Tractable Inference in Hybrid Bayesian Networks with Deterministic Conditionals using Re-approximations

Rafael Rumí, Antonio Salmerón
Department of Statistics and Applied Mathematics
University of Almería, Spain
{rrumi,antonio.salmeron}@ual.es

Prakash P. Shenoy
School of Business
University of Kansas, USA
pshenoy@ku.edu

Abstract

In this paper we study the problem of inference in hybrid Bayesian networks containing deterministic conditionals. The difficulties in handling deterministic conditionals for continuous variables can make inference intractable even for small networks. We describe the use of re-approximations to reduce the complexity of the potentials that arise in the intermediate steps of the inference process. We show how the idea behind re-approximations can be applied to the frameworks of mixtures of polynomials and mixtures of truncated exponentials. Finally, we illustrate our approach by solving a small stochastic PERT network modeled as an hybrid Bayesian network.

1 Introduction

Hybrid Bayesian networks are Bayesian networks (BNs) that include a mix of discrete and continuous random variables. The first proposal of an exact algorithm for hybrid Bayesian networks was developed for the case in which the conditional distributions of all continuous variables in the network are conditional linear Gaussian (CLG) (Lauritzen, 1992; Lauritzen and Jensen, 2001), and that discrete variables do not have continuous parents. This implies that the marginal distribution of each continuous variable is a *mixture of Gaussians* (MoG). Some limitations of the MoG model are the CLG assumptions for the conditionals of continuous variables, and the assumption that discrete variables do not have continuous parents.

A more general technique, based on the use of *mixtures of truncated exponentials* (MTEs), was proposed in (Moral et al., 2001). The MTE model does not impose any topological restriction on its corresponding networks, and is compatible with any efficient algorithm for exact inference that requires only the combination and marginalization operations, such as the Shenoy-Shafer (Shenoy and Shafer, 1990; Shenoy and West, 2011a) and variable elimination methods (Zhang and Poole, 1996). Furthermore, MTEs have shown a remarkable ability for fitting many commonly used univariate probability density functions (PDFs) (Cobb et al., 2006; Langseth et al., 2010).

A more recent proposal for dealing with hybrid Bayesian networks is based on the use of *mixtures of polynomials* (MOPs) (Shenoy and West, 2011b). Like MTEs, MOPs have high expressive power, but the latter are superior in dealing with deterministic conditionals for continuous variables (Shenoy and West, 2011b; Shenoy and West, 2011a). A conditional distribution for a variable is said to be *deterministic* if its variances are all zeroes for each state of its parents. Also, a MOP approximation of a PDF can be easily found using Lagrange interpolating polynomial with Chebyshev points (Shenoy,

2012). MTEs and MOPs can be seen as special cases of the recently proposed mixtures of truncated basis functions (MoTBFs) (Langseth et al., 2012).

In this paper we discuss the use of a *re-approximation* strategy for making inference tractable. The idea is based on simplifying the potentials that arise in the intermediate steps of the inference process. This simplification is achieved by reducing the number of pieces of MOP/MTE potentials, and also by reducing the degree of MOPs and the number of exponential terms of MTEs. We compare the performance of MOPs and MTEs in this context, through an example arising from a stochastic PERT network (Cinicioglu and Shenoy, 2009).

2 MTEs and MOPs

In this section we formally define the MTE and MOP models, which will be used throughout the paper. We will use uppercase letters to denote random variables, and boldfaced uppercase letters to denote random vectors, e.g. $\mathbf{X} = \{X_1, \ldots, X_n\}$, and its domain will be written as $\Omega_{\mathbf{X}}$. By lowercase letters x (or $\mathbf{x}$) we denote some element of Ω_X (or $\Omega_{\mathbf{X}}$). The MTE model (Moral et al., 2001) is defined as follows.

