EVALUATION OF HYBRID BAYESIAN NETWORKS USING ANALYTICAL DENSITY REPRESENTATIONS

Oliver C. Schrempf ^{*,1} **Uwe D. Hanebeck** *

* *Universität Karlsruhe*
Institute of Computer Design and Fault Tolerance
Kaiserstrasse 12
76128 Karlsruhe, Germany
e-mail: schrempf@ira.uka.de, Uwe.Hanebeck@ieee.org

Abstract: In this article, a new mechanism is described for modeling and evaluating Hybrid Dynamic Bayesian networks. The approach uses Gaussian mixtures and Dirac mixtures as messages to calculate marginal densities. As these densities are approximated by means of Gaussian mixtures, any desired precision is possible.
The presented approach removes the restrictions of sample based evaluation of Bayesian networks since it uses an analytical approximation scheme for probability densities which systematically minimizes the distance between the exact and the approximate density.

Keywords: Probabilistic models, Hybrid systems, Nonlinear systems

1. INTRODUCTION

The application of Bayesian networks is evolving since their first appearance in (Pearl, 1988). Their stochastic foundation provides a method for building models for systems with an uncertain behavior. Such a model is usually found by identifying parts of the system which can be represented by random variables. Joint probabilities over these random variables are then used to represent the system's behavior.
The term *random variable* is usually used for scalar values only. For the sake of simplicity this paper only deals with scalar values, but it is easy to extend the presented approach to vector valued systems.

Bayesian networks are considered to be an efficient representation of joint probabilities, exploiting the causal background of a domain. This is achieved by representing the causal structure of a domain by means of a directed acyclic graph (DAG). Each random variable is depicted by a node in this graph and every edge stands for a direct dependency between two variables. Probabilistically, this dependency is expressed by means of a likelihood function. Bayesian networks have the big advantage, that not all possible combinations of variables and their states have to be addressed to represent the joint probability. It is sufficient to consider the conditional densities of the variables given their parents in the graph.

The first Bayesian networks were limited to discrete domains and their conditional densities were modeled by conditional tables. Pearl's approach to evaluate the network by means of message passing (Pearl, 1988) was extended for continuous networks in (Driver and Morrell, 1995). They used Gaussian mixtures, which are sums of weighted Gaussian densities, to approximate the likelihood functions and to represent the messages.

The treatment of hybrid Bayesian networks today is mainly influenced by the articles (Lauritzen, 1992; Lauritzen and Wermuth, 1989; Olesen, 1993), which use so called cg-potentials. The drawback of this approach is the mere use of the first two moments

[1] This work was supported in part by the German Research Foundation (DFG) within the Collaborative Research Center SFB 588 on "Humanoid robots – learning and cooperating multimodal robots".

(mean and variance) to characterize continuous densