENSOR RANK–ONE DECOMPOSITION

In this section we will present the main theoretical results. Recall that determining the minimal rank-one decomposition is an NP-hard problem [10]. However, we will provide closed-form solution for the problem of finding a minimal rank-one decomposition of some special tensors that play an important role, since they correspond to CPTs that are often used in Bayesian network models in real applications.

The class of tensors of our special interest are tensors ψ_f that represent a functional dependence of one variable Y on variables $X_1, \ldots, X_m$. Let $\mathcal{X} = \mathcal{X}_1 \times \ldots \mathcal{X}_m$ and $x = (x_1, \ldots, x_m) \in \mathcal{X}$. Further, let

$$I(expr) \quad = \quad \begin{cases} 1 & \text{if } expr \text{ is true} \\ 0 & \text{otherwise.} \end{cases}$$

Then for a function $f : \mathcal{X} \mapsto \mathcal{Y}$ the tensor is defined for all $(x, y) \in \mathcal{X} \times \mathcal{Y}$ as $\psi_f(x, y) = I(y = f(x))$.

Let $r = \text{rank}(\psi_f)$ and

$$\psi_f \quad = \quad \sum_{b=1}^{r} \xi_b \otimes (\otimes_{i=1}^{m} \varphi_{i,b}) \,, \tag{3}$$

where $\xi_b : \mathcal{Y} \mapsto \mathbb{R}$ and $\varphi_{i,b} : \mathcal{X}_i \mapsto \mathbb{R}$ for all $b \in \{1, \ldots, r\}$. Formula (3) is called a minimal tensor rank-one decomposition of ψ_f.

First, we will provide a lower bound on the rank of tensors from this class. This bound will be later used to prove the minimality of certain rank-one decompositions.

Lemma 1. Let function $f : \mathcal{X} \mapsto \mathcal{Y}$ and $\psi_f : \mathcal{X} \times \mathcal{Y} \mapsto \{0, 1\}$ be a tensor representing the functional dependence given by f. Then $\text{rank}(\psi_f) \geq |\mathcal{Y}|$.

P r o o f . For a minimal tensor rank-one decomposition of ψ_f it holds for all $(x, y) \in \mathcal{X} \times \mathcal{Y}$ that

$$\psi_f(x, y) \quad = \quad \sum_{b=1}^{r} \xi_b(y) \cdot \prod_{i=1}^{m} \varphi_{i,b}(x_i), \tag{4}$$

where $r = \text{rank}(\psi_f)$. Consider the matrices[5]

$$\begin{aligned} \boldsymbol{W} &= \{\psi_f(x, y)\}_{x \in \mathcal{X}}^{y \in \mathcal{Y}} \,, \\ \boldsymbol{U} &= \{\xi_b(y)\}_{b \in \{1,\ldots,r\}}^{y \in \mathcal{Y}} \,, \\ \boldsymbol{V} &= \left\{ \prod_{i=1}^{m} \varphi_{i,b}(x_i) \right\}_{x \in \mathcal{X}}^{b \in \{1,\ldots,r\}} \,. \end{aligned}$$

Equation (4) can be rewritten as

$$\boldsymbol{W} = \boldsymbol{U}\boldsymbol{V}.$$

Each row of $\boldsymbol{W}$ contains at least one nonzero entry, since each $y \in \mathcal{Y}$ is in the range of f. Moreover, each column of W contains exactly one nonzero entry, since f is

[5]The upper index labels the rows and the lower index labels the columns.

a function. Hence, no row is a linear combination of other rows. Therefore there are $|\mathcal{Y}|$ independent rows in $\boldsymbol{W}$ and $\text{rank}(\boldsymbol{W}) = |\mathcal{Y}|$. Since the rank of a matrix product cannot be higher than the rank of matrices in the product [6], we get that $\text{rank}(\boldsymbol{W}) \leq \text{rank}(\boldsymbol{U}) \leq r$. Altogether, $|\mathcal{Y}| \leq r$. □

3.1. Maximum and minimum

An additive decomposition of max was originally proposed in [5]. This result is included in this section for completeness and we add a result about its optimality. The proofs are constructive, i. e., they provide a minimal tensor rank-one decomposition for *max* and *min*.

Let us assume that $\mathcal{X}_i = [a_i, b_i]$ is an interval of integers for each $i = 1, \ldots, m$. Clearly, the range $\mathcal{Y}$ of max on $\mathcal{X}_1 \times \ldots \times \mathcal{X}_m$ is $[\max_{i=1}^m a_i, \max_{i=1}^m b_i]$ and the range $\mathcal{Y}$ of min is $[\min_{i=1}^m a_i, \min_{i=1}^m b_i]$.

Theorem 1. If $f(x) = \max\{x_1, \ldots, x_m\}$ and $x_i \in [a_i, b_i]$ for $i = 1, \ldots, m$ then $\text{rank}(\psi_f) = |\mathcal{Y}|$.

P r o o f. Let for $i \in \{1, \ldots, m\}$, $x_i \in \mathcal{X}_i$, and $b \in \mathcal{Y}$

$$\varphi_{i,b}(x_i) \quad = \quad \begin{cases} 1 & x_i \leq b \\ 0 & \text{otherwise} \end{cases}$$

and for $y \in \mathcal{Y}$, $b \in \mathcal{Y}$

$$\xi_b(y) \quad = \quad \begin{cases} +1 & b = y \\ -1 & b = y - 1 \\ 0 & \text{otherwise.} \end{cases}$$

Let $\omega(x, y) = I(\max(x_1, \ldots, x_m) \leq y)$. Put

$$\omega(x, y) \quad = \quad \prod_{i=1}^{m} \varphi_{i,y}(x_i).$$

Therefore,

$$\psi_f(x, y) \quad = \quad \omega(x, y) - \omega(x, y - 1),$$

where $\omega(x, y_{min} - 1)$ is considered to be zero. Altogether,

$$\psi_f(x, y) \quad = \quad \prod_{i=1}^{m} \varphi_{i,y}(x_i) - \prod_{i=1}^{m} \varphi_{i,y-1}(x_i).$$

Since the product $\xi_b(y) \cdot \prod_{i=1}^m \varphi_{i,b}(x_i)$ is nonzero only for $b = y$ and $b = y - 1$, we have

$$\begin{aligned} \psi_f(x, y) \quad &= \quad \xi_y(y) \cdot \prod_{i=1}^{m} \varphi_{i,y}(x_i) \;+\; \xi_{y-1}(y) \cdot \prod_{i=1}^{m} \varphi_{i,y-1}(x_i) \\ &= \quad \sum_{b \in \mathcal{Y}} \xi_b(y) \cdot \prod_{i=1}^{m} \varphi_{i,b}(x_i)\,. \end{aligned}$$

Taking $b' = b + y_{min} - 1$ we get the required decomposition

$$\psi_f \quad = \quad \sum_{b'=1}^{|\mathcal{Y}|} \xi_{b'} \otimes (\otimes_{i=1}^{m} \varphi_{i,b'}) .$$

By Lemma 1, this is a minimal tensor rank-one decomposition of ψ_f. □

Theorem 2. If $f(x) = \min\{x_1, \ldots, x_m\}$ and $x_i \in [a_i, b_i]$ for $i = 1, \ldots, m$ then $\mathrm{rank}(\psi_f) = |\mathcal{Y}|$.

P r o o f. Let for $i \in \{1, \ldots, m\}$, $x_i \in \mathcal{X}_i$, and $b \in \mathcal{Y}$

$$\varphi_{i,b}(x_i) \quad = \quad \begin{cases} 1 & x_i \geq b \\ 0 & \text{otherwise} \end{cases}$$

and for $y \in \mathcal{Y}$, $b \in \mathcal{Y}$

$$\xi_b(y) \quad = \quad \begin{cases} +1 & b = y \\ -1 & b = y + 1 \\ 0 & \text{otherwise.} \end{cases}$$

and follow an analogous argument as in the proof of Theorem 1 to obtain a tensor rank-one decomposition of ψ_f. Again, by Lemma 1, this is a minimal tensor rank-one decomposition. □

Remark. If for $i \in \{1, \ldots, m\}$ $\mathcal{X}_i = \{0, 1\}$, then the functions $\max\{x_1, \ldots, x_m\}$ and $\min\{x_1, \ldots, x_m\}$ correspond to logical disjunction $x_1 \vee \ldots \vee x_m$ and logical conjunction $x_1 \wedge \ldots \wedge x_m$, respectively. In Example 2 we illustrate how this can be generalized to Boolean expressions consisting of negations and disjunctions.

