Knowledge Engineering for Bayesian Networks: How Common Are Noisy-MAX Distributions in Practice?

Adam Zagorecki and Marek J. Druzdzel

Abstract—**One problem faced in knowledge engineering for Bayesian networks is the exponential growth of the number of parameters in their conditional probability tables (CPTs). The most common practical solution is the application of so-called canonical gates and, among them, the noisy-OR (or their generalization, the noisy-MAX) gates, which take advantage of independence of causal interactions and provide a logarithmic reduction of the number of parameters required to specify a CPT. In this paper, we propose an algorithm that fits a noisy-MAX distribution to an existing CPT and we apply this algorithm to search for noisy-MAX gates in three existing practical Bayesian network models: ALARM, HAILFINDER and HEPAR II. We show that the noisy-MAX gate provides a surprisingly good fit for as many as 50% of CPTs in two of these networks. We observed this in both distributions elicited from experts and those learned from data. The importance of this finding is that it provides an empirical justification for the use of the noisy-MAX gate as a powerful knowledge engineering tool.**

I. INTRODUCTION

BAYESIAN networks (BNs) [1] provide a convenient and sound framework for encoding uncertain knowledge and for reasoning under uncertainty. A BN consists of an acyclic directed graph encoding a factorization of the joint probability distribution over a set of random variables and a set of conditional probability distributions. The structure of the graph represents the variables and independencies among them, while the probability distributions over the individual variables conditioned on their direct predecessors (parents) represent individual components of the factorization.

When a node of a BN and all its parents are discrete (this is most common in practice due to the existence of general purpose efficient algorithms for reasoning with discrete variables), the conditional probability distributions are stored in *conditional probability tables* (CPTs) indexed by all possible combinations of states of the parents. The CPT of a node with ten binary parents will be indexed by $2^{10} = 1,024$ combinations of parents' states and will contain as many conditional probability distributions. This poses considerable difficulties for knowledge engineering, for learning BNs from data, and for BN inference algorithms. An ingenious practical solution to this problem has been the application of parametric conditional distributions, such as the noisy-OR gates or their multi-valued generalization, the noisy-MAX gates. By taking advantage of *independence of causal influences* (ICI), these gates offer a reduction in the number of parameters required to specify a conditional probability distribution from exponential to linear in the number of parents.

The two most widely applied ICI distributions are the binary noisy-OR model [2], [3] and its extension to multi-valued variables, the noisy-MAX model [4], [5]. Noisy-OR and noisy-MAX gates have proven their worth in many real-life applications (e.g., [6–8]). Their foremost advantage is a small number of parameters that are sufficient to specify the entire CPT. This leads to a significant reduction of effort in knowledge elicitation from experts [4], [6], improves the quality of distributions learned from data [9], and reduces the spatial and temporal complexity of algorithms for Bayesian networks [10–12].

Our research aims at better understanding the applicability of noisy-MAX gates in practical BN models. We achieve this by developing a technique for fitting noisy-MAX relationships to existing, fully specified CPTs. Having this technique, we can examine CPTs in existing practical BN models for whether they can be suitably approximated by the noisy-MAX model. Models that we selected for our study first of all contain many nodes with multiple parents and with CPTs obtained by full specification without apparent systematic patterns. Obviously, models that are built on noisy-OR gates are not of interest here. It turns out that finding models which fulfill the above criteria is relatively difficult and we were able to find only three: ALARM [13], HAILFINDER [14] and HEPAR II [9]. We apply two measures of distance between two CPTs — one based on Euclidean distance and one based on Kullback-Leibler divergence. We prove that Euclidean distance between any CPT and a CPT that is generated from a set of noisy-MAX parameters has exactly one minimum. We apply this result to an algorithm that, given a CPT, finds a noisy-MAX distribution that provides the best fit to it. We show that the noisy-MAX gate provides a surprisingly good fit for a significant percentage of distributions in these networks. We observed this both in distributions elicited from experts and in those learned from data, and for two measures of distance between distributions. We tested the robustness of this result by fitting the noisy-MAX distribution to randomly generated CPTs and

Adam Zagorecki is with the Operational and Decision Analysis Group, Department of Informatics and Systems Engineering, Cranfield University, Defence Academy of the United Kingdom, Shrivenham, SN6 8LA, United Kingdom,(phone: +44 (0) 1793-785-293, e-mail: a.zagorecki@cranfield.ac.uk)

Marek Druzdzel is with the Decision Systems Laboratory, School of Information Sciences and Intelligent Systems Program, University of Pittsburgh, Pittsburgh, PA 15260, USA and Faculty of Computer Science, Białystok University of Technology, Wiejska 45A, 15-351 Białystok, Poland. (e-mail: marek@sis.pitt.edu)

Marek Druzdzel was supported in part by the National Institute of Health under grant number U01HL101066-01 and by Intel Research.

observed that the fit in this case is poor. Obtaining a randomly generated CPT that can be reasonably approximated by a noisy-MAX gate is extremely unlikely, which leads us to the conclusion that our results are not a coincidence. Finally, we investigate the effect of approximating CPTs in BNs by means of noisy-MAX gates on the accuracy of posterior probability distributions obtained by means of these BNs.

We envisage two applications of our results. The first is providing a justification for refocusing knowledge engineering effort from obtaining an exponentially growing number of numerical probabilities to a much smaller number of noisy-MAX parameters. While a parametric distribution may be only an approximation to a set of general conditional probability distributions, the precision that goes with the latter is often only theoretical. In practice, obtaining large numbers of numerical probabilities is likely to lead to expert exhaustion and result in poor quality estimates. Focusing the expert's effort on a small number of parameters of a corresponding parametric distribution should lead to a better quality model. The second application of our results is in approximate algorithms for Bayesian networks. Whenever the fit is good, a CPT can be replaced by an ICI gate, leading to potentially significant savings in computation [10], [11].

The remainder of this paper is organized as follows. Section II reviews those properties of BNs that are relevant to our work. Section III introduces the noisy-OR and the noisy-MAX gates. Section IV proposes our algorithm for fitting noisy-MAX parameters to an arbitrary CPT. Section V presents the results of our experiments that test goodness of fit of noisy-MAX gates to CPTs in several existing practical networks. Section VI proposes an explanation of the observed results.

II. Bayesian Networks

A Bayesian network is a compact representation of a joint probability distribution over a finite set of random variables. Its qualitative part is an acyclic directed graph, in which vertices represent random variables and edges indicate direct statistical relationships among these variables. Its quantitative part consists of probability distributions associated with variables (vertices in the graph).

A BN captures a joint probability distribution over the variables that it models by means of factorization, using the chain rule of probability, and taking advantage of statistical independencies between variables in a domain. However, explicit representation of independencies alone is insufficient for creating complex BN models in practice. The main reason for this is the number of distributions in conditional probability tables that grow exponentially with the number of parents of a node in a graph. We will introduce BNs and explain the problem with large CPTs informally by means of a simple example, a BN modeling several possible causes of problems with starting a car engine (Fig. 1).

