On a Possibility of Gradual Model-Learning

Radim Jiroušek RADIM@UTIA.CAS.CZ
Czech Acad Sci, Inst Inform Th & Autom, Praha
(and) Faculty of Management, Jindřichův Hradec

Abstract

In this paper, the term of gradual learning describes the process, in which an n-dimensional model is constructed in n steps; each step increases the dimensionality of the constructed model by one. The approach is explained using the apparatus of compositional models since its algebraic properties seem to serve the purpose best. The paper shows also the equivalence of compositional models and Bayesian networks, and thus the paper gives a hint that the approach applies to the graphical model as well.

Keywords: Compositional model; multidimensionality; conditional independence; model-learning; Bayesian network.

1. Introduction

As it can be guessed from the title, the paper does not present an algorithm for gradual model learning, it just proves that such algorithms may exist. All the more, it does not discuss the computational efficiency of such procedures. The term of gradual learning should not be confused with the "Incremental Model Learning", which is stably used in the machine learning community to describe those methods that use new input data to modify (hopefully to improve) the existing model (see e.g., (Utgoff, 1989)). In this paper, we suggest an idea of an approach to model learning that, to construct a multidimensional model, starts with a one-dimensional model, then two-dimensional one, and so on, until the constructed model reaches the required dimensionality. Thus, it is based on the idea that a model construction can be viewed as an inverse process to model reduction.

If used for Bayesian network learning, this approach starts with a node representing the probability distribution for one variable, one source-node of the final Bayesian network. Then, the process adds another variable assigned to a child of the first variable, and so on, until the full Bayesian network is constructed. There is not a simpler way to get an optimum Bayesian network than that just described. This statement holds only if one has two oracles at their disposal. The first oracle advises, which variable should be selected at each step, the other oracle advises, which nodes should be the parents of the currently added variable.

Unfortunately, not having such oracles at our disposal, the process of getting the optimum Bayesian network turns to be difficult (NP-complete (Chickering, 1996)). During the last thirty years, abundant literature on different approaches to Bayesian network learning (not relying on the above-mentioned oracles) was published. They use a great variety of tools from those based on information theory (Heckerman et al., 1995) and minimum description length principle (Lam and Bacchus, 1994) to those employing seemingly unrelated fields of mathematics (Leung and Lee, 1994). From the latter group of papers, let us cite the paper by Park and Klabjan (2017), which solves exactly the problem that is expected from the second oracle: the knowledge of an ordering of the variables, which is topological with respect the resulting Bayesian network. As pointed out

by one of the anonymous reviewers, the below-presented approach shows some similarity with the dynamic programming approaches to model learning starting with (Singh and Moore, 2005).

In this paper, we want to show that the oracle determining the topological order of the variables may be bypassed. It may be bypassed even though it seems to be crucial for the goal of getting an optimal (or suboptimal) model. This is based on a very simple idea: Having a Bayesian network, we can subsequently delete its terminal nodes until the network diminishes. The order of deleting the nodes is not arbitrary, one may delete only terminal nodes, and this is why we need the oracle when realizing the inverse process. But, Ross Shachter (1986, 1988) presented a procedure based on two rules of *node deletion* and *edge reversal* that allow for the reduction of a Bayesian network, which deletes the variables in an arbitrary order. Therefore, there should also be an inverse process that gradually increases the Bayesian network adding the variables in an arbitrary order.

The other oracle mentioned above advises how the variables are to be interconnected in the model. It means, using the terminology of probability theory, it gives evidence about the conditional independence relations among the variables. Some authors, like e.g., Švorc and Vomlel (2019), rely on expert knowledge. Data-based model learning processes usually use different statistical tests for this purpose. To avoid problems that are not in the focus of this paper, like those highlighted by Edwards and Havránek (1987), who coped with the fact that the results of statistical testing may be misleading[1], we assume that these relations are known. Thus, instead of asking the statistics for help, we can keep an idea of the oracle advising us about the relations of conditional independence among the variables. Thus, we assume that the relations approved by the oracle meet all the required theoretical properties.

To describe the proposed way of model learning, we do not use the terminology of Bayesian networks and graphs. The reader familiar with the Shachter's approach surely understands that the inverse process of his *edge reversal* procedure (including *parents inheritance* rule) would be rather difficult to describe. Instead, we believe the description of this study will be more lucid if we use the terminology of probabilistic compositional models. Therefore, in the following Section, we introduce the necessary notions and notation and show the equivalence of Bayesian networks and compositional models. Section 3 is a brief introduction to compositional model theory, and Section 4 introduces the operator that serves the same purpose as the Shachter's edge reversal rule. We call it an anticipating operator, and it makes the necessary modifications of the model structure possible. The description of the model construction process, illustrated with examples, is the content of Section 5.

2. Basic Notions and Notation

In this paper, we denote random variables by lower-case characters from the end of the Latin alphabet (u, v, w, ...). All the considered variables are assumed to be finite-valued. $\mathbb{X}_u$, $\mathbb{X}_v$, ... denote the finite sets of values of variables u, v, ... Sets of variables are denoted by upper-case characters K, L, V, ... Thus, K may be, say, $\{u, v, w\}$. By a *state* of variables K we understand any combination of values of the respective variables, i.e., in the considered case $K = \{u, w, w\}$, a state is an

1. Edwards and Havránek coped with problems arising from the fact that results of statistical tests need not be consistent with principles of probabilistic modeling. If a model corresponds to the data structure (system of conditional independence relations) well, then all its submodels (defining only a subset of conditional independence relations) should correspond to the data as well. And yet, it need not hold for statistical tests of models (for details, the reader is referred to (Edwards and Havránek, 1987)).

element of a Cartesian product $\mathbb{X}_K = \mathbb{X}_u \times \mathbb{X}_v \times \mathbb{X}_w$. For a state $\mathbf{a} \in \mathbb{X}_K$ and $L \subset K$, $\mathbf{a}^{\downarrow L}$ denote a *projection* of $\mathbf{a} \in \mathbb{X}_K$ into $\mathbb{X}_L$, i.e., $\mathbf{a}^{\downarrow L}$ is the state from $\mathbb{X}_L$ that is got from $\mathbf{a}$ by dropping out all the values of variables from $K \setminus L$.

