Speeding up approximate MAP by applying domain knowledge about relevant variables

Johan Kwisthout johan.kwisthout@donders.ru.nl
Radboud University, Donders Institute for Brain, Cognition, and Behaviour

Abstract

The MAP problem in Bayesian networks is notoriously intractable, even when approximated. In an earlier paper we introduced the Most Frugal Explanation heuristic approach to solving MAP, by partitioning the set of intermediate variables (neither observed nor part of the MAP variables) into a set of relevant variables, which are marginalized out, and irrelevant variables, which will be assigned a sampled value from their domain. In this study we explore whether knowledge about which variables are relevant for a particular query (i.e., domain knowledge) speeds up computation sufficiently to beat both exact MAP as well as approximate MAP while giving reasonably accurate results. Our results are inconclusive, but also show that this probably depends on the specifics of the MAP query, most prominently the number of MAP variables.

Keywords: MAP problem; approximate MAP; heuristic; algorithm comparison; background knowledge.

1. Introduction

In a Bayesian network, the Maximum A Posteriori (MAP) problem is the computational problem to infer the most probable explanation given evidence, i.e., the mode of a posterior distribution. In a decision support system, where the underlying statistical model is a Bayesian network (e.g., Dey and Stori, 2005; Geenen et al., 2006; Kuang et al., 2017; Liu et al., 2018), the MAP problem typically establishes the diagnosis or advice that is best supported by the available evidence and as such is a crucial component of such systems. Computing the MAP problem is computationally very demanding in larger networks; the problem is $\mathsf{NP}^{\mathsf{PP}}$-hard (Park and Darwiche, 2004) and remains NP-hard under various structural constraints (de Campos, 2020; Kwisthout, 2011) as well as under a variety of approximation approaches (Kwisthout, 2015a). This unfavourable complexity not only hinders practical application; it also implies that even the state-of-the-art approximation algorithm (Annealed MAP; Yuan et al. (2004)) will have difficulty on at least *some* problem instances.

To partially overcome this challenge, in earlier work we proposed a *heuristic* approach to MAP, based on the observation that in many real-world inference queries, only a small subset of the variables really contributes to the decision. In the Most Frugal Explanation heuristic (Kwisthout, 2015b) the set of intermediate variables in the network (those variables that are neither to be explained nor contain observations) is partitioned, based on background information about their role in the inference process, in relevant and irrelevant variables. The relevant variables are marginalized over, the irrelevant variables are sampled from. By this method, and under some assumptions about the probability distribution, a reasonably well approximation of the MAP explanation can be offered with hopefully

less resources. We showed that the quality of the explanation (that is, its deviation to the ground truth MAP explanation in terms of structural distance, rank, and probability) strongly depends on the accuracy of the partitioning in relevant and irrelevant variables, as well as on the probability landscape.

In this follow-up paper we are interested in the question to what extent *domain knowledge* about which variables are and are not relevant in a particular query can help speeding up the MAP computation with reasonable accuracy. To that end, we compared the Most Frugal Explanation heuristic with Exact MAP and Annealed MAP on a number of benchmark networks, measuring the amount of time needed for computation and the deviation from the actual MAP explanation. In general, for an arbitrary computation only a small proportion of the intermediate variables will be relevant (Druzdzel and Suermondt, 1994). Knowledge of the causal structure modelled by the Bayesian network, previous experience, or clinical evidence can help to assess this relevance. In addition, in the absence of an authoritative source of information, one can *compute*, on the fly, whether a particular variable is instrumental for a particular inference question, or use a pre-computed assessment in a lookup table. This computation aims to assess the likeliness (expressed as *intrinsic relevance*, see Section 2) that a variable, were it observed to one of the possible values in its domain, would change the outcome of the MAP problem. In this paper we investigate both aspects: Is knowledge about relevance helpful for establishing MAP explanations in situations where we 1) have access to a 'lookup table' and 2) approximately compute this on the fly as part of the heuristic? Obviously, the second method takes more time as it is integrated in the heuristic. More specifically, our research questions and working hypotheses are the following:

- **RQ 1:** If we have access to pre-computed knowledge about which intermediate variables are irrelevant (have a low likeliness of changing the outcome of the MAP problem), could this speed up approximate inference, with comparable accuracy, compared to the state-of-the-art approximation algorithm?

- **Hypothesis 1:** Yes, on large networks with a huge number of intermediate variables, and with many of them being irrelevant, it will. In either other case probably not.