This simple network models three causes that can prevent an engine from starting: (1) dead battery, (2) dirty connectors between the battery and the rest of the electrical system, which prevent current from flowing, and (3) a short in the wiring caused by water (e.g., after a rainy day and driving through pools of water). It is relatively easy to obtain prior probability distributions for nodes *Dead Battery*, *Dirty Connectors*, and *Shortcut Caused by Water* from an expert. More problematic is obtaining a CPT for the variable *Engine does not Start*. This requires an explicit specification of eight conditional probability distributions — one for each combination of states of the parent nodes. The exponential growth of CPTs in the number of parent nodes is a major problem with knowledge engineering for BNs. The resulting large number of probabilities makes the elicitation process time-consuming, as it increases the costs (expert's time easily translates into money) and decreases the quality of estimates. In addition, in our example, the expert may have difficulties with estimating the probability that the engine does not start, given that the battery is charged, but connectors are dirty and there is water in electrical system. This is because some combinations of parent states may be very unlikely and typically the expert may have no experience with them.

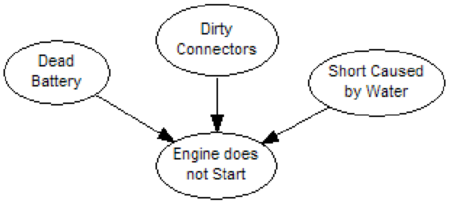

Fig. 1. An example BN modeling causes of problems with starting a car engine.

The knowledge engineer should notice in this situation, that it is not necessary to specify the entire CPT for the variable *Engine does not Start*. He can take advantage of the observation that the three causes operate independently in preventing the engine from starting. This type of independence is typically referred to as ICI. In this case, he can apply the noisy-MAX model because the three modeled causes act independently on the ability of the engine to start (none of them interferes with other causes in producing the effect) and presence of one cause is sufficient to prevent the engine from starting. The ICI assumption allows us to reduce the number of probabilities required to specify the CPT from exponential to linear in the number of parents. In the literature, we can find different approaches to the problem of simplifying specification of CPTs. Examples based on the ICI assumption include: temporal and atemporal representations [12], and independence of causal inputs [15]. Examples of non-ICI approaches to the problem include linear potential functions [16] or context-specific independence [17]. An explicit representation of ICI can lead to considerable savings in belief updating algorithms for BNs [11], [12]. In this paper, we concentrate on the most commonly used model for ICI — the noisy-OR model and its generalization, the noisy-MAX model.

III. Noisy-OR and Noisy-MAX

In what follows, we will represent random variables by upper-case letters (e.g., X) and their values by indexed lower-case letters, (e.g., x). We use $Rng(X)$ to denote the range

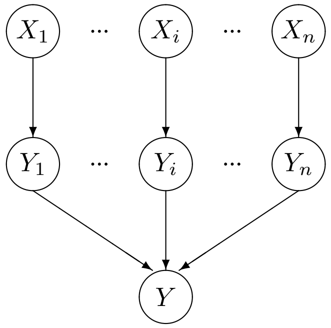

Fig. 2. Direct modeling of noisy-OR

(the set of possible values) of a variable X. We assume that all variables are discrete.

An interaction can be approximated by a noisy-MAX gate if it meets the following three assumptions [6]: (1) the child node and all its parents must be variables indicating the degree of presence of an anomaly, (2) each of the parent nodes must represent a cause that can produce the effect (the child node) in the absence of the other causes, and (3) there may be no significant synergies among the causes.

Let there be n binary nodes $X_1, \ldots, X_n$, each with values from $Rng(X_i) = \{x_i, \overline{x}_i\}$. Let the variables X_i be the parents of an effect variable Y that assumes values y and $\overline{y}$. The noisy-OR model can be thought of as a probabilistic extension of the deterministic OR model of interaction of inputs in producing the output. The noisy-OR model assumes that the presence of each single cause X_i is able to produce the presence of the effect Y and that its ability to produce that effect is independent of the presence of other causes. However, the presence of a cause X_i does not guarantee that the effect Y will occur.

A useful concept for modeling the noisy-OR by means of the deterministic OR, is that of the *inhibitor* nodes [1], [12], which model the probabilistic relation between each cause and the effect individually. The *inhibitor* nodes Y_i introduce *noise* — the probabilistic effect of X_i on Y. Each inhibitor node Y_i takes the same values as Y. Fig. 2 presents an explicit graphical representation of the example network of Fig. 1 that includes inhibitor nodes. The CPT of Y defines how those individual effects Y_i combine to produce Y. For the noisy-OR model, the CPT of node Y is equivalent to a deterministic OR. The CPT of each Y_i is of the form: $\Pr(y_i|x_i) = p_i$ where $p_i \in [0, 1]$, and $\Pr(y_i|\overline{x}_i) = 0$. The noisy-MAX model [4], [5] is basically an extension of the noisy-OR model to multi-valued variables. The noisy-MAX assumes that the variable Y has n_y states and that these states are ordered. In practice, the states are ordered according to increasing (or decreasing) severity, for example: *none*, *low*, *medium*, *high*. The inhibitor nodes Y_i take values from the same domain as Y and their states follow the same ordering. Every parent variable X_i has n_i values.

For the convenience of further discussion, we introduce a notation that is similar to the (ijk) coordinates notation commonly used in the Bayesian network literature. We use q_{ijk} to denote the element of CPT of inhibitor node Y_i that corresponds to the j-th value of parent node X_i and k-th value of Y_i: $\forall_{ijk}\ q_{ijk} = \Pr(y_i^k|x_i^j)$. Probabilities q_{ijk} are noisy-MAX parameters. The inhibitor variables Y_i have the same range as Y and their CPTs are constrained in the following way:

$$q_{ijk} = \begin{cases} 1 & \text{for } j = 1, k = 1 \\ 0 & \text{for } j = 1, k \neq 1 \\ p \in [0 \ldots 1] & \text{for } j \neq 1\ . \end{cases}$$

The CPT of Y is a deterministic MAX defined by the ordering relation over states of Y. It is common practice to add a *leak* term to the noisy-OR and noisy-MAX models. The leak is an auxiliary cause that serves the purpose of modeling the influence of causes that are not explicitly included in the model. However the leak can be mathematically modeled as an additional cause X_{leak} and the corresponding inhibitor node Y_{leak}, making all consequent discussion applicable to the case with the leak cause present. In our experiments, we always assume that the noisy-MAX model includes the leak probability.

IV. CONVERTING A CPT INTO A NOISY-MAX GATE

In this section, we propose an algorithm that fits a noisy-MAX distribution to an arbitrary CPT. In other words, it identifies the set of noisy-MAX parameters that produces a CPT that is the *closest* to the original CPT. Let C_Y be the CPT of a node Y that has n parent variables $X_1, \ldots, X_n$. Let y^j be the j-th value of Y and also of Y_i, $i = 1, \ldots, n$. Each variable X_i can take one of n_{X_i} possible values.