Probability distributions are denoted by characters of Greek alphabet (κ, λ, μ, $\pi, \ldots$). Recall that it means that $\kappa(K) : \mathbb{X}_K \longrightarrow [0, 1]$, for which $\sum_{\mathbf{a}\in\mathbb{X}_\mathbf{K}} \kappa(\mathbf{a}) = 1$. For a probability distribution $\kappa(K)$, and a subset of variables $L \subset K$, $\kappa^{\downarrow L}$ denote a *marginal distribution* of κ defined for each $\mathbf{a} \in \mathbb{X}_L$ by the formula

$$\kappa^{\downarrow L}(\mathbf{a}) = \sum_{\mathbf{c}\in\mathbb{X}_K : \mathbf{c}^{\downarrow L} = \mathbf{a}} \kappa(\mathbf{c}). \tag{1}$$

Consider two distributions $\kappa(K)$ and $\lambda(L)$. We say that κ and λ are *consistent* if $\kappa^{\downarrow K\cap L} = \lambda^{\downarrow K\cap L}$. For two probability distributions defined for the same group of variables, say $\pi(K)$, $\kappa(K)$, we say that κ *dominates* π (in symbol $\pi \ll \kappa$) if

$$\forall\, \mathbf{a} \in \mathbb{X}_K \quad (\kappa(\mathbf{a}) = 0 \implies \pi(\mathbf{a}) = 0)\,.$$

Consider a probability distribution $\pi(V)$, and three disjoint subsets of variables K, L, M ($K \cup L \cup M \subseteq V$). Let K and L be nonempty. Symbol $\pi^{K|M}$ is used to denote the respective conditional distribution of variables K given M, for which $\pi^{K|M} \cdot \kappa^{\downarrow M} = \kappa^{\downarrow K\cup M}$. We say that groups of variables K and L are *conditionally independent given M for distribution π* if

$$\pi^{\downarrow K\cup L\cup M} \cdot \pi^{\downarrow M} = \pi^{\downarrow K\cup M} \cdot \pi^{\downarrow L\cup M}, \tag{2}$$

in symbol $K \perp\!\!\!\perp_\pi L|M$. Thus, $(\pi^{\downarrow K\cup L\cup M} = \pi^{\downarrow K\cup M} \cdot \pi^{L|M})$ implies $K \perp\!\!\!\perp_\pi L|M$. In case of $M = \emptyset$ we use only $K \perp\!\!\!\perp_\pi L$ and speak about an unconditional independence.

Using this notation, a Bayesian network is a couple $(\mathbf{G}, \mathcal{S})$, where $\mathbf{G}$ is an acyclic directed graph; its set of variables coincides with the set of variables V, and there is an edge $(u \to v)$ in $\mathbf{G}$ if $u \in pa(v)$. $\mathcal{S}$ denotes a system of conditional probability distributions

$$\mathcal{S} = \{\pi^{\{u\}|pa(u)} : \forall u \in V\}.$$

This Bayesian network represents the probability distribution

$$\prod_{u\in V} \pi^{\{u\}|pa(u)}. \tag{3}$$

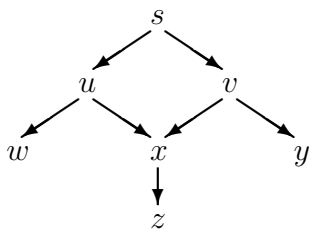

Figure 1: Bayesian network representing $\pi(s, u, v, w, x, y, z)$, $pa(s) = \emptyset$, $pa(u) = pa(v) = \{s\}$, $pa(w) = \{u\}$, $pa(x) = \{u, v\}$, $pa(y) = \{v\}$, $pa(z) = \{x\}$.

3. Compositional models

Definition 1 *For arbitrary two distributions $\kappa(K)$ and $\lambda(L)$, for which $\lambda^{\downarrow K\cap L}$ dominates $\kappa^{\downarrow K\cap L}$, their* composition *is for each $\mathbf{a} \in \mathbb{X}_{K\cup L}$ given by the following formula*[2]

$$(\kappa \triangleright \lambda)(\mathbf{a}) = \frac{\kappa(\mathbf{a}^{\downarrow K})\lambda(\mathbf{a}^{\downarrow L})}{\lambda^{\downarrow K\cap L}(\mathbf{a}^{\downarrow K\cap L})}. \tag{4}$$

In case that $\kappa^{\downarrow K\cap L} \not\ll \lambda^{\downarrow K\cap L}$ the composition remains undefined.

2. Define $\frac{0\cdot 0}{0} = 0$.

By a *compositional model*, we understand a multidimensional probability distribution assembled from a sequence of low-dimensional distributions with the help of the introduced operator of composition, i.e., $\kappa_1 \triangleright \kappa_2 \triangleright \ldots \triangleright \kappa_n$. Since the operator of composition is not associative, this expression is ambiguous. To avoid this ambiguity, let us make a convention that we omit the parentheses if the operators are to be performed from left to right:

$$\kappa_1 \triangleright \kappa_2 \triangleright \ldots \triangleright \kappa_n = (\ldots((\kappa_1 \triangleright \kappa_2) \triangleright \kappa_3) \triangleright \ldots \triangleright \kappa_{n-1}) \triangleright \kappa_n. \tag{5}$$

For distributions $\kappa_1(K_1), \kappa_2(K_2), \ldots, \kappa_n(K_n)$, Formula (5) can be rewritten into the form

$$\kappa_1 \triangleright \kappa_2 \triangleright \ldots \triangleright \kappa_n = \prod_{i=1}^{n} \kappa_i^{K_i \setminus (K_1 \cup \ldots, \cup K_{i-1}) | K_i \cap (K_1 \cup \ldots, \cup K_{i-1})}.$$

It means that every distribution represented in a form of a Bayesian network (see Formula (3)) can also be represented in a form of a compositional model:

$$\prod_{u \in V} \pi^{\{u\}|pa(u)} = \pi(\{u_1\}) \triangleright \pi(\{u_2\} \cup pa(u_2)) \triangleright \ldots \triangleright \pi(\{u_n\} \cup pa(u_n)),$$

for any topological ordering (the ordering, in which the parents are always before their children) of nodes of the considered Bayesian network.