Definition 1. Let $\mathbf{X}$ be a mixed n-dimensional random vector. Let $\mathbf{Y} = (Y_1, \ldots, Y_d)^{\mathsf{T}}$ and $\mathbf{Z} = (Z_1, \ldots, Z_c)^{\mathsf{T}}$ be the discrete and continuous parts of $\mathbf{X}$, respectively, with $c + d = n$. We say that a function $f : \Omega_{\mathbf{X}} \mapsto \mathbb{R}_0^+$ is a *mixture of truncated exponentials (MTE) potential* if for each fixed value $\mathbf{y} \in \Omega_{\mathbf{Y}}$ of the discrete variables $\mathbf{Y}$, the potential over the continuous variables $\mathbf{Z}$ is defined as:

$$f(\mathbf{z}) = a_0 + \sum_{i=1}^{m} a_i \exp\left\{\mathbf{b}_i^{\mathsf{T}}\mathbf{z}\right\}, \qquad (1)$$

for all $\mathbf{z} \in \Omega_{\mathbf{Z}}$, where $a_i \in \mathbb{R}$ and $\mathbf{b}_i \in \mathbb{R}^c$, $i = 1, \ldots, m$. We also say that f is an MTE potential if there is a partition $D_1, \ldots, D_k$ of $\Omega_{\mathbf{Z}}$ into hypercubes and in each one of them, f is defined as in Eq. (1). In this case, we say f is a k-piece, m-term MTE potential.

Mixtures of polynomials (MOPs) were proposed as modeling tools for hybrid Bayesian networks in (Shenoy and West, 2011b). The original definition is similar to MTEs in the sense that they are piecewise functions defined on hypercubes (a more general definition is given in (Shenoy, 2012), where the hypercube condition is relaxed, however it is not used in this paper). The details are as follows.

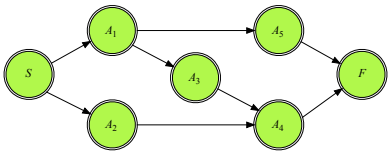

Figure 1: A PERT network with five activities.

Definition 2. Let $\mathbf{X}, \mathbf{Y}$ and $\mathbf{Z}$ be as in Def. 1. We say that a function $f : \Omega_{\mathbf{X}} \mapsto \mathbb{R}_0^+$ is a *mixture of polynomials (MOP) potential* if for each fixed value $\mathbf{y} \in \Omega_{\mathbf{Y}}$ of the discrete variables $\mathbf{Y}$, the potential over the continuous variables $\mathbf{Z}$ is defined as:

$$f(\mathbf{z}) = P(\mathbf{z}), \qquad (2)$$

for all $\mathbf{z} \in \Omega_{\mathbf{Z}}$, where $P(\mathbf{z})$ is a multivariate polynomial in variables $\mathbf{Z} = (Z_1, \ldots, Z_c)^{\mathsf{T}}$. We also say that f is a MOP potential if there is a partition $D_1, \ldots, D_k$ of $\Omega_{\mathbf{Z}}$ into *hypercubes* and in each one of them, f is defined as in Eq. (2).

3 A PERT Hybrid Bayesian Network

PERT stands for *Program Evaluation and Review Technique*, and is one of the commonly used project management techniques (Malcolm et al., 1959). PERT networks are directed acyclic networks where the nodes represent duration of activities and the arcs represent precedence constraints in the sense that before we can start any activity, all the parent activities have to be completed. The term *stochastic* refers to the fact that the duration of activities are modeled as continuous random variables.

Fig. 1 shows a PERT network with 5 activities $A_1, \ldots, A_5$. Nodes S and F represent the start and finish times of the project. The links among activities mean that an activity cannot be started until after all its predecessors have been completed. Assume we are informed that

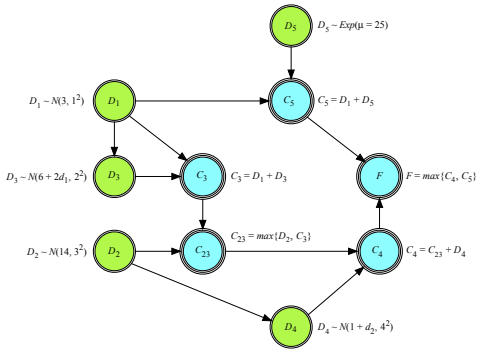

Figure 2: A Bayesian network representing the PERT network in Fig. 1.

