Universiteit
van
Amsterdam

IAS technical report IAS-UVA-06-01

Inference Meta Models: A New Perspective On Belief Propagation With Bayesian Networks

Gregor Pavlin, Jan Nunnink, and Frans Groen
Intelligent Systems Laboratory Amsterdam,
University of Amsterdam
The Netherlands

We investigate properties of Bayesian networks (BNs) in the context of robust state estimation. We focus on problems where state estimation can be viewed as a classification of the possible states, which in turn is based on the fusion of heterogeneous and noisy information. We introduce a coarse perspective of the inference processes and show that classification with BNs can be very robust, even if we use models and evidence associated with significant uncertainties. By making coarse and realistic assumptions we can (i) formulate asymptotic properties of the classification performance, (ii) identify situations in which Bayesian fusion supports robust inference and (iii) introduce techniques that support detection of potentially misleading inference results at runtime. The presented coarse grained analysis from the runtime perspective is relevant for an important class of real world fusion problems where it is difficult to obtain domain models that precisely describe the true probability distributions over different states.

Keywords: Bayesian networks, robust information fusion, heterogeneous information.

Dagstuhl Seminar Proceedings 05381
Form and Content in Sensor Networks
http://drops.dagstuhl.de/opus/volltexte/2006/756

IAS
intelligent autonomous systems

Contents

1 Introduction **1**

2 State Estimation with Bayesian networks **1**
2.1 Estimation Accuracy 2
2.2 Bayesian networks 2
2.3 Factorization 3

3 Inference Processes **4**
3.1 Prediction 5
3.2 Diagnostic Inference 5
3.3 Robustness of Inference Processes 6

4 Factor Accuracy **7**
4.1 Updating Tendencies 8
4.2 True Distributions and Inference 9

5 Inference Meta Model **11**
5.1 Inference Faults 12
5.2 A Coarse Perspective on Inference 12
5.3 Reinforcement Counter Distributions 13
5.4 Robust Inference 14
5.5 Reinforcement Propagation 15

6 Applications **16**
6.1 Design of Robust Inference Systems 17
6.2 Coping with Imprecise Models by Using an Alternative Belief Propagation Method 18
6.3 Runtime Analysis of the Inference Quality 19

7 Discussion **21**
7.1 Causal Models with Simple Topologies 21
7.2 Extending the IMM to More Complex Topologies 24
7.3 Related Work 25
7.4 Further Research 26

Intelligent Autonomous Systems
Informatics Institute, Faculty of Science
University of Amsterdam
Kruislaan 403, 1098 SJ Amsterdam
The Netherlands
Tel (fax): +31 20 525 7461 (7490)
http://www.science.uva.nl/research/ias/

Corresponding author:
Gregor Pavlin
tel: +31 20 525 7555
gpavlin@science.uva.nl
http://www.science.uva.nl/~gpavlin/

1 Introduction

Modern situation assessment and controlling applications often require efficient fusion of large amounts of heterogeneous and uncertain information. In addition, fusion results are often mission critical. It turns out that Bayesian networks (BN) [22] are suitable for a significant class of such applications, since they facilitate modeling of very heterogeneous types of uncertain information and support efficient belief propagation techniques. BNs are based on solid theoretical foundations which facilitate (i) analysis of the robustness of fusion systems and (ii) monitoring of the fusion quality.

We assume domains where situations can be described through sets of discrete random variables. A situation corresponds to a set of hidden and observed states that the nature 'sampled' from some true distribution over the combinations of possible states. Thus, in a particular situation certain states materialized while others did not, which corresponds to a point-mass distribution over the possible states. Consequently, the state estimation can be reduced to a classification of the possible combinations of relevant states. We assume that there exist mappings between hidden states of interest and optimal decisions/actions. *In this context, we consider classification of the states accurate if it is equivalent to the truth in the sense that knowing the truth would not change the action based on the classification.*

We focus on classification based on the estimated probability distributions (i.e. beliefs) over the hidden states. These distributions are estimated with the help of BNs, which facilitate systematic fusion of information about observations with the prior knowledge about the stochastic processes. BNs define mappings between observations and hypotheses about hidden events and, consequently, BNs have a significant impact on the classification accuracy. In general, one of the most challenging problems associated with BNs is determination of adequate modeling parameters [7].

We emphasize a fundamental difference between the model accuracy and the estimation accuracy. In general, a BN is a generalization over many possible situations that captures the probability distributions over the possible events in the observed domain. However, even a perfect generalization does not necessarily support accurate classification in a particular situation. For example, consider a domain in which 90% of fires cause smoke. While it is common that fires cause smoke, in rare cases we might have a fire but no smoke. By applying diagnostic inference we could use smoke detector reports to reason about the existence of a fire. Such inference is based on a sensor model, a generalization which describes the probability that a fire will cause smoke. Consequently, observing the absence of smoke would in such a rare case decrease our belief in the presence of fire, leading our belief away from the truth, even if the used BN were a perfect generalization.

In this paper we expose properties of BNs which are very relevant for the design of robust information fusion systems in real world applications. We show that certain types of BNs support robust inference. In addition, we introduce the Inference Meta Model (IMM), a new runtime perspective on inference in BNs which supports analysis of the inherent fusion robustness and can provide additional information on the fusion quality.

2 State Estimation with Bayesian networks

In general, human decision makers or artificial intelligent systems make use of mappings between the constellations of relevant states and actions. We assume that the relevant states of the environment can be captured sufficiently well by finite sets of discrete variables. Thus, each combination of variable instantiations corresponds to a certain choice of actions.

Moreover, in real world applications we can often directly observe only a fraction of the variables of interest. Consequently, we have to estimate the states of interest with the help

of models that describe relations between the observed and hidden variables, i.e. variables representing events that cannot be observed directly. In addition, in real world applications we usually deal with stochastic domains. In other words, we often do not know with certainty which states of the hidden variables materialized. Instead, we associate each possible state of a variable with a hypothesis that the state materialized. Each hypothesis is associated with a score, a posterior probability determined with the help of probabilistic causal models that map constellations of observed states to probability distributions over hidden states.

We assume that the hypothesis whose score exceeded a certain threshold corresponds to the truth. Thus, the state estimation process can be reduced to a classification problem.

2.1 Estimation Accuracy

We define accurate state estimation in the decision making context. Suppose that each constellation of states is associated with an optimal decision d_i. If the decision maker knew that state h_i materialized she would make the decision d_i corresponding to that state. However, she cannot directly observe the true state. Instead, she is supplied with a posterior probability distribution $\hat{P}(h_i|\mathcal{E})$ over the possible states of variable H that is based on the current observations $\mathcal{E}$. Moreover, for each possible state h_i we define a threshold θ_{h_i} in such a way that only one of the possible thresholds can be exceeded at a time. If the estimated $\hat{P}(h_i|\mathcal{E}) > \theta_{h_i}$ then decision d_i is made as though the true state would be h_i. In this decision making context we define accurate state estimation:

Definition 1 (Accurate Distribution) *A posterior distribution $\hat{P}(H|\mathcal{E})$ is considered* accurate *iff there exists a decision threshold θ_{h_i} such that $\hat{P}(h_i|\mathcal{E}) > \theta_{h_i}$ and $h_i = h^*$.*

Thus, the threshold corresponding to the true state h^* is exceeded if $\hat{P}(H|\mathcal{E})$ gets sufficiently close to the true distribution $P(H)$. In other words, the state estimation can be reduced to a classification of the possible combinations of relevant states.

Obviously, the classification quality is related to the divergence between the estimated and the true distributions. Throughout this paper we use the Kullback-Leibler divergence and assume that there exists a constant δ corresponding to a decision threshold θ_{h_i}, such that $\hat{P}(H|\mathcal{E})$ will result in the correct decision if $\mathrm{KL}(P(H) \parallel \hat{P}(H|\mathcal{E})) < \delta$.

Note, in this paper $\hat{P}(.)$ refers to modeling parameters and estimated probabilities, while $P(.)$ without a hat denotes true probabilities in the modeled world.

2.2 Bayesian networks

We assume that $\hat{P}(H|\mathcal{E})$ is computed with the help of Bayesian networks (BNs), which support theoretically rigorous modeling and belief propagation. A Bayesian network is defined as a tuple $\langle \mathcal{D}, P \rangle$, where $\mathcal{D} = \langle \mathcal{V}, E \rangle$ is a directed a-cyclic graph defining a domain $\mathcal{V} = \{V_1, \ldots, V_n\}$ and a set of directed edges $\langle V_i, V_j \rangle \in E$ over the domain. The joint probability distribution over the domain $\mathcal{V}$ is defined as

$$\hat{P}(\mathcal{V}) = \prod_{V_i \in \mathcal{V}} \hat{P}(V_i|\pi(V_i)),$$

where $\hat{P}(V_i|\pi(V_i))$ is the conditional probability table (CPT) for node V_i given its parents $\pi(V_i)$ in the graph. In this paper, we assume that each node represents a discrete variable. In general, probability distributions over arbitrary sets of discrete variables can be computed through appropriate marginalization of $P(\mathcal{V})$ and they are described through real-valued tables called potentials[1] [12].

[1] Note that CPTs are also potentials

BNs can be used as causal models [23, 17] that describe probabilistic relations between different hidden phenomena and heterogeneous sensory observations (see example in figure 1). In a BN we choose a hypothesis node H with states h_i and compute probability distribution $\hat{P}(H|\mathcal{E})$ over H for a given evidence pattern $\mathcal{E}$ (e.g. sensory observations). Evidence $\mathcal{E}$ corresponds to a certain constellation of node instantiations and subsequent inference (i.e. information fusion) results in a distribution $\hat{P}(H|\mathcal{E})$ that determines a "score" $\hat{P}(h_i|\mathcal{E})$ for each hypothesis $h_i \in H$. Moreover, given H we can define a *conditionally independent network fragment*:

Definition 2 (Conditionally Independent Network Fragment) *Given a BN and a classification variable H, i^{th}* conditionally independent network fragment $\mathcal{F}_i^H$ *is a set of nodes that include node H and are d-separated from other parts of a BN by H. All nodes within $\mathcal{F}_i^H$ are dependent given the variable H.*

2.3 Factorization

D-separation implies conditional independence between the modeled variables, which corresponds to a specific factorization of the estimated posterior probability distribution $\hat{P}(H|\mathcal{E})$. Namely, $\bigcap_i \mathcal{F}_i^H = \{H\}$, which means that the potentials corresponding to a particular network fragment $\mathcal{F}_i^H$ do not share any variables with the potentials associated with other network fragments, except the hypothesis variable H. Thus, each network fragment $\mathcal{F}_i^H$ is associated with a factor $\phi_i(H)$ resulting from a marginalization of all variables from this fragment except H and the evidence variables from $\mathcal{F}_i^H$ that were instantiated according to the evidence $\mathcal{E}_i$. This is reflected in the following factorization:

$$
\begin{aligned}
\hat{P}(H,\mathcal{E}) = \sum_{\mathcal{V}\setminus H} \hat{P}(\mathcal{V}) \prod_{e_k\in\mathcal{E}} e_k = \\
&\left.\sum_{\mathcal{V}_0\setminus H} \prod_{V_i\in\mathcal{V}_0} \hat{P}(V_i|\pi(V_i)) \prod_{e_k\in\mathcal{E}_0} e_k \quad\right\} \phi_0(H) \\
&\cdot \left.\sum_{\mathcal{V}_1\setminus H} \prod_{V_i\in\mathcal{V}_1\setminus H} \hat{P}(V_i|\pi(V_i)) \prod_{e_k\in\mathcal{E}_1} e_k \quad\right\} \phi_1(H) \\
&\cdots \\
&\cdot \left.\sum_{\mathcal{V}_m\setminus H} \prod_{V_i\in\mathcal{V}_m\setminus H} \hat{P}(V_i|\pi(V_i)) \prod_{e_k\in\mathcal{E}_m} e_k, \quad\right\} \phi_m(H)
\end{aligned}
\tag{1}
$$

$\mathcal{V}_0$ denotes all nodes from the network fragment $\mathcal{F}_0^H$ that includes all predecessor nodes of H, while $\mathcal{V}_i$ $(i = 1,\ldots,m)$ is the set of nodes contained in the fragments consisting of H's successors only. In addition, $\prod_{e_k\in\mathcal{E}_i} e_k$ denotes the instantiations of the evidence nodes in the $i-th$ network fragment $\mathcal{F}_i^H$ (see [12]). Since H d-separates all sets $\mathcal{V}_i$ (see Definition 2) we can identify conditionally independent factors $\phi_i(H)$ $(i = 0,\ldots,m)$ whose product determines the resulting joint probability. Each factor $\phi_i(H)$, is a function that yields a value $\phi_i(h_i)$ for each state h_i of H. In other words, $\phi_i(H)$ is a vector of scalars corresponding to the states of H. *Each factor $\phi_i(H)$ corresponds to an independent opinion over H based on a subset $\mathcal{E}_i \subseteq \mathcal{E}$ of all observations $\mathcal{E}$.*

By considering the d-separation, we can further distinguish between *Predictive* and *Diagnostic* conditionally independent network fragments.

Definition 3 (Predictive Network Fragment) *Given a probabilistic causal model and a hypothesis variable H, a* Predictive conditionally independent network fragment $\mathcal{F}_i^H$ *relative to H includes (1) all ancestors $\pi^*(H)$ of H and (2) variables for which there exists at least one path to H via ancestor nodes $\pi^*(H)$.*

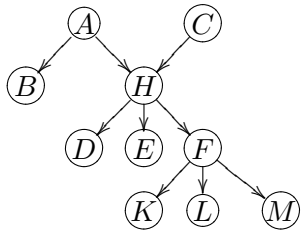

Figure 1: A causal model relating hypotheses represented by node H and different types of observations captured by nodes B, D, E, K, L and M.

In general, given definition 3, we can show that in any BN we can find at most one predictive fragment if the predecessors of H do not form special Independence of Causal Influence models (ICI), such as noisy-OR gates [11, 22].

Definition 4 (Diagnostic Network Fragment) *Given a probabilistic causal model and a class variable H, a* Diagnostic conditionally independent network fragment $\mathcal{F}_i^H$ *relative to variable H does not include any predecessors of H.*

By considering causality, we see that *Diagnostic conditionally independent network fragments* provide retrospective support for the belief over H. In other words, factors corresponding to such fragments update belief over H by considering only the evidence nodes that H d-separates from all H's predecessors. As we will show in the following discussion, this has important implications w.r.t. the factorization and classification robustness.

For the sake of clarity, in this paper we limit our discussion to domains that can be described with BNs featuring poly-tree topologies [2]. Consequently, a predictive fragment can never contain a descendant from the classification variable H and each child of H corresponds to a specific diagnostic fragment. For example, given the DAG shown in Figure 1 and an evidence set $\mathcal{E} = \{b_1, d_2, e_1, k_2, l_1, m_1\}$ we obtain the following factorization:

$$\hat{P}(h_i, \mathcal{E}) = \overbrace{\sum_A \hat{P}(A)\hat{P}(b_1|A) \sum_C \hat{P}(C)\hat{P}(h_i|A,C)}^{\phi_0(h_i)} \cdot \underbrace{\sum_F \hat{P}(F|H)\hat{P}(k_2|F)\hat{P}(l_1|F)\hat{P}(m_1|F)}_{\phi_1(h_i)} \underbrace{\hat{P}(d_2|h_i)}_{\phi_2(h_i)} \underbrace{\hat{P}(e_1|h_i)}_{\phi_3(h_i)} \tag{2}$$

In this example a single predictive fragment $\mathcal{F}_0^H$ consists of variables A, B, C and H, while there are three diagnostic fragments $\mathcal{F}_1^H$, $\mathcal{F}_2^H$ and $\mathcal{F}_3^H$, each corresponding to a child of H. Moreover, variable instantiations in fragments $\mathcal{F}_0^H$, $\mathcal{F}_1^H$, $\mathcal{F}_2^H$ and $\mathcal{F}_3^H$ were based on evidence subsets $\mathcal{E}_0 = \{b_1\}$, $\mathcal{E}_1 = \{d_2\}$, $\mathcal{E}_2 = \{e_1\}$ and $\mathcal{E}_3 = \{k_2, l_1, m_1\}$, respectively. Note also that the Predictive fragment $\mathcal{F}_0^H$ is associated with a single factor $\phi_0(H)$.

3 Inference Processes

In general, probabilistic inference (also called belief propagation) in BNs can be viewed as a series of multiplication and marginalization steps that combine predefined modeling parameters

[2] The discussion can be extended to more general topologies which, however, is out of scope of this paper.

according to the observed evidence. Moreover, belief propagation in BNs is a combination of predictive and diagnostic inference processes [22]. In this section we discuss the two types of inference in a decision making context and analyze their robustness with respect to modeling inaccuracies.

3.1 Prediction

Predictive inference is reasoning about states of a hidden variable H that can materialize as a consequence of observed events $\mathcal{E}$. Given a probabilistic causal model, we infer the probability distribution $\hat{P}(H|\mathcal{E})$ over hidden states of the hypothesis variable H by considering observed instantiations of the variables from the set of ancestors π^*_H of H. Thus, we reason in the causal direction about the outcome of a stochastic causal process, which can be viewed as a sampling process on some true distribution $P(H|\mathcal{E})$. Note that $P(H|\mathcal{E})$ corresponds to a particular materialization of the states of variables from the set of H's ancestors π^*_H.

For example, consider a network fragment consisting of a hypothesis node H and n parents E_i (see Figure 2). Node H is associated with a CPT capturing $\hat{P}(H|E)$. By instantiating parents with evidence $\mathcal{E} = \{e_1, \ldots, e_n\}$, we express the distribution over the states of node H with $\hat{P}(H|e_1, \ldots, e_n)$, which is a column in $\hat{P}(H|E)$. Parents E in this example represent a single predictive network fragment and, according to the factorization properties emphasized in the previous section, we see that this corresponds to a single factor, i.e. $\hat{P}(H|e_1, \ldots, e_n) = \phi_0(H)$.

3.2 Diagnostic Inference

Diagnostic inference (or retrospective support [22]) is reasoning about hidden events that already took place and were followed by observations. Such inference is based on reversal of the causal relations captured by diagnostic network fragments. Moreover, in diagnostic reasoning we know that exactly one of the possible events took place. Therefore, the true distribution must be one of the possible point mass distributions:

$$P(h_i) = \begin{cases} 1 & \text{if } h_i = h^* \\ 0 & \text{otherwise} \end{cases} \tag{3}$$

In this context, classification based on diagnostic inference can be viewed as a choice of one of the true point mass distributions.

Moreover, in BNs with tree topologies all children of the classification variable H are conditionally independent given H. Consequently, according to definition 4, each child node of H corresponds to exactly one diagnostic factor. For example, consider a simple model with a hypothesis node H which is a root of n branches with evidence nodes (see Figure 3). The posterior distribution over the states of H is given by:

$$\hat{P}(H|\mathcal{E}) = \alpha \hat{P}(H) \prod_{e_j \in \mathcal{E}} \hat{P}(e_j|H), \tag{4}$$

where $\mathcal{E} = \{e_1, \ldots, e_n\}$ is the evidence set, e_j denotes the instantiated state of child E_j and α is a normalizing constant.

The likelihoods capture a generative model, which describes the distributions over effects of a certain cause. The likelihoods represent generalizations obtained through sampling in many different possible situations. As we will show later, the fact that diagnostic inference implements reasoning about a state corresponding to a point mass distribution has important implications with respect to the inference robustness.

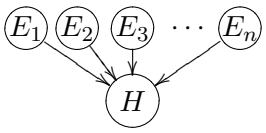

Figure 2: Predictive BN.

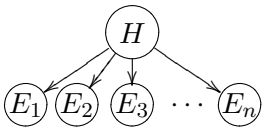

Figure 3: Diagnostic BN.

3.3 Robustness of Inference Processes

The robustness of inference processes can be expressed as the size of the parameter domain that guarantees a sufficiently small KL divergence between the posterior and the true distribution with high probability; i.e. the greater the domain from which the designer or the learning algorithm can choose adequate modeling parameters, the greater is the chance that inference will be accurate in different situations.

We can show that the choice of evidence nodes in a poly-tree influences the inherent inference robustness. In general, the predictive and diagnostic inference processes in tree like structures are very different with respect to the way the evidence is incorporated into the factorization. Namely, all ancestors of H and variables connected to H via its ancestors are summarized through a single predictive factor. Diagnostic inference, on the other hand, can be realized through several factors, each corresponding to a child of H.

Again, we assume that the estimation accuracy is related to the KL divergence between the true distribution over states of a hypothesis node $P(H)$ and the posterior distribution $\hat{P}(H|\mathcal{E})$ given the evidence set $\mathcal{E}$. We first consider a simple network in Figure 2, which consists of binary nodes. Also, let's assume a particular instantiation $\{e_1, \ldots, e_n\}$ of the n parent nodes (hard evidence) corresponding to a single distribution vector from the CPT. Suppose that the true probability $P(h) = 0.7$. We plot the corresponding $\mathrm{KL}(P(H) \parallel \hat{P}(H|e_1, \ldots, e_n))$ as a function of the relevant modeling parameter (see Figure 4). The figure shows that a sufficiently small divergence can be achieved if $\hat{P}(h|e_1, \ldots, e_n) \in [0.65, 0.75]$, which is a rather narrow interval.

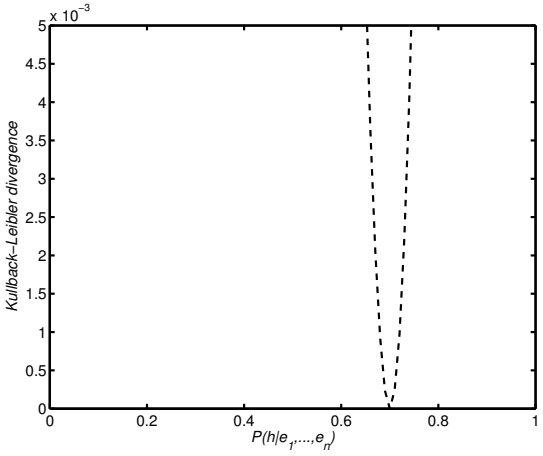

Figure 4: Divergence between the true and the posterior distribution for different parameters of a simple 'predictive' BN that guarantee a correct decision; i.e. $\mathrm{KL}(P(h) \parallel \hat{P}(h|\mathcal{E})) < 0.005$.

Next, consider an example of diagnostic inference based on a naive BN from Figure 3 where all n children, are associated with identical CPTs. Since we assumed binary variables, the CPTs can be specified by two parameters $\hat{P}(e|h)$ and $\hat{P}(e|\overline{h})$. We investigate the effect of changing

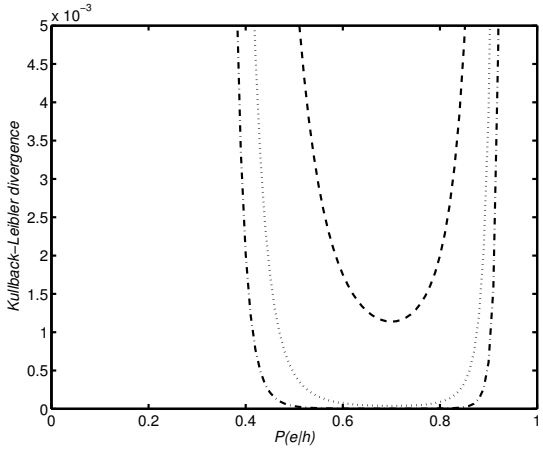

Figure 5: Divergence between the true and the posterior distribution for different parameters $\hat{P}(e|h)$ of a naive BN that guarantee a correct decision; i.e. $\mathrm{KL}(P(h) \parallel \hat{P}(h|\mathcal{E})) < 0.005$. Different curves correspond to the following numbers of children nodes: 20 (dashed), 30 (dotted) and 40 (dash-dotted).

$\hat{P}(e|h)$ and fix $\hat{P}(e|\overline{h}) = 0.3$ which is equal to the true conditional distribution. We assume that the true probability $P(h) = 1$. Figure 5 depicts the divergence for different values of $\hat{P}(e|h)$, where each curve represents a different number of children n. On the horizontal axis we can identify intervals for values of $\hat{P}(e|h)$, for which the divergence $\mathrm{KL}(P(h) \parallel \hat{P}(h|\mathcal{E})) < 0.005$. From this diagram it is apparent that the intervals, from which we can choose adequate $\hat{P}(e|h)$, grow with the number of children. In other words, diagnostic inference becomes inherently robust if we use BNs with sufficiently large branching factors. In such cases we can pass the correct decision threshold under a wide choice of modeling parameters. This implies that the likelihood of choosing inadequate modeling parameters in a given situation is reduced. Contrary to the predictive inference example, we see that the redundancy with respect to the evidence nodes does improve the robustness.

While predictive inference is sufficiently accurate only if we can obtain parameters that precisely describe the true distributions over events of interest, we see that parameter precision is not crucial for diagnostic inference. In other words, the redundancy of parameters plays an important role w.r.t. the robustness.

4 Factor Accuracy

Examples from the preceding section suggest that inference in BNs can be robust if the underlying process models have topologies featuring many conditionally independent factors. We explain these properties with the help of a coarse runtime perspective. We investigate under which conditions the factors support accurate fusion. We show that inference processes can be very robust if the CPTs merely capture simple relations between the true conditional probability distributions and the BN topology corresponds to many factors in the posterior factorization. We argue that because of this property the fusion can be inherently robust since such relations can be identified easily by the designers or machine learning algorithms.

4.1 Updating Tendencies

In order to be able to analyze the impact of the modeling parameters on the classification with BNs, we focus our attention on inference processes. Consider again the example from the previous section (see Figure 1). Recall that each instantiation of a network fragment that is d-separated from the other parts of the network by H corresponds to a factor in the expression describing the distribution over H. For each such conditionally independent network fragment we can observe, that if we multiply the conditional equation with the corresponding factor and normalize over all states of H, the posterior probability of one state will increase the most. For example suppose the parameters were $P(f_2|h_1) = 0.8$ and $P(f_2|h_2) = 0.3$. Observation of $F = f_2$, thus increased the posterior of h_1 the most. One could say that for observation $F = f_2$ state h_1 'wins'. Obviously, the state that wins sufficiently often will end up with the highest posterior probability. This suggests that it is not the exact factor values, but the relations between them that matter most with respect to the estimation accuracy. Therefore, for each factor $\phi_i(H)$ we introduce a factor reinforcement:

Definition 5 (Factor Reinforcement) *Assume a classification variable H and a fragment $\mathcal{F}_i^H$. Given some instantiation $\mathcal{E}_i$ of the evidence variables within $\mathcal{F}_i^H$, we can compute a factor $\phi_i(h_j)$ for each state h_j of variable H and determine the corresponding factor reinforcement r_i^H as follows:*

$$r_i^H = \arg\max_{h_j} \phi_i(h_j). \tag{5}$$

Note that factor $\phi_i(H)$ either captures the likelihood of states of H, if it corresponds to a diagnostic fragment, or it represents a prior over H if it corresponds to a predictive fragment.

In other words, reinforcement r_i^H is a function that returns the state h_j of variable H, whose probability is increased the most (i.e. reinforced) by instantiating nodes of the fragment $\mathcal{F}_i^H$ corresponding to factor $\phi_i(H)_i$. For example, given factorization (2), we obtain four reinforcements: $r_0^H = \arg\max_{h_i} \phi_0(h_i)$, $r_1^H = \arg\max_{h_i} \phi_1(h_i)$, $r_2^H = \arg\max_{h_i} \phi_2(h_i)$ and $r_3^H = \arg\max_{h_i} \phi_3(h_i)$.

Moreover, we can define an accurate reinforcement:

Definition 6 (Accurate Reinforcement) *Let H be a classification variable and let h^* be its hidden true value. A reinforcement r_i^H contributed by factor $\phi_i(H)$ is accurate in a particular situation iff*

$$h^* = r_i^H. \tag{6}$$

In other words, the true state of H is reinforced. We illustrate accurate reinforcements with an example. We assume binary variables H and E related through $\hat{P}(E|H)$ (i.e. a CPT) containing modeling parameters $\hat{P}(e_1|h_1) = 0.7$ and $\hat{P}(e_1|h_2) = 0.2$. Given these parameters and observation of $E = e_1$, the subsequent inference is based on the multiplication with factors $\phi_i(h_1) = \hat{P}(e_1|h_1)$ and $\phi_i(h_2) = \hat{P}(e_1|h_2)$, which yields reinforcement $r_i^H = h_1$. If h_1 is indeed the true value of H (i.e. the ground truth) then belief propagation through the network fragment corresponding to ϕ_i reinforces the true value and we consider the reinforcement accurate (see Definition 6). Consequently, we consider modeling parameters $\hat{P}(e_1|h_1)$ and $\hat{P}(e_1|h_2)$ adequate. Moreover, one can see that in this particular case we will obtain an accurate reinforcement as long as the parameters in $\hat{P}(E|H)$ satisfy condition $\hat{P}(e_1|h_1) > \hat{P}(e_1|h_2)$, which defines *intervals* for adequate parameter values.

If the true probability distribution $P(H)$ is a point mass distribution, then we can show an interesting property of the factors that satisfy this condition:

Proposition 1 *Let's assume a posterior distribution* $\hat{P}(H|\mathcal{E}) = \alpha \prod_i \phi_i(H)$. *If this factorization is expanded through a multiplication with a new factor* $\phi'(H)$ *that for a given instantiation* $\mathcal{E}'$ *satisfies condition (6), then the resulting posterior* $\hat{P}(H|\mathcal{E} \cup \mathcal{E}') = \alpha\phi'(H) \prod_i \phi_i(H)$ *satisfies the relation*

$$KL(P(H) \parallel \hat{P}(H|\mathcal{E})) > KL(P(H) \parallel \hat{P}(H|\mathcal{E} \cup \mathcal{E}')), \tag{7}$$

independently of the assumed modeling parameters corresponding to other factors. In other words, the estimated distribution approaches the true distribution $P(H)$.

Proof We write a ratio of posterior probabilities in order to get rid of the normalizing constants

$$\forall h_k \neq h^* : \frac{\hat{P}(h^*|\mathcal{E} \cup \mathcal{E}')}{\hat{P}(h_k|\mathcal{E} \cup \mathcal{E}')} = \frac{\phi'(h^*)}{\phi'(h_k)} \prod_i \frac{\phi_i(h^*)}{\phi_i(h_k)}. \tag{8}$$

If factor $\phi'(H)$ satisfies condition (6) we see that $\forall h_k \neq h^* : \frac{\phi'(h^*)}{\phi'(h_k)} > 1$. Consequently, $\frac{\hat{P}(h^*|\mathcal{E}\cup\mathcal{E}')}{\hat{P}(h_k|\mathcal{E}\cup\mathcal{E}')} > \frac{\hat{P}(h^*|\mathcal{E})}{\hat{P}(h_k|\mathcal{E})}$ for all $h_k \neq h^*$. Since we assume that the true distribution $P(H)$ is a point mass distribution, the ratio of true probabilities $\frac{P(h^*)}{P(h_k)} = \infty$ for all $h_k \neq h^*$. Thus, the posterior ratio (8) approaches the true ratio $\frac{P(h^*)}{P(h_k)}$ independently of the magnitude of $\frac{\phi'(h^*)}{\phi'(h_k)}$. In other words, the probability of the correct hypothesis is increased: i.e. $\hat{P}(h^*|\mathcal{E} \cup \mathcal{E}') > \hat{P}(h^*|\mathcal{E})$. Because $P(H)$ is a point mass distribution, the divergence is reduced to $KL(P(H) \parallel \hat{P}(H|\mathcal{E})) = P(h^*)log\frac{P(h^*)}{\hat{P}(h^*|\mathcal{E})}$ and relation (7) obviously holds true. □

In other words, proposition 1 implies that, in a given situation, an inference step results in correct belief updating for very different parameters as long as the parameters satisfy very simple relations.

4.2 True Distributions and Inference

A factor $\phi_i(H)$ is obtained by combining parameters from one or more CPTs from the corresponding network fragment $\mathcal{F}_i^H$. This combination depends on the evidence which in turn depends on the true distributions over modeled events. Thus, with a certain probability we encounter a situation in which factor $\phi_i(H)$ is adequate, i.e. it satisfies condition (6). We can show that this probability depends on the true distributions and simple relations between the true distributions and the CPT parameters.

We can facilitate further analysis, by using the concept of factor reinforcements to characterize the influence of a single CPT. For the sake of clarity, we focus on diagnostic inference only. For example, consider a CPT $\hat{P}(E|C)$ relating variables C and E. If we assume that one of the two variables was instantiated, we can compute a reinforcement at the other related variable. For the instantiation $E = e^*$, we would obtain a reinforcement $r_i^C = \arg\max_{c_j} \phi_i(c_j)$ at C, where $\phi_i(C) = \hat{P}(e^*|C)$. Thus, in such a case factors are identical to CPT parameters and an adequate CPT can be defined:

Definition 7 (Adequate CPT) *A CPT* $\hat{P}(E|C)$ *is adequate in a given situation if the following is true: if one variable were instantiated to the true state, then the belief propagation based on this CPT would reinforce the true state of the other variable. i.e. parameters in* $\hat{P}(E|C)$ *satisfy (6) for a given instantiation.*

In other words, an *adequate CPT supports accurate inference*. If all CPTs from $\mathcal{F}_i^H$ were adequate in a given situation, then also ϕ_i would be adequate. This is often not the case, however. Whether a factor ϕ_i is adequate depends on which CPTs from the corresponding fragment are

inadequate. Obviously, the higher is the probability that any CPT from a fragment $\mathcal{F}_i^H$ is adequate, the higher is the probability that $\mathcal{F}_i^H$ is adequate as well. In further discussion we express the probability that a CPT is adequate. For the sake of clarity, we focus on diagnostic inference only.

By taking a closer look at the relations between the true and modeled distributions over the events of interest, we can express the lower bound p_{re} for the probability that the parameters of a CPT relating two random variables C and E satisfy condition (6) in a particular situation. In order to facilitate further discussion, we first introduce sets of effects that can be characterized through relations between the true probabilities $P(e_k|c_i)$:

$$B^*_{c_i} = \{e_k|\forall c_j \neq c_i : P(e_k|c_i) > P(e_k|c_j)\}. \tag{9}$$

Each set $B^*_{c_i}$ contains the effects for which the likelihood of the cause c_i is greater than the likelihood of any other possible state of C. In other words, each $B^*_{c_i}$ describes a partition of the input space for which a classifier based on a single CPT $\hat{P}(E|C)$ that perfectly describes the true conditional distributions between C and E, i.e. $\hat{P}(E|C) = P(E|C)$, would be optimal [9] if the class prior distribution $P(C)$ were uniform.

Moreover, for each possible cause, i.e. a state c_i, we can express the probability P_{c_i} that an effect from the set $B^*_{c_i}$ will take place:

$$P_{c_i} = \sum_{e_j \in B^*_{c_i}} P(e_j|c_i), \tag{10}$$

P_{c_i} is the probability that given state $C = c_i$ a classifier based on $P(E|C)$, uniform $P(C)$ and classification threshold 0.5 will correctly classify a case.

In general, the modeling parameters $\hat{P}(E|C)$ will not be identical to the true distributions $P(E|C)$. Thus, the input space of a classifier using a single CPT $\hat{P}(E|C)$ and assuming uniform class prior distribution is described through the set $B_{c_i} = \{e_k|\forall c_j \neq c_i : \hat{P}(e_k|c_i) > \hat{P}(e_k|c_j)\}$ instead, where $\hat{P}(E|C) \neq P(E|C)$. Obviously, the probability of a correct classification with such a classifier is P_{c_i} if $B^*_{c_i} = B_{c_i}$. By considering this and the definitions of $B^*_{c_i}$ and B_{c_i} we can show the following property:

Proposition 2 *Let's assume the modeling parameters $\hat{P}(e_j|c_i)$, the true distribution $P(e_j|c_i)$ and the corresponding P_{c_i} for state $C = c_i$. Given any state c_i, P_{c_i} is also the probability that we will encounter a situation in which the modeling parameters $\hat{P}(e_j|c_i)$ support an accurate reinforcement (i.e. satisfy condition (6)) if the following relations are satisfied:*

$$\forall e_j : argmax_i \hat{P}(e_j|c_i) = argmax_i P(e_j|c_i). \tag{11}$$

In addition, with the help of probability P_{c_i}, we can define well distributed events as follows:

Definition 8 (Well Distributed Events) *If for all $c_i \in C$ relation $P_{c_i} > 0.5$ is satisfied, we consider events C and E* well distributed.

By considering this definition and proposition 2 we can derive the following important corollary:

Corollary 3 *Let's define the lower bound $p_{re} = \min_i(P_{c_i})$ on the probability, that we will encounter a situation in which a CPT relating C and E is such that the condition (6) is satisfied. $p_{re} > 0.5$ if the modeling parameters satisfy condition (11) and the events C and E are well distributed, i.e. $\forall c_i \in C : P_{c_i} > 0.5$.*

(a)

	c_1	c_2
e_1	0.7	0.4
e_2	0.2	0.3
e_3	0.1	0.3

(b)

	c_1	c_2
e_1	0.8	0.4
e_2	0.2	0.6

(c)

	c_1	c_2
e_1	0.7	0.6
e_2	0.3	0.4

Figure 6: Conditional Probability Tables (CPT).

We illustrate this with an example. We assume a simple model consisting of two nodes C and E which are related through a CPT $\hat{P}(E|C)$. In addition, we assume that we have a perfect model. This means that the modeling parameters $\hat{P}(E|C)$ are identical to the true distributions $P(E|C)$ given by the CPT in table 6.a.

Given that c_2 is the (hidden) true state of C (i.e. $c^* = c_2$), we would obtain an accurate factor reinforcement if we observed either e_2 or e_3. Thus, for state c_2 we obtain the following evidence set $B_{c_2} = \{e_2, e_3\}$. The probability P_{c_2} that either of these observations is caused by c_2 is $P_{c_2} = \mathcal{P}(e_2 \vee e_3|h_2) = 0.6$. Similarly, if c_1 were the true state of C, then observation of e_1 would result in a factor corresponding to an accurate reinforcement; i.e. $B_{c_1} = \{e_1\}$. The probability of observing e_1 given event c_1 is $P_{c_2} = \mathcal{P}(e_1|c_1) = 0.7$. Thus, whichever the true state of C, for this example we get $p_{re} > 0.6$. Note also, that in this example the events were well distributed and $p_{re} > 0.6$ for many different parameters as long as condition (11) is satisfied.

However, we can encounter also cases where the relations between the events are such that the lower bound for p_{re} is less than 0.5. For example, assume random variables C and E which are related by a CPT, whose parameters are identical to the true distribution depicted in table 6.c (i.e. we have a perfect model). By considering Definition 6 we see that in the case that c_2 occurs we will in 40% of the cases observe e_2, which will yield a correct reinforcement. In other words, there exist domains in which the true distribution over modeled events is such that the expected classification performance can be very poor, even if we had perfect models. If we consider binary variables, we see that this is the case if for both possible causes the true probability of getting the same effect is greater than 0.5.

Note that proposition 2 and corollary 3 imply $p_{re} > 0.5$ if (i) the events are well-distributed and (ii) coarse relations (11) between the modeling parameters and true probability distributions are satisfied. Since relations (11) are very simple, it is plausible to assume that in many domains designers or machine learning algorithms can specify BNs where for most CPTs $p_{re} > 0.5$. This is especially the case if we use variables with few states. For example, assume the true distribution in table 6.b. If we specified modeling parameters such that $\hat{P}(e_1|c_1) > 0.5$ and $\hat{P}(e_1|c_2) < 0.5$, then we would obtain a modeling component (i.e. a CPT) for which $p_{re} = 0.6$.

As we show later, the estimation process can be very robust if we can assume that $p_{re} > 0.5$ for every CPT in a BN with poly-tree topology featuring sufficiently large branching factors.

5 Inference Meta Model

By considering the properties of inference processes in BNs discussed in the previous section, we introduce an *Inference Meta Model* (IMM), which describes inference processes from a coarse perspective. IMM facilitates analysis of inference processes and represents a basis for theoretically rigorous approaches to the monitoring of information fusion quality. IMM is a collection of definitions and theorems that capture relevant aspects of inference with BNs:

1. Causes of inference faults and their impact on the fusion quality (Section 5.1).

2. Asymptotic properties of updating tendencies with respect to the fusion robustness (Section 5.2).

5.1 Inference Faults

While a single factor satisfying condition (6) does not guarantee accurate fusion results, we see that factors which do not satisfy this condition inevitably increase the KL divergence between the true and the estimated distributions. In other words, factors that do not satisfy (6) are in a given situation inadequate and result in erroneous belief updates. We can identify several causes of such inadequacies.

Definition 9 (Fault type 1) *BNs are generalizations of large sets of training examples corresponding to different relevant situations from the target domains. Consequently, even though a model might be accurate in the sense that it describes the generalized world perfectly, in a rare situation it might not support correct mapping between the instantiated evidence node and the hypothesis, i.e. condition (6) is not satisfied. In other words, such a model is inadequate in the current situation and causes faulty inference. Consider the previous example based on the true probability $P(E|C)$ shown in table 6.a. If we had perfect modeling parameters, i.e. $\hat{P}(E|C) = P(E|C)$, then the inference step based on the CPT $\hat{P}(E|C)$ would violate condition (6) and decrease the belief in the true state (e.g. $C = c_1$), if the world were in the (rare) state $\{C = c_1, E = e_2\}$.*

Definition 10 (Fault type 2) *The generalization can inaccurately capture relations between the conditional probabilities, such that relation (11) is not satisfied. Such modeling faults increase the chance that the model is inadequate in a particular situation.*

Definition 11 (Fault type 3) *Information providers can be erroneous.*

5.2 A Coarse Perspective on Inference

Condition (6) relates belief updating tendencies to accuracy, and ignores the magnitude of the updates. We extend the notion of updating tendencies in order to translate inference processes to a perspective where belief updating is reduced to counting of state updates, which facilitates the analysis of the fusion processes.

As next, we define for each state x_i of a random variable X a reinforcement counter:

Definition 12 (Reinforcement Counter) *Reinforcement counter n_i^X of state x_i of variable X is defined as follows:*

$$n_i^X = ||\{r_j^X | \forall r_j^X : x_i = r_j^X\}||,$$

where $||\cdot||$ denotes the cardinality of a set and r_j^X denotes the reinforcement from the j-th fragment rooted in X. In other words, n_i^X counts reinforcements of the state x_i. Moreover, each variable X with m states is associated with a set of reinforcement counters $\mathcal{N}^X = \{n_1, \ldots, n_m\}$.

We can further simplify the inference process by assuming that the state x_i with the greatest reinforcement counter n_i^X is the true state. The assumed true state is captured by the *Reinforcement Summary*:

Definition 13 (Reinforcement Summary) *We define a* reinforcement summary s^X *of node X as follows:*

$$s^X = x_{i_{max}}, \tag{12}$$

where $i_{max} = \arg\max_i n_i^X$ is the index of the state of X that is associated with the greatest reinforcement counter. In other words, Reinforcement Summary *is the state associated with the greatest reinforcement counter $n_{max}^X = \max_i(n_i^X)$. If X is a leaf node then s^X is an instantiation of X.*

Reinforcement summary s^X represents the state of node X that best 'explains' the current set of observations, when ignoring the belief magnitudes. We can illustrate these definitions with an example. Let's assume three independent fragments relative to variable H and a sequence of factor reinforcements $r_1^H = h_1$, $r_2^H = h_1$ and $r_3^H = h_2$. If node H had 3 states, then we would obtain counters $\mathcal{N}^H = \{2, 1, 0\}$ and the reinforcement summary $s^H = h_1$.

5.3 Reinforcement Counter Distributions

By considering reinforcement counters we can reduce belief updating to a counting problem. We can show that under certain conditions there exist probability distributions over combinations of reinforcement counters, which suggests that the reinforcement counters can be used for a coarse grained belief propagation as well as a robust analysis of the inference processes.

If we define the lower bound $p_{\mathcal{F}}^X$ for the probability that any diagnostic network fragment $\mathcal{F}_i^X$ rooted in variable H supports an accurate reinforcement in a particular situation, then we can show the following:

Proposition 4 *Given a classification node H and an odd number of independent network fragments k for which $p_{\mathcal{F}}^H > 0.5$, the probability p_s^H that the true state corresponds to the maximum reinforcement counter n_{max}^H is greater than* 0.5.

Proof We can express the lower bound for the probability p_s^H that we will encounter a situation in which s^H, the state of H associated with the maximum reinforcement counter, is indeed the hidden true state of H:

$$\bar{p}_s^H = \sum_{m=\lceil k/2 \rceil}^{k} \binom{k}{m} (p_{\mathcal{F}}^H)^m (1 - p_{\mathcal{F}}^H)^{k-m}, \tag{13}$$

where m is the number of accurate reinforcements (see Definition 6). In the case of binary nodes $n_{max}^H = m \geq \lceil k/2 \rceil$. By considering properties of binomial distributions, $m \geq \lceil k/2 \rceil$ and the lower bound $p_{\mathcal{F}}^H > 0.5$, we know that the following relations must hold for odd numbers of network fragments k

$$p_s^H \geq \bar{p}_s^H \geq p_{\mathcal{F}}^H > 0.5. \tag{14}$$

Also, by considering the binomial distribution captured by (13), we can easily show that for odd k's which are greater than 1, a more restrictive relation holds true: $p_s^H \geq \bar{p}_s^H > p_{\mathcal{F}}^H > 0.5$. □

Moreover, if we use multi-value variables, then $\bar{p}_s^H$ is the probability of encountering a *subset* of situations in which the maximum counter is associated with the true state. Thus, if we use (13) for networks with multi-value nodes, we impose a conservative requirement that $n_{max}^H = m \geq \lceil k/2 \rceil$. In other words,if we use multi state variables, the probability $\bar{p}_s^H$ is even greater than in the binary cases. In addition, we can easily identify the following properties:

Corollary 5 *Given a BN with classification node H and a sufficiently great number of network fragments k for which $p_{\mathcal{F}}^H > 0.5$, then probability $p_s^H \geq \bar{p}_s^H > 0.5$ also for* even *numbers of network fragments k. In addition, $lim_{k \to \infty} p_s^H = 1$.*

5.4 Robust Inference

Modeling parameters determined by designers or learning algorithms usually do not describe the true probability distributions precisely. On the other hand, classification with a BN in a particular situation is correct if the designers or the learning algorithms choose CPT parameters from a parameter domain such that the correct classification threshold is exceeded; i.e. the KL divergence between the estimated and the true probability distribution is sufficiently small. Clearly, the greater the domain from which we can choose adequate parameters for any situation, the greater is the chance that the inference will be accurate. In this section we show that the robustness depends on the modeling topology.

With the help of the reinforcement counters, we can explain under which circumstances inference processes in BNs will be robust with respect to the variation of parameters. While the classification accuracy depends on the actual combination of observations and parameter values, we can show that the robustness improves with increasing numbers of factors ϕ_i if they satisfy very weak conditions. We can identify the following porperty:

Proposition 6 *The domain of parameters for which the correct classification threshold is exceeded in a particular situation grows with the increasing number of conditionally independent fragments $\mathcal{F}_i^H$ that support accurate factor reinforcements r_i^H in more than 50% of possible situations. Thus, the more independent fragments are introduced, the greater variety of parameters will support correct classification.*

Proof We can explain this property by using reinforcement counters. Let's first assume a δ such that estimated distribution $\hat{P}(H|\mathcal{E})$ over node H will result in a correct classification if the Kullback-Leibler divergence satisfies $KL(P(H) \parallel \hat{P}(H|\mathcal{E})) < \delta$, where $P(H)$ is the true distribution. Note that $P(H)$ is a point mass distribution, reflecting the fact that exactly one of the possible states of H materialized. For the classification variable we also define n_a^H, the number of factors contributing accurate reinforcements (see definition 6) and n_e^H, the number of factors introducing inaccurate reinforcements. Thus, the total number of independent factors is $n_a^H + n_e^H$.

We also assume that the posterior probability $\hat{P}(H|E)$ remains approximately constant if the numbers of accurate and inaccurate reinforcements are the same, i.e. $n_a^H = n_e^H$. This is the case if for every factor ϕ_i corresponding to an inaccurate reinforcement we can find a factor ϕ_j corresponding to an accurate reinforcement such that the factors cancel out after the normalization.

According to proposition 1 we see that $n_a^H - n_e^H$ accurate reinforcements will reduce $KL(P(H) \parallel \hat{P}(H|E))$, irrespectively of the magnitude of the corresponding factors ϕ_i. However, for a given δ, these factors must be large enough to satisfy $KL(P(H) \parallel \hat{P}(H|E)) < \delta$ in $n_a^H - n_e^H$ steps.

If the number of fragments is increased we obtain new reinforcement counters $n_{a'}^H$ and $n_{e'}^H$. According to proposition 4 it is more likely that $(n_{a'}^H - n_{e'}^H) > (n_a^H - n_e^H)$ than $(n_{a'}^H - n_{e'}^H) < (n_a^H - n_e^H)$ if $p_{\mathcal{F}}^H > 0.5$ for every network fragment. Moreover, according to corollary 5, this probability grows with increasing number of fragments.

In other words, by increasing the number of factors, the absolute difference $n_a^H - n_e^H$ grows as well. Consequently, we can use a greater variety of factor values in order to satisfy $KL(P(H) \parallel \hat{P}(H|E)) < \delta$; due to larger $n_a^H - n_e^H$ we can satisfy $KL(P(H) \parallel \hat{P}(H|E)) < \delta$ also with factors that result in smaller belief updates, i.e. smaller changes of the estimate $\hat{P}(H|E)$. This means that the intervals from which we can choose adequate parameters grow, which increases the likelihood of having factors ϕ_i that will in a given situation support accurate classification. Thus, *increasing the number of factors in $\hat{P}(H|E)$ that satisfy $p_{\mathcal{F}}^H > 0.5$ improves the robustness*, since the classification is less sensitive to the parameter precision and the impact of inference

faults of type 1 can effectively be mitigated. Properties captured by this proposition are reflected in the examples from section 3.3. □

5.5 Reinforcement Propagation

Reinforcement counters support a coarse grained approach to belief propagation in BNs that feature many conditionally independent network fragments.

For the sake of clarity, we consider only reinforcements contributed through factors ϕ_i implementing diagnostic inference steps (see section 3). This simplification is justified for BNs with poly-tree topologies with significant branching factors. In such a network, the predictive network fragment has an insignificant influence on the posterior over the classification variable H. Consequently, we assume that the state estimation is based on a BN with a simple tree topology where H corresponds to the root node.

Let's assume a set $\mathcal{V}_L$ consisting of leaf nodes that were instantiated through hard evidence. Moreover, we find a set $\mathcal{P}_L = \{\pi(X)|X \in \mathcal{V}_L\}$ consisting of all parents of nodes from set $\mathcal{V}_L$. For each parent $Y \in \mathcal{P}_L$ we determine the reinforcement summary s^Y resulting from the propagation from its children contained in the set $\sigma(Y) \subseteq \mathcal{V}_L$. Every parent node is then instantiated as if the state returned by s^Y were observed. If nodes from $\mathcal{P}_L$ have parents, we set $\mathcal{V}_L \leftarrow \mathcal{P}_L$ and find a new set of parents and the procedure is repeated until the reinforcement summary is determined at the root node H. This procedure can be summarized by the following algorithm:

Algorithm 1: Reinforcement Propagation Algorithm

1 Collect all instantiated leaf nodes in the set $\mathcal{V}_L$;
2 Find $\mathcal{P}_L$, a set of all parents of the nodes in $\mathcal{V}_L$;
3 **if** $\mathcal{P}_L \neq \oslash$ **then**
4 **for** *each node* $Y \in \mathcal{P}_L$ **do**
5 find set $\sigma(Y)$ of all children of Y;
6 **for** *each node* $X_i \in \sigma(Y)$ **do**
7 Compute reinforcement r_i^Y at node Y resulting from the instantiation of X_i;
8 **end**
9 Compute reinforcement summary s^Y at node Y;
10 Instantiate node Y as if s^Y were observed (hard evidence);
11 **end**
12 Make instantiated parent nodes from $\mathcal{P}_L$ elements of $\mathcal{V}_L$: $\mathcal{V}_L \leftarrow \mathcal{P}_L$;
13 Go to step 2;
14 **else**
15 stop;
16 **end**

By running this algorithm, we obtain s^X for all unobserved variables. Moreover, we can show that this algorithm has an interesting property if the probability $p_{re} > 0.5$ for all CPTs in a BN with tree topology:

Proposition 7 *Given a BN with binary nodes and odd branching factors k, for every non-terminal node X the probability* p_s^X *that the true state corresponds to the maximum reinforcement counter* n_{max}^H *is greater than* 0.5.

Proof We can show this by considering proposition 4. Namely, in BNs with tree topologies each child node of some node X corresponds to a fragment $\mathcal{F}_i^X$ relative to X. In other words, the branching factor at node X is identical to the number of diagnostic fragments rooted in X.

We can use (13) to compute $\bar{p}_s^X$ for every node X except the leaves. In addition, tree topologies facilitate the formulation of the probability $p_{\mathcal{F}}^X$; i.e. the lower bound that any of the network fragments rooted in X supports correct mapping in a given situation. If we compute $\bar{p}_s^X$ for the parents of leaf nodes (i.e. terminal nodes), we simply set $p_{\mathcal{F}}^X$ equal to p_{re}^*, corresponding to a child node of X associated with the minimum p_{re}. However, for the hidden nodes with non-terminal children we must consider the fact that according to the presented algorithm the instantiation of every child is based on its reinforcement summary. In other words, in contrast to the leaf nodes, the instantiations of non-terminal children can be erroneous. Therefore, we formulate $p_{\mathcal{F}}^X$ at node X as a function of p_{re}^* and $\bar{p}_s^X$:

$$p_{\mathcal{F}}^X = \bar{p}_s^Y \; p_{re}^* + \alpha(1 - \bar{p}_s^Y)(1 - p_{re}^*), \tag{15}$$

where $\bar{p}_s^Y$ is the lower bound on the probability that summary s^Y at X's child node Y is accurate in a given situation. The second term represents the probability of encountering a situation where the reinforcement would be inaccurate given a correct state of Y, but we chose an incorrect state y_i such that the errors cancel out and the true state of H gets reinforced. Note that (15) is valid also for the leaf nodes, where we set $\bar{p}_s^Y = 1$, since the observations are deterministic. By recursively applying operations (15) and (13) and starting with the leaf nodes, we can compute $\bar{p}_s^X$ for all non-terminal nodes in a tree.

If we have binary variables, then $\alpha = 1$ and we can easily show that for any $p_{re}^* > 0.5$ the probability $p_{\mathcal{F}}^X > 0.5$ at any non-terminal node. Consequently, according to proposition 4, we see that in a BN with binary variables featuring a tree topology and any odd branching factor, the probability $p_s^X > 0.5$ for any non-terminal node X, since $p_s^H \geq \bar{p}_s^H \geq p_{\mathcal{F}}^H > 0.5$. □

By considering the properties of binomial distributions we can derive the following corollary:

Corollary 8 *Given a BN with a tree topology, $p_{re} > 0.5$ for all CPTs and sufficiently great branching factor k, then the probability $p_s^X > 0.5$ also in the case of multi-state nodes and* even *branching factors k. Moreover, $lim_{k\to\infty} p_s^X = 1$ for every node X and for each fragment $\mathcal{F}_i^X$, $lim_{k\to\infty} p_{\mathcal{F}_i}^X = p_{re}$, where p_{re} corresponds to the CPT connecting X and the rest of fragment $\mathcal{F}_i^X$.*

In other words, the distribution over constellations of reinforcement counters at any variable is a function of p_{re}^* and the branching factors k, which is reflected in the experimental results shown in section 6.2.

Note that if we use networks with binary nodes then the presented approach to the propagation of reinforcements can be viewed as a hierarchical system of decoders (see for example figure 7) that implement repetition coding technique known from the information theory [18]. Determination of the reinforcement summary s^X and its use for the instantiation at node X corresponds to the majority voting. Probability $1 - p_{re}$, on the other hand, corresponds to the failure ratio over a binary noisy channel. It is well known that by using the repetition coding technique the decoding accuracy at the receiver asymptotically approaches the true point-mass distribution corresponding to the source state.

6 Applications

The presented IMM supports analysis and techniques that are relevant for the development of robust fusion systems for real world applications. In particular, by considering IMM we can derive design rules that support building of fusion systems that can cope with imprecise models and uncertain observations. In addition, IMM provides a theory that justifies simple yet effective runtime analysis of fusion processes.

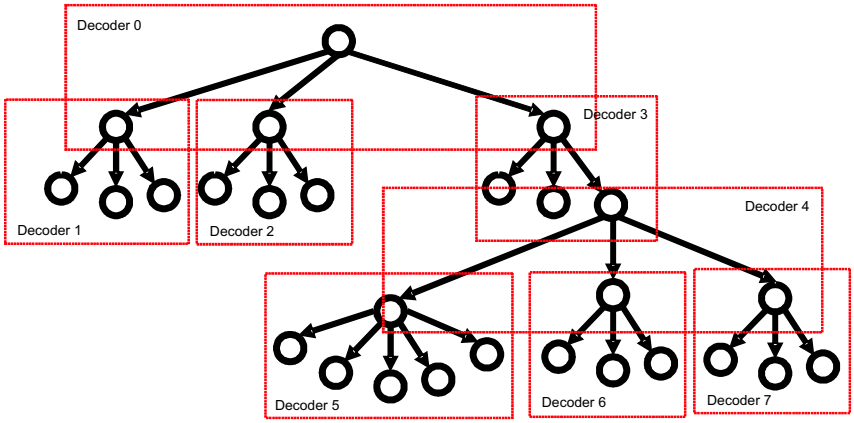

Figure 7: A hierarchy of decoders.

6.1 Design of Robust Inference Systems

The IMM provides a guidance for the design of inherently robust fusion systems if we can specify CPTs such that the corresponding $p_{re} > 0.5$. In section 4.2 we showed that this assumption is realistic. Namely, $p_{re} > 0.5$ if the modeling parameters capture very coarse relations between the true probabilities, i.e. we avoid faults of type 2, and events are not ill distributed. Since these relations are very coarse, we can assume that in many cases they can easily be identified by designers or machine learning algorithms.

Given assumption $p_{re} > 0.5$, proposition 6 implies that the inherent robustness of the state estimation with BNs improves through addition of conditionally independent network fragments, given some hypothesis variable H. As we build a fusion system we must specify modeling fragments for each information source. For many information sources, as for example sensors, we can assume that they are conditionally independent. Consequently, by adding new sources to the system we increase the number of fragments at different levels of the BN that supports a meaningful fusion. Such a BN basically describes how observed events can cause different states of H, which in turn can cause further observations.

For example, in the model from figure 1, we could consider nodes D and E as two observations from different types of sensors. These two observations were caused by some phenomenon corresponding to some state of H. On the other hand, the same phenomenon will cause a hidden event F with a certain probability, while F will result in three observations represented by nodes K, L and M. In this way we introduced three diagnostic fragments relative to H which results in factorization (2). Obviously, if we had more sensors that can detect event H, we could introduce more conditionally independent factors.

If the observation types D and E were obtained with sensors that can detect event A instead, we would obtain a different topology depicted in figure 8. The corresponding factorization would contain less factors. Namely, by incorporating such information sources we introduce new nodes to the predictive fragment. Consequently, the number of fragments relative to H would not increase by adding new information sources. For example for the evidence set $\mathcal{E} = \{b_1, d_2, e_1, k_2, l_1, m_1\}$ we would obtain the following factorization:

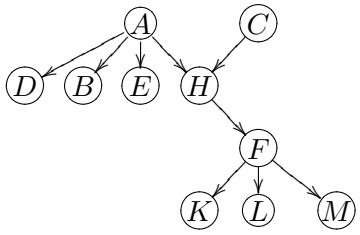

Figure 8: A causal model relating hypotheses represented by node H and different types of observations captured by nodes B, D, E, K, L and M.

$$\hat{P}(h_i,\mathcal{E}) = \overbrace{\sum_A \hat{P}(A)\hat{P}(b_1|A)P(d_2|A)P(e_1|A)\sum_C \hat{P}(C)\hat{P}(h_i|A,C)}^{\phi_0(h_i)} \cdot \underbrace{\sum_F \hat{P}(F|H)\hat{P}(k_2|F)\hat{P}(l_1|F)\hat{P}(m_1|F)}_{\phi_1(h_i)} \tag{16}$$

In this case the incorporation of new information sources did not increase the number of independent factors relative to H and we cannot exploit the property captured by proposition 6. Such a network requires precise parameters $\hat{P}(H|A,C)$ relating variables A, C and H and it cannot compensate faults of type 1, which requires factor redundancy.

This example illustrates that, given limited resources, we should incorporate information sources that increase the number of conditionally independent factors of $\hat{P}(H|\mathcal{E})$, which improves inherent fusion robustness as well as the effectiveness of different fusion monitoring approaches (see the following sections). In other words, IMM can be considered in mission critical sensor management tasks. Namely, often situation assessment requires different types of information. However, due to temporal limitations and scarce resources, it might be difficult or impossible to gather and process all possible observations in a given time frame. For example, in a gas disaster scenario the decision makers would require heterogeneous information from mobile labs that is obtained through more or less advanced measurements. However, the measurements cannot be made simultaneously and they take time. Consequently, due to the time pressure and a small number of mobile labs, it makes sense to first make the measurements that will contribute to reliable assessment the most. If the relations between the measurements and hidden events of interest H can be described through a BN, then we can make first the measurements corresponding to diagnostic fragments relative to H. In this way we would optimize the sensing process with respect to the robustness.

6.2 Coping with Imprecise Models by Using an Alternative Belief Propagation Method

Proposition 7 and corollary 8 suggest that we can use reinforcement propagation algorithm as a coarse alternative to usual belief propagation. In this way we can achieve very robust diagnostic inference with asymptotic properties if a few coarse assumptions are satisfied, even if we use imprecise models and noisy evidence. This is achieved if we use networks with many conditionally independent fragments (see definition 2) and $p_{re} > 0.5$, which is reflected in the experimental results shown in figure 9. The curves show how p_s^X changes as a function of p_{re} at a root node in simple tree networks with four levels and different branching factors. In this experiment, the

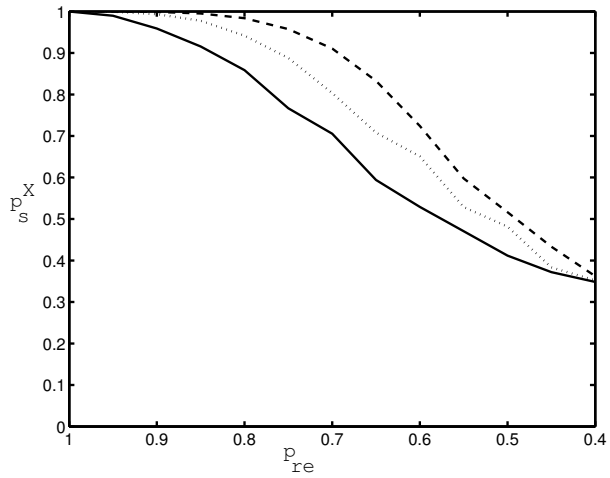

Figure 9: p_s^X, the probability that the maximum counter is associated with the correct state, as a function of p_{re} and branching factors k: 3 (solid), 5 (dotted) and 7 (dashed).

variables had three states. From the depicted curves it is apparent that for sufficient branching factors the classification accuracy can be very high, despite significant modeling uncertainties and noisy evidence. Note for example a high classification accuracy for $p_{re} = 0.7$ and branching factors 7. The same accuracy can be achieved with very different CPT parameters as long as they capture simple relations (see section 4.2).

Such a simplified belief propagation approach is very relevant for many real world applications where states of hidden variables are inferred through interpretation (i.e. fusion) of information obtained from large numbers of different sources, such as sensors and humans. While in such settings it is usually very difficult to obtain precise descriptions of the true probability distributions, the domains can be adequately captured through tree like BNs with high branching factors. In such applications each information source is associated with a conditionally independent fragment given the monitored phenomenon (see section 7.1).

On the other hand, we can often make the realistic assumption that the learning algorithms or human experts can identify simple relations between the probabilities in the true distributions (see section 4.2) and specify CPT parameters such that $p_{re} > 0.5$. For example, assume the true conditional probabilities over variables E and C in 6.b. The corresponding $p_{re} = 0.6$ as long as the modeled probabilities satisfy relations (11). Thus, the CPT contributes to accurate classification since it will be adequate in more than 50% of cases; e.g. given observation $E = e_1$, the CPT $\hat{P}(E|C)$ will support correct reinforcement at C as long as the used parameters satisfy relations $\hat{P}(e_1|c_1) > 0.5$ and $\hat{P}(e_1|c_2) < 0.5$.

6.3 Runtime Analysis of the Inference Quality

Even if we had a perfect causal model BN, precisely describing the true distributions in the domain, we could encounter situations in which this BN would not support accurate state classification for the given observations. Namely, the CPTs in such a model are generalizations, which do not support correct mapping of an observation if it is a rare case. In other words, in stochastic domains we always have to deal with the faults of type 1 (see definition 9). There are situations where a significant portion of observations corresponds to rare cases, which can result in erroneous estimation; i.e. an incorrect state of hypothesis variable H will be associated with the greatest posterior probability.

However, if we can assume that $p_{re} > 0.5$, we can show that in a given situation the maximum counter n_{max}^H corresponds to the true state of H with a certain probability that grows with

the number of network fragments that incremented n^H_{max}. We can exploit these properties and consider a reinforcement counter as additional information on the estimation quality. Namely, we consider a fusion result $\hat{P}(H|\mathcal{E})$ potentially misleading, if the state with the maximum posterior probability is not associated with a reinforcement counter that exceeds some threshold τ. This simple approach is implemented by the filtering Algorithm 2 that provides a flagging mechanism which can reduce the chance of using misleading fusion results.

Algorithm 2: Filtering Algorithm

```
Inputs: BN, evidence E;
Compute P̂(H|E) over hypothesis variable H;
Compute set of reinforcement counters N^H = {n^H_1, ..., n^H_m};
Determine the index of the state with the maximum posterior probability:
  i_max = argmax_i P̂(h_i|E);
if n^H_{i_max} < τ then
    Consider P̂(H|E) misleading → activate the flag for a potentially misleading result;
end
```

Proposition 4 and corollary 5 suggest that this algorithm is effective if it is used with BNs featuring many independent fragments (see definition 2) and CPTs where $p_{re} > 0.5$. For example, if $\tau = \lceil k/2 \rceil$, where k is the number of fragments, the probability of encountering a case in which the result will correctly be classified as reliable or misleading is greater than 0.5. This probability approaches 1 as k grows. By increasing τ we improve effectiveness of removing false positives (i.e. critical events classified as non-critical) while, the portion of true positives considered as misleading grows.

These properties are reflected in experiments. Data was sampled from a generative model, a BN with a tree topology, 4 levels and branching factor 7. Each sampled set of observations was fed to a classification network which was identical to the generative model; thus we used a perfect model for the classification. Random sampling occasionally resulted in constellations of observations corresponding to rare situations. For the sake of simplicity all CPTs were associated with the same p_{re}. Diagrams in figure 10 show the impact of the presented filtering algorithm on the percentage of true/false positives. The horizontal axis represents threshold τ while the vertical axis represents the percentage of true positives out of all positives. In other words, the vertical distance between a curve and the horizontal line at 100% represents false positives, which should be reduced. The curves show that the filtering effectiveness is a function of p_{re} and threshold τ. For example, the curve with diamonds corresponds to $p_{re} = 0.7$. By using $\tau = 4$ the percentage of false positives was reduced by approximately $\Delta F_p \approx 70\%$, while 176 out of 1000 cases were considered potentially misleading. In this experiment the prior probability of the critical state was 0.33. In general, the effectivness of this algorithm grows with p_{re}, which is reflected in figure 10. The effectiveness grows also with the number of independent factors, which corresponds to growing branching factors in tree-like topologies.

Such filtering technique is useful in applications where a failure to detect critical events (i.e. false positives) could have devastating consequences while reacting to false alarms is less costly. For example, a fire in a remotely observed section of a chemical plant can be detected with relatively unreliable sensors. A failure to detect the fire could result in a catastrophe while an activation of a flag indicating a potentially misleading state estimation could prompt the operator to zoom in with a camera or send a remotely controlled robot. In this case, the damage caused by a false positive outweighs the use of an alternative, more expensive mode of observation involving manual work or scarce resources.

Algorithm 2 implements a coarse rejection mechanism, which is relevant for the domains

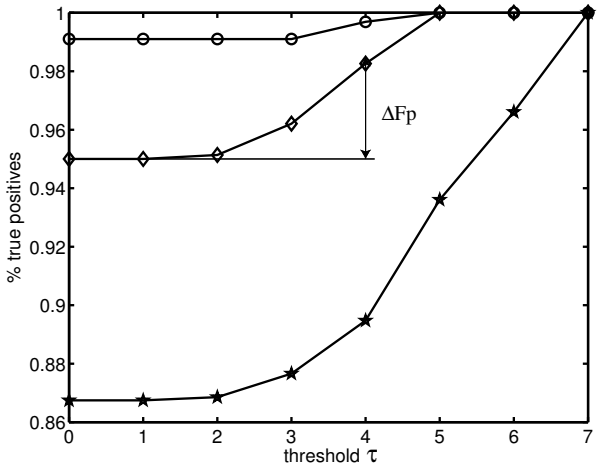

Figure 10: True positives as a function of different thresholds τ. Stars, diamonds and circles correspond to $p_{re} = 0.6$, $p_{re} = 0.7$ and $p_{re} = 0.8$, respectively.

where the available data quantities do not allow reliable determination of the optimal classification/rejection thresholds based on the ROC curves [9].