- **RQ 2:** If we assess, by sampling, during the execution of the algorithm which intermediate variables are relevant, could this speed up approximate inference, with comparable accuracy, compared to the state-of-the-art approximation algorithm?

- **Hypothesis 2:** It might, under the same conditions as in RQ 1, if the assessment of the relevance of the variables is computationally cheap. Otherwise, probably not.

The latter condition is probably necessary. Computing intrinsic relevance is an NP-hard problem (Kwisthout, 2015b). In this study we investigate how costly this computation is in a number of benchmark cases, to see whether this may be a viable option at all. The remainder of this paper is structured as follows. In Section 2 we formally introduce the MAP problem in Bayesian networks, describe Annealed MAP and the Most Frugal Explanation heuristic, and share our notational conventions. In Section 3 we explicate the experimental setup of this study, describe the characteristics of the benchmark networks,

and motivate some of the choices made in the study. The results are given and discussed in Section 4. We conclude the paper in Section 5.

2. Preliminaries

A Bayesian network $\mathcal{B}$ is a probabilistic graphical model that describes a set of stochastic variables, a joint probability distribution over these variables, and the conditional independences that hold in this distribution (Pearl, 1988). $\mathcal{B}$ includes a directed acyclic graph $\mathbf{G}_\mathcal{B} = (\mathbf{V}, \mathbf{A})$, modelling the variables and conditional independences in the network, and a set of parameter probabilities Pr in the form of conditional probability tables (CPTs), capturing the strengths of the stochastic relationships between the variables. The network efficiently factorizes a joint probability distribution $\Pr(\mathbf{V}) = \prod_{i=1}^{n} \Pr(V_i \mid \pi(V_i))$ over its variables, where $\pi(V_i)$ denotes the parents of V_i in $\mathbf{G}_\mathcal{B}$. As notational convention we will use upper case letters to denote individual nodes in the network, upper case bold letters to denote sets of nodes, lower case letters to denote value assignments to nodes, and lower case bold letters to denote joint value assignments to sets of nodes. The set of values for a particular variable (and by extension, set of variables) V is denoted as $\Omega(V)$.

Given a partitioning of the variables in the network into explanation variables $\mathbf{H}$, evidence variables $\mathbf{E}$, and intermediate variables $\mathbf{I}$, the MAP problem is the computational problem to establish the best explanation $\mathbf{h}$ to $\mathbf{H}$ given an observation $\mathbf{e}$ to $\mathbf{E}$. More formally, the MAP problem is the problem to find $\text{argmax}_\mathbf{h}\Pr(\mathbf{H} = \mathbf{h}, \mathbf{E} = \mathbf{e})$. The MAP problem is $\mathsf{NP}^{\mathsf{PP}}$-hard and thus highly intractable (Park and Darwiche, 2004). It stays NP-hard even in trees with cardinality 3 (de Campos, 2020), yet enjoys a fixed parameter tractable algorithm for several parameter sets, e.g., when the tree-width of the moralization of the network, the cardinality of the variables, and the size of $\mathbf{H}$ is bounded.

Among the approximation algorithms are for example Anytime Approximate MAP (Mauá and de Campos, 2012), Annealed MAP (Yuan et al., 2004), and P-Loc (Park and Darwiche, 2001). In this paper we focus on comparison with Annealed MAP, a local search algorithm which can, with good accuracy, approximate relatively large benchmark networks. Approximate MAP, however, is NP-hard as well if no additional constraints are imposed on the input (Kwisthout, 2011, 2015b; Park and Darwiche, 2004).

In Kwisthout (2015b) we introduced a novel approach towards MAP where we exploit a generic property of probability distributions: often-times, when making an inference only a small subset of all variables are actually instrumental in the computation (Druzdzel and Suermondt, 1994). Assuming a partition of the intermediate variables $\mathbf{I}$ in a set of relevant variables $\mathbf{I}^+$ and irrelevant variables $\mathbf{I}^-$, the MFE heuristic approaches MAP by sampling over the irrelevant variables and marginalizing only over the relevant variables. This heuristic, because of this sampling, cannot deal well with deterministic variables as this might lead to conflicting evidence. The algorithm for MFE is given below, taken from (Kwisthout, 2015b, p. 64).