It is not difficult to show that each distribution represented in a form of a compositional model can also be represented in a form of a Bayesian network. The reader can find a simple algorithm realizing such transform in (Jiroušek, 2004). There is still another way to show that for a compositional model

$$\kappa_1(K_1) \triangleright \kappa_2(K_2) \triangleright \ldots \triangleright \kappa_n(K_n) \tag{6}$$

there exists an equivalent Bayesian network. The reader can show it using the old results of Andersson et al. (1997), and Studený (1997). Namely, considering Formula (6), one can construct an acyclic chain graph $\mathbf{G}_C$ with the set of nodes V. An undirected edge $\{u, v\}$ is in $\mathbf{G}_C$ if there is κ_i such that $\{u, v\} \subseteq K_i \setminus (K_1 \cup \ldots \cup K_{i-1})$ (it means that the components of the chain graph $\mathbf{G}_C$ are sets $K_i \setminus (K_1 \cup \ldots \cup K_{i-1})$, for all $i = 1, \ldots, n$). To specify directed edges of $\mathbf{G}_C$, one has to find for all nodes u the first distribution, in which the variable appears among the arguments. Let it be $\kappa_{[u]}$ (i.e., $[u] = \min\{i : u \in K_i\}$). Then the directed edge $(v \to u)$ is in $\mathbf{G}_C$ if $v \in K_{[u]} \cap (K_1 \cup \ldots \cup K_{[u]-1})$. Thus, $\mathbf{G}_C$ is obviously a chain graph (the components can be ordered in the way that for each directed edge $(v \to u)$, v belongs to the component that is before the component containing u). And this chain graph is an essential graph for any Bayesian network equivalent to the compositional model (6). This graphical representation of compositional models will be used in Example 3.

4. Anticipating operator of composition

As said in Introduction, the key idea of this paper can be more easily articulated using the technique of compositional models than using the terminology of Bayesian networks. Nevertheless, if the reader feels it more appropriate, they can translate it into the language of Bayesian networks. This is why we concluded the preceding Section by mentioning the equivalence of these two theoretical frameworks.

Recall that the studied process is inverse to the process of model reduction, which deletes at each step one variable. Thus, the latter process marginalizes at each step one variable out. This is why we believe that the next Section will be easier to understand, if we first describe the marginalization procedure for compositional models that, in a way, corresponds to the process of marginalization proposed for Bayesian networks by Ross Shachter (1986, 1988). Since we are interested in the elimination of one selected variable from the considered model, we introduce a special symbol: for a variable $u \in K$ and distribution $\kappa(K)$, its marginal $\kappa^{\downarrow K\setminus\{u\}}$ will also be denoted simply by κ^{-u}.

As already said above, the operator of composition is generally not associative, and it is neither commutative. To counterbalance this computational drawback, let us introduce its generalization (Jiroušek, 2011).

Definition 2 *Consider an arbitrary set of variables M and two distributions $\kappa(K)$, $\lambda(L)$. Their* anticipating composition *is given by the formula*

$$\kappa \circledcirc_M \lambda = (\lambda^{\downarrow(M\setminus K)\cap L} \cdot \kappa) \triangleright \lambda = (\lambda^{\downarrow(M\setminus K)\cap L} \triangleright \kappa) \triangleright \lambda. \tag{7}$$

The operator $\circledcirc_M$ is called an anticipating operator of composition.

Note that $\kappa \circledcirc_\emptyset \lambda = \kappa \triangleright \lambda$. Thus, it is clear that it may happen that the result of the composition remains undefined. However, it follows immediately from the respective definitions that if $\kappa \triangleright \lambda$ is defined then also $\kappa \circledcirc_M \lambda$ is defined. Both $\kappa \triangleright \lambda$ and $\kappa \circledcirc_M \lambda$ are distributions defined for the same set of variables.

In the following theorem we summarize the properties (proved in (Jiroušek, 2011)) of the operators of composition necessary in the following exposition.

Theorem 3 *Suppose $\kappa(K)$, $\lambda(L)$ and $\mu(M)$ are probability distributions. The following statements hold under the assumption that the respective compositions are defined:*

1. (Domain): *$\kappa \triangleright \lambda$ is a probability distribution for $K \cup L$.*
2. (Composition preserves first marginal): $(\kappa \triangleright \lambda)^{\downarrow K} = \kappa$.
3. (Reduction): *If $L \subseteq K$ then,* $\kappa \triangleright \lambda = \kappa$.
4. (Extension): *If $M \subseteq K$ then,* $\kappa^{\downarrow M} \triangleright \kappa = \kappa$.
5. (Commutativity under consistency): *κ and λ are consistent if and only if* $\kappa \triangleright \lambda = \lambda \triangleright \kappa$.
6. (Restricted associativity): *If $K \supseteq (L \cap M)$, or $L \supseteq (K \cap M)$ then,* $(\kappa \triangleright \lambda) \triangleright \mu = \kappa \triangleright (\lambda \triangleright \mu)$.
7. (Anticipating associativity): $(\mu \triangleright \kappa) \triangleright \lambda = \mu \triangleright (\kappa \circledcirc_M \lambda)$.
8. (Stepwise composition): *If $(K \cap L) \subseteq M \subseteq L$ then,* $(\kappa \triangleright \lambda^{\downarrow M}) \triangleright \lambda = \kappa \triangleright \lambda$.
9. (Exchangeability): *If $K \supseteq (L \cap M)$ then,* $(\kappa \triangleright \lambda) \triangleright \mu = (\kappa \triangleright \mu) \triangleright \lambda$.
10. (Simple marginalization): *If $(K \cap L) \subseteq M \subseteq K \cup L$ then,* $(\kappa \triangleright \lambda)^{\downarrow M} = \kappa^{\downarrow K\cap M} \triangleright \lambda^{\downarrow L\cap M}$.
11. (Factorization): *Let $M \supseteq K \cup L$. Then, $(K \setminus L) \perp\!\!\!\perp_\mu (L \setminus K) | (K \cap L)$ if and only if* $\mu^{\downarrow K\cup L} = \mu^{\downarrow K} \triangleright \mu^{\downarrow L}$.