Let $\mathbf{X}$ be a n-dimensional variable composed of a vector of variables $X_1, \ldots, X_n$. Let $\mathbf{x}^i = (x_1^i, \ldots, x_n^i)$ denote the i-th value of n-dimensional variable $\mathbf{X}$ (an instantiation of parent variables). The number of possible values of $\mathbf{X}$, denoted by m is the product of the numbers of possible values of the X_is, i.e., $m = \prod_{i=1}^{n} n_{X_i}$. There exist several measures of similarity of two probability distributions (for a good overview we refer the reader to [18]), of which two are commonly used: Euclidean distance and Kullback-Leibler (KL) divergence. The main difference between the two is that the Euclidean distance is based on absolute differences and, hence, is insensitive to large relative differences in very small probabilities, which can be a major drawback in some contexts. The KL divergence addresses this problem but, on the other hand, the problem with KL divergence is that it is undefined for cases when the estimated probability is zero and the goal probability is non-zero. In our experiments, we used a common approach that amounts to replacing zeros with a constant close to zero.

The problem of defining a distance between two CPTs is somewhat more complicated, because a CPT is a set of probability distributions. The easiest approach is to define distance between two CPTs as a sum of distances of corresponding probability distributions in both CPTs. However, in practice not all distributions in the CPT are equally important. This is because typically some of the configurations of parents' states are far more likely than the others. In our example, the probability of the instantiation of parents: *dead battery*, *dirty connectors*, and *short caused by water* present is far less likely than all three parent variables in states *ok*. Thus, in some situations, we may want to use a measure of distance between two CPTs that takes this into account.

Definition 1 (Euclidean distance between two CPTs): The distance D_E between two CPTs, $\Pr_A(Y|\mathbf{X})$ and $\Pr_B(Y|\mathbf{X})$ is a weighted sum of Euclidean distances between their corresponding probability distributions:

$$\begin{aligned} &D_E(\Pr_A(Y|\mathbf{X}), \Pr_B(Y|\mathbf{X})) = \\ &= \sum_{i=1}^{m} w_{\mathbf{x}^i} \sum_{j=1}^{n_Y} \left(\Pr_A(y^j|\mathbf{x}^i) - \Pr_B(y^j|\mathbf{x}^i) \right)^2 , \end{aligned} \tag{1}$$

where $w_{\mathbf{x}^i} = P(\mathbf{x}^i)$ is a weighting constant for each distribution in the CPT.

Analogously we define distance between two CPTs based on Kullback-Leibler divergence as follows:

Definition 2 (KL divergence between CPTs): The divergence D_{KL} between the goal CPT $\Pr_A(Y|\mathbf{X})$ and its approximation $\Pr_B(Y|\mathbf{X})$ is a weighted sum of KL divergences between their corresponding probability distributions:

$$\begin{aligned} &D_{KL}(\Pr_A(Y|\mathbf{X}), \Pr_B(Y|\mathbf{X})) = \\ &\sum_{i=1}^{m} w_{\mathbf{x}^i} \sum_{j=1}^{n_Y} \Pr_A(y^j|\mathbf{x}^i) \log \frac{\Pr_A(y^j|\mathbf{x}^i)}{\Pr_B(y^j|\mathbf{x}^i)} , \end{aligned} \tag{2}$$

where $w_{\mathbf{x}^i} = P(\mathbf{x}^i)$ is a weighting constant for each distribution in the CPT.

In case of the KL divergence, it is possible to show that under the assumption that the weights $w_{\mathbf{x}^i} = P(\mathbf{x}^i)$ the defined measure of divergence between two CPTs is equivalent to the KL divergence defined for the joint probability distribution over $Y \cup \mathbf{X}$.

Definition 3 (MAX-based CPT): A MAX-based CPT $\Pr_q(Y|\mathbf{X})$ is a CPT constructed from a set of noisy-MAX parameters $\mathbf{q}$ using the noisy-MAX equations.

Let q_{ijk} be defined as follows:

$$q_{ijk} = \Pr(Y_k = y^j | X_k = x_k^i) .$$

Our goal is to find for a given CPT $\Pr_{cpt}(Y|\mathbf{X})$, a set of noisy-MAX parameters $\mathbf{q}$ that minimizes the Euclidean distance between the original CPT and the MAX-based CPT $\Pr_q(Y|\mathbf{X})$. For simplicity, we will use θ_{ij} to denote the element of the CPT that corresponds to the i-th element of $\mathbf{X}$ and j-th state of Y:

$$\theta_{ij} = \Pr(Y = y^j | \mathbf{X} = \mathbf{x}^i) .$$

When this parameter is given in a CPT, we use upper index *cpt* (e.g., θ_{ij}^{cpt}), and when the parameter was obtained from the MAX-based CPT, we use upper index *max* (e.g., θ_{ij}^{max}). We can now rewrite Eq. 1 as:

$$\sum_{i=1}^{m} w_{\mathbf{x}^i} \sum_{j=1}^{n_Y} \left(\theta_{ij}^{cpt} - \theta_{ij}^{max} \right)^2 .$$

Because subsequent discussion relies heavily on cumulative probabilities, we introduce cumulative probability distributions based on the parameters θ_{ij} and q_{ijk}. We define Θ_{ij} as:

$$\Theta_{ij} = \begin{cases} \sum_{l=1}^{j} \theta_{il} & \text{for } j \neq 0 \\ 0 & \text{for } j = 0 \end{cases} ,$$

which constructs a cumulative probability distribution function for $\Pr(Y|\mathbf{x}^i)$. It is easy to notice, that $\theta_{ij} = \Theta_{ij} - \Theta_{i(j-1)}$. The next step is to express the MAX-based CPT parameters θ_{ij}^{max} in terms of the noisy-MAX parameters. In similar manner, we define the cumulative probability distribution of the noisy-MAX parameters Q_{ij} as:

$$\begin{aligned} Q_{ijk} &= \Pr(Y_k \leq y^j | X_k = x_k^i) \\ &= \sum_{l=1}^{j} \Pr(Y_k = y^l | X_k = x_k^i) \\ &= \sum_{l=1}^{j} q_{ilk} , \end{aligned}$$

and for j=0 define $Q_{ijk} = 0$.

Pradhan et al. [7] proposed an algorithm exploiting cumulative probability distributions for efficient calculation of the MAX-based CPT that computes parameters of the MAX-based CPT as follows:

$$\begin{aligned} \Theta_{ij}^{max} &= \Pr(Y \leq y^j | \mathbf{X} = \mathbf{x}^i) \\ &= \prod_{k=1}^{n} \Pr(Y_k \leq y^j | X_k = x_k^i) \\ &= \prod_{k=1}^{n} Q_{ijk} , \end{aligned} \tag{3}$$

In Eq. 3 the values of parent node X_k are components of vector $\mathbf{x}^i$. Eq. 4 shows how to compute the element θ_{ij}^{max} from the noisy-MAX parameters:

$$\begin{aligned} \theta_{ij}^{max} &= \Theta_{ij}^{max} - \Theta_{i(j-1)}^{max} \\ &= \prod_{k=1}^{n} Q_{ijk} - \prod_{k=1}^{n} Q_{i(j-1)k} \\ &= \prod_{k=1}^{n} \sum_{l=1}^{j} q_{ilk} - \prod_{k=1}^{n} \sum_{l=1}^{j-1} q_{ilk} . \end{aligned} \tag{4}$$

However, parameters θ_{ij}^{max} have to obey the axioms of probability, which means that we have only $n_Y - 1$ independent terms and not n_Y, as the notation suggests. Hence, we can express θ_{ij}^{max} in the following way:

$$\theta_{ij}^{max} = \begin{cases} \prod_{k=1}^{n} \sum_{l=1}^{j} q_{ilk} - \prod_{k=1}^{n} \sum_{l=1}^{j-1} q_{ilk} & \text{for } j \neq n_Y \\ 1 - \prod_{k=1}^{n} \sum_{l=1}^{n_Y - 1} q_{ilk} & \text{for } j = n_Y . \end{cases}$$

Theorem 1: The distance D_E between an arbitrary CPT $\Pr_{cpt}(Y|\mathbf{X})$ and a MAX-based CPT $\Pr_q(Y|\mathbf{X})$ of noisy-MAX parameters $\mathbf{q}$ as a function $\mathbf{q}$ has exactly one minimum.

Proof: We prove that for each noisy-MAX parameter $q \in \mathbf{q}$, the first derivative of D_E has exactly one zero point. We will subsequently show that the second derivative is always positive, which indicates that D_E has exactly one minimum. The first derivative of D_E over q is

$$\frac{\partial}{\partial q}\sum_{i=1}^{m} w_{\mathbf{x}^i}\sum_{j=1}^{n_Y-1}\left(\theta_{ij}^{cpt}-\prod_{k=1}^{n}\sum_{l=1}^{j}q_{ilk}+\prod_{k=1}^{n}\sum_{l=1}^{j-1}q_{ilk}\right)^2$$

$$+\frac{\partial}{\partial q}\sum_{i=1}^{m} w_{\mathbf{x}^i}\left(-\sum_{j=1}^{n_Y-1}\theta_{ij}^{cpt}+\prod_{k=1}^{n}\sum_{l=1}^{n_Y-1}q_{ilk}\right)^2 .$$

Each of the three products contains at most one term q and, hence, the expression takes the following form:

$$\frac{\partial}{\partial q}\sum_{i,j}\left(A_{ij}+B_{ij}q\right)^2 , \tag{5}$$

where A_{ij} and B_{ij} are constants. At least some of the terms B_{ij} have to be non-zero (because external sum in Eq. 5 runs over all elements of the CPT). The derivative

$$\frac{\partial}{\partial q}\sum_{i,j}\left(A_{ij}+B_{ij}q\right)^2 = 2\sum_{i,j}\left(A_{ij}B_{ij}\right)+2q\sum_{i,j}B_{ij}^2$$

is a non-trivial linear function of q. The second order derivative is equal to $2\sum_{i,j}B_{ij}^2$ and always takes positive values. Therefore, there exists exactly one local minimum of the original function. ■

In our approach, we try to identify the set of noisy-MAX parameters that minimizes the distances D_E or D_{KL} for a given CPT. The problem amounts to finding the minimum of the distance as a multidimensional function of the noisy-MAX parameters. Theorem 1 states that for the Euclidean distance, there exists exactly one local minimum. Therefore, we can use any mathematical optimization method ensuring convergence to a single minimum. In case of KL divergence, we have no guarantee that there exists exactly one local minimum. However, the assumption that there exist only one local minimum is conservative, because if there exist multiple local minima, the algorithm may find a sub-optimal solution. Finding the global minimum in this case would make our results stronger.

We implemented a simple hill-climbing algorithm outlined in Fig. 3 that takes a CPT as its input and produces noisy-MAX parameters and a measure of fit as its output. In every step of the inner loop (3b), we introduce a change in the noisy-MAX parameters by adding/subtracting a small value of *step* from a single noisy-MAX parameter (procedure *ChangeMAX*). When, one parameter is changed, the other parameters have to be changed as well in order to obey constraints imposed by the probability axioms. In our algorithm, we distribute the change proportionally to the value of each parameter. The procedure *CalculateDistance* returns a measure of distance between two CPTs.

```
Procedure NoisyMaxParametersFromCpt
  Input:   Set of CPT parameters $C$, $\varepsilon$.
  Output:  Set of noisy-MAX parameters $M^*$,
           distance $d^*$.
  1) $M^* \leftarrow$ Initialize, $step \leftarrow$ Initialize
  2) $d^* \leftarrow$ CalculateDistance($M^*$,$C$)
  3) do
       a) $M \leftarrow M^*$, $d \leftarrow d^*$,$m^* \leftarrow NULL$.
       b) for each $m_i^* \in M^*$, do for $+step$ and $-step$
            i) $M \leftarrow$ ChangeMAX($m_i^*, M^*, step$)
           ii) $d \leftarrow$ CalculateDistance($M$,$C$)
          iii) if ($d < d^*$) then $d^* \leftarrow d$,
                 $m^* \leftarrow m_i^*, step^* \leftarrow step$.
       c) if ($m^* \neq NULL$)
          then $M^* \leftarrow$ ChangeMAX($m^*, M^*, step^*$)
          else $step \leftarrow$ decrease step.
     until ($d^* < \varepsilon$)
```

Fig. 3. Algorithm for fitting noisy-MAX parameters to a CPT

V. How Common Are Noisy-MAX Gates in Real Models

We decided to test several sizeable real world models in which probabilities were specified by an expert, learned from data, or obtained by a combination of both. Three models that fit the needs of our experiments were available to us: ALARM [14], HAILFINDER [13] and HEPAR II [9]. Basic characteristics of these networks are presented in the Table I (each of these models can be accessed at http://genie.sis.pitt.edu/networks.html). We verified by contacting the authors of these models that none of the CPTs in these networks were specified using the noisy-OR/MAX assumption. For each of the networks, we first identified all nodes that had at least two parents and then we applied our conversion algorithm to these nodes. HEPAR contains 31 candidate nodes, while ALARM and HAILFINDER contain 17 and 19 such nodes respectively. We applied the algorithm of Fig. 3 to each of the nodes using both D_E and D_{KL} measures and $\varepsilon = 10^{-4}$.

	Alarm	Hepar	Hailfinder
Number of nodes	37	71	56
Average in-degree	1.24	1.68	1.18
Maximal in-degree	4	6	4
Average outcomes count	2.84	2.32	3.98
Maximal outcomes count	4	4	11
Average CPT size	20.3	27.3	66.8
Maximal CPT size	108	384	1188

TABLE I
CHARACTERISTICS OF THE THREE NETWORKS.

It is important to note that the procedure, as described above, assumes that states of variables are already appropriately ordered to meet the noisy-MAX assumptions. Not surprisingly, for conditional probability distributions created without the noisy-MAX model in mind this is necessarily true. We resolved this problem by making the assumption that the order of values in nodes is always ascending or descending (i.e., states are never ordered as $\{$*hi*, *low*, *med*$\}$ but rather $\{$*low*, *med*, *hi*$\}$ or $\{$*hi*, *med*, *low*$\}$) and tested both the ascending and

Original CPT			Fitted CPT		
0.98	0.01	0.01	0.985	0.011	0.004
0.40	0.59	0.01	0.349	0.647	0.004
0.30	0.40	0.30	0.347	0.368	0.284
0.98	0.01	0.01	0.978	0.009	0.013
0.01	0.98	0.01	0.012	0.975	0.013
0.01	0.01	0.98	0.009	0.010	0.981

TABLE II
EXAMPLE OF FIT OF NOISY-MAX PARAMETERS TO CPT. NODE HRBP FROM THE ALARM NETWORK.

the descending order for each variable (parents and the child) in looking for the best fit.

A. Simple and Weighted Fit

Because we were interested in the question *How common are the noisy-MAX interactions in practical models?*, we decided to fit the noisy-MAX model to CPTs without taking into account the probabilities of parent instantiations. In other words, we assumed that the constants $w_{\mathbf{p}_i}$ in Eqs. 1 and 2 are always equal to 1. We decided to report these results together with the weighted distances. In the sequel, we will refer to uniformly weighted distributions as *simple* and the weighted according to probabilities of parent combinations as *weighted*.

We used two criteria to measure the goodness of fit between a CPT and its MAX-based equivalent: (1) *Average*, the average Euclidean distance (simple, i.e., not weighted by probabilities of parent instantiations) between the two corresponding parameters and (2) *MAX*, the maximal absolute value of difference between two corresponding parameters, which is an indicator of the single worst parameter fit for a given CPT.

Fig. 4 shows the results for the three tested networks for Euclidean and KL divergence measures (without weighting by parents states probabilities) respectively. In this and all subsequent figures, the x-axis shows the percentage rank of each node according to the node's fit to the noisy-MAX distribution (from the best to the worst fit). The y axis shows the Euclidean distance of the node's CPT from the node's MAX-based CPT. It shows, thus, either the average or the maximum distance between the parameters of the two CPTs. Using the Euclidean distance in all plots (we refer to it as *absolute distance*) allowed us to have an intuitive common denominator that has a clear interpretation on the probability scale.

The figures show the distance for all networks on one plot. The nodes in each of the networks are sorted according to the corresponding distance (*Average* or *MAX*) and the scale is converted to percentages. We can see for the *MAX* distance that as many as roughly 50% of the variables in two of the networks the greatest difference between two corresponding values in the compared CPTs was less than 0.1. We present the fit using weighted measures of distance D_E and D_{KL} in Fig. 5. We present an example of a fitted CPT with MAX distance around 0.05 in Table II.

B. Influence of the Size of CPT

We checked whether there is any dependence between the size of a CPT and the goodness of fit. Figure 6 shows goodness

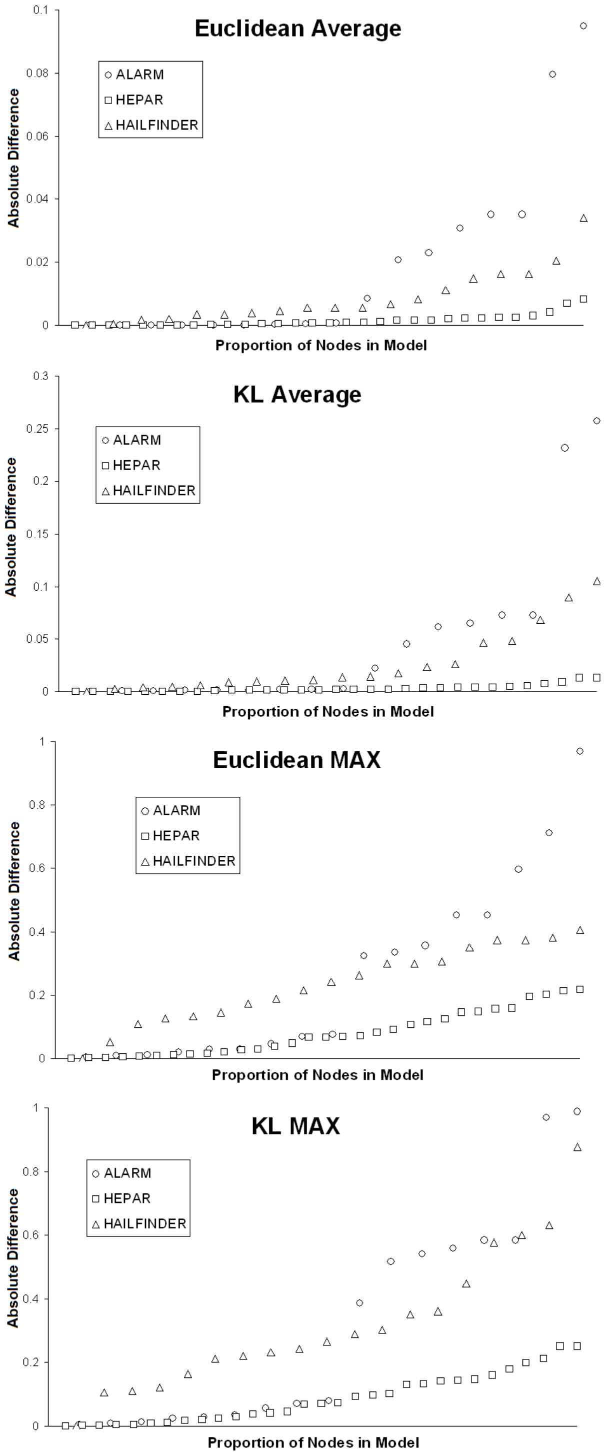

Fig. 4. The *Average* and *MAX* distance for the nodes of the three analyzed networks obtained using non-weighted distances. The horizontal axes show the fraction of the nodes, while the vertical axes show the absolute distance between parameters, a measure of the quality of the fit.