In order to formally assess the relevance of intermediate variables, the concept *Intrinsic relevance* was defined in Kwisthout (2015b) as the fraction of joint value assignments $\mathbf{i}$ in $\Omega(\mathbf{I} \setminus \{I\})$ for which $\text{argmax}_\mathbf{h}\Pr(\mathbf{h}, \mathbf{e}, \mathbf{i}, i)$ is not identical for all $i \in \Omega(I)$.

Algorithm 1 Compute the Most Frugal Explanation

Sampled-MFE($\mathcal{B}, \mathbf{H}, \mathbf{I}^+, \mathbf{I}^-, \mathbf{e}, N$)

1: **for** $n = 1$ to N **do**
2: Choose $\mathbf{i} \in \mathbf{I}^-$ at random
3: Determine $\mathbf{h} = \text{argmax}_{\mathbf{h}} \Pr(\mathbf{H} = \mathbf{h}, \mathbf{i}, \mathbf{e})$
4: Collate the joint value assignments $\mathbf{h}$
5: **end for**
6: Decide upon the joint value assignment $\mathbf{h}_{\text{maj}}$ that was picked most often
7: **return** $\mathbf{h}_{\text{maj}}$

3. Methods

In order to evaluate the Most Frugal Explanation (hereafter MFE) we studied the performance of this heuristic in comparison with Exact MAP via the Junction Tree algorithm (hereafter MAP) and the Annealed MAP algorithm (hereafter ANN) on several benchmark Bayesian networks (Alarm, Andes, Barley, and Hailfinder). We implemented[1] both MFE and ANN in C++ using the LibDAI library (Mooij, 2010) and compared running time and accuracy in terms of the number of variables in the explanation set that differed from the actual MAP explanation. We ran additional tests to check for differences between different accuracy measures, to assess the impact of the hypothesis space, and to compare theoretical and run-time results.

3.1 Experimental setup

We ran our algorithms on an HP Compaq Elite 8300 CMT desktop computer, with Intel Core i7-3770 CPU running at 3.40 GHz and 16 GB of memory, running Debian GNU/Linux 10. We partitioned the variables of the Bayesian network into hypothesis variables, evidence variables, and intermediate variables as described per benchmark network below. For each network we randomly assigned ten joint value assignments to the evidence nodes and simulated each MAP query five times to average out perturbations in running time due to external factors (such as OS activity). We computed the Hamming distance between the MAP explanation and the heuristic approaches (hereafter denoted as 'error') and averaged running time and error over the $5 \times 10 = 50$ simulations. Data was locally stored and processed; raw data, scripts, and processed data are available[2] at the Donders Data Repository for colleagues to inspect and reuse.

Per benchmark network, we ported the .bif files from the BNLearn repository[3] to factor graphs using an in-house tool[4] since the LibDAI library requires factor graphs as input. As MFE cannot deal properly with deterministic variables, we manually adjusted these variables to have values very close to 1 and 0; each 0 entry in a CPT was replaced with 0.000000001 and the 1 entry matched such that the distribution adds up to 1. We use

1. `https://gitlab.socsci.ru.nl/j.kwisthout/most-frugal-explanations`.
2. `https://doi.org/10.34973/7p10-m012`
3. `https://www.bnlearn.com/bnrepository/`
4. Bif2Fg, `https://gitlab.socsci.ru.nl/j.kwisthout/most-frugal-explanations`; note that this tool assumes a specific ordering of the CPT entries to work.

the orginal factor graphs for MAP and ANN and the adjusted factor graphs for MFE computations[5].

For the MFE heuristic we simulated two variants: one where the partitioning into relevant and irrelevant variables was given (by pre-computation) as to simulate background knowledge, and one where the partitioning was part of the heuristic approach. In the first case, as pre-computation we approximated the intrinsic relevance by 1000 samples or, in case that was computationally infeasible, by 100 or even 10, as indicated per benchmark network below; the threshold for inclusion in the set of relevant variables was set to 0.1. In the second case, the approximation was part of the heuristic; we sampled thrice and deemed a variable as relevant if the intrinsic relevance was non-zero[6]. In both variants, the algorithm marginalized out the relevant variables and assigned a random value to the irrelevant variables.