The reader interested in other theoretical issues concerning the operator of composition is referred to (Jiroušek, 2011) and the papers cited there.

Consider a compositional model $\pi = \kappa_1(K_1) \triangleright \kappa_2(K_2) \triangleright \ldots \triangleright \kappa_n(K_n)$, and any variable $u \in K_1 \cup \ldots \cup K_n$. The marginalization over variable u means to find a new compositional model that corresponds to π^{-u}. It is shown in (Bína et al., 2020) that this model can always be obtained by the application of one or several from the following three rules, each of which somehow redefines the input model.

Marginalization rules

Variable deletion. If $u \in K_j$, and $u \notin K_i$ for all $i = 1, 2, \ldots, j-1, j+1, \ldots, n$, then marginalize variable u out of distribution κ_j, i.e., $\pi^{-u} = \kappa_1 \triangleright \ldots \triangleright \kappa_j^{-u} \triangleright \ldots \triangleright \kappa_n$.

Distribution deletion. If there exists index j such that $K_j \subseteq K_1 \cup \ldots \cup K_{j-1}$ then delete κ_j from the model, i.e., $\pi = \kappa_1 \triangleright \ldots \kappa_{j-1} \triangleright \kappa_{j+1} \triangleright \ldots \triangleright \kappa_n$.

Decrease of variable occurrences. For variable $u \in (K_1 \cup \ldots \cup K_n)$ find indices j and k ($j < k$) such that $u \in K_j \cap K_k$, and $u \notin K_i$ for all $i = 1, 2, \ldots, j-1, j+1, \ldots, k-1$, and set $M = (K_1 \cup \ldots \cup K_{k-1}) \setminus \{u\}$, then
$\pi = \kappa_1 \triangleright \ldots \triangleright \kappa_{j-1} \triangleright \kappa_j^{-u} \triangleright \kappa_{j+1} \triangleright \ldots \triangleright \kappa_{k-1} \triangleright (\kappa_j \circledcirc_M \kappa_k) \triangleright \kappa_{k+1} \triangleright \ldots \triangleright \kappa_n$.

For the theoretical support of Decrease-of-variable-occurrences rule see (Bína et al., 2020). Variable-deletion rule follows from Property 10 of Theorem 3, and Distribution-deletion rule is nothing else that the application of Property 3 of Theorem 3.

The reduction of a model (i.e., the subsequent computations of its marginal distribution deleting one variable at each step) can be performed by deleting variables in an arbitrary order. Each step of this reduction process, i.e. the computation of the respective marginal distribution, is realized by the application of the above-presented rules in a proper order. If the selected variable appears among the arguments of only one distribution κ_j, then it can be deleted by Variable-deletion rule. In opposite case, one has to first apply (possibly several times) Decrease-of-variable-occurrences rule before it can be deleted by Variable-deletion rule. The application of these two rules may induce the applicability of Distribution-deletion rule.

Example 1 *Consider model*

$$\pi(u_1, u_2, u_3, u_4) = \nu_1(u_1) \triangleright \nu_2(u_3) \triangleright \nu_3(u_1, u_3, u_4) \triangleright \nu_4(u_2, u_4), \tag{8}$$

and assume we want to reduce the model deleting respectively u_4, u_3, u_2, u_1. To marginalize over variable u_4, we have to first apply Decrease-of-variable-occurrences rule obtaining

$$\begin{aligned}
\pi(u_1, u_2, u_3, u_4) &= \nu_1(u_1) \triangleright \nu_2(u_3) \triangleright \nu_3^{-u_4}(u_1, u_3) \triangleright \big(\nu_3(u_1, u_3, u_4) \circledcirc_{\{u_1,u_3\}} \nu_4(u_2, u_4)\big) \\
&= \nu_1(u_1) \triangleright \nu_2(u_3) \triangleright \big(\nu_3(u_1, u_3, u_4) \circledcirc_{\{u_1,u_3\}} \nu_4(u_2, u_4)\big)\,, \\
&= \nu_1(u_1) \triangleright \nu_2(u_3) \triangleright (\nu_3(u_1, u_3, u_4) \triangleright \nu_4(u_2, u_4))\,,
\end{aligned}$$

where the second equation holds due to Property 3 of Theorem 3, and the last one holds due to Formula (7). Thus, applying Variable-deletion rule we get

$$\pi^{-u_4}(u_1, u_2, u_3) = \nu_1(u_1) \triangleright \nu_2(u_3) \triangleright \nu_5(u_1, u_2, u_3),$$

where

$$\nu_5(u_1,u_2,u_3)=\left(\nu_3(u_1,u_3,u_4)\triangleright\nu_4(u_2,u_4)\right)^{-u_4}.$$

To continue in the process of model reduction, we marginalize the model over variable u_3. To do it, we proceed in an analogous way:

$$\begin{aligned}\pi^{-u_4}(u_1,u_2,u_3) &= \nu_1(u_1)\triangleright\left(\nu_2(u_3)\circledcirc_{\{u_1\}}\nu_5(u_1,u_2,u_3)\right)\\ &= \nu_1(u_1)\triangleright\left((\nu_5^{\downarrow\{u_1\}}(u_1)\triangleright\nu_2(u_3))\triangleright\nu_5(u_1,u_2,u_3)\right).\end{aligned}\tag{9}$$

Thus, Variable deletion rule is applicable to Expression (9), and denoting

$$\nu_6(u_1,u_2)=\left((\nu_5^{\downarrow\{u_1\}}(u_1)\triangleright\nu_2(u_3))\triangleright\nu_5(u_1,u_2,u_3)\right)^{-u_3}$$

we get

$$\pi^{\downarrow\{u_1,u_2\}}(u_1,u_2)=\left(\pi^{-u_4}\right)^{-u_3}(u_1,u_2)=\nu_1(u_1)\triangleright\nu_6(u_1,u_2),$$

and the remaining reduction using just Variable-deletion rule is trivial.