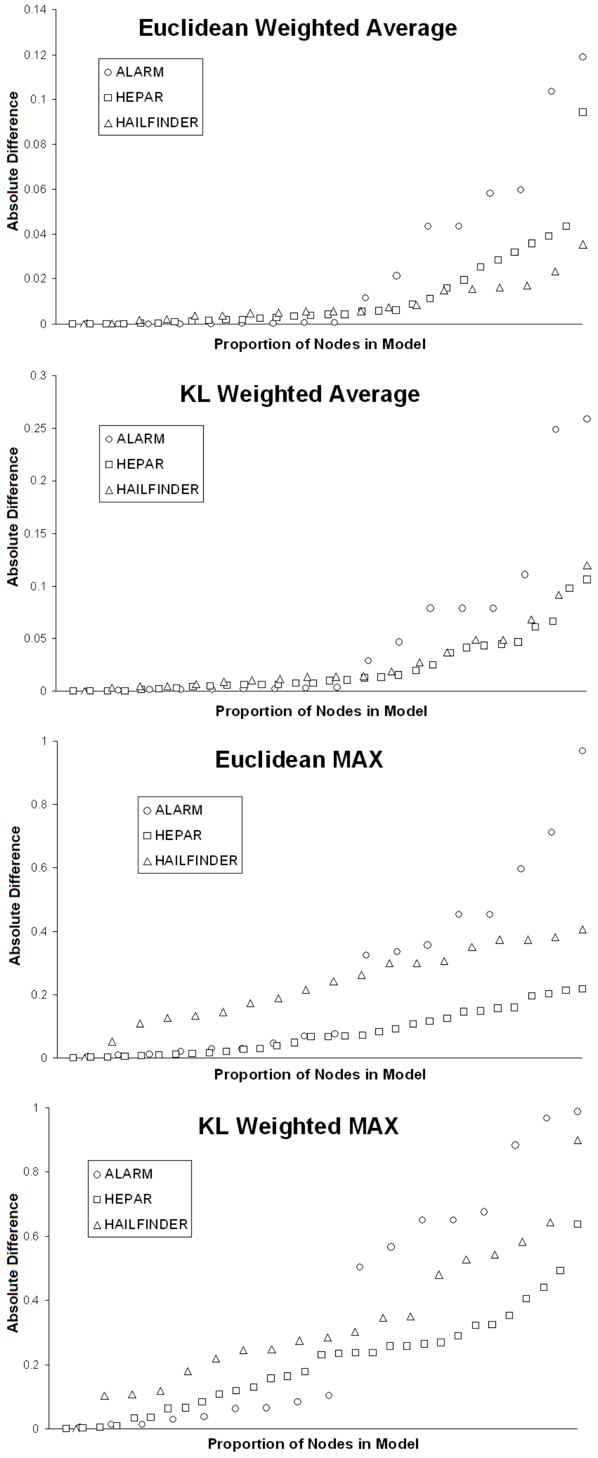

Fig. 5. The *Average* and *MAX* distance for the nodes of the three analyzed networks obtained using weighted distances. The horizontal axes show the fraction of the nodes, while the vertical axes show the absolute distance between parameters, a measure of the quality of the fit.

Number of Parents	2	3	4	5	6
Alarm	14	2	1	-	-
Hepar	16	8	3	3	1
Hailfinder	13	4	2	-	-

TABLE III
DISTRIBUTION OF PARENTS FOR NODES IN THE EXPERIMENTS.

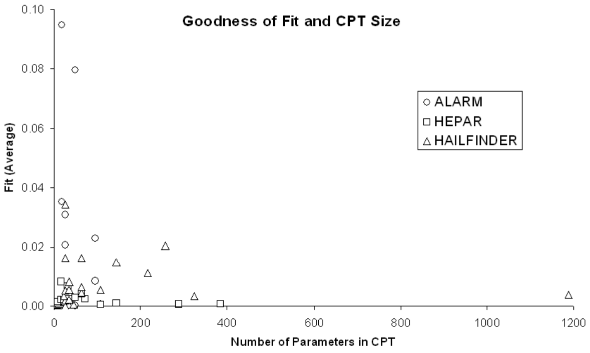

Fig. 6. Relation between number of parameters in CPT and goodness of fit.

of fit as a function of the size of CPT (measured in the number of parameters). There seem to be no obvious patterns in the plot. Table III shows the distribution of the number of parents for nodes under consideration. Generally, large CPTs tend to fit the noisy-MAX model just as well as smaller CPTs, although there are too few large CPTs in our networks to draw definitive conclusions.

C. Random CPTs

One possible explanation of our findings is that the noisy-MAX model is likely to fit well any randomly selected CPT. We decided to verify this by generating CPTs for binary nodes, with 2-5 parents (10,000 CPTs for every number of parents, for a total of 40,000 CPTs), whose parameters were sampled from the uniform distribution. To fit the noisy-MAX parameters we used Euclidean distance (a non-weighted version). Fig. 7 shows the results. On the x-axis there are generated CPTs sorted according to their fit to the noisy-OR using average and MAX measures. Except for the cases with two parents, the results are qualitatively different from the results we obtained using real models. They clearly indicate that a close approximation of a randomly generated CPT by the noisy-OR is highly improbable. Additionally, these results can provide an empirical basis for interpretation of values of distance measures. A relatively good fit for nodes with two binary parents is not surprising, because in this case we are dealing with approximating a function of four parameters with a function of three parameters. There are only 7 out of 16 noisy-MAX nodes with two parents that involve only binary variables (i.e., noisy-OR nodes) in the HEPAR II model.

D. Implications on Inference

Small differences in the conditional probabilities do not necessarily imply small differences in the posterior probability distributions, the main output of Bayesian networks. We

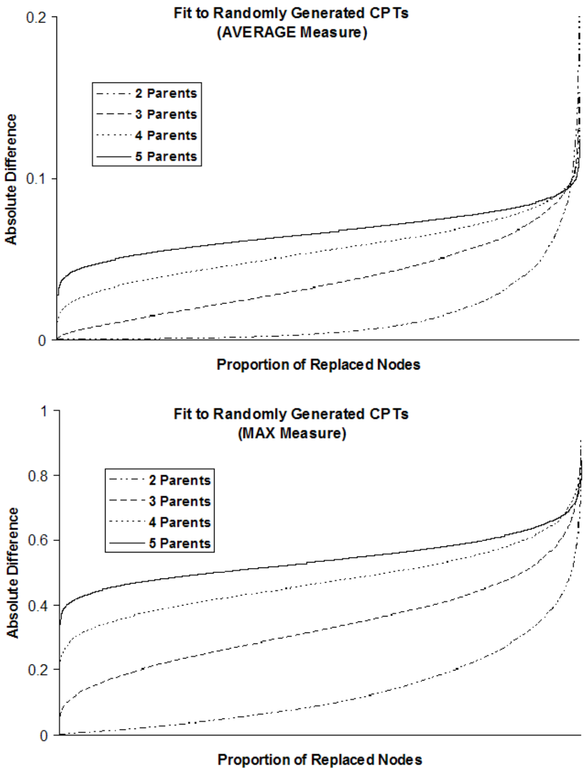

Fig. 7. The *Average* and *MAX* distances for randomly generated CPTs.

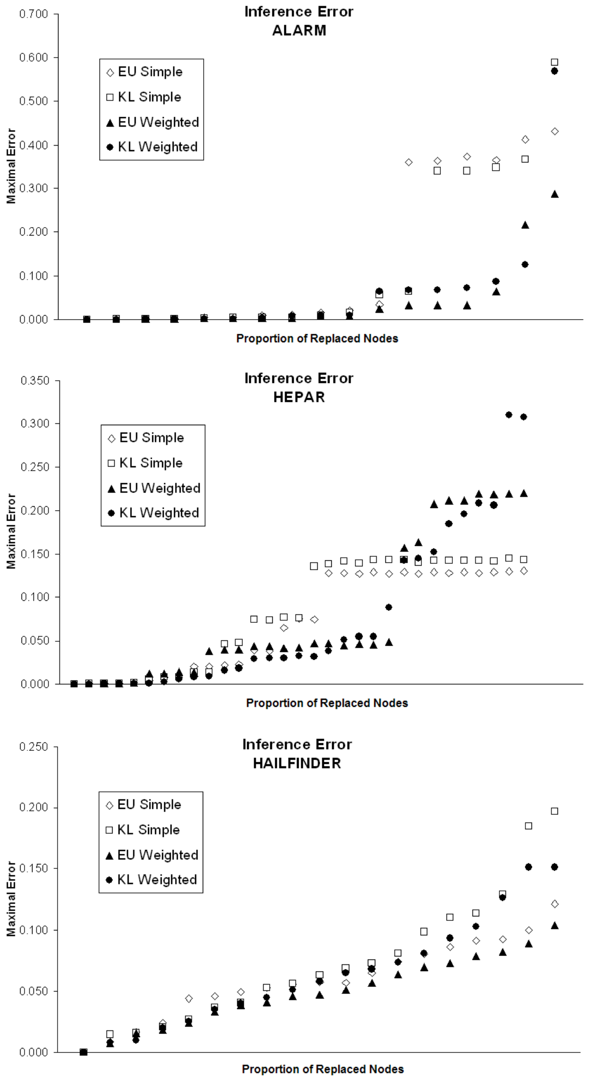

Fig. 8. Accuracy of the posterior probabilities for the three networks. Evidence sampled from the posterior distribution.