The LibDAI library has no functionality to compute MAP efficiently; only inference (using the junction tree algorithm (Lauritzen and Spiegelhalter, 1988)) and MPE (that is, when there are no intermediate variables). As a workaround we computed MAP by computing the joint distribution over the MAP variables and then searching for the maximum value. Obviously this workaround approach does not utilize the independences between the hypothesis variables, leading to an inefficient implementation that hugely effects the results; see Section 4.3 for a further analysis. To mitigate this, we decided to limit the size of the hypothesis set; rather than the 20 hypothesis variables in Yuan et al. (2004) we selected five variables in Andes and Hailfinder and four in Barley. For Hailfinder we explored the effect of including more variables (specifically seven and ten) in the hypothesis set[7]. Note that approximate MAP is intractable, even when a singleton binary hypothesis variable is used (Kwisthout, 2015a).

3.1.1 Alarm

The Alarm network (Beinlich et al., 1989) consists of 37 discrete random variables which have a natural partitioning into hypothesis variables (eight diagnostic variables), evidence variables (sixteen observable findings) and intermediate variables (the remaining thirteen variables). Variables can take on two, three, or at most four values; there are 46 arcs and the in-degree is at most four. In our simulations we used the natural partitioning as indicated above. Pre-computation of the intrinsic relevance of the thirteen intermediate variables was done using 1000 samples each; pre-computation time varied between 11 and 78 seconds per variable.

3.1.2 Barley

The Barley network (Kristensen and Rasmussen, 2002) has 48 nodes with 84 arcs, with in-degree at most four. However, the cardinality of the variables is significantly larger, with

5. Note that none of the algorithms exploits determinism, and the values used have a negligible impact.
6. Given the actual distribution, any threshold between 0 and 0.9 would have led to identical results in these experiments. The threshold for inclusion, however, is a tunable parameter in MFE in general.
7. One reviewer suggested that the number of candidate hypotheses, rather than the size of the hypothesis set (i.e., the number of variables), is the crucial variable. That is a correct observation; however, given minor differences between the hypothesis variables we actually considered, the size of the hypothesis set is the most dominant factor determining the number of candidate hypothesis.

one node having no less than 67 states. This renders exact MAP infeasible on the Barley network. In line with Yuan et al. (2004) we interpreted the ten root variables as hypothesis variables and the eight leaf variables as evidence variables, which reasonably matches the layout of this network (Kristensen and Rasmussen, 2002, p.206). Due to the impossibility to compute either MAP or MFE over the total hypothesis set due to the library constraints, we selected the first four variables as our hypothesis set. From piloting it became pretty clear that pre-computation of intrinsic relevance using 1000 samples was infeasible, as a single sample already took about 90 seconds due to the intractability of MAP computation. We approximated intrinsic relevance using 10 samples per variable.

3.1.3 Andes

Andes (Conati et al., 1997) has 223 binary nodes with 338 arcs, and in-degree at most six; the CPTs are to a large extent rolled out from canonical models (leaky noisy or/and models). There are 89 root nodes and 22 leaf nodes; for these simulations, the first five root variables make up the hypothesis set. Furthermore, as Andes has several deterministic variables, for the MFE computation we adjusted these variables as indicated above. Due to the large number of intermediate variables we opted for 100 samples per variable to approximate intrinsic relevance.

3.1.4 Hailfinder

Finally, Hailfinder (Abramson et al., 1996) has 56 nodes with 66 arcs with maximum in-degree four. This network has 17 root nodes and 13 leaf nodes; the cardinality ranges from two to eleven. Again the size of the hypothesis set renders exact MAP intractable in practice. To experiment with the effect of the size of the hypothesis set on the behaviour of the different algorithms we ran simulations using the first 10, 7, and 5 root variables as hypothesis set. Also for Hailfinder we adjusted the deterministic variables as indicated above. In contrast to MAP inference, computing the intrinsic relevance using 1000 samples was feasible, taking from 10 to 60 seconds.

4. Results

This result section is structured as follows. We start with presenting the main results, namely running time and accuracy of MAP, ANN, MFE with sampling, and MFE with pre-computation (MFE+), for Alarm, Barley, Andes, and Hailfinder (with five hypothesis variables) in Table 1 and graphically in Figure 1. We then further investigate the effect on hypothesis set size for Hailfinder (Figure 2) and present a preliminary interpretation of these results. In Sub-section 4.1 we investigate, using the Alarm network, whether the Hamming distance results sufficiently generalize to other accuracy measures for MAP. In Sub-section 4.2 we look at the number of variables that are actually considered to be relevant (based on the pre-computation) per network, and in Sub-section 4.3 we use this information to compare actual running time with some theoretical analyses of the number of elementary operations needed. In Sub-section 4.4 we discuss the overall simulations.