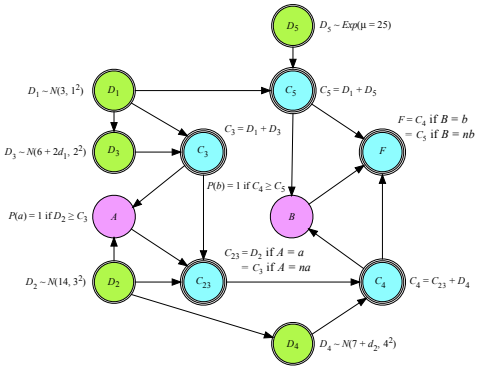

Figure 3: A hybrid Bayesian network representing the PERT network in Fig. 1. Notice that variables A and B are discrete.

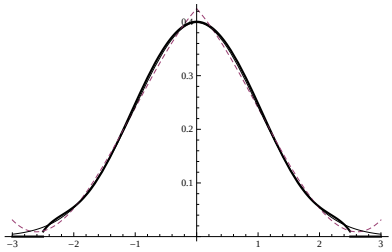

Figure 4: Comparison of the Normal PDF (black), the MTE approximation (thick-black) and the MOP approximation (dashed-black).

the durations of A_1 and A_3 are positively correlated, and the same is true with A_2 and A_4. Then, this PERT network can be transformed into a Bayesian network as described below.

Let D_i and C_i denote the duration and the completion time of the activity i, respectively. The activity nodes in the PERT network are replaced with activity completion times in the BN. Next, activity durations are added with a link from D_i to C_i, so that each activity will be represented by two nodes, its duration D_i and its completion time C_i. Notice that the completion times of the activities with no predecessors will be the same as their durations.

We assume that the project start time is zero and each activity is started as soon as all the preceding activities are completed. Thus, F represents the completion time of the project. The resulting PERT BN is given in Fig. 2.

Notice that the conditionals for the variables C_3, C_{23}, C_4, C_5 and F are deterministic. On the other hand, variables $D_1, \ldots, D_5$ are continuous random variables, and their corresponding conditional distributions are depicted next to their corresponding nodes in Fig. 2. The parameters μ (mean) and σ^2 (variance) of the normal distribution are in units of *days* and $days^2$, respectively. The parameter μ (mean) of the exponential distribution is in units of *days*.

Before solving the network, we must deal with the max function in Fig. 2. We convert this max deterministic function to a linear function as proposed in (Shenoy and West, 2011a), transforming the Bayesian network in Fig. 2 into a hybrid Bayesian network, depicted in Fig. 3.

3.1 Solving the Network

The problem of inference in hybrid BNs with deterministic conditionals has been studied in (Shenoy and West, 2011a) and a solution to this PERT network was found in (Shenoy et al., 2011), using the MTEs and MOPs paradigms. When using MOPs, we use a 2-piece 3-degree MOP approximation to the PDF of $N(0, 1)$ (Shenoy, 2012). In the case of MTEs, we use a 1-piece 7-terms MTE approximation to the PDF of $N(0, 1)$ (Langseth et al., 2010). The quality of these approximations can be seen in Fig. 4. Using these approximations, we obtain the corresponding approximations to any Gaussian distribution with parameters σ^2 and μ.