7 Discussion

We introduced IMM which describes information fusion in BNs from a coarse, runtime perspective. We emphasize the difference between the generalization accuracy and the fusion (i.e. classification) accuracy. Fusion is based on models that are generalizations over many possible situations. Consequently, even if we used a perfect model, in a rare situation, a particular set of observations could result in erroneous classification. From the user point of view, the classification accuracy is more important than the generalization accuracy.

IMM exposes important properties of BNs that are relevant for the construction of inherently robust information fusion systems. IMM is based on very coarse and plausible assumptions (see section 4.2). With the help of the IMM we show that inference in BNs can be very insensitive to the parameter values and can have asymptotic properties with respect to the classification accuracy. In this way we can, in certain cases, relax the problem of obtaining appropriate modeling parameters [7]. This means that the fusion can be very robust, which is especially relevant in the domains where it is difficult to obtain precise models due to the lack of sufficient training data or expertise. Also, the implications of the IMM agree to a great extent with the experimental results reported in [11].

In addition, IMM introduces a reinforcement propagation algorithm that can be used as an alternative to the common approaches to inference in BNs and supports detection of potentially misleading fusion results.

7.1 Causal Models with Simple Topologies

In this paper we proposed methods that assume BNs with great numbers of conditionally independent network fragments (see definition 2) which implies that:

- The presented IMM is limited to BNs with relatively simple topologies.
- Huge quantities of evidence must be propagated through large networks.

In this context, we should ask (i) whether such limited topologies still support modeling of real world domains and (ii) whether the required computing and communication capacities are feasible.

It turns out that simple topologies are relevant for a significant class of modern situation assessment and controlling applications that require processing of large amounts of heterogeneous information that can be accessed through the existing sensing and communication infrastructure.

We illustrate this with the help of a simple example. Let's assume a fire detection system that makes use of different types of sensors, such as smoke detectors, thermometers and cameras detecting flames. Each sensor measures a particular physical quantity. For example, smoke detectors could measure conductivity of the ionized air in their vicinity. Electronic circuits evaluate air conductivity and generate streams of sensor reports. Moreover, the j-th report from the i-th sensor is represented by a random variable R^i_j. A report signaling the presence or absence of some phenomenon, such as smoke, is characterized by $R^i_j = true$ or $R^i_j = false$, respectively. The smoke concentration exceeding some threshold for a certain period of time (i.e. a time slice) would cause a change of the air conductivity, such that the electronic circuit would measure a high conductivity within that time slice. In such a case the sensor is considered to work properly if $P(R^i_j = true)$, the probability that a report from this sensor will indicate the presence of smoke, is greater than $P(R^i_j = false)$, the probability that a report will signal the absence of smoke, i.e. $P(R^i_j = true) > P(R^i_j = false)$. If the smoke were absent, a sensor that works correctly would generate reports for which the distribution $P(R^i_j)$ would satisfy relation $P(R^i_j = true) < P(R^i_j = false)$. However, the probability distribution $P(R^i_j)$ does not depend only on the presence of the phenomenon that we are trying to infer. For example, the sensor electronics might fail, the sensor could be covered by ice, low temperatures could influence the air conductivity, etc. In other words, the distribution $P(R^i_j)$ over reports depends on many different factors which are often not well known.

In order to avoid detailed modeling of the processes resulting in sensor reports, we introduce the *sensor propensity* [3] concept that represents two types (classes) of situations characterized by combinations of the states of the electronic components and the states of the sensor's immediate environment (e.g. conductivity of the ionized air). We represent the *sensor propensity* by a binary variable S. $S = true$ denotes the class of state combinations which influence the sensing process in such a way that the probability of obtaining a sensor report confirming some phenomenon is greater than 0.5; i.e. $P(R^i_j = true|S = true) > P(R^i_j = false|S = true)$. On the other hand, the class of situations corresponding to $S = false$ would influence the distribution over sensor reports $P(R|S)$, such that $P(R^i_j = true|S = false) < P(R^i_j = false|S = false)$. In other words, the sensor propensity denotes the sensor's tendency of producing observation sequences in which the majority of reports indicates either the presence or the absence of some phenomenon.

By representing sensor propensity through a binary node we can use simple causal models to describe the relations between the hidden phenomenon, sensor propensity and the observation sequences. For example, for smoke detectors such relations are captured by highlighted nodes in figure 11, where node *Smoke* represents the presence/absence of smoke, binary nodes S_i represent propensity of the i-th sensor, while nodes R^i_j correspond to sensor reports obtained from the i-th sensor during a particular time slice.

Clearly, it can be very difficult to obtain parameters for the CPTs $\hat{P}(S|Smoke)$ and $\hat{P}(R^i_j|S)$ that would precisely describe the true distributions over the states of propensity S_i and existence of $Fire$. However, according to corollary 3 in section 4.2, we know that a network fragment $\mathcal{F}^H_i$ is likely to contribute to correct estimation if the CPTs corresponding to nodes of $\mathcal{F}^H_i$ capture very simple relations between the true conditional probabilities. In this context a smoke

[3] observation tendency

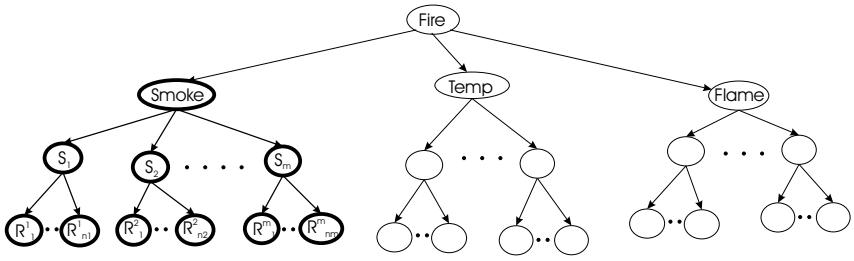

Figure 11: A causal model capturing relations between the hidden phenomenon $Fire$ and different types of sensor observations. Highlighted portion of the graph captures causal relations between $Smoke$, sensor propensities S_i and sensor reports R^i_j.

detector is useful for the state estimation if the corresponding modeling parameters $\hat{P}(R^i_j|S)$ and $\hat{P}(S|Smoke)$ satisfy condition (11) and the true distributions are well distributed thus simultaneously satisfying relations: $P(R^i_j = true|S = false) < P(R^i_j = false|S = false)$, $P(R^i_j = true|S = true) > P(R^i_j = false|S = true)$, $P(S = true|Smoke = true) > P(S = true|Smoke = false)$ and $P(S = false|Smoke = true) < P(S = false|Smoke = false)$. In other words, a modeling fragment provides a useful estimation expertise without precisely describing the true distributions. A fragment is useful if merely very simple relations between the true distributions are satisfied and identified by the designer or a learning algorithm.

With each sensor we introduce an independent partial causal process which is initiated through some hidden phenomenon. For example, by introducing a new smoke detector the presence of smoke will initiate different processes in the sensor's circuitry which will eventually produce sensor reports. We can assume that such a sensor does not influence its environment and other sensors; a smoke detector does not influence the smoke concentration or any other parameters of the environment that could in turn influence other sensors.

Such independencies are reflected in the corresponding causal models. Namely, for each sensor we expand a BN by introducing a propensity node and the corresponding sensor report nodes. For example, in figure 11 the m^{th} smoke detector is associated with a propensity node S_m and report nodes R^m_j. Nodes $Smoke$, S_m and R^m_j represent a network fragment $\mathcal{F}^{Smoke}_m$, which is conditionally independent of other network components given node $Smoke$. Obviously, by using many information sources reporting about heterogeneous phenomena we can obtain BN topologies with many independent fragments.

Moreover, we assume domains where hidden states of the environment do not change during a certain observation time interval, i.e. a single time slice. For example, fire is either present or absent throughout an entire time slice. We call such phenomena quasi-static. On the other hand, within a finite time slice, we obtain sequences of observations which are influenced by the hidden quasi-static phenomena. Such observations are represented through leaf nodes. In contrast to hidden variables representing quasi-static phenomena, each observation corresponds to a particular time instant. Also, for a significant class of sensors, we can assume that sequences of observations result from first order Markov processes. As it was shown in [5], such a process can be modeled through a set of branches rooted in a single node corresponding to a quasi static phenomenon. In other words, for each observation from a sequence obtained within a single time slice, a new leaf node is appended to a common parent node in a BN.

In addition, often simple models assuming conditional independence can provide accurate classification even in the cases where the true processes are more complex and the models ignore certain dependencies [30, 6]. In fact, experiments showed that classification with very simple BNs often outperformed estimation based on complex models with topologies describing the true

dependencies accurately. In other words, despite the simplicity, the models assumed by IMM can be adequate in relatively complex domains. In addition, if we can translate the classification to a counting problem, we can use methods based on majority voting techniques with well known properties [15].

Moreover, inherently robust inference systems based on many independent network fragments require processing of large quantities of evidence in large networks, which introduces large communication and processing burden. Fortunately, due to the factorization properties, we can cope with these challenges by distributing modeling and belief propagation throughout networks of processing nodes [21, 29, 20].

7.2 Extending the IMM to More Complex Topologies

It seems that under certain conditions the reinforcement propagation and filtering algorithms could be adapted to more general topologies.

Inference based on reinforcements could be extended to more general topologies containing ICI components, such as noisy-OR gates [11, 22]; i.e. given some hypothesis node H, there exist evidence nodes E_i which are (i) ancestors of H and (ii) causally independent. It seems that in such cases the robustness of the estimation of the states of H improves with the number of evidence nodes E_i. Namely, the true distribution over $P(H|e_1, \ldots, e_n)$ approaches some point mass distribution as the number in independent causes increases. Consequently, we might be able to introduce predictive reinforcements with similar properties as diagnostic reinforcements w.r.t. the updating tendencies (see proposition 1).

In addition, we could transform multiply connected DAGs into trees consisting of hypernodes. The hypernodes in such trees would represent compound states, combinations of states from the original multiply connected network. Such a construction can be guided by a Junction tree [12], which could facilitate identification of conditionally independent hypernodes. After the states of hypernodes would be determined, we could compute CPTs relating complex states of the hypernodes by using common approaches to belief propagation in multiply connected BNs [12]. Thus, we would obtain a singly connected BN with nodes representing variables with many states. In other words, the basic principles of the IMM can be applied to more general topologies as long as they feature sufficiently large numbers of conditionally independent network fragments. In such cases, however, the determination of p_{re} as well as the justification of the assumption $p_{re} > 0.5$ might not be as straight forward as is the case with tree topologies. Namely, the conditional distributions between the states of the resulting hypernodes might not be well distributed (see section 4.2) and due to many possible states in each hypernode it might be difficult to identify relations (11).

In other words, the rationale from section 4.2 might not be justified in the case of multiply connected BNs. In this section we showed that we can relatively easily identify simple relations (11) between the true probabilities over related events if the CPTs do not have many parameters. This is the case if modeling variables do not have many states and the causal processes can be described through BNs with simple tree topologies. In such cases the assumption $p_{re} > 0.5$ is plausible and by recursively using (15) we can compute the lower bound on the probability $p^H_{\mathcal{F}_i}$ that a particular factor ϕ_i corresponding to fragment $\mathcal{F}^H_i$ supports an accurate reinforcement. In other words, tree topologies allow a modular approach to the determination of $p^H_{\mathcal{F}_i}$, which requires rather coarse assumptions about simple relations. In multiply connected networks, however, relations are more complex and many parameters might be required to describe them. Consequently, if such networks are transformed to simple trees with hypernodes, each hypernode can have many possible states. In other words, in multiply connected networks it can be difficult to justify $p^H_{\mathcal{F}_i} > 0.5$ and, consequently, the presented algorithms might not have asymptotic properties. Clearly, if we had sufficient amounts of data, $p^H_{\mathcal{F}_i} > 0.5$ could be verified

experimentally for each modeling fragment and the results from this paper are valid.