The results are difficult to interpret. For this hypothesis size, MAP is quicker than ANN and MFE and as quick as MFE+! It seems that the initial costs and the estimation of effect

Network	Alarm		Barley		Andes		Hailfinder (5)	
	RT	Err	RT	Err	RT	Err	RT	Err
MAP	0.0135	-	1.4973	-	0.1556	-	0.0051	-
ANN	0.1536	0.08	120.9530	0.46	6.3393	0.04	0.2891	0.70
MFE	0.1546	0.42	2420.1568	1.38	188.1614	0.26	1.5000	1.42
MFE+	0.0133	0.70	1.4669	1.66	0.1511	0.56	0.0061	1.84

Table 1: Summary of the main results for running time (RT) and error (Err) for the four approaches on four benchmark networks.

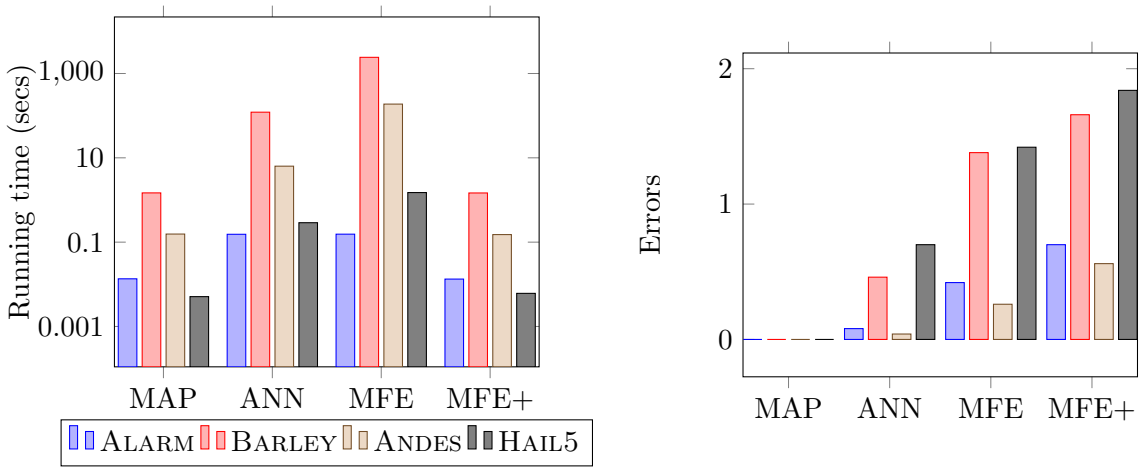

Figure 1: Graphical depiction of the main results for running time (RT) and error (Err) for the four approaches on four benchmark networks. In the left panel from left to right the average running time in seconds for exact MAP, Annealed MAP, MFE with sampled relevance, and MFE with pre-computed relevance (note the log scale); in the right panel the average errors.

on local changes for ANN are too large to be competitive for MAP for small hypothesis sizes; for MFE the sampling is way too costly, which may be due to an unreasonable large MAP computation time due to the limitations of the library. We further explored this in Section 4.3. For MFE+, we see that the limited amount of marginalisation operations has little effect on the running time.

When comparing errors, we note that, to our surprise, in MFE the sampling over 3 samples led to less errors than sampling over 1000 samples in the pre-compute stage. The only reasons we can think of that explain this would be that by either the sampling result is incorrect, or that the heuristic sometimes (even with perfect knowledge) leads to a different MFE explanation than the MAP explanation, as also observed in Kwisthout (2015b), such that the lower number of samples incidentally leads to an explanation closer to MAP.

As a follow-up simulation we explored the effect of a larger hypothesis space. For the Hailfinder network we ran the same simulations, but now also with the first seven and

the first ten root variables designated as hypothesis variables. The results are depicted in Figure 2.