5. Model learning process

Recall the underlying idea of this paper. Like the model reduction, the model learning procedure can process the variables in an arbitrary order, say $V=\{u_1,u_2,\ldots,u_n\}$. Denote by π the multidimensional distribution, for which the compositional model is looked for. The model is composed from low-dimensional marginals of π. The initialization step of the learning process is easy: it consists of defining a one-dimensional model $\kappa_1(u_1)=\pi(u_1)$. Then, at each step, the model extends by one variable. It means that after finishing $\ell-1$ steps, we have a compositional model $\pi(u_1,u_2,\ldots,u_{\ell-1})=\kappa_1(K_1)\triangleright\ldots\triangleright\kappa_m(K_m)$, and the goal of the next step is to redefine $\kappa_1(K_1)\triangleright\ldots\triangleright\kappa_m(K_m)$ so that it represents (ℓ)-dimensional distribution $\pi(u_1,u_2,\ldots,u_\ell)$. The realization of this (ℓ)-th step starts with considering the model

$$\pi(u_1,u_2,\ldots,u_{\ell-1},u_\ell)=\kappa_1(K_1)\triangleright\ldots\triangleright\kappa_m(K_m)\triangleright\pi(L),\tag{10}$$

where L is the smallest subset of $K_1\cup\ldots\cup K_m\cup\{u_\ell\}$, for which $u_\ell\in L$, and

$$u_\ell\perp\!\!\!\perp(K_1\cup\ldots\cup K_m)\setminus L|L\setminus\{u_\ell\}.$$

Then, within this step, the structure of this (ℓ)-dimensional model (10) must be improved. Based on the knowledge of conditional independence relations among variables L, we start exchanging the positions of distributions within the model given by Formula (10). The goal is to move the parts, from which $\pi(L)$ is composed, as much to left as possible. Before describing it in more detail, let us illustrate the idea of this process with a simple example.

Example 2 *Assume, the goal is to construct the model from Example 1:*

$$\nu_1(u_1)\triangleright\nu_2(u_3)\triangleright\nu_3(u_1,u_3,u_4)\triangleright\nu_4(u_2,u_4).$$

Therefore, the oracle, when being asked, produces the conditional independence relations in compliance with this model, i.e., the oracle knows that $u_1\perp\!\!\!\perp u_3$, $u_2\perp\!\!\!\perp\{u_1,u_3\}|u_4$. The variables are indexed corresponding to the order, in which the variables are to be added to the model. Thus, the index corresponds to the number of the step, in which the variable is added. At each step we denote the resulting model $\pi(u_1,u_2,\ldots,u_\ell)=\kappa_1(K_1)\triangleright\ldots\triangleright\kappa_m(K_m)$ (naturally with different m). It means that each step redefines both m and distributions $\kappa_1,\ldots,\kappa_m$.

Step 1. $\pi(u_1) = \kappa_1(u_1)$.

Step 2. *When adding variable u_2 one has to ask the oracle, whether u_1 and u_2 are mutually dependent, or not. Learning that the variables are dependent one gets the model $\kappa_1(u_1) \triangleright \pi(u_1, u_2)$. Therefore, we can consider only a simple model $\pi(u_1, u_2) = \kappa_1(u_1, u_2)$ (see Property 4 of Theorem 3).*

Step 3. *When adding variable u_3 one has to ask the oracle, what is the smallest subset $L \subset \{u_1, u_2, u_3\}$ such that $u_3 \in L$, $u_3 \perp\!\!\!\perp \{u_1, u_2\} \setminus L | L \setminus \{u_3\}$. The answer is $L = \{u_1, u_2, u_3\}$ because even though $u_3 \perp\!\!\!\perp u_1$, $u_3 \not\perp\!\!\!\perp u_1 | u_2$, $u_3 \not\perp\!\!\!\perp u_2 | u_1$, and $u_3 \not\perp\!\!\!\perp \{u_1, u_2\}$. Therefore, like in the previous step,*

$$\pi(u_1, u_2, u_3) = \kappa_1(u_1, u_2) \triangleright \pi(u_1, u_2, u_3) = \pi(u_1, u_2, u_3).$$

A subsequent question directed to the oracle aims at the detection whether $\pi(u_1, u_2, u_3)$ is or is not an (anticipating) composition of its marginals. Therefore we have to ask what is the independence structure of the set $L = \{u_1, u_2, u_3\}$. Learning from the oracle that $u_1 \perp\!\!\!\perp u_3$, one has to include this information into the form how $\pi(u_1, u_2, u_3)$ is expressed:

$$\pi(u_1, u_2, u_3) = \pi^{\downarrow\{u_3\}}(u_3) \circledcirc_{\{u_1\}} \pi(u_1, u_2, u_3) = \left(\pi^{\downarrow\{u_1\}}(u_1) \triangleright \pi^{\downarrow\{u_3\}}(u_3)\right) \triangleright \pi(u_1, u_2, u_3),$$

which is, for the purpose of the next step denoted as $\kappa_1(u_1) \triangleright \kappa_2(u_3) \triangleright \kappa_3(u_1, u_2, u_3)$.