To define the conditional distributions, we followed the mixed tree approach (Moral et al., 2003) for both paradigms. For example, let $g_2(d_1, d_3)$ be the conditional distribution of

$D_3 \mid d_1$ (the distribution function associated to D_3 in Fig. 3). To define an approximation of $g_2(d_1, d_3)$ we partition the domain of D_1 into several pieces and assume that d_1 is a constant in each piece equal to the mid-point of the piece:

$$g_{2c}(d_1, d_3) = \begin{cases} f(\frac{d_3-8}{2})/2 & \text{if } 0 \leq d_1 < 2, \\ f(\frac{d_3-12}{2})/2 & \text{if } 2 \leq d_1 < 4, \\ f(\frac{d_3-16}{2})/2 & \text{if } 4 \leq d_1 \leq 6. \end{cases} \quad (3)$$

The number of pieces in which to split the domain (see Eq. (3) above) is a key point to control the computational complexity of the problem. In (Shenoy et al., 2011), the maximum number of pieces in a mixed tree approach was two, which leads to inaccurate solutions. Because we were using a re-approximation strategy (described in Section 4), the number of pieces in the mixed tree approach were increased to three, with the corresponding gain in accuracy.

The goal is to compute the marginal density for F. To that end, we choose a sequence of the remaining variables in the network, $D_5\, D_3\, D_1\, D_4\, D_2\, C_{23}\, C_3\, C_4\, C_5$, and marginalize the variables following this sequence, to compute the marginal density for F.

Without the re-approximation procedures described in this paper, no solution was obtained to this problem, due to the growth in complexity during the combination and marginalization operations.

4 Re-approximation of MOPs/MTEs

In the process of solving the network, we may get pieces for which the density defines a very low or even null probability value. Also, it may happen that the density in several contiguous pieces can be represented as a density in a single larger piece without loss in accuracy. Finally, a third source of complexity may arise from the existence of too many exponential terms (in MTEs) with low contribution to the actual density or a higher degree than necessary (in MOPs). We will describe two methods for re-approximating MOPs and MTEs, consisting of dropping pieces and re-estimating the parameters.

4.1 Re-approximation by Dropping Pieces

A simple method of re-approximation is to drop pieces that are defined on lower-dimensional regions. Since there are no probabilities associated with such pieces, dropping them causes no loss of accuracy. In some cases, there may also be pieces with very low probability associated with them. In this case, we can also drop them, as long as the total associated probability is below some threshold (e.g., 0.05) so that the loss of accuracy remains limited. It is necessary to renormalize the potentials after dropping pieces with positive probabilities.

In the solution of the PERT problem described in Sec. 3, after marginalizing D_1, we obtain a MOP denoted by $f_4(c_3, c_5)$ representing the joint PDF of C_3 and C_5, with 23 pieces, 9 of which defined on 1-dimensional regions. Thus, we can safely drop these pieces resulting in a 14-piece MOP. Further examination of the remaining 14 pieces reveals that 6 of them have very small probabilities (0.000017 to 0.026) associated with them, amounting to a total of ≈ 0.044. After dropping these 6 pieces and renormalizing the potential, we obtain an 8-piece MOP $f_{4r}(c_3, c_5)$ that can be used in place of $f_4(c_3, c_5)$. Fig. 5 shows plots of f_4 and f_{4r}.

When solving the network using MTEs, we also get the same potential $f_4(c_3, c_5)$ with 11-pieces. Of these, 4 are singleton pieces, which can be dropped without loss of accuracy. Three of the remaining pieces have very low probabilities (0.00005 to 0.019) with a total associated probability of 0.026. After dropping these three pieces and re-normalizing the density, we obtain a 4-piece MTE $f_{4s}(c_3, c_5)$ that can be used in place of $f_4(c_3, c_5)$. Fig. 6 shows f_4 and f_{4s}.

4.2 Re-approximation of MOPs using LIP with Chebyshev Points

Another method for re-approximating MOPs is to use Lagrange interpolating polynomials (LIPs) with Chebyshev points (Shenoy, 2012). To illustrate this, consider $g_6(\cdot)$, which is defined on non-singleton intervals $(0, 6)$, $(6, 9)$, $(9, 12]$, $(12, 15)$, $(15, 18]$, $(18, 21)$,