decided to test the accuracy of models in which a fraction of nodes has been converted into noisy-MAX gates. For each of the tested networks, we converted an increasing number of nodes (this was the independent variable) in the order of their fit to noisy-MAX gates using Euclidean and KL measures in both weighted and simple variants. In this way, after each node had been converted, a new model was created. For each such model we generated evidence for a randomly selected 10% of nodes in the network. In generating evidence, we repetitively selected a state of each evidence node according to the node's posterior probability distribution in the model, conditional on the evidence entered so far. We subsequently calculated the posterior probability distributions over the remaining nodes. We compared these posterior probabilities to those obtained in the original model (with the same evidence), which we treated as the gold standard. We repeated the procedure described above 1,000 times for each of the three models.

Fig. 8 shows the results of tests for accuracy of posterior probabilities for the three networks. The x-axis shows the percentage of nodes converted (always those with the best fit are converted). On the y-axis, there is absolute average maximal error between posterior probabilities for 1,000 trials. We observe a consistent tendency that the accuracy of the posterior probabilities is decreasing with the decreasing goodness of fit of the noisy-MAX to the CPT. From these results one can conclude that the weighted KL divergence is superior to other distances, when it comes to CPTs which are good fits to the noisy-MAX (these at the left hand side of x-axis). The nodes on the right hand side are usually not of much interest, as they represent nodes that are not a good fit. The nature of maximal error can explain the weighted distances' worse performance in the case of the HEPAR II network. The HEPAR II network has many small probabilities, and these lead to orders of magnitude differences in probabilities of parents' instantiations. The weighted distances work well on the average, but their performance can be worse for less likely scenarios and this additionally can be amplified by using the maximal measure, which captures the worst case scenario.

E. Comparison with Expert's Knowledge

In the later version of the HEPAR II model, a domain expert's knowledge was used to identify variables that are candidates for the noisy-MAX distributions [9]. We had access to this later version of the model and we could compare the results obtained by means of our algorithm to the variables indicated by the expert. The comparison of nodes indicated by the expert and fit to the noisy-MAX according to the algorithm are presented in Table IV. Apparently, the expert's

name	fit	expert	name	fit	expert
carcinoma	0.000	yes	pain ruq	0.073	yes
hbsag anti	0.001		inr	0.092	yes
hcv anti	0.001	yes	transfusion	0.098	yes
Cirrhosis	0.005		bleeding	0.101	
Hyperbil	0.005		ChHepatitis	0.132	yes
spiders	0.010		cholesterol	0.133	yes
Steatosis	0.011		ast	0.142	yes
pain	0.018	yes	bilirubin	0.145	yes
nausea	0.020	yes	alt	0.148	yes
hbeag	0.024	yes	ESR	0.161	yes
THepatitis	0.030	yes	ggtp	0.178	yes
anorexia	0.038	yes	injections	0.200	yes
hbc anti	0.040	yes	phosphatase	0.213	yes
pressure ruq	0.045	yes	hepatomegaly	0.250	yes
fatigue	0.067	yes	PBC	0.250	
hbsag	0.069	yes			

TABLE IV

VARIABLES IN THE HEPAR MODEL IDENTIFIED BY A DOMAIN EXPERT AS NOISY-MAX AND THEIR FIT TO THE NOISY-MAX DISTRIBUTIONS ACCORDING TO THE ALGORITHM.

strategy was to convert as many variables as possible with large CPTs to the noisy-MAX, even though they were not a good fit. The surprising part is that the expert did not identify variables that were the best fit – she listed only 4 of the 10 variables with the best fit. Upon a closer investigation, we concluded that the remaining 6 nodes that were not identified by the expert are indeed a good fit to the noisy-MAX, but they required reversing the order of states, a manipulation that was probably beyond the expert's modeling skills. The other interesting observation was that the expert rejected using the noisy-MAX when *gender* node was a parent — because it seemed to be a disabling factor which can not be modeled by the noisy-MAX. But it turned out that after reversing the child node's states, the interaction can be modeled successfully by the noisy-MAX model. The conclusion is that experts can have difficulty in identifying noisy-MAX relations when it comes to non-trivial meaning of variable outcomes.

F. Learning from Data

Even though this work does not touch on the learning from data aspect, it may be worth noting that the results for the weighted KL-divergence can provide some insights into learning the noisy-MAX models from data. The log-likelihood methods can be used to find a set of noisy-MAX parameters that minimizes the weighed KL-divergence. As discussed in [19], for a certain class of BN that includes the noisy-MAX, the set of parameters that maximize the log-likelihood of the data also maximizes the conditional log-likelihood.

VI. DISCUSSION AND CONCLUSIONS

In this paper, we introduced two measures of distance between two CPTs – one based on the Euclidean distance and one based on the KL divergence. We proved that Euclidean distance between any CPT and a MAX-based CPT, as a function of the noisy-MAX parameters of the latter, has exactly one minimum. We applied this result to an algorithm that given a CPT finds a noisy-MAX distribution that provides the best fit to it. Subsequently, we analyzed CPTs in three existing Bayesian network models using both measures. Our experimental results suggest that the noisy-MAX gate may provide a surprisingly good fit for as many as 50% of CPTs in two of the three analyzed networks. We demonstrated that this result was unlikely to be observed in randomly generated CPTs. It should be made clear that this results does not need to hold in the general case. In fact, different BN models can have diverse characteristics, both in terms of graphical structure and encoded probability distributions, making it difficult to generalize results of studies like this one. However, the fact remains that based on a sample of three models, a substantial number of local probability distributions can be fitted to noisy-MAX distributions.

We tested the influence of accuracy of the approximation of CPTs by noisy-MAX gates on the accuracy of posterior probabilities, showing that converting some CPTs to noisy-MAX gates can provide a good approximation to the original models. Our results provide strong empirical support for the practical value of the noisy-MAX models. We showed that the relation defined by the noisy-MAX often approximates interactions in the modeled domain reasonably well.

We might expect such result in networks that were elicited from human experts (HAILFINDER and ALARM). One of the reasons for that may be that humans tend to simplify their picture of the world by conceptualizing independencies among causal mechanisms. The fact that we observed as many as 50% noisy-MAX gates in a model whose parameters were learned from a data set (HEPAR II) is puzzling. In fact, the goodness of fit for the HEPAR II network was better than that of the HAILFINDER network. It seems to us, based on this result, that the independence of causal interactions may be fairly common in real-world probability distributions.