Example 2. In order to achieve a minimal tensor rank-one decomposition of

$$\psi(x_1, x_2, y) = I(y = (x_1 \vee \neg x_2))$$

with variable B having two states 0 and 1, it is sufficient to use functions:

$$\begin{aligned} \varphi_{1,b}(x_1) &= I(x_1 \leq b) \\ \varphi_{2,b}(x_2) &= I(\neg x_2 \leq b) \\ \xi_b(y) &= \begin{cases} +1 & y = b \\ -1 & y = 1,\ b = 0 \\ 0 & y = 0,\ b = 1 \end{cases} \end{aligned}$$

3.2. Addition

In this section, we assume an integer r_i for each $i = 1, \ldots, m$ and assume that $\mathcal{X}_i$ is the interval of integers $[0, r_i]$. This assumption is made for simplicity and without loss of generality. If $\mathcal{X}_i$ are intervals of integers, which do not start at zero, it is possible to transform the variables by subtracting the lower bounds of the intervals to obtain variables satisfying the assumption. Moreover, let $f : \mathbb{N}^m \to \mathbb{N}$ be a function, such that $f(x) = f_0(\sum_{i=1}^{m} x_i)$ where $f_0 : \mathbb{N} \to \mathbb{N}$. Let $\mathcal{A}$ be the interval of integers $[0, \sum_{i=1}^{m} r_i]$. Clearly, $\mathcal{A}$ is the range of $\sum_{i=1}^{m} x_i$.

Theorem 3. Let f_0, f and $\mathcal{A}$ be as above. Then $\mathrm{rank}(\psi_f) \leq |\mathcal{A}|$. Moreover, if f_0 is the identity function, then $\mathrm{rank}(\psi_f) = |\mathcal{A}|$.

P r o o f. Consider the Vandermonde matrix

$$\begin{pmatrix} \alpha_1^0 & \alpha_2^0 & \dots & \alpha_{|\mathcal{A}|}^0 \\ \alpha_1^1 & \alpha_2^1 & \dots & \alpha_{|\mathcal{A}|}^1 \\ \dots & & & \\ \alpha_1^{|\mathcal{A}|-1} & \alpha_2^{|\mathcal{A}|-1} & \dots & \alpha_{|\mathcal{A}|}^{|\mathcal{A}|-1} \end{pmatrix}, \tag{5}$$

where $\alpha_1, \dots, \alpha_{|\mathcal{A}|}$ are pairwise distinct real numbers and the upper index is the exponent. This matrix is non-singular, since the corresponding Vandermonde determinant is non-zero [6]. The system of equations

$$\beta_t \quad = \quad \sum_{b=1}^{|\mathcal{A}|} z_b \cdot \alpha_b^t, \quad t \in \mathcal{A} \tag{6}$$

has the Vandermonde matrix as its system matrix and since the Vandermonde determinant is non-zero it has always a solution for variables $z_b, b \in \mathcal{A}$.

For a fixed $y \in \mathcal{Y}$, define $\xi_b(y)$ to be the solution of system (6) with $\beta_t = I(y = f_0(t))$. Therefore it holds for a fixed $y \in \mathcal{Y}$ and for all $t \in \mathcal{A}$ that

$$I(y = f_0(t)) \quad = \quad \sum_{b=1}^{|\mathcal{A}|} \xi_b(y) \cdot \alpha_b^t. \tag{7}$$

By substituting $t = \sum_{i=1}^{m} x_i$ and taking $\varphi_{i,b}(x_i) = \alpha_b^{x_i}$ for $b = 1, \dots, |\mathcal{A}|$ we obtain that for all combinations of the values of x and y

$$I(y = f_0(\sum_{i=1}^{m} x_i)) \quad = \quad \sum_{b=1}^{|\mathcal{A}|} \xi_b(y) \cdot \prod_{i=1}^{m} \varphi_{i,b}(x_i).$$

This proves the first assertion of the theorem.

