Importance Sampling for General Hybrid Bayesian Networks

Changhe Yuan
Department of Computer Science and Engineering
Mississippi State University
Mississippi State, MS 39762
cyuan@cse.msstate.edu

Marek J. Druzdzel
Decision Systems Laboratory
School of Information Sciences and
Intelligent Systems Program
University of Pittsburgh
Pittsburgh, PA 15260
marek@sis.pitt.edu

Abstract

Some real problems are more naturally modeled by hybrid Bayesian networks that consist of mixtures of continuous and discrete variables with their interactions described by equations and continuous probability distributions. However, inference in such general hybrid models is hard. Therefore, existing approaches either only deal with special instances, such as *Conditional Linear Gaussians* (CLGs), or approximate a general model with a restricted version and then perform inference on the simpler model. However, results thus obtained highly depend on the quality of the approximations. This paper describes an importance sampling-based algorithm that directly deals with hybrid Bayesian networks constructed in the most general settings and guarantees to converge to the correct answers given enough time.

1 Introduction

This paper addresses inference in general hybrid Bayesian networks that contain mixtures of discrete and continuous variables and mixtures of deterministic and probabilistic relations. Since the general case is difficult, existing research often focuses on special instances, such as *Conditional Linear Gaussians* (CLGs) or *Augmented* CLGs [5, 10, 11, 12, 18]. Recent research focused on developing methodologies for more general non-Gaussian models. One approach uses *Mixture of Truncated Exponentials* (MTE) to approximate arbitrary probability distributions [2, 14]. Another approach approximates arbitrary hybrid models using CLGs [20]. Although the inference step can be done exactly for these approaches, they may encounter numerical instability when trying to improve the quality of the approximate models for better inference in large models.[1] Other approaches for inference in hybrid Bayesian networks include dynamic discretization [9] and *junction tree algorithm with sample potentials* [8].

Importance sampling-based algorithms, such as *likelihood weighting* (LW) [3, 19], AIS-BN [1], *Dynamic* IS [15], and EPIS-BN [25], have proven effective in solving discrete models. Monte Carlo sampling puts minimum restriction on the representation of the models, which makes it a natural choice for inference in general hybrid Bayesian networks. Add-ons to spreadsheet programs, such as Excel, apply Monte Carlo sampling in solving spreadsheet models. Still, Excel offers complete modeling freedom and allows users to specify any interaction among its cells (these can be viewed as variables). Another group of modeling tools that offer a complete modeling freedom are visual spreadsheets, with Analytica [16] being a prominent example. However, the main shortcoming of both spreadsheets and visual spreadsheets is that they only allow forward sampling and have no capability to deal with evidential reasoning. Evidential reasoning is hard, mainly because the a-priori probability of observed variables in extremely low. Importance sampling has proven its worth precisely in cases with extremely unlikely evidence.

In this paper we propose an importance sampling-based algorithm to deal with evidential reasoning in general hybrid Bayesian networks. More specifically, we extend the EPIS-BN algorithm proposed by Yuan and Druzdzel [25] to the most general setting. Although the extension seems natural, several nontrivial difficulties unique to hybrid models need to be overcome for the new algorithm to work. Firstly, we propose a technique called *delayed importance function generation* that applies *Hybrid Loopy Belief Propagation* (HLBP) [24] to calculate an importance function. This technique allows the use of arbitrary conditional relations and makes the general representation in Section 2 possible. Secondly, we propose another technique called *soft arc reversal* to draw importance samples when a deterministic variable has been observed. This technique makes importance sampling a viable approach for hybrid models.

The remainder of this paper is structured as follows. In Section 2, we discuss a general representation of hybrid

[1] Personal communication with Dr. Barry Cobb

Bayesian networks. In Section 3, we discuss the HEPIS-BN algorithm. Finally, we present results of an empirical evaluation of the algorithm in Section 4.

2 General Representation of Hybrid Bayesian Networks

In order not to limit the modeling power of a Bayesian network-based tool, we should make the representation of hybrid Bayesian networks as general as possible. The representation should not only allow mixtures of discrete and continuous variables, but also allow arbitrary orderings between them, including discrete variables with continuous parents. The representation should also allow linear or nonlinear deterministic equations and arbitrary probability distributions. In our work, we use a representation defined as follows.

A hybrid Bayesian network contains a mixture of discrete and continuous nodes and can be factorized as a product of *hybrid conditional probability tables* (HCPTs), one for each variable conditional on its parents. An HCPT is defined as follows.

Definition 1. *For every node X, its parents PA(X) are divided into two disjoint sets: discrete parents DPA(X) and continuous parents CPA(X). Then, its* hybrid conditional probability table (HCPT) $P(X|PA(X))$ *is a table indexed by its discrete parents DPA(X) and with each entry representing one of the following conditional relations:*

1. *If X is a* discrete variable with only discrete parents, *a discrete probability distribution;*
2. *If X is a* discrete variable with continuous parents, *a discrete probability distribution dependent on CPA(X);*
3. *If X is a* continuous and deterministic variable, *a deterministic equation dependent on CPA(X).*
4. *If X is a* continuous and stochastic variable, *a deterministic equation dependent on CPA(X) plus a noise term having an arbitrary continuous probability distribution with parameters dependent on CPA(X) as well;*

Let us use several concrete examples to illustrate the power of this representation. First, Item 1 reduces to a *conditional probability table* (CPT) as in discrete Bayesian networks.

Item 2 models the conditional relation of a discrete node with both discrete and continuous parents. It does not restrict the representation to any specific form but only specifies the property that it should have. For example, one allowable representation is the softmax function [13]. Let A be a discrete node with possible values $a_1, a_2, ..., a_m$, and let $X_1, X_2, ..., X_k$ be its continuous parents. We have

$$P(A = a_i|x_1, x_2, ..., x_k) = \frac{exp\left(w_0^{(i)} + \sum_{l=1}^{k} w_l^{(i)} x_l\right)}{\sum_{j=1}^{m} exp\left(w_0^{(j)} + \sum_{l=1}^{k} w_l^{(j)} x_l\right)}. \quad (1)$$

In case A has discrete parents, we define a different softmax function for every configuration of the discrete parents. Once the parents are instantiated, the conditional relation becomes a concrete discrete distribution. Note that the representation allows any other form with the property. Another example would be different discrete distributions for different ranges of an arbitrary function of the continuous parents.

Item 3 models the deterministic relation between a continuous variable and its parents. For each instantiation of its discrete parents, we specify a different linear or nonlinear deterministic equation.

Item 4 is a generalization of Item 3. The conditional relation contains two parts, a deterministic equation plus a noise term. As an example, suppose a continuous variable Y has continuous parents X_1 and X_2. We can specify the conditional relation as follows:

$$Y = f(X_1, X_2) + N(g(X_1, X_2), h(X_1, X_2)), \quad (2)$$

where f, g, h are arbitrary functions, and $N(\mu, \sigma)$ is a random noise from the normal distribution with mean μ and standard deviation σ.

Following a popular convention, we use discrete variables only as indices. We can easily relax this assumption and allow discrete variables to behave as numerical variables. There is only one entry in a node's HCPT if it has no discrete parents. Also, the equation part in Representation 4 only shifts the location of the noise term. To simplify our discussion, we later treat the conditional relation a distribution as a whole.

3 Importance Sampling for General Hybrid Bayesian Networks

In this section, we propose the HEPIS-BN algorithm. We first review the main idea of EPIS-BN, the basis of our new algorithm. Then we discuss how to apply the *Hybrid Loopy Belief Propagation* (HLBP) [24] algorithm using *delayed importance function generation* to calculate the importance function. Finally we discuss how to deal with evidence in the HEPIS-BN algorithm.

3.1 The EPIS-BN Algorithm: A Review

The EPIS-BN algorithm is an importance sampling-based algorithm for Bayesian networks proposed by Yuan and

Druzdzel in [23, 25]. The importance function of EPIS-BN is defined as:

$$\rho(\mathbf{X}\backslash\mathbf{E}) = \prod_{i=1}^{n} P(X_i|\text{PA}(X_i), \mathbf{E}) \ , \quad (3)$$

where each $P(X_i|\text{PA}(X_i), \mathbf{E})$ is an *importance conditional probability table* (ICPT) [1]. The following theorem shows that we can calculate the ICPTs exactly in polytrees [25].

Theorem 1. *Let X_i be a variable in a polytree, and $\mathbf{E}$ be the set of evidence variables. The exact* ICPT *$P(X_i|PA(X_i), \mathbf{E})$ for X_i is*

$$\alpha(PA(X_i))P(X_i|PA(X_i))\lambda(X_i) \ , \quad (4)$$

where $\alpha(PA(X_i))$ is a normalizing constant dependent on $PA(X_i)$, and $\lambda(X_i)$ is the message to X_i sent from its descendants.

Based on the observation that LBP provides surprisingly good results for many networks with loops [17], Yuan and Druzdzel [25] propose to use a small number of LBP to estimate the importance functions for loopy networks in the EPIS-BN algorithm. Experimental results in [25] show that subsequent importance sampling provides an insurance against those cases in which LBP does not converge to the right posterior . The resulting algorithm, EPIS-BN, achieves a considerable improvement over the previous state-of-the-art algorithm, AIS-BN [1].

In order to generalize the EPIS-BN algorithm to deal with hybrid Bayesian networks, we need to perform belief propagation in hybrid models. We use the *Hybrid Loopy Belief Propagation* (HLBP) algorithm [24] for this purpose. Since no closed-form solutions exist for the LBP messages in general hybrid models, HLBP represents the messages using mixture of Gaussians (MoGs) and formulates their calculation as Monte Carlo integration problems. Readers who are interested in HLBP can refer to [24] for more details.

3.2 Delayed Importance Function Generation

We now discuss how to apply HLBP to calculate an importance function. Similarly to EPIS-BN, we are interested in making use of the λ messages estimated by HLBP to calculate the ICPTs. Since variables are all discrete in discrete Bayesian networks, we can multiply the λ messages with CPTs to get all the ICPTs. It is much more complex in hybrid Bayesian networks. We discuss the following several scenarios separately.

Discrete variable with only discrete parents: This scenario reduces to the same case as in discrete Bayesian networks.

Discrete variable with continuous parents: In this case, we have a discrete λ message. However, the conditional relation is not determined before the continuous parents are instantiated. To deal with the problem, we propose a technique called *delayed importance function generation.* The main idea is to delay combining the λ message with the conditional relation until the continuous parents are all instantiated. Importance sampling instantiates the nodes in their topological order. Therefore parent nodes are always instantiated before their children. Once the parents of a discrete variable are determined, the conditional relation becomes a discrete probability distribution which can be multiplied by the incoming $lambda$ messages to produce the importance function.

Continuous variable with only discrete parents: All the conditional relations in this case are continuous probability distributions with fixed parameters. We can approximate the conditional distributions with MoGs [22] and multiply with the λ message to produce new MoGs using either Gibbs sampling [21] or importance sampling [24] to get the ICPT.

Continuous variable with continuous parents: Again, the conditional relation is not determined before the continuous parents are instantiated. We also delay computing the importance function until we have values for the parents. Once the parents are instantiated, we can approximate the conditional distribution by an MoG and then multiply by the λ messages to get the importance function.

There is one situation under which the λ messages are useless. As soon as the parents of a deterministic node are instantiated, the node itself is also determined. We simply evaluate the deterministic relation to get its value. Therefore, the importance function and the original distribution are the same for this node. Its λ messages can be simply discarded.

3.3 The Algorithm

Now, let us discuss the HEPIS-BN algorithm, which essentially boils down to drawing a single sample from the importance function. The importance function that we have now is expressed as a set of ICPTs for some variables and a set of HCPTs and $\lambda(x)$ messages for the others.

The algorithm works as follows. We first order all the nodes in their topological order and initialize the weight of the current sample to be 1.0. Then we sample for each node X in the ordering according to the following several scenarios.

Stochastic variable with no evidence: If X's ICPT is already computed, we find the correct importance function $I(x)$ in ICPT using its parents' values and draw a sample. Otherwise, we can find the correct conditional probability distribution $P(x|u)$, evaluate its parameters using the continuous parents, and multiply it with the λ message to get the importance function $I(x)$. We then sample from $I(x)$

and update the weight as follows:

$$w = w * \frac{P(x|u)}{I(x)} \ . \tag{5}$$

Deterministic variable with no evidence: We can simply evaluate the deterministic relation and get value for X. There is no need to adjust the weight.