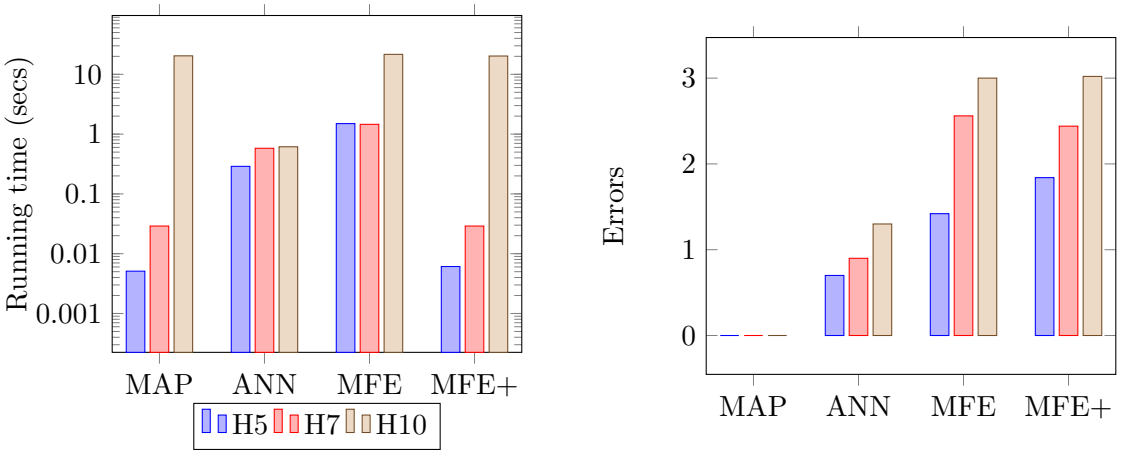

Figure 2: Results for the HAILFINDER network with 5, 7, and 10 hypothesis nodes. In the left panel from left to right the average running time in seconds for exact MAP, Annealed MAP, MFE with sampled relevance, and MFE with pre-computed relevance (note the log-scale); in the right panel the average errors. Note that for 10 hypothesis nodes the inefficiency of the MAP computation dominates the running times.

4.1 Other error measures

In Kwisthout (2015b), three error measures were used, in line with the three notions of approximate MAP established in Kwisthout (2015a); in addition to the Hamming distance between ground truth MAP and the approximate explanation, the *ratio* of their probabilities as well as the *rank* of the explanation (i.e., the number k such that the approximate explanation is the kth most probable explanation) were used. These measures can in principle deviate, hence, we ran a sanity check test to assure that the Hamming distance gives a reasonable impression of the quality of the approximate explanation. The results can be found in Figure 3.

ANN scores best on all three measures, then MFE, then MFE+. This suggests that indeed the Hamming distance gives a good indicator of the accuracy of the approximation.

4.2 Size of relevant variables set

We refer to Figure 4. In general: With more hypothesis variables, the ratio increases (see the increase in HAILFINDER but also the large ratio in *Alarm*). For smaller sets, the number of relevant variables is really low (see for example *Andes*), which ought to lead to a more efficient algorithm; reducing factors with a large set of sampled values potentially greatly reduces the resulting factor size and discards the marginalization computation. However,

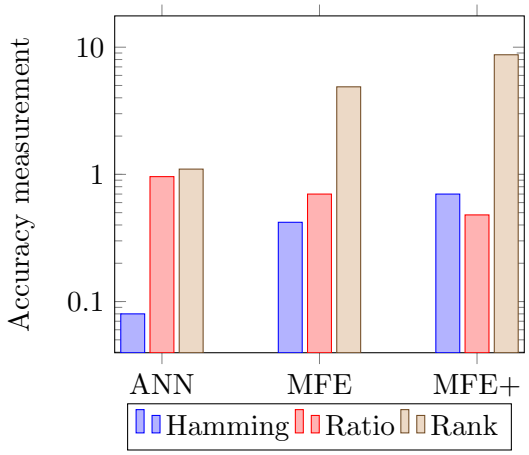

Figure 3: Comparison between Hamming distance, ratio, and rank of the explanations. Note that for distance and rank lower is better, whereas for ratio a value closer to 1 is better.

in reality, it does not, at least not for these simulations. We explore this in the next Sub-section.

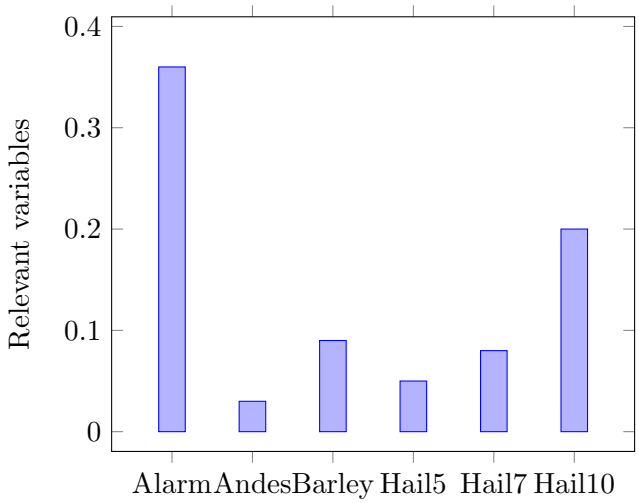

Figure 4: Ratio of relevant variables out of all intermediate variables.