If f_0 is the identity, then the range of f is the whole $\mathcal{A}$. It follows from Lemma 1 that $\mathrm{rank}(\psi_f) \geq |\mathcal{A}|$ and therefore the above decomposition is minimal. □

Example 3. Let $\mathcal{X}_i = \{0, 1\}$ for $i = 1, 2$, $f(x_1, x_2) = x_1 + x_2$ and $\mathcal{Y} = \{0, 1, 2\}$. We have

$$\begin{aligned} \psi_f(x_1, x_2, y) \quad &= \quad I(y = x_1 + x_2) \\ &= \quad \begin{pmatrix} \begin{pmatrix} 1 \\ 0 \\ 0 \end{pmatrix} & \begin{pmatrix} 0 \\ 1 \\ 0 \end{pmatrix} \\ \begin{pmatrix} 0 \\ 1 \\ 0 \end{pmatrix} & \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix} \end{pmatrix} \end{aligned}$$

As in the proof of Theorem 3, we assume $\varphi_{i,b}(x_i) = \alpha_b^{x_i}$ for $i = 1, 2$ and distinct α_b, $b = 1, 2, 3$. For simplicity of notation, let us assume $\alpha_1 = \alpha$, $\alpha_2 = \beta$, $\alpha_3 = \gamma$, $\xi_1(y) = u_y$, $\xi_2(y) = v_y$, and $\xi_3(y) = w_y$. Let us substitute these $\varphi_{i,b}(x_i)$ and $\xi_b(y)$ into (3) and rewrite it using tensor product as follows.

$$\begin{aligned}\psi_f(x_1, x_2, y) &= (\alpha^0, \alpha^1) \otimes (\alpha^0, \alpha^1) \otimes (u_0, u_1, u_2) \\ &\quad +(\beta^0, \beta^1) \otimes (\beta^0, \beta^1) \otimes (v_0, v_1, v_2) \\ &\quad +(\gamma^0, \gamma^1) \otimes (\gamma^0, \gamma^1) \otimes (w_0, w_1, w_2).\end{aligned}$$

For each $y = 0, 1, 2$ we require

$$\begin{pmatrix} I(y=0) & I(y=1) \\ I(y=1) & I(y=2) \end{pmatrix} = u_y \cdot \begin{pmatrix} \alpha^0 & \alpha^1 \\ \alpha^1 & \alpha^2 \end{pmatrix} + v_y \cdot \begin{pmatrix} \beta^0 & \beta^1 \\ \beta^1 & \beta^2 \end{pmatrix} + w_y \cdot \begin{pmatrix} \gamma^0 & \gamma^1 \\ \gamma^1 & \gamma^2 \end{pmatrix},$$

which defines a system of three linear equations with three variables u_y, v_y, w_y

$$\begin{pmatrix} I(y=0) \\ I(y=1) \\ I(y=2) \end{pmatrix} = \begin{pmatrix} \alpha^0 & \beta^0 & \gamma^0 \\ \alpha^1 & \beta^1 & \gamma^1 \\ \alpha^2 & \beta^2 & \gamma^2 \end{pmatrix} \cdot \begin{pmatrix} u_y \\ v_y \\ w_y \end{pmatrix}.$$

If α, β, and γ are pairwise distinct real numbers then the corresponding Vandermonde determinant is non-zero and a solution exists. The solution for $\alpha = 1, \beta = 2, \gamma = 3$ is

$$\begin{aligned}\psi_f(x_1, x_2, y) &= (1,1) \otimes (1,1) \otimes \left(3, -\frac{5}{2}, \frac{1}{2}\right) \\ &\quad + (1,2) \otimes (1,2) \otimes (-3, 4, -1) \\ &\quad + (1,3) \otimes (1,3) \otimes \left(1, -\frac{3}{2}, \frac{1}{2}\right).\end{aligned}$$

3.3. Generalized addition

In this section, we present a tensor rank-one decomposition of ψ_f, where f is defined as $f(x) = f_0(\sum_{i=1}^m f_i(x_i))$. Let $\mathcal{A}$ be the set of all possible values of $\sum_{i=1}^m f_i(x_i)$. The rank of ψ_f depends on the nature of functions f_i, more exactly, on the range of the values of $\sum_{i=1}^m f_i(x_i)$. The decomposition is useful, if this range is substantially smaller than $|\mathcal{X}_1| \cdot \ldots \cdot |\mathcal{X}_m|$.

Theorem 4. If $f(x) = f_0(\sum_{i=1}^m f_i(x_i))$, where f_i are integer valued functions, then $\text{rank}(\psi_f) \leq |\mathcal{A}|$.