Stochastic variable with evidence: We simply take the evidence e as though it is the sample and adjust the weight as follows:

$$w = w * P(e|u) \ . \tag{6}$$

Deterministic variable with evidence: This case needs special care. Note that traditional sampling methods instantiate a network in its topological order. However, when a deterministic node is observed, the values of the parents together with the evidence may conflict with the evidence. The difficulty was often ignored for discrete Bayesian networks because not many states are possible and we can still get valid samples. However, it is extremely unlikely to hit a sample that satisfies a deterministic equation in hybrid Bayesian networks. To address the problem, we propose a technique that we call *soft arc reversal*: we draw samples for all the parents except one and solve the remaining parent based on other parents' values. The name is due to the fact that the technique is similar to performing a physical arc reversal on the network.

We explain it using a concrete example.

Example:

$P(C\|A,B)$
$C = A + 2 * B$

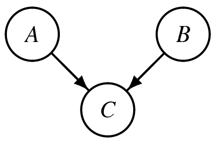


Figure 1: A simple hybrid Bayesian network.

Suppose we have a small hybrid Bayesian network with three continuous variables A, B, and C as in Figure 1. C is a deterministic node dependent on A and B. We also assume that C is observed to be 4.0. A conventional sampling algorithm would sample A and B first and accept the evidence of C as the sample. However, such a sample would almost certainly have zero probability, because it is extremely unlikely to get values for A and B that satisfy the deterministic equation $P(C|A, B)$. Note that once we get the value for either A or B, the value of the other variable is already determined. For example, suppose we get sample 2.0 for A. Then B can only take value 1.0. Therefore, instead of sampling for B, we should solve for it and take 1.0 as the sample.

Algorithm: HEPIS-BN
Input: Hybrid Bayesian network B, a set of evidence variables **E**, and a set of non-evidence variables **X**;
Output: The marginal distributions of non-evidence variables.

1. Order the nodes in the topological order of the network.
2. Perform hybrid loopy belief propagation (HLBP) for several steps.
3. Precompute the ICPTs for all the nodes with only discrete parents using messages by HLBP.
4. Draw a desirable number of samples for the network according to Section.
5. Estimate the posterior marginal probability distributions for unobserved variables using the weighted samples.

Figure 2: The HEPIS-BN algorithm.

We need to consider several issues when deciding which of the parents to choose to be the new child. First, since we want to use the values of the other parents to solve for the chosen one, we need an equation solver. In our implementation, we use the *Newton's method for solving nonlinear set of equations* [6]. However, not all equations are solvable by this equation solver or any equation solver for that matter. We may want to choose the parent that is easiest to solve. This can be tested using a preprocessing step. In more difficult cases, we have to resort to modeler's help and ask for specifying which parent to solve or even specify the inverse functions manually. When there are multiple choices, one useful heuristic is to choose the parent with the largest variance. There are also circumstances under which we need to resort to upper level soft arc reversal if all the parents are deterministic as well.

In the end, we get an independent weighted sample for the network. What remains is to repeat the process until we get enough samples. We outline the HEPIS-BN algorithm in Figure 2.

4 Experimental Results

We tested the HEPIS-BN algorithm on three commonly used hybrid Bayesian networks: emission network, augmented emission network, and augmented crop network [10, 12] shown in Figure 3. For the emission networks, we use the same parameterizations as in [12]. For the augmented crop network, we added a deterministic node TotalPrice to the original crop network and parameterized it as in Table 1 to demonstrate the capability of soft arc reversal. We also transformed the emission network into a dynamic model to test the algorithms on a large model.

To evaluate how well HEPIS-BN performs, we discretized the ranges of continuous variables to 50 intervals and then calculated the Hellinger's distance [7] between the results of HEPIS-BN and the exact solutions obtained by a massive amount of computation (likelihood weighting (LW) with 100M samples) as the error for HEPIS-BN. We tried to compare our algorithm against another accessible algorithm to us, the Gibbs sampling implemented in BUGS [4], but encountered convergence difficulties similar to those reported in [12], so we compared our algorithm mainly against LW. Also, HEPIS-BN achieves a precision much better than HLBP. We omit the results for clarity. All the results are the average of 50 runs of the experiments.