4.3 Running time vs. theoretical analysis

For the Alarm network, we see that on average four out of the eleven relevant variables are relevant. We closely examined a particular run to get some insight in the discrepancy

between theoretical results and actual runtime. Table 2 shows the characteristics of the particular MAP query and MFE+ heuristic.

variable	value	variable	value	variable	value
HISTORY	FALSE	CVP	HIGH	PCWP	HIGH
HRBP	HIGH	HREKG	NORMAL	HRSAT	NORMAL
TPR	HIGH	EXPCO2	HIGH	MINVOL	ZERO
FIO2	LOW	SAO2	LOW	PAP	NORMAL
PRESS	NORMAL	MINVOLSET	LOW	CO	LOW
BP	LOW				

Table 2: Example evidence setting.

For this query, LVEDVOLUME, STROKEVOLUME, VENTTUBE, and VENTALV were found relevant; we sample over ERRLOWOUTPUT, ERRCAUTER, PVSAT, SHUNT, VENTMACH, VENTLUNG, ARTCO2, CATECHOL, and HR. The biggest factor, once they are reduced with the evidence and sampled intermediate variables, now contains KINKEDTUBE, INTUBATION, and VENTTUBE or $2 \times 3 \times 4 = 24$ entries, rather than four variables with 92 entries. When comparing MAP to MFE+ we see that they LibDAI library spends $2.1ms$ on a junction tree run and $12.2ms$ computing the marginal distribution in MAP, with $0.8ms$, respectively $8.2ms$ for MFE+. The library does not utilize the fact that, after marginalization, the biggest factor contains two hypothesis variables with 12 entries (rather than all hypothesis variables with 384 entries), which may explain why we see less effect on the running times than what could be expected in theory.

4.4 Discussion

Based on the above results, we conclude that background knowledge *may* help, and perhaps *should* help in theory, but it does not really show in the current study using the LibDAI library. This is partially due to the inefficient MAP computation used as sub-routine in MFE; in addition, marginalization might be less costly for variables with small domains than sampling might be due to various overhead. When comparing actual reported running times (for Alarm and Hailfinder) we see that Yuan et al. (2004) is about 2.5 times faster than our results, despite running on an older computer with less memory. Perhaps the tight integration with the SMILE inference engine in this implementation can account for this.

5. Conclusion

When reflection back on the original research questions, it is clear that when used as approximation algorithm, MFE is unlikely to beat the state-of-the-art approximation as already for a single sample and for very few relevant variables the assessment of relevance is very costly save in the easiest situations. However, background knowledge (either precomputed or by estimation of experts, inasmuch as this is possible) may still be useful, although our results show little improvement over computing MAP exactly; however, we argued that this may be partially due to the inefficient MAP computation using the LibDAI library. We see that there is some variation on the relevant variables given the specific evidence, but

there is also quite a lot of overlap so many variables may not be that relevant anyway for designated sets of hypothesis and evidence variables.

In their study comparing Annealed MAP with P-LOC and P-SYS, Yuan et al. (2004) uses 20 MAP variables, which favours local search approaches over approaches (such as MFE) that work on the entire hypothesis set simultaneously; however, MAP is intractable in general even for a singleton binary MAP variable, implying that there are hard instances with a limited number of hypothesis variables. It would be interesting to compare the approaches on such instances.

For future work we obviously would like to either patch the LibDAI library with a 'true' Junction-tree based MAP algorithm, or use another library that has this feature, to be able to offer MFE a fairer comparison with other approximation algorithms. We also aim to implement other approximation algorithms, such as Anytime Approximate MAP and P-Loc and compare MFE on other - and larger - networks. Finally, the MFE algorithm should be able to circumvent incompatible evidence due to deterministic variables so that the heuristic also works on such networks without patching.