Proof. Without a loss of generality, we may assume that $f_i(x_i) \geq 0$ for $i = 1, \ldots, m$ and that zero is in the range of f_i. If not, this may be achieved by using

$f_i(x_i) - \min_{z_i} f_i(z_i)$ instead of f_i and modifying f_0 so that f does not change. The proof is analogous to the proof of the Theorem 3, where $t = \sum_{i=1}^{m} f_i(x_i)$ and $\varphi_{i,b}(x_i) = \alpha_b^{f(x_i)}$ for $b = 1, \ldots, |\mathcal{A}|$. □

3.4. Exclusive-or (parity) function

Let $\oplus$ denote the addition modulo two, which is also known as the exclusive-or operation. Parity is often used in coders and decoders. We conjecture tensor rank-one decomposition may substantially speed up exact inference in probabilistic graphical models used to model decoders for noisy channels. By the parity or exclusive-or function, we will understand the function $x_1 \oplus \ldots \oplus x_m$.

Theorem 5. Let $\mathcal{X}_i = \mathcal{Y} = \{0, 1\}$ for $i = 1, \ldots, m$ and $f(x) = x_1 \oplus \ldots \oplus x_m$. Then $\mathrm{rank}(\psi_f) = 2$.

P r o o f. The exclusive-or function may easily be expressed as a product, if the values $\{0, 1\}$ are replaced by $\{1, -1\}$ using substitution $0 \mapsto 1$, $1 \mapsto -1$. An odd number of ones in the 0/1 representation is equivalent to a negative product of the corresponding values in the $1/-1$ representation. Expressing the required transformations in the form of a linear transformation, we obtain

$$x_1 \oplus \ldots \oplus x_m = \frac{1}{2}(1 - (1 - 2x_1) \ldots (1 - 2x_m)).$$

Since $\psi_f(x, y) = I(y \oplus x_1 \oplus \ldots \oplus x_m = 0) = y \oplus x_1 \oplus \ldots \oplus x_m \oplus 1$, we have

$$\psi_f(x, y) \quad = \quad \frac{1}{2}(1 + (1 - 2y)(1 - 2x_1) \ldots (1 - 2x_m)).$$

Hence, ψ_f may be expressed as a sum of two functions, the first of which is the constant $\frac{1}{2}$ and the second is $(\frac{1}{2} - y)(1 - 2x_1) \ldots (1 - 2x_m)$. It is now easy to express ψ_f in the form of (3), if we use tensors defined as follows. Let for $i \in \{1, \ldots, m\}$, $x_i \in \{0, 1\}$, and $b \in \{1, 2\}$

$$\varphi_{i,b}(x_i) \quad = \quad \begin{cases} 1 & b = 1 \\ 1 - 2x_i & b = 2 \end{cases}$$

and

$$\xi_b(y) \quad = \quad \begin{cases} \frac{1}{2} & b = 1 \\ \frac{1}{2} - y & b = 2\,. \end{cases}$$

It follows from Lemma 1 that this defines a minimal tensor rank-one decomposition of exclusive-or. □

3.5. Noisy functional dependence

For every $i = 1, \ldots, m$ we define a dummy variable X_i' taking values x_i' from set $\mathcal{X}_i' = \mathcal{X}_i$. The noisy functional dependence of Y on $X = (X_1, \ldots, X_m)$ is defined by

$$\psi_{f,\varkappa_1,\ldots,\varkappa_m}(x, y) \quad = \quad \sum_{x'} \psi_f(x', y) \cdot \prod_{i=1}^{m} \varkappa_i(x_i, x_i'), \tag{8}$$

where ψ_f is tensor that represent a functional dependence $y = f(x')$ and for $i = 1, \ldots, m$ tensors $\varkappa_i$ represent the noise for variable X_i. Note that models like noisy-or, noisy-and, etc., fall within the scope of the above definition. Actually, the definition covers the whole class of models known as models of independence of causal influence (ICI) [8].

Theorem 6. Let tensor $\psi_{f,\varkappa_1,\ldots,\varkappa_m}$ represent the noisy functional dependence f defined by formula (8). Then $\mathrm{rank}(\psi_{f,\varkappa_1,\ldots,\varkappa_m}) \leq \mathrm{rank}(\psi_f)$.