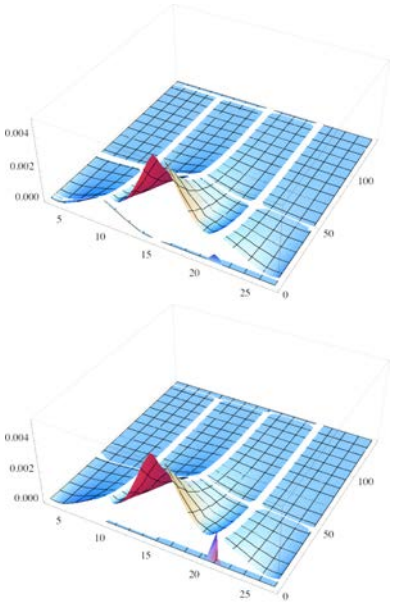

Figure 5: MOPs corresponding to f_4 (top) and f_{4r} (bottom).

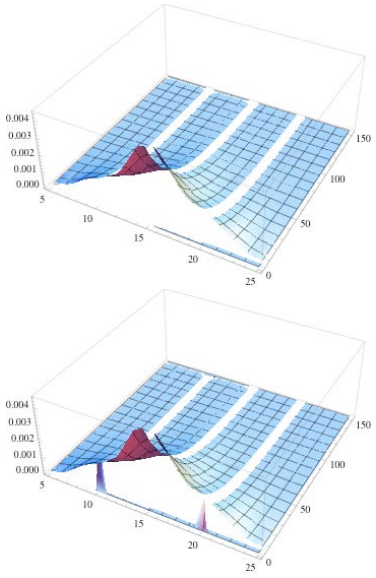

Figure 6: MTEs corresponding to f_4 (top) and f_{4s} (bottom).

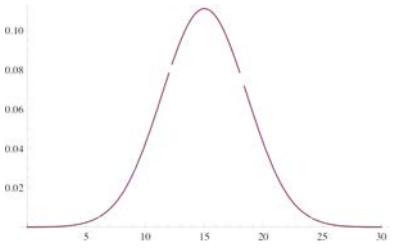

Figure 7: A graph of $g_{6r}(\cdot)$ overlaid on the graph of $g_6(\cdot)$ for MOPs.

$(21, 24)$, $(24, 30)$. We will describe how to re-approximate $g_6(\cdot)$ using just 5 pieces, on intervals $[0, 6)$, $[6, 12)$, $[12, 18)$, $[18, 24)$, $[24, 30]$. Currently, we do not have a theory for the choice of pieces, so we follow a heuristic strategy consisting of merging pieces with the restriction of keeping the sizes of the pieces approximately equal.

For each interval, we compute n Chebyshev points with n representing a tradeoff between complexity and accuracy of the resulting polynomial. A good starting choice is $n = 4$. We compute the 3-degree LIP polynomial that passes through the 4 points. We have to verify that the LIP polynomial is non-negative on the interval. If not, we increase n. If the function being approximated is non-negative over the interval, there are guarantees to find an interpolating polynomial that is non-negative for some n. This is because when we increase the number of Chebyshev points by 1, the maximum error between the LIP polynomial and the polynomial being approximated is reduced by a factor of 2. If the smallest n that results in a polynomial that is non-negative is too high, we reduce the width of the interval (i.e., use more pieces) and re-start.

Using $n = 5$ for all five pieces results in a non negative MOP for $g_6(\cdot)$. Next we normalize the resulting 5-piece, 4-degree MOP so that the total area under the MOP is 1. Let $g_{6r}(\cdot)$ denote the 5-piece, 4-degree MOP found using the above procedure. Fig. 7 shows $g_{6r}(\cdot)$ overlaid on $g_6(\cdot)$.

The LIP method applies for two or higher dimensional functions. There also exists Chebyshev points theory for two-dimensional regions (Xu, 1996). For regions in 3 or higher dimensions, we can use some extensions of the one-dimensional Chebyshev point theory. One problem with using LIPs with Chebyshev points in two or higher dimensions is non-negativity. It may be necessary to increase the degree of the

MOP potential to a very high number making computations numerically unstable. In such cases, we can resort to the idea behind mixed-tree approximations, and use a one-dimensional LIP method with Chebyshev points to approximate a two or higher dimensional function.