References

B. Abramson, J. Brown, W. Edwards, A. Murphy, and R. Winkler. Hailfinder: A Bayesian system for forecasting severe weather. *International Journal of Forecasting*, 12(1):57–71, 1996.

I. Beinlich, G. Suermondt, R. Chavez, and G. Cooper. The ALARM monitoring system: A case study with two probabilistic inference techniques for belief networks. In *Proceedings of the Second European Conference on AI and Medicine*, pages 247–256. Springer-Verlag, 1989.

C. Conati, A. Gertner, K. VanLehn, and M. Druzdzel. On-line student modeling for coached problem solving using Bayesian networks. In *Proceedings of the 6th International Conference on User Modeling*, pages 231–242. Springer-Verlag, 1997.

C. P. de Campos. Almost no news on the complexity of MAP in Bayesian networks. In M. Jaeger and T. D. Nielsen, editors, *International Conference on Probabilistic Graphical Models, PGM 2020, 23-25 September 2020, Aalborg, Denmark*, volume 138 of *Proceedings of Machine Learning Research*, pages 149–160. PMLR, 2020.

S. Dey and J. A. Stori. A Bayesian network approach to root cause diagnosis of process variations. *International Journal of Machine Tools and Manufacture*, 45(1):75–91, 2005.

M. J. Druzdzel and H. J. Suermondt. Relevance in probabilistic models: "backyards" in a "small world". In *In Working notes of the AAAI–1994 Fall Symposium Series: Relevance*, pages 60–63, 1994.

P. L. Geenen, A. R. W. Elbers, L. C. van der Gaag, and W. L. A. van der Loeffen. Development of a probabilistic network for clinical detection of classical swine fever. In *Proceedings of the Eleventh Symposium of the International Society for Veterinary Epidemiology and Economics*, pages 667–669, 2006.

K. Kristensen and I. Rasmussen. The use of a Bayesian network in the design of a decision support system for growing malting barley without use of pesticides. *Computers and Electronics in Agriculture*, 33:197–217, 2002.

D. Kuang, R. Yang, X. Chen, G. Lao, F. Wu, X. Huang, R. Lv, L. Zhang, C. Song, and S. Ou. Depression recognition according to heart rate variability using Bayesian networks. *Journal of psychiatric research*, 95:282–287, 2017.

J. Kwisthout. Most Probable Explanations in Bayesian networks: Complexity and tractability. *International Journal of Approximate Reasoning*, 52(9):1452–1469, 2011.

J. Kwisthout. Tree-width and the computational complexity of MAP approximations in Bayesian networks. *Journal of Artificial Intelligence Research*, 53:699–720, 2015a.

J. Kwisthout. Most frugal explanations in Bayesian networks. *Artificial Intelligence*, 218: 56–73, 2015b.

S. L. Lauritzen and D. J. Spiegelhalter. Local computations with probabilities on graphical structures and their application to expert systems. *Journal of the Royal Statistical Society*, 50(2):157–224, 1988.

S. Liu, J. Zeng, H. Gong, H. Yang, J. Zhai, Y. Cao, J. Liu, Y. Luo, Y. Li, L. Maguire, et al. Quantitative analysis of breast cancer diagnosis using a probabilistic modelling approach. *Computers in biology and medicine*, 92:168–175, 2018.

D. D. Mauá and C. P. de Campos. Anytime marginal MAP inference. In *Proceedings of the 29th International Conference on Machine Learning, ICML 2012, Edinburgh, Scotland, UK, June 26 - July 1, 2012*. icml.cc / Omnipress, 2012.

J. Mooij. libDAI: A free and open source C++ library for discrete approximate inference in graphical models. *Journal of Machine Learning Research*, 11:2169–2173, Aug. 2010.

J. Park and A. Darwiche. Approximating MAP using local search. In *Proceedings of the 17th Conference on Uncertainty in Artificial Intelligence(UAI)*, pages 403–410. Morgan Kaufmann Publishers, 2001.

J. D. Park and A. Darwiche. Complexity results and approximation settings for MAP explanations. *Journal of Artificial Intelligence Research*, 21:101–133, 2004.

J. Pearl. *Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference*. Morgan Kaufmann, Palo Alto, CA, 1988.

C. Yuan, T. Lu, and M. J. Druzdzel. Annealed MAP. In D. Chickering and J. Halpern, editors, *Proceedings of the Twentieth Conference in Uncertainty in Artificial Intelligence*, pages 628–635. AUA, 2004.