We envisage two applications of the proposed technique. The first is using our algorithm to discover noisy-MAX relationships in initial versions of CPTs elicited from experts, or directly from data when such are available, and then refocus knowledge engineering effort to noisy-MAX distributions. The second is in approximate algorithms for Bayesian networks. Whenever a fit is good, a CPT can be replaced by an ICI gate, leading to potentially significant savings in computation. It is possible that a small change in the CPT parameters can affect the accuracy of such a transformed network. Therefore, we suggest using sensitivity analysis techniques to test possible effects of converting nodes to the noisy-MAX.

ACKNOWLEDGMENTS

While we assume full responsibility for any possible shortcomings of this paper, we would like to thank Javier Díez for valuable comments that improved our presentation. We would also like to thank Tsai-Ching Lu for stimulating discussions on the topic of ICI and Agnieszka Oniśko for making versions of HEPAR II model available to us. All experimental data have been obtained using SMILE, a Bayesian inference engine developed at Decision Systems Laboratory and available at `http://genie.sis.pitt.edu/`.

References

[1] J. Pearl, *Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference*. San Mateo, CA: Morgan Kaufmann Publishers, Inc., 1988.

[2] I. Good, "A causal calculus (I)," *British Journal of Philosophy of Science*, vol. 11, pp. 305–318, 1961.

[3] Y. Peng and J. A. Reggia, "Plausibility of diagnostic hypotheses," in *Proceedings of the 5th National Conference on AI (AAAI–86)*, Philadelphia, 1986, pp. 140–145.

[4] M. Henrion, "Some practical issues in constructing belief networks," in *Uncertainty in Artificial Intelligence 3*, L. Kanal, T. Levitt, and J. Lemmer, Eds. New York, N. Y.: Elsevier Science Publishing Company, Inc., 1989, pp. 161–173.

[5] F. J. Díez, "Parameter adjustement in Bayes networks. The generalized noisy OR–gate," in *Proceedings of the 9th Conference on Uncertainty in Artificial Intelligence*. Washington D.C.: Morgan Kaufmann, San Mateo, CA, 1993, pp. 99–105.

[6] F. J. Díez, J. Mira, E. Iturralde, and S. Zubillaga, "DIAVAL, a Bayesian expert system for echocardiography," *Artificial Intelligence in Medicine*, vol. 10, pp. 59–73, 1997.

[7] M. Pradhan, G. Provan, B. Middleton, and M. Henrion, "Knowledge engineering for large belief networks," in *Proceedings of the Tenth Annual Conference on Uncertainty in Artificial Intelligence (UAI–94)*. San Francisco, CA: Morgan Kaufmann Publishers, 1994, pp. 484–490.

[8] M. Shwe, B. Middleton, D. Heckerman, M. Henrion, E. Horvitz, H. Lehmann, and 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*, vol. 30, no. 4, pp. 241–255, 1991.

[9] A. Oniśko, M. J. Druzdzel, and H. Wasyluk, "Learning Bayesian network parameters from small data sets: Application of Noisy-OR gates," *International Journal of Approximate Reasoning*, vol. 27, no. 2, pp. 165–182, 2001.

[10] F. J. Díez and S. F. Galán, "Efficient computation for the noisy MAX," *Int. J. Int. Syst.*, vol. 18, no. 2, pp. 165–177, 2004.

[11] N. Zhang and D. Poole, "Exploiting causal independence in Bayesian network inference," *Journal of Artificial Intelligence Research*, vol. 5, pp. 301–328, 1996.

[12] D. Heckerman and J. S. Breese, "A new look at causal independence," in *Proceedings of the Tenth Annual Conference on Uncertainty in Artificial Intelligence (UAI–94)*. San Francisco, CA: Morgan Kaufmann Publishers, 1994, pp. 286–292.

[13] I. Beinlich, H. Suermondt, R. Chavez, and G. Cooper, "The ALARM monitoring system: A case study with two probabilistic inference techniques for belief networks," in *Proceedings of the Second European Conference on Artificial Intelligence in Medical Care*, London, 1989, pp. 247–256.

[14] B. Abramson, J. Brown, W. Edwards, A. Murphy, and R. Winkler, "Hailfinder: A Bayesian system for forecasting severe weather," in *International Journal of Forecasting*, Amsterdam, 1996, pp. 57–71.

[15] S. Srinivas, "A generalization of the noisy-OR model," in *Proceedings of the Ninth Annual Conference on Uncertainty in Artificial Intelligence (UAI–93)*. San Francisco, CA: Morgan Kaufmann Publishers, 1993, pp. 208–215.

[16] E. Santos Jr., "On linear potential functions for approximating Bayesian computations," *Journal of the ACM*, vol. 43, no. 3, pp. 399–430, 1996.

[17] C. Boutilier, N. Friedman, M. Goldszmidt, and D. Koller, "Context-specific independence in Bayesian networks," in *Proceedings of the Twelfth Annual Conference on Uncertainty in Artificial Intelligence (UAI–96)*. San Francisco, CA: Morgan Kaufmann Publishers, 1996, pp. 115–123.

[18] L. Lee, "On the effectiveness of the skew divergence for statistical language analysis," in *Artificial Intelligence and Statistics 2001*, 2001, pp. 65–72.

[19] T. Roos, H. Wettig, P. Grünwald, P. Myllymäki, and H. Tirri, "On discriminative Bayesian network classifiers and logistic regression," *Machine Leanrning*, vol. 59, no. 3, pp. 267–296, 2005.

Adam Zagorecki received his M.Sc. in computer science from Bialystok University of Technology (1999), and Ph.D. in information science from University of Pittsburgh (2010).

He is currently a Research Fellow at the Operational and Decision Analysis Group,Department of Informatics and Systems Engineering, Cranfield University, UK. He has held Post-doctoral Fellow position at Graduate School of Public and International Affairs, University of Pittsburgh and Senior Research Programmer at Robotics Institute, Carnegie Mellon University. His research interests include probabilistic methods, decision support systems, operational analysis, and machine learning. His expertise lies in Bayesian networks: knowledge elicitation and learning from data.

Marek J. Druzdzel is a senior member of IEEE since 2004. He received his M.Sc. degrees in computer science (1985) and electrical engineering (1987) from Delft University of Technology in The Netherlands (both with distinction) and his Ph.D. in engineering and public policy (1992) from Carnegie Mellon University, Pittsburgh, PA, USA.

He is currently an associate professor in the School of Information Sciences and Intelligent Systems Program at the University of Pittsburgh, USA, where he heads the Decision Systems Laboratory, an interdisciplinary research group focusing on normative decision support systems. He is also a visiting professor in the Department of Computer Science, Bialystok University of Technology, Poland. His research interests concentrate on probabilistic and decision theoretic reasoning in decision support systems. He is the principal designer of GeNIe, a Bayesian modeling system with thousands of users World-wide and available at http://genie.sis.pitt.edu/. More details about his research interests and publications can be found at http://www.pitt.edu/ druzdzel.