Step 4. *When adding variable u_4 one has to ask the oracle, again, what is the smallest subset $L \subset \{u_1, u_2, u_3, u_4\}$ such that $u_4 \in L$, and $u_4 \perp\!\!\!\perp \{u_1, u_2, u_3\} \setminus L | L \setminus \{u_4\}$. This time, the answer is $L = \{u_1, u_2, u_3, u_4\}$. Therefore $\pi(u_1, u_2, u_3, u_4) = \kappa_1(u_1) \triangleright \kappa_2(u_3) \triangleright \kappa_3(u_1, u_2, u_3) \triangleright \pi(u_1, u_2, u_3, u_4)$. Again, we have to find out whether $\pi(u_1, u_2, u_3, u_4)$ might be an anticipating composition of its marginals, or, in other words, what are the independence relations holding for the set of variables $L = \{u_1, u_2, u_3, u_4\}$. Learning from the oracle that $u_2 \perp\!\!\!\perp \{u_1, u_3\} | u_4$, one has to employ this information:*

$$\pi(u_1, u_2, u_3, u_4) = \pi(u_1, u_3, u_4) \triangleright \pi(u_2, u_4) = \pi(u_1, u_3, u_4) \circledcirc_{\{u_1, u_3\}} \pi(u_2, u_4).$$

Thus,

$$\begin{aligned}
\pi(u_1, u_2, u_3, u_4) &= \kappa_1(u_1) \triangleright \kappa_2(u_3) \triangleright \kappa_3(u_1, u_2, u_3) \triangleright \pi(u_1, u_2, u_3, u_4) \\
&= \kappa_1(u_1) \triangleright \kappa_2(u_3) \triangleright \kappa_3(u_1, u_2, u_3) \triangleright \left(\pi(u_1, u_3, u_4) \circledcirc_{\{u_1, u_3\}} \pi(u_2, u_4)\right) \\
&= \kappa_1(u_1) \triangleright \kappa_2(u_3) \triangleright \left(\kappa_3(u_1, u_2, u_3) \triangleright \left(\pi(u_1, u_3, u_4) \circledcirc_{\{u_1, u_3\}} \pi(u_2, u_4)\right)\right) \\
&= \kappa_1(u_1) \triangleright \kappa_2(u_3) \triangleright \left(\left(\pi(u_1, u_3, u_4) \circledcirc_{\{u_1, u_3\}} \pi(u_2, u_4)\right) \triangleright \kappa_3(u_1, u_2, u_3)\right) \\
&= \kappa_1(u_1) \triangleright \kappa_2(u_3) \triangleright \left(\pi(u_1, u_3, u_4) \circledcirc_{\{u_1, u_3\}} \pi(u_2, u_4)\right) \\
&= \kappa_1(u_1) \triangleright \kappa_2(u_3) \triangleright \pi(u_1, u_3, u_4) \triangleright \pi(u_2, u_4).
\end{aligned}$$

Perhaps, the reader comprehended that the above-realized modifications are possible because of Properties 6, 5, and 7 of Theorem 3.

From the above-presented example, one can see that the gradual model-learning procedure is simple, if one knows how to modify the structure of the model described by Formula (10) at the ℓ-th

step. Nevertheless, neither this task is difficult when one realizes that it is nothing else than the multiple inverse application of Decrease-of-variable-occurrences rule. It means that the modification of a structure is possible if $\pi(L)$ is an anticipating composition of its marginals. First, however, it is advantageous to move the marginal $\pi(L)$ in the Formula (10) as much as possible to the left, i.e., to find the smallest k, for which

$$\begin{aligned}\pi(u_1,\ldots,u_\ell,u_{\ell+1}) &= \kappa_1(K_1)\triangleright\ldots\triangleright\kappa_m(K_m)\triangleright\pi(L)\\ &= \kappa_1(K_1)\triangleright\ldots\triangleright\kappa_k(K_k)\triangleright\pi(L)\triangleright\kappa_{k+1}(K_{k+1})\triangleright\ldots\triangleright\kappa_m(K_m).\end{aligned}\tag{11}$$

Realize that this $k\leq\tilde{k}=\min\left\{i\in\{1,\ldots,m\}:\forall j(i\leq j\leq m)(K_1\cup\ldots\cup K_j)\supseteq K_{j+1}\cap L\right\}$. Namely, for $\tilde{k}$ one can apply Property 9 of Theorem 3 $(m-\tilde{k})$-times obtaining

$$\begin{aligned}\pi(u_1,\ldots,u_\ell,u_{\ell+1}) &= \kappa_1(K_1)\triangleright\ldots\triangleright\kappa_m(K_m)\triangleright\pi(L)\\ &= \kappa_1(K_1)\triangleright\ldots\triangleright\kappa_{m-1}(K_{m-1})\triangleright\pi(L)\triangleright\kappa_m(K_m)=\ldots\\ &= \kappa_1(K_1)\triangleright\ldots\triangleright\kappa_{\tilde{k}}(K_{\tilde{k}})\triangleright\pi(L)\triangleright\kappa_{\tilde{k}+1}(K_{\tilde{k}+1})\triangleright\ldots\triangleright\kappa_m(K_m).\end{aligned}$$

Quite often, the required smallest $k<\tilde{k}$, and can be found by the subsequent application of other properties of Theorem 3. If $\pi(L)$ cannot be expressed as an anticipating composition of its marginals, then the result of the ℓ-th step, i.e., ℓ-dimensional model is expressed in the form of Formula (11).

In opposite case, further modifications of Formula (11) are possible. The only task requiring some programmer's wit[3] is to find subsets of variables N and K such that

$$\pi(L)=\pi(N)\circledcirc_M\pi(K)\tag{12}$$

for $M=K_1\cup\ldots\cup K_k$. Having such decomposition of $\pi(L)$ one can go on with the modifications of the following expression

$$\begin{aligned}\pi(u_1\ldots,u_\ell,u_{\ell+1}) &= \kappa_1(K_1)\triangleright\ldots\triangleright\kappa_k(K_k)\triangleright\big(\pi(N)\circledcirc_M\pi(K)\big)\triangleright\kappa_{k+1}(K_{k+1})\triangleright\ldots\triangleright\kappa_m(K_m)\\ &= \kappa_1(K_1)\triangleright\ldots\triangleright\kappa_k(K_k)\triangleright\pi(N)\triangleright\pi(K)\triangleright\kappa_{k+1}(K_{k+1})\triangleright\ldots\triangleright\kappa_m(K_m),\end{aligned}$$

and its submodel $\kappa_1(K_1)\triangleright\ldots\triangleright\kappa_k(K_k)\triangleright\pi(N)$ can further be modified exactly in the same way as described above for the model from Formula (10).