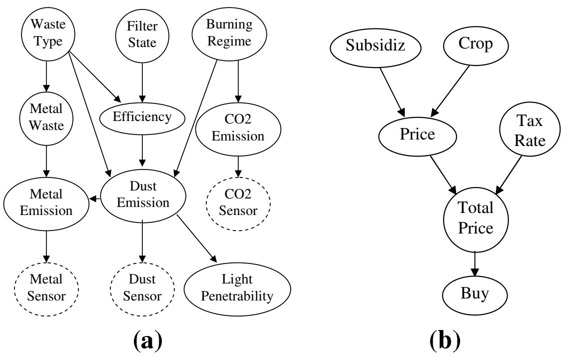


Figure 3: (a) Emission network (without dashed nodes) and augmented emission network (with dashed nodes) (b) augmented crop network.

Variable		Distribution
Subsidize (S)		(0.3,0.7)
Crop (C)		N(5, 1)
Price (P)	S	10-N(C, 1)
	$\neg S$	20-N(C, 1)
TaxRate (T)		N(0.5, 0.1)
TotalPrice (TP)		P*(1+T)
Buy (B)		(α=$\frac{exp(TP-14.0)}{1+exp(TP-14.0)}$, 1-$\alpha$)

Table 1: Parameterizations of the augmented crop network.

We observe that typically only a few steps of HLBP are necessary for HEPIS-BN to achieve a good performance, so we set the propagation length to 4. Since we only use HLBP to calculate an importance function, we need not strive for good results from HLBP. We only use 250 samples to estimate HLBP messages and two components for each MoG.

4.1 Results on the Emission Network

Some existing approaches only produce mean and variance as the solutions for the posterior probability distributions. However, we would like to note that mean and variance alone provide only limited information about the actual posterior probability distributions. Figure 4 shows the posterior probability distribution of node DustEmission when observing CO2Emission to be -1.6, Penetrability to be 0.5, and WasteType to be 1. We also plot in the same figure the corresponding normal approximation with mean 3.77 and variance 1.74. We see that the normal approximation does not reflect the true posterior. While the actual posterior distribution has a multimodal shape, the normal approximation does not tell us where the mass really is. On the contrary, LW and HEPIS-BN with 4K samples were both able to correctly estimate the shape of the posterior probability distribution, with HEPIS-BN demonstrating better performance than LW.

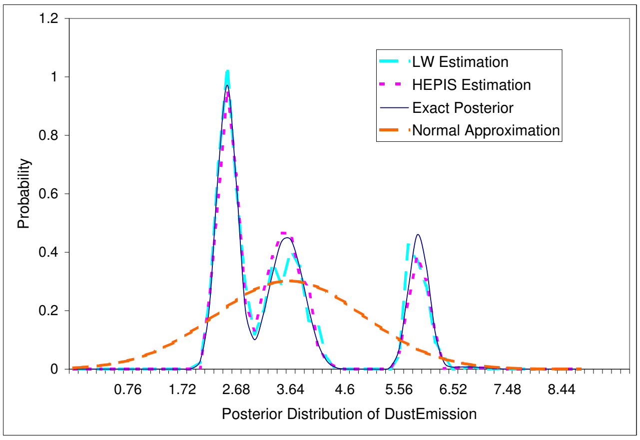
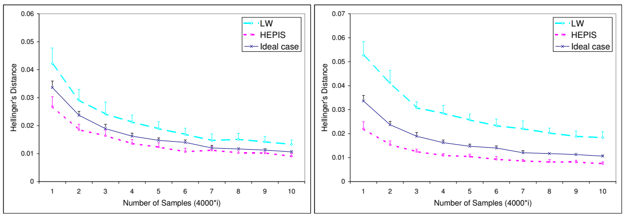


Figure 4: The posterior probability distributions estimated by LW, HEPIS-BN, normal approximation of DustEmission when observing CO2Emission to be -1.6, Penetrability to be 0.5 and WasteType to be 1 in emission network.