P r o o f. Let $r = \mathrm{rank}(\psi_f)$. Then

$$\psi_f(x', y) \quad = \quad \sum_{b=1}^{r} \xi_b(y) \cdot \prod_{i=1}^{m} \varphi_{i,b}(x'_i).$$

Substituting this to formula (8) we get

$$\begin{aligned}
\psi_{f,\varkappa_1,\ldots,\varkappa_m}(x, y) &= \sum_{x'} \sum_{b=1}^{r} \xi_b(y) \cdot \prod_{i=1}^{m} (\varphi_{i,b}(x'_i) \cdot \varkappa_i(x_i, x'_i)) \\
&= \sum_{b=1}^{r} \xi_b(y) \prod_{i=1}^{m} \sum_{x'_i} (\varphi_{i,b}(x'_i) \cdot \varkappa_i(x_i, x'_i)) \\
&= \sum_{b=1}^{r} \xi_b(y) \cdot \prod_{i=1}^{m} \varphi'_{b,i}(x_i),
\end{aligned}$$

where $\varphi'_{b,i}(x_i) = \sum_{x'_i} (\varphi_{i,b}(x'_i) \cdot \varkappa_i(x_i, x'_i))$.

The last equation proves that $\mathrm{rank}(\psi_{f,\varkappa_1,\ldots,\varkappa_m}) \leq \mathrm{rank}(\psi_f)$. □

4. NUMERICAL METHOD

In the previous section we showed how a minimal tensor rank-one decomposition can be found for some special CPTs. But since it would be useful to get tensor rank-one decompositions for CPTs with an unknown closed-form solution we tested several numerical methods.

Recall that a tensor rank-one approximation of length s (Definition 4) is a series $\{\varrho_b\}_{b=1}^{s}$ of rank-one tensors ϱ_b that is a tensor rank-one decomposition of a tensor $\hat{\psi}$ with $\mathrm{rank}(\hat{\psi}) = s$. If $\hat{\psi}$ minimizes $\sum_x (\psi(x) - \hat{\psi}(x))^2$ we say that it is a *best tensor rank-one approximation of length s*.

Note that if $s = \mathrm{rank}(\psi)$ then the minimal value of $\sum_x (\psi(x) - \hat{\psi}(x))^2$ is zero and a best tensor rank-one approximation of length s is also a minimal tensor rank-one decomposition of ψ. Therefore, one can search numerically for a minimal tensor rank-one decomposition by solving the task from Definition 4 starting with $s = 1$ and then incrementing s by one until $\sum_x (\psi(x) - \hat{\psi}(x))^2$ is sufficiently close to zero.

We performed tests with several gradient methods. The best performance was achieved with Polak–Ribière conjugate gradient method [15] that used the Newton method in one dimension. We performed experiments for tensors corresponding to

the exclusive-or and maximum functions of three binary variables. The tensor rank of these functions is two therefore we were able to verify whether for $s = 2$ the algorithm found a tensor rank-one decomposition of these tensors.

The initial values for the algorithm were random numbers from interval $[-0.5, +0.5]$. In most cases, the algorithm converged to vectors that were tensor rank-one decomposition. However, sometimes we needed to restart the algorithm from another starting values since it got stuck in a local minima. Figures 7 and 8 illustrate the convergence using three sample runs. The displayed value is one value of $\hat{\psi}$ as it changes with the progress of the algorithm.

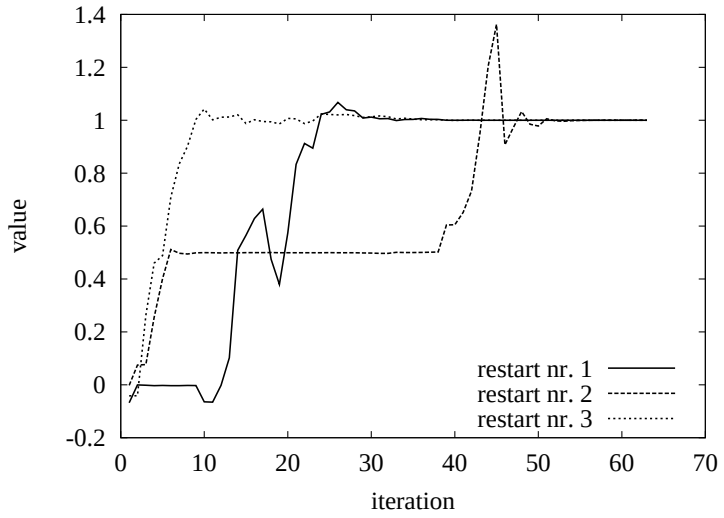

Fig. 7. Development of one value of ψ' in case of decomposition of xor.