7.3 Related Work

Several researchers have addressed the parameter robustness problem by using convex sets to capture uncertainties of the CPT parameters [10, 4, 27]. However, representations based on convex sets require complex approaches to belief propagation, such as linear programing and can result in not very informative distributions. The IMM introduced in this paper, on the other hand, suggests that inference with BNs can be inherently robust without explicitly considering the parameter uncertainties. In the case of topologies with many independent network fragments, CPTs with very different parameters support accurate classification without any modification of the representation. Namely, contrary to approaches [10, 27, 4], we can show that, given certain topologies, the inference can be very robust even with crisp CPT parameters, if they capture simple greater than/smaller than relations between the true probabilities (see section 4.2); e.g. for binary nodes the experts must merely specify whether a particular parameter is greater or smaller than 0.5. Consequently, it seems that specification of such ordering relations might be easier for an expert or machine learning process than specification of sets of possible parameter values. In other words, the presented approach supports fusion with asymptotic properties for a significant class of models while it requires very coarse assumptions.

Moreover,the reinforcement propagation introduced in IMM seems to be similar to the inference in Qualitative Probabilistic Networks (QPN) [8, 28]. Both approaches use a more abstract view on the conditional probability distributions. Similarly to the QPN approach, we assume that designers or machine learning processes can identify a few coarse grained relations between the natural distributions. But there are significant differences. The QPN approach is suitable for arbitrarily complex topologies, while the reinforcement propagation algorithm is limited to simpler topologies. However, due to a very coarse representation of distributions, the QPN approach becomes inconclusive in cases where different network fragments relative to some classification variable H introduce conflicting updates of the distribution over H [26]. In stochastic domains, this is quite common and the chance of having conflicting influences grows with the number of conditionally independent network fragments. In order to be able to cope with such problems the basic QPN principles were extended by sophisticated representation and updating algorithms [19, 25]. However, it seems that implementation of such approaches is relatively complex and results might be difficult to interpret. For example, the algorithm described in [26] considers the evidence entering order and ignores intercausal influences, while approach in [19] considers relative and absolute magnitudes of influence and introduces complex operations. On the other hand, the reinforcement propagation algorithm introduced in this paper can cope with conflicting evidence in a very robust way since it is based on different assumptions. Contrary to QPNs, we do not assume any preprocessing of the BNs in order to obtain a coarser grained representation of the distributions between the different events of interest. In QPNs relations between the true distributions are encoded through different types of influence and combined through special operators. Instead, we use common BNs in conjunction with a coarse grained inference algorithm which takes into account frequencies of factor reinforcements (see section 4). In contrast to the QPN approach we assume a lower bound on the probability of obtaining a correct factor reinforcement in a given situation. Given coarse grained assumptions that (i) the modeled events are well distributed and (ii) that the designers or machine learning algorithms can identify simple relations in the true distributions correctly, we can consider the frequencies of reinforcements as indications of the true states. Consequently, we can formulate asymptotic properties of classification processes and determine lower bounds on the classification accuracy as well as the lower bounds on the quality checking effectiveness (see section 6.3). In other words, QPN and IMM are complementary. QPN seems to be useful as an intermediate stage

in the design process [24], while the IMM supports robust runtime fusion analysis and accurate estimation with asymptotic properties based on very imprecise models.

Also, the presented robustness analysis is complementary to the common approaches to fine grained sensitivity analysis [2, 3, 1]. Contrary to these approaches, we take into account the relations between the true distributions and the modeling parameters and do not consider the entire network topology along with the instantiations. In this context, the IMM provides coarse guidelines for (i) the early design phase of robust fusion systems and (ii) sensor management at runtime.

Several researchers also addressed the problem of detecting potentially misleading inference results [14, 16, 13]. These approaches are based on the *data conflict* measure and straw models. While they support more general models than the filtering *Algorithm 2* presented in section 6.3, they are based on assumptions that might be difficult to justify in real world settings. This is due to the fact that these approaches consider magnitude of belief updates. On the other hand, *Algorithm 2* is based on updating tendencies, which requires rather coarse assumptions. We use the magnitude of the greatest reinforcement counter as a consistency measure. By considering properties of the reinforcement propagation algorithm, we can show that the effectiveness of *Algorithm 2* improves asymptotically as the number of conditionally independent network fragments increases.

7.4 Further Research

Currently, we are investigating how the reinforcement counters could be used for the detection of inadequate modeling components and erroneous information sources. Namely, patterns of reinforcement counters seem to provide a suitable consistency measure. Our future work will focus on a thorough investigation of the robustness of IMM with respect to erroneous independence assumptions as well as possibilities of using the IMM theory in the context of machine learning techniques. It seems that IMM could provide a plausible rationale for a meaningful state space discretization that would support more efficient learning of the CPT parameters.

References

[1] E. Castillo, J. M. Gutiérrez, and A. S. Hadi. Sensitivity analysis in discrete Bayesian Networks. *IEEE Transactions on Systems, Man, and Cybernetics. Part A: Systems and Humans*, 27:412–423, 1997.

[2] H. Chan and A. Darwiche. Sensitivity analysis in bayesian networks: from single to multiple parameters. In *Proceedings of the 20th conference on Uncertainty in Artificial Intelligence*, pages 67–75, 2004.

[3] V. M. H. Coupé, N. Peek, J. Ottenkamp, and J. D. F. Habbema. Using sensitivity analysis for efficient quantification of a belief network. *Artificial Intelligence in Medicine*, 17(3):223–247, 1999.

[4] F. G. Cozman. Robustness analysis of Bayesian networks with local convex sets of distributions. In *Conference on Uncertainty in Artificial Intelligence*, pages 108–115. Morgan Kaufmann, 1997.

[5] P. de Oude, B. Ottens, and G. Pavlin. Information fusion in distributed probabilistic networks. In *Artificial Intelligence and Applications*, pages 195–201, Innsbruck, Austria, 2005.

[6] P. Domingos and M. J. Pazzani. Beyond independence: Conditions for the optimality of the simple bayesian classifier. In *International Conference on Machine Learning*, pages 105–112, 1996.

[7] M. J. Druzdzel and L. C. van der Gaag. Building probabilistic networks: 'Where do the numbers come from?' Guest editors' introduction. *IEEE Transactions on Knowledge and Data Engineering*, 12(4):481–486, 2000.

[8] Marek Druzdzel and Max Henrion. Efficient reasoning in qualitative probabilistic networks. In Richard Fikes and Wendy Lehnert, editors, *Proceedings of the Eleventh National Conference on Artificial Intelligence*, pages 548–553, Menlo Park, CA, 1993. AAAI Press.

[9] R. O. Duda and P. E. Hart. *Pattern classification and scene analysis*. John Wiley & Sons, New York, 1973.

[10] E. Fagiuoli and M. Zaffalon. 2U: an exact interval propagation algorithm for polytrees with binary variables. *Artificial Intelligence*, 106(1):77–107, 1998.

[11] M. Henrion, M. Pradhan, B. Del Favero, K. Huang, G. Provan, and P. O'Rorke. Why is diagnosis using belief networks insensitive to imprecision in probabilities? In *Twelfth Conference on Uncertainty in Artificial Intelligence*, pages 307–314. Morgan Kaufmann, 1996.

[12] F. V. Jensen. *Bayesian Networks and Decision Graphs*. Springer-Verlag, New York, 2001.

[13] F. V. Jensen, B. Chamberlain, T. Nordahl, and F. Jensen. Analysis in hugin of data conflict. In *In Proc. Sixth International Conference on Uncertainty in Artificial Intelligence*, pages 519–528, 1990.

[14] Y.-G. Kim and M. Valtorta. On the detection of conflicts in diagnostic bayesian networks using abstraction. In *Proceedings of the Eleventh International Conference on Uncertainty in Artificial Intelligence*, pages 362–367, 1995.

[15] L. Lam and C. Y. Suen. Application of majority voting to pattern recognition: An analysis of its behavior and performance. *IEEE Transactions on Systems, Man, and Cybernetics - Part A: Systems and Humans*, 27(5):553–568, 1997.

[16] K. Laskey. Conflict and surprise: Heuristics for model revision. In *In Proc. Seventh International Conference on Uncertainty in Artificial Intelligence*, pages 197–204, 1991.

[17] S. Lauritzen. Causal inference from graphical models, 2001.

[18] David J. C. MacKay. *Information Theory, Inference, and Learning Algorithms*. Cambridge University Press, 2003. available from http://www.inference.phy.cam.ac.uk/mackay/itila/.

[19] S. Parsons. Refining reasoning in qualitative probabilistic networks. In *Proceedings of the 11th Conference on Uncertainty in Artificial Intelligence*, 1995.

[20] M. Paskin and C. Guestrin. A robust architecture for distributed inference in sensor networks. *Technical Report IRBTR -03-039, Intel Research*, 2003.

[21] Gregor Pavlin, Patrick de Oude, Marinus Maris, and Thomas Hood. Distributed perception networks: An architecture for information fusion systems based on causal probabilistic models. In *International Conference on Multisensor Fusion and Integration for Intelligent Systems*, Heidelberg, Germany, 2006.

[22] J. Pearl. *Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference*. Morgan Kaufmann, 1988.

[23] Judea Pearl. Graphs, causality, and structural equation models. Technical Report 980004, 31, 1998.

[24] S. Renooij and L. C. van der Gaag. From qualitative to quantitative probabilistic networks. In *Proceedings of the Eighteenth Conference on Uncertainty in Artificial Intelligence*, pages 422–429, 2002.

[25] Silja Renooij, Simon Parsons, and Linda C. van der Gaag. Context-specific sign-propagation in qualitative probabilistic networks. In *IJCAI*, pages 667–672, 2001.

[26] L.C. van der Gaag S. Renooij and S. Parsons. Propagation of multiple observations in qpns revisited. In *Proceedings of the Fifteenth European Conference on Artificial Intelligence, IOS Press, Amsterdam*, pages 665 – 669.

[27] B. Tessem. Interval probability propagation. *International Journal of Approximate Reasoning*, 7(2):95–120, 1992.

[28] M.P. Wellman. Fundamental concepts of qualitative probabilistic networks. In *Artificial Intelligence*, volume 44, pages 257–303, 1990.

[29] Y. Xiang and V. Lesser. Justifying multiply sectioned bayesian networks. In *Proc. of the 6th Int. Conf. on Multi-agent Systems*, pages 349–356, Boston, 2000.

[30] Huajie Zhang and Charles X. Ling. Geometric properties of naive bayes in nominal domains. In *EMCL '01: Proceedings of the 12th European Conference on Machine Learning*, pages 588–599, London, UK, 2001. Springer-Verlag.

Acknowledgements

The presented work was done within the COMBINED Systems project at the DECIS Lab. Funding was provided by the technology program of the Dutch Ministry of Economic Affairs.

IAS reports

This report is in the series of IAS technical reports. The series editor is Bas Terwijn (bterwijn@science.uva.nl). Within this series the following titles appeared:

Zoran Zivkovic and Olaf Booij *How did we built our hyperbolic mirror omni-directional camera - practical issues and basic geometry.* Technical Report IAS-UVA-05-04, Informatics Institute, University of Amsterdam, The Netherlands, December 2005

A. Diplaros and T. Gevers and N. Vlassis *An efficient spatially constrained EM algorithm for image segmentation.* Technical Report IAS-UVA-05-03, Informatics Institute, University of Amsterdam, The Netherlands, December 2005

J.J. Verbeek *Rodent behavior annotation from video.* Technical Report IAS-UVA-05-02, Informatics Institute, University of Amsterdam, The Netherlands, November 2005.

All IAS technical reports are available for download at the IAS website, http://www.science.uva.nl/research/ias/publications/reports/.