We compared the convergence rates and plotted the error curves of both algorithms with $40K$ samples in Figure 5(a). We also plotted the results of LW on emission network with no evidence (the ideal case for importance sampling). Although results thus obtained are not strict lower bounds, they can at least serve as an indication of the limiting precision of importance sampling on the network. We observe that HEPIS had a precision even better than the ideal case. However, since HEPIS-BN is more complicated than LW, it requires more running time, roughly twice as much as LW on the emission network in our implementation.

We also used a more unlikely evidence with CO2Emission at -0.1, Penetrability at 0.5 and WasteType at 1 to test the robustness of LW and HEPIS-BN. It is more unlikely because when we set Penetrability to 0.5 and WasteType



(a) **(b)**

Figure 5: Convergence curve comparison of LW and HEPIS-BN on emission network with evidence (a) CO2Emission= -1.6, Penetrability $= 0.5$ and WasteType $= 1$ (b) CO2Emission $= -0.1$, Penetrability $= 0.5$ and WasteType $= 1$, together with Ideal case (LW on emission with no evidence).

to 1, the posterior probability distribution of CO2Emission has a mean of -1.55. The error curves are shown in Figure 5(b). We can see that LW clearly performed worse, but HEPIS-BN seemed quite robust to the likelihood of the evidence. To get a more clear understanding, we gradually changed the observed value of CO2Emission and ran both algorithms on these cases. The results are plotted in Figure 6. We observe that LW kept deteriorating in face of unlikely evidence, while HEPIS-BN's performance was fairly stable as expected.

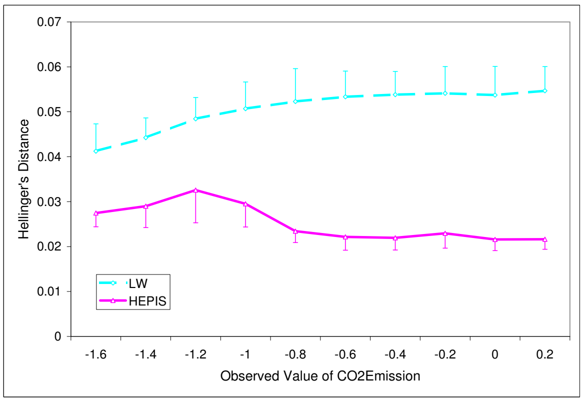


Figure 6: The influence of the observed value of CO2Emission on the precision of LW and HEPIS-BN after observing Penetrability to 0.5 and WasteType to 1 in emission network.

The emission network is only a small network. We anticipate that the advantage of HEPIS-BN will be more evident in real models, which are typically much more complex. To test the hypothesis, we transformed the emission network into a dynamic model with three slices and a total of 27 variables, and observed CO2Emission at -0.1 and Penetrability at 0.5 in the first and third slice of the model. The results of the algorithms are shown in Figure 7. While LW showed a much worse precision, the precision of HEPIS-BN was still close the that of the ideal case. Therefore, although HEPIS-BN typically requires more running time per sample, it will outperform LW on complex models. We are currently building a much more complex network modelling the university budget planning problem to test the algorithm.

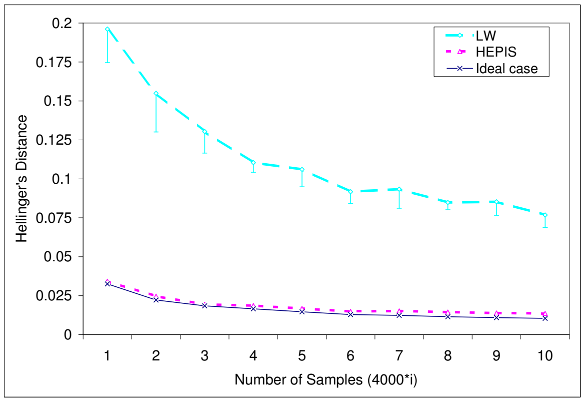
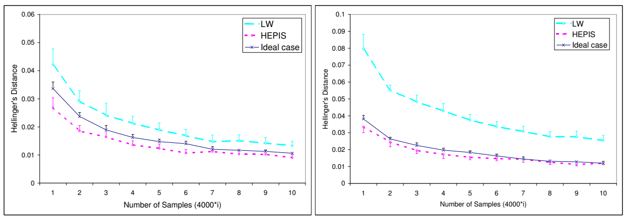


Figure 7: Error curves of LW and HEPIS-BN on the dynamic emission network.