To illustrate this, consider $f_7(c_3, c_4)$ representing the conditional PDF of C_4 given c_3 in the PERT hybrid BN discussed in Section 3. f_7 is computed as a 49-piece, 7-degree MOP on the domain $(3 \leq c_3 \leq 27) \times (1.125 \leq c_5 \leq 137.5)$. We will approximate this potential by a 8-piece, 7-degree MOP f_{7r} as follows. Each of the 49 pieces of f_7 are re-defined on a 4-way partition of $(3 \leq c_3 \leq 27)$: $[3, 11)$, $[11, 17)$, $[17, 23)$, and $[23, 27]$. Consider the region $3 \leq c_3 < 11$. We approximate $f_7(7, c_4)$ (a 1-dimensional function, $c_3 = 7$ is the mid-point of the interval) using LIP with Chebyshev points as discussed earlier by a 2-piece, 7-degree MOP and use this approximation for the region $(3 \leq c_3 < 11) \times (11 \leq c_5 \leq 59)$. By doing this for all four pieces of the partition of the domain of C_3, we obtain a 8-piece, 7-degree MOP approximation f_{7r} of the two-dimensional MOP f_7. Fig. 8 displays f_7 and f_{7r}.

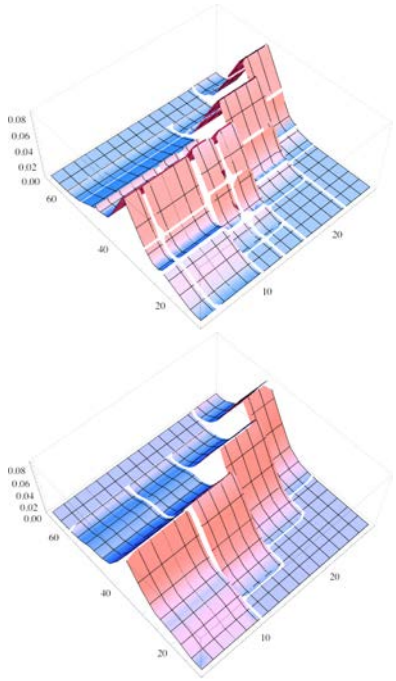

Figure 8: 3-D plots of f_7 (top) and f_{7r} (bottom) for MOPs.

4.3 Re-approximation of MTEs using Least Squares

In this section, we show how to re-approximate MTEs by reducing the number of pieces and exponential terms. The idea behind this method is similar to the one used for MOPs, but using a different mathematical tool. In this method, to re-approximate a function f, we use the command *FindFit* implemented in *Mathematica*®, which performs a numerical least-squares fit. A key point in this procedure is selecting appropriate starting values for the set of parameters to estimate. Since the MTE approximation to the Gaussian distribution is accurate, we will use the parameter estimates of this approximation as starting values in the *FindFit* method.

The steps to approximate f are as follows. 1: Divide the domain of f in a number of pieces lower than the original one. 2: For each new piece, compute the mean and standard deviation using the original density, and fit an MTE to the corresponding Gaussian density with those parameters. 3: Select a sample of equally distributed values of X in the current piece (25 values were selected in this example). 4: Build a 2-dimensional sample $(x_i, f(x_i))$. 5: Define an MTE potential in the corresponding piece, fixing the number of terms. 6: Use $(x_i, f(x_i))$ and the starting values to compute the estimates of the parameters of the MTE potential. 7: Re-normalize each piece so that the probability of the re-approximated potential is the same as the original for each piece.

Notice that this procedure is flexible. For example, if the function being re-approximate is not too complicated, we can set a different number of terms in different regions, as we actually do in this PERT network in some re-approximations. We use 6-terms for difficult approximations, usually central regions, and 2-terms for simpler regions, usually tails. Like MOPs, there is not yet a straightforward way to divide the domain, so we followed the strategy of dividing it if the shape of the function differs from Gaussian shape. This is also related to a

successful selection of the starting points, since it requires that the density in the piece being re-approximated somehow resembles a Gaussian shape, which is not always the case.