Let us illustrate this idea with a simple example.

Example 3 *Because of the lack of space, consider just the realization of the seventh step of the application of the gradual model construction. Consider a situation when the model*

$$\pi(u_1,\ldots,u_6)=\kappa_1(u_1,u_3)\triangleright\kappa_2(u_3,u_4)\triangleright\kappa_3(u_1,u_2,u_4)\triangleright\kappa_4(u_1,u_2,u_5)\triangleright\kappa_5(u_5,u_6)\tag{13}$$

is to be extended by variable u_7. Let this new variable be conditionally independent of $\{u_3,u_6\}$ given all the remaining variables: $u_7\perp\!\!\!\perp\{u_3,u_6\}|\{u_1,u_2,u_4,u_5\}$. Thus, the goal of this step is to

3. Assuming L is rather small, the task can usually be solved also by a brute force. If the task has more solutions, some heuristics should be used. One possibility is to prefer couples with the largest N because it rises chances that $\pi(N)$ can, again, be represented in the form of an anticipating composition.

modify the model

$$\begin{aligned}
&\pi(u_1,\ldots,u_7)\\
&\quad=\kappa_1(u_1,u_3)\triangleright\kappa_2(u_3,u_4)\triangleright\kappa_3(u_1,u_2,u_4)\triangleright\kappa_4(u_1,u_2,u_5)\triangleright\kappa_5(u_5,u_6)\triangleright\pi(u_1,u_2,u_4,u_5,u_7)\\
&\quad=\kappa_1(u_1,u_3)\triangleright\kappa_2(u_3,u_4)\triangleright\kappa_3(u_1,u_2,u_4)\triangleright\kappa_4(u_1,u_2,u_5)\triangleright\pi(u_1,u_2,u_4,u_5,u_7)\triangleright\kappa_5(u_5,u_6)\\
&\quad=\kappa_1(u_1,u_3)\triangleright\kappa_2(u_3,u_4)\triangleright\kappa_3(u_1,u_2,u_4)\triangleright\pi(u_1,u_2,u_4,u_5,u_7)\triangleright\kappa_5(u_5,u_6)\\
&\quad=\kappa_1(u_1,u_3)\triangleright\kappa_2(u_3,u_4)\triangleright\pi(u_1,u_2,u_4,u_5,u_7)\triangleright\kappa_5(u_5,u_6),
\end{aligned}\tag{14}$$

where the last two modifications are the applications of Property 8 of Theorem 3 (κ_3 and κ_4 are marginals of π). At this stage of the model-learning process, we have to ask what is the independence structure of $\pi(u_1,u_2,u_4,u_5,u_7)$. Learning that $u_5\perp\!\!\!\perp\{u_2,u_4\}|\{u_1,u_7\}$, due to Property 11 of Theorem 3, we know that

$$\begin{aligned}
\pi(u_1,u_2,u_4,u_5,u_7) &= \pi(u_1,u_2,u_4,u_7)\triangleright\pi(u_1,u_5,u_7)\\
&= \pi(u_1,u_2,u_4,u_7)\circledcirc_{\{u_1,u_3,u_4\}}\pi(u_1,u_5,u_7).
\end{aligned}$$

Thus,

$$\begin{aligned}
\pi(u_1,\ldots,u_7) &= \kappa_1(u_1,u_3)\triangleright\kappa_2(u_3,u_4)\\
&\qquad\triangleright\Big(\pi(u_1,u_2,u_4,u_7)\circledcirc_{\{u_1,u_3,u_4\}}\pi(u_1,u_5,u_7)\Big)\triangleright\kappa_5(u_5,u_6)\\
&= \kappa_1(u_1,u_3)\triangleright\kappa_2(u_3,u_4)\triangleright\pi(u_1,u_2,u_4,u_7)\triangleright\pi(u_1,u_5,u_7)\triangleright\kappa_5(u_5,u_6).
\end{aligned}\tag{15}$$

Again, we are at the stage of the model-learning process when we have to ask what is the independence structure of a newly introduced distribution, this time it is $\pi(u_1,u_2,u_4,u_7)$. Learning that $u_4\perp\!\!\!\perp u_7|\{u_1,u_2\}$, due to Property 11 of Theorem 3, we consider

$$\pi(u_1,u_2,u_4,u_7)=\pi(u_1,u_2,u_4)\triangleright\pi(u_1,u_2,u_7)=\pi(u_1,u_2,u_4)\circledcirc_{\{u_1,u_2,u_3\}}\pi(u_1,u_2,u_7).$$

Using this, one obtains the result of the described 7th step

$$\begin{aligned}
\pi(u_1,\ldots,u_7) &= \kappa_1(u_1,u_3)\triangleright\kappa_2(u_3,u_4)\triangleright\Big(\pi(u_1,u_2,u_4)\circledcirc_{\{u_1,u_2,u_3\}}\pi(u_1,u_2,u_7)\Big)\\
&\qquad\triangleright\pi(u_1,u_5,u_7)\triangleright\kappa_5(u_5,u_6)\\
&= \kappa_1(u_1,u_3)\triangleright\kappa_2(u_3,u_4)\triangleright\kappa_3(u_1,u_2,u_4)\triangleright\pi(u_1,u_2,u_7)\triangleright\pi(u_1,u_5,u_7)\triangleright\kappa_5(u_5,u_6).
\end{aligned}\tag{16}$$

The reader preferring a graphical representation of model structures can find the essential graphs corresponding to the models from Expressions (13)–(16) in Figure 2.

Let us conclude the example by repeating that the transition from the model described by Formula (14) to model described by Formula (15) was possible because $u_5\perp\!\!\!\perp\{u_2,u_4\}|\{u_1,u_7\}$, and the transition from the model from Formula (15) to model described by Formula (16) was possible due to $u_4\perp\!\!\!\perp u_7|\{u_1,u_2\}$.