4.2 Results on Other Networks

We also plotted the convergence results on the augmented emission network with evidence of CO2Sensor and DustSensor being true in Figure 8(a). We used the same evidence as in [12]. Again, we observe that HEPIS-BN performs better than LW.



(a) **(b)**

Figure 8: Error curves LW and HEPIS-BN (a) augmented emission network when CO2Sensor and DustSensor are observed to be true (b) crop network when totalprice is observed to be 18.0.

We discussed how to deal with deterministic nodes with evidence in importance sampling. To verify that the proposed technique works properly, we revised the crop network and added a nonlinear deterministic node TotalPrice to it (as in Figure 8(b)) and let TotalPrice to observed at 18.0. Note that the classic LW does not work in such case. We had to enhance it with the *soft arc reversal* technique in Section 3.2. We ran the same experiment as in last subsection, and plotted out the error curves in Figure 8(b). We again observed that HEPIS-BN performed better than LW on this network and its precision was comparable to the ideal case.

5 Conclusion

In this paper, we propose the HEPIS-BN algorithm, an importance sampling algorithm for general hybrid Bayesian networks that applies HLBP to calculate importance function. We also propose two novel techniques, delayed importance function generation and soft arc reversal, to deal with difficulties that are unique to hybrid models. We tested the algorithm on three benchmark hybrid models, and we observed that HEPIS-BN not only yields a much better accuracy than LW, but also was more stable in face of unlikely evidence, which make HEPIS-BN a promising approach for addressing inference tasks in much larger hybrid models.

There are two major advantages of our approach. First, it allows the maximum freedom in the representation of the hybrid Bayesian networks. It allows linear or nonlinear equations and arbitrary probability distributions and accommodates naturally the situation where discrete variables have continuous parents. Second, the algorithm directly deals with a hybrid model and guarantees to converge to the correct posterior probability distributions.

Acknowledgements

This research was supported by the Air Force Office of Scientific Research grants F49620–03–1–0187 and FA9550–06–1–0243 and by Intel Research. All experimental data have been obtained using SMILE, a Bayesian inference engine developed at the Decision Systems Laboratory and available at `http://genie.sis.pitt.edu`.

References

[1] J. Cheng and M. J. Druzdzel. BN-AIS: An adaptive importance sampling algorithm for evidential reasoning in large Bayesian networks. *Journal of Artificial Intelligence Research*, 13:155–188, 2000.

[2] B. R. Cobb and P. P. Shenoy. Nonlinear deterministic relationships in Bayesian networks. In *Proceedings of Eighth European Conference on Symbolic and Quantitative Approaches to Reasoning with Uncertainty (ECSQARU-05)*, pages 27–38, Berlin, 2005. Springer-Verlag.

[3] R. Fung and K.-C. Chang. Weighing and integrating evidence for stochastic simulation in Bayesian networks. In M. Henrion, R. Shachter, L. Kanal, and J. Lemmer, editors, *Uncertainty in Artificial Intelligence 5*, pages 209–219, New York, N. Y., 1989. Elsevier Science Publishing Company, Inc.

[4] W. Gilks, A. Thomas, and D. Spiegelhalter. A language and program for complex Bayesian modelling. *The Statistician*, 43:169–178, 1994.

[5] V. Gogate and R. Dechter. Approximate inference algorithms for hybrid Bayesian networks with discrete constraints. In *Proceedings of the 21th Annual Conference on Uncertainty in Artificial Intelligence (UAI–05)*, AUAI Press Corvallis, Oregon, 2005.

[6] C. T. Kelley. *Solving Nonlinear Equations with Newton's Method.* 2003.

[7] G. Kokolakis and P. Nanopoulos. Bayesian multivariate micro-aggregation under the Hellinger's distance criterion. *Research in official statistics*, 4(1):117–126, 2001.

[8] D. Koller, U. Lerner, and D. Angelov. A general algorithm for approximate inference and its application to hybrid Bayes nets. In *Proceedings of the Fifteenth Annual Conference on Uncertainty in Artificial Intelligence (UAI-99)*, pages 324–333. Morgan Kaufmann Publishers Inc., 1999.