In the case of re-approximating 2 or higher dimensional potentials, we use the same procedure as MOPs, i.e., using mixed trees and fixing one dimension whilst re-approximating the function for the remaining dimensions. In the case of MTEs this is not difficult, since this is the approach selected to define the conditional distributions (parent variables can only affect the partition of the domain, not the expression of the density itself).

Such a procedure was carried out to re-approximate $f_7(c_3, c_4)$, a 54-piece potential (many of them singleton points) and between 7 and 65 terms (depending on the hypercube). The result is an approximate potential with 7 pieces, and with 6 terms for 3 of the pieces and 2 terms for the remaining 4 pieces. In Fig. 9 we can see the original MTE potential and the corresponding re-approximated potential.

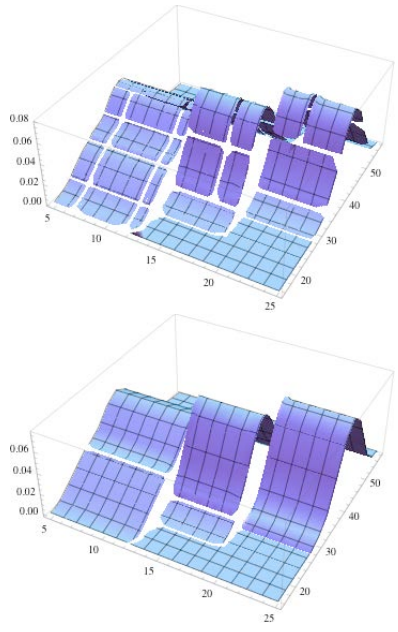

Figure 9: 3-D plots of f_7 (top) and f_{7r} approximated (bottom) for MTEs.

Notice that that the re-approximation strategy can lead to very accurate solutions, in which the complexity reduction is minimal, or to very

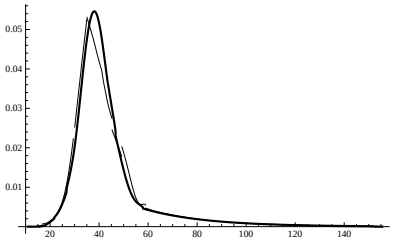

Figure 10: Plot of MOP (black) and MTE (thick-black) density for F.

smooth approximations, but with a big decrease in the complexity. It depends on the number of pieces selected and the number of exponential terms included (or degree of the polynomial in the MOP case). Finding a tradeoff between these two extremes is still an open question.

Implementing re-approximations in the elimination procedure to solve the PERT network lead to two different solutions for F. Using MOPs, we obtained a solution in 468.54 seconds, in which $E(F) = 43.44$ and $Var(F) = 246.87$. Without re-approximations we were not able to obtain a solution. Using MTEs, we obtained a solution in 481.90 seconds, in which $E(F) = 43.78$ and $Var(F) = 260.74$. Solving the same problem using MTEs without re-approximations took 1,600 seconds (Shenoy et al., 2011). Fig. 10 shows graphs of the MTE and the MOP densities for F.

5 Summary and Conclusions

The main contribution of this paper is a re-approximation strategy for making computations with MOPs and MTEs tractable in hybrid BN with deterministic conditionals. The significance of this contribution is that we can now solve some hybrid BNs that we could not solve before. The re-approximations results in a reduction in the sizes of the potentials involved in the solution process, and also results in a reduction in the complexity of the potentials (in terms of degree of the polynomials and number of exponential terms). The mathematical tools used for re-approximation are LIP and numerical least squares, and both of them can be implemented using software such

as *Mathematica*®, which broadens the potential users of hybrid BNs in communities beyond computer science and statistics.

The re-approximation strategy can serve as a basis for designing algorithms that scale up to solve large networks. Our future research will further investigate this issue. Another line of future work is to extend the re-approximation strategy by using the approximation method presented in (Langseth et al., 2012) in the context of MoTBFs.