Naturally, in the above-presented simple example we could consider only a small number of variables. So it happened that the both independence relations making the modifications possible (and highlighted at the end of the example), contained only the variables from the respective marginal to be decomposed. Generally, when modifying a model

$$\kappa_1(K_1)\triangleright\ldots\triangleright\kappa_k(K_k)\triangleright\pi(L)\triangleright\ldots$$

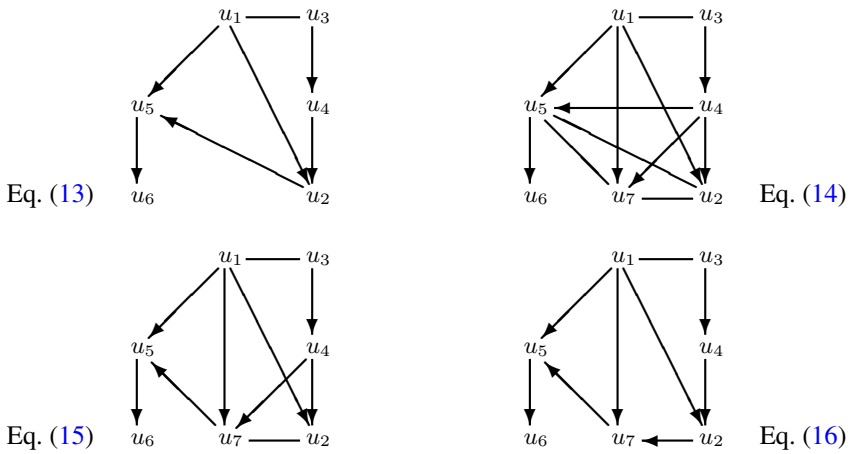

Figure 2: Essential Graphs of Bayesian networks equivalent to compositional models from Example 3.

we have to consider independence relations guaranteeing the validity of Formula (12), i.e., relations $(K \setminus \tilde{M}) \perp\!\!\!\perp (N \setminus \tilde{M}) | \tilde{M}$, where $\tilde{M} \subseteq (L \cup K_1 \cup \ldots \cup K_k)$ (in the example $\tilde{M}$ was a subset of L, only), and

$$\pi(N) \circledcirc_{\tilde{M} \setminus L} \pi(K) = \pi(N) \circledcirc_M \pi(K),$$

for $M = K_1 \cup \ldots \cup K_k$. Therefore, the efficient algorithmization of this step is not a trivial task.

6. Conclusions

The paper shows that there is a possibility to increase the dimension of a multidimensional model by one without throwing away the information included in the smaller model. For this, we suggest a procedure, which is inverse to the process of model reduction based on the Shachter's node deletion and edge reversal rules. We describe the procedure using the apparatus of compositional models, proving thus that this technique can sometimes easily explain what would be hard to express in the language of graphs.

The paper does not describe a machine-learning algorithm. It only explains the idea of the modification process that forms the core of the described approach. Nevertheless, we believe that using the apparatus developed by Kratochvíl (2013), the algorithmization of the procedure should not be a great problem.

Though we use the terminology of compositional models, the results apply to Bayesian networks, too. This is illustrated at the end of Example 3 (Figure 2), and the description of the corresponding process using the terminology of graphical models may be a challenge for graphical experts.

Acknowledgements

The research was financially supported by the Czech National Science Foundation under grant no. 19-06569S.

References

Steen A Andersson, David Madigan, and Michael D Perlman. On the Markov equivalence of chain graphs, undirected graphs, and acyclic digraphs. *Scandinavian Journal of Statistics*, 24(1):81–102, 1997.

Vladislav Bína, Radim Jiroušek, and Václav Kratochvíl. Foundations of compositional models: Inference. *International Journal of General Systems* (submited), 2020.

David Maxwell Chickering. Learning Bayesian networks is NP-complete. In *Learning from data*, pages 121–130. Springer, 1996.

David Edwards and Tomáš Havránek. A fast model selection procedure for large families of models. *Journal of the American Statistical Association*, 82(397):205–213, 1987.

David Heckerman, Dan Geiger, and David M Chickering. Learning Bayesian networks: The combination of knowledge and statistical data. *Machine learning*, 20(3):197–243, 1995.

R Jiroušek. What is the difference between Bayesian networks and compositional models. In *Proc. 7th Czech-Japan Seminar on Data Analysis and Decision Making under Uncertainty (H. Noguchi, H. Ishii, M. Inuiguchi, eds.), Awaji Yumebutai ICC*, pages 191–196, 2004.

Radim Jiroušek. Foundations of compositional model theory. *International Journal of General Systems*, 40(6):623–678, 2011.

Václav Kratochvíl. Probabilistic compositional models: Solution of an equivalence problem. *Int. J. Approx. Reasoning*, 54(5):590–601, 2013.

Wai Lam and Fahiem Bacchus. Learning Bayesian belief networks: An approach based on the MDL principle. *Computational intelligence*, 10(3):269–293, 1994.

Janny Leung and Jon Lee. More facets from fences for linear ordering and acyclic subgraph polytopes. *Discrete Applied Mathematics*, 50(2):185–200, 1994.

Young Woong Park and Diego Klabjan. Bayesian network learning via topological order. *The Journal of Machine Learning Research*, 18(1):3451–3482, 2017.

Ross D Shachter. Evaluating influence diagrams. *Operations research*, 34(6):871–882, 1986.

Ross D Shachter. Probabilistic inference and influence diagrams. *Operations research*, 36(4):589–604, 1988.

Ajit P Singh and Andrew W Moore. *Finding optimal Bayesian networks by dynamic programming*. Citeseer, 2005.

Milan Studený. A recovery algorithm for chain graphs. *International Journal of Approximate Reasoning*, 17(2-3):265–293, 1997.

Paul E Utgoff. Incremental induction of decision trees. *Machine learning*, 4(2):161–186, 1989.

Jan Švorc and Jiří Vomlel. Bayesian networks for the analysis of subjective well-being. In *22nd Czech-Japan Seminar on Data Anal. and Decision Making*, pages 175–188. MatfyzPress, 2019.