We also applied the numerical algorithm to tensors for which we do not know the rank. Namely, it was the family of tensors corresponding to CPTs representing Boolean functions – the Boolean threshold functions and exactly-t functions.

Definition 5. Boolean *threshold function* is a Boolean function f such that

$$f(x) \quad = \quad \begin{cases} 0 & \text{if } |x| < t \\ 1 & \text{if } |x| \geq t. \end{cases}$$

where $t \in \{0, \ldots, m+1\}$ is the threshold value.

Note that if $t = 0$ then it corresponds to the *true* function, if $t = m + 1$ it is the *false* function, if $t = 1$ then it is the *or* function, and if $t = m$ then it is the *and* function.

Definition 6. Let f_{t+1} be the Boolean threshold function with threshold $t + 1$ and f_{t+2} be the Boolean threshold function with threshold $t + 2$. Then for any $t \in \{1, m-1\}$ the *exactly-t function* is function $g_t(x) = f_{t+1}(x) - f_{t+2}(x)$.

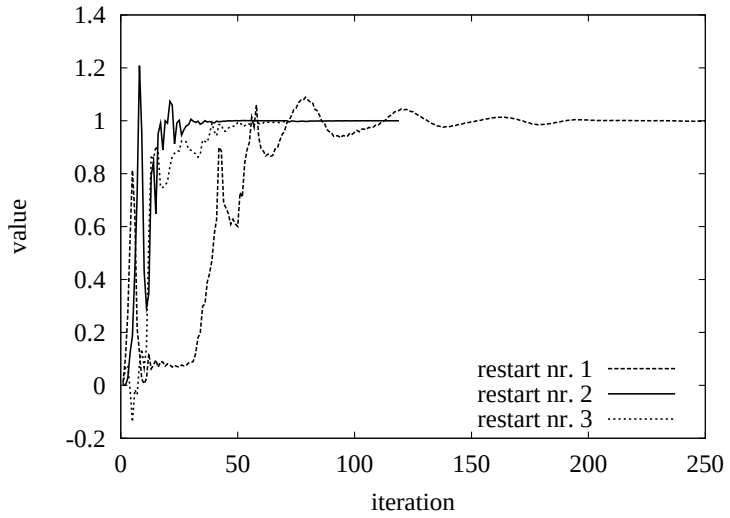

Fig. 8. Development of one value of ψ' in case of decomposition of max.

To compute numerically the rank of tensors corresponding to the threshold function, the exactly-t function, and a general randomly generated Boolean functions we again used the Polak–Ribière conjugate gradient method. For each $s = 1, 2, \ldots$ we searched for the best tensor rank-one approximation of length s until we found an approximation that was sufficiently close to the tensor of a given Boolean function (we again started the algorithm from ten different randomly generated starting points). As a stopping criterion we used the condition

$$\sum_x (\psi(x) - \hat{\psi}(x))^2 \quad < \quad 10^{-5}.$$

The lowest value of s for which the above condition was met we regarded to be the rank of the tensor of the given Boolean function.

Table 1. Rank of the threshold Boolean functions with threshold t compared with the average rank and the rank interval of randomly generated Boolean functions.

t	0	1	2	3	4	5	6	random
$m = 2$	1	2	2	1	1	1	1	1.92 [1,2]
$m = 3$	1	2	3	2	1	1	1	2.78 [2,3]
$m = 4$	1	2	3	3	2	1	1	3.94 [3,5]
$m = 5$	1	2	3	4	3	2	1	6.62 [5,8]

In Table 1 we compare the rank of the threshold Boolean functions with threshold t compared with the average rank and the rank interval of randomly generated

Boolean functions. We did the computations for different number of variables m. The average was computed from tensor rank values of fifty randomly generated Boolean functions. The values in brackets correspond to the interval of the rank values. Note that this does not mean that there do not exist Boolean functions with lower or higher rank. In Table 2 we present similar comparisons for the exactly-t function.

Table 2. Rank of the exactly-t function compared with the average rank and the rank interval of randomly generated Boolean functions.

t	0	1	2	3	4	5	6	random
$m = 2$	2	2	2	1	1	1	1	1.92 [1,2]
$m = 3$	2	3	3	2	1	1	1	2.78 [2,3]
$m = 4$	2	3	4	3	2	1	1	3.94 [3,5]
$m = 5$	2	3	4	4	3	2	1	6.62 [5,8]

From the tables one can see that the rank of the threshold Boolean functions (for $m > 3$) and also the exactly-t Boolean functions (for $m > 4$) is lower than rank of a randomly generated Boolean functions. The lower the rank, the more compact is the tensor rank-one decomposition, which implies that probabilistic inference with these models can be performed more efficiently.