Acknowledgments

This research has been funded in part by the Spanish Ministry of Economy and Competitiveness through projects TIN2010-20900-C04-01,02, by Junta de Andalucía through project P11-TIC-7821, by the European Regional Development Funds (FEDER), and by a sabbatical from the University of Kansas.

References

E. N. Cinicioglu and P. P. Shenoy. 2009. Using mixtures of truncated exponentials for solving stochastic PERT networks. In J. Vejnarová and T. Kroupa, editors, *Proceedings of the Eighth Workshop on Uncertainty Processing*, pages 269–283. Univ. of Economics, Prague.

B. R. Cobb, P. P. Shenoy, and R. Rumí. 2006. Approximating probability density functions with mixtures of truncated exponentials. *Statistics and Computing*, 16(3):293–308.

H. Langseth, T. D. Nielsen, R. Rumí, and A. Salmerón. 2010. Parameter estimation and model selection for mixtures of truncated exponentials. *International Journal of Approximate Reasoning*, 51(5):485–498.

H. Langseth, T. D. Nielsen, R. Rumí, and A. Salmerón. 2012. Mixtures of truncated basis functions. *International Journal of Approximate Reasoning*, 53(2):212–227.

S. L. Lauritzen and F. Jensen. 2001. Stable local computation with conditional Gaussian distributions. *Statistics and Computing*, 11(2):191–203.

S. L. Lauritzen. 1992. Propagation of probabilities, means and variances in mixed graphical association models. *Journal of the American Statistical Association*, 87(420):1098–1108.

D. G. Malcolm, J. H. Roseboom, C. E. Clark, and W. Fazar. 1959. Application of a technique for research and development program evaluation. *Operations Research*, 7(5):646–669.

S. Moral, R. Rumí, and A. Salmerón. 2001. Mixtures of truncated exponentials in hybrid Bayesian networks. In S. Benferhat and P. Besnard, editors, *Symbolic and Quantitative Approaches to Reasoning with Uncertainty*, Lecture Notes in Artificial Intelligence 2143, pages 135–143. Springer, Berlin.

S. Moral, R. Rumí, and A. Salmerón. 2003. Approximating conditional MTE distributions by means of mixed trees. In T. D. Nielsen and N. L. Zhang, editors, *Symbolic and Quantitative Approaches to Reasoning with Uncertainty*, Lecture Notes in Artificial Intelligence 2711, pages 173–183. Springer, Berlin.

P. P. Shenoy and G. Shafer. 1990. Axioms for probability and belief function propagation. In R. D. Shachter, T. S. Levitt, J. F. Lemmer, and L. N. Kanal, editors, *Uncertainty in Artificial Intelligence 4*, pages 169–198. North Holland, Amsterdam.

P. P. Shenoy and J. C. West. 2011a. Extended Shenoy-Shafer architecture for inference in hybrid Bayesian networks with deterministic conditionals. *International Journal of Approximate Reasoning*, 52(6):805–818.

P. P. Shenoy and J. C. West. 2011b. Inference in hybrid Bayesian networks using mixtures of polynomials. *International Journal of Approximate Reasoning*, 52(5):641–657.

P. P. Shenoy, R. Rumí, and A. Salmerón. 2011. Some practical issues in inference in hybrid Bayesian networks with deterministic conditionals. In S. Ventura et al., editor, *Proceeding of the 11th International Conference on Intelligent Systems Design and Applications*, pages 605–610. IEEE Research, Piscataway, NJ.

P. P. Shenoy. 2012. Two issues in using mixtures of polynomials for inference in hybrid Bayesian networks. *International Journal of Approximate Reasoning*, 53(5):847–866.

Y. Xu. 1996. Lagrange interpolation on Chebyshev points of two variables. *Journal of Approximation Theory*, 87(2):220–238.

N. L. Zhang and D. Poole. 1996. Exploiting causal independence in Bayesian network inference. *Journal of Artificial Intelligence Research*, 5:301–328.