[9] A. Kozlov and D. Koller. Nonuniform dynamic discretization in hybrid networks. In *Proceedings of the 13th Annual Conference on Uncertainty in Artificial Intelligence (UAI-97)*, pages 314–325, San Francisco, CA, 1997. Morgan Kaufmann.

[10] S. Lauritzen. Propagation of probabilities, means and variances in mixed graphical association models. *Journal of the American Statistical Association*, 87:1098–1108, 1992.

[11] S. Lauritzen. and F. Jensen. Stable local computation with conditional Gaussian distributions. *Statistics and Computing*, 11:191–203, 2001.

[12] U. Lerner, E. Segal, and D. Koller. Exact inference in networks with discrete children of continuous parents. In *Proceedings of the Seventeenth Annual Conference on Uncertainty in Artificial Intelligence (UAI-01)*, pages 319–328. Morgan Kaufmann Publishers Inc., 2001.

[13] U. N. Lerner. *Hybrid Bayesian Networks for Reasoning about Complex Systems.* PhD thesis, Stanford University, Computer Science Department, 2002.

[14] S. Moral, R. Rumi, and A. Salmeron. Mixtures of truncated exponentials in hybrid Bayesian networks. In *Proceedings of Fifth European Conference on Symbolic and Quantitative Approaches to Reasoning with Uncertainty (ECSQARU-01)*, pages 156–167, 2001.

[15] S. Moral and A. Salmeron. Dynamic importance sampling computation in Bayesian networks. In *Proceedings of Seventh European Conference on Symbolic and Quantitative Approaches to Reasoning with Uncertainty (ECSQARU-03)*, pages 137–148, 2003.

[16] M. G. Morgan and M. Henrion. *Uncertainty: A Guide to Dealing with Uncertainty in Quantitative Risk and Policy Analysis*. Cambridge University Press, New York, 1990.

[17] K. Murphy, Y. Weiss, and M. Jordan. Loopy belief propagation for approximate inference: An empirical study. In *Proceedings of the Fifteenth Annual Conference on Uncertainty in Artificial Intelligence (UAI–99)*, pages 467–475, San Francisco, CA, 1999. Morgan Kaufmann Publishers.

[18] K. P. Murphy. A variational approximation for Bayesian networks with discrete and continuous latent variables. In *Proceedings of the Fifteenth Annual Conference on Uncertainty in Artificial Intelligence (UAI-99)*, pages 457–466. Morgan Kaufmann Publishers Inc., 1999.

[19] R. D. Shachter and M. A. Peot. Simulation approaches to general probabilistic inference on belief networks. In M. Henrion, R. Shachter, L. Kanal, and J. Lemmer, editors, *Uncertainty in Artificial Intelligence 5*, pages 221–231, New York, N. Y., 1989. Elsevier Science Publishing Company, Inc.

[20] P. Shenoy. Inference in hybrid Bayesian networks using mixtures of gaussians. In *Proceedings of the 22nd Annual Conference on Uncertainty in Artificial Intelligence (UAI-06)*, pages 428–436, Corvallis, OR, 2006. AUAI Press.

[21] E. Sudderth, A. Ihler, W. Freeman, and A. Willsky. Nonparametric belief propagation. In *Proceedings of 2003 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR-03)*, 2003.

[22] D. Titterington, A. Smith, and U. Makov. *Statistical Analysis of Finite Mixture Distribution*. John Wiley and Sons, NY, 1985.

[23] C. Yuan and M. J. Druzdzel. An importance sampling algorithm based on evidence pre-propagation. In *Proceedings of the 19th Conference on Uncertainty in Artificial Intelligence (UAI–03)*, pages 624–631, Morgan Kaufmann Publishers San Francisco, California, 2003.

[24] C. Yuan and M. J. Druzdzel. Hybrid loopy belief propagation. In *Proceedings of the Third European Workshop on Probabilistic Graphical Models (PGM'06)*, pages 317–324, Prague, Czech Republic, 2006.

[25] C. Yuan and M. J. Druzdzel. Importance sampling algorithms for Bayesian networks: Principles and performance. *Mathematical and Computer Modelling*, 43:1189–1207, 2006.