5. CONCLUSIONS

In many applications of Bayesian networks, special types of relations between variables are used. The transformation applied in this paper exploits functional dependence in order to achieve substantially more efficient probabilistic inference. The transformation is based on introducing an auxiliary variable. Fewer states of the auxiliary variable mean more efficient probabilistic inference. We observed the correspondence between the problem of minimal number of states of the auxiliary variable and the problem of the rank of a tensor. We proposed a new factorization for the addition and parity functions and their noisy counterparts. The factorization with an auxiliary variable can be also used to approximate large potentials. This could be used in an approximate propagation method when the exact inference is not possible.

ACKNOWLEDGMENTS

The authors were supported by the Ministry of Education of the Czech Republic under the project nr. 1M0545 (P. Savicky) and nr. 1M0572 (J. Vomlel). J. Vomlel was also supported by the Grant Agency of the Czech Republic under the project nr. 201/04/0393.

(Received June 12, 2006.)

REFERENCES

[1] M. Chavira and A. Darwiche: Compiling Bayesian networks with local structure. In: Proc. 19th Internat. Joint Conference on Artificial Intelligence (IJCAI), Edinburgh 2005, pp. 1306–1312.

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

[3] L. De Lathauwer and B. De Moor: From matrix to tensor: multilinear algebra and signal processing. In: 4th Int. Conf. on Mathematics in Signal Processing, Part I, IMA Conf. Series, Warwick 1996, pp. 1–11.

[4] L. De Lathauwer, B. De Moor, and J. Vandewalle: On the best Rank-1 and Rank-$(R_1, R_2, \ldots, R_N)$ approximation of higher-order tensors. SIAM Journal on Matrix Analysis and Applications *21* (2000), 4, 1324–1342.

[5] F. J. Díez and S. F. Galán: An efficient factorization for the noisy MAX. Internat. J. Intelligent Systems *18* (2003), 2, 165–177.

[6] G. H. Golub and C. F. Van Loan: Matrix Computations. Johns Hopkins University Press, Baltimore 1996. Third edition.

[7] D. Heckerman: A tractable inference algorithm for diagnosing multiple diseases. In: Proc. Fifth Annual Conf. on Uncertainty in AI (M. Henrion, R. D. Shachter, L. N. Kanal, and J. F. Lemmer, eds.), 1989, pp. 163–171.

[8] D. Heckerman: Causal independence for knowledge acquisition and inference. In: Proc. Ninth Conf. on Uncertainty in AI (D. Heckerman and A. Mamdani, eds.), 1993, pp. 122-127.

[9] D. Heckerman and J. S. Breese: A new look at causal independence. In: Proc. Tenth Conf. on Uncertainty in AI (R. Lopez de Mantaras and D. Poole, eds.), 1994, pp. 286–292.

[10] J. Håstad: Tensor Rank is NP-complete. Journal of Algorithms *11* (1990), 644–654.

[11] F. V. Jensen: Bayesian Networks and Decision Graphs. (Statistics for Engineering and Information Science), Springer Verlag, New York – Berlin – Heidelberg 2001.

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

[13] S. L. Lauritzen: Graphical Models. Clarendon Press, Oxford 1996.

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

[15] E. Polak: Computational Methods in Optimization: A Unified Approach. Academic Press, 1971.

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

[17] J. Vomlel: Exploiting functional dependence in Bayesian network inference. In: Proc. Eightteenth Conf. on Uncertainty in AI (UAI), Morgan Kaufmann Publishers, 2002, pp. 528–535.

[18] N. L. Zhang and D. Poole: Exploiting causal independence in Bayesian network inference. J. Artificial Intelligence Research *5* (1996), 301–328.

Petr Savicky, Institute of Computer Science — Academy of Sciences of the Czech Republic, Pod Vodárenskou věží 2, 182 07 Praha 8. Czech Republic.
e-mail: savicky@cs.cas.cz

Jiří Vomlel, Institute of Information Theory and Automation — Academy of Sciences of the Czech Republic, Pod Vodárenskou věží 4, 182 08 Praha 8. Czech Republic.
e-mail: vomlel@utia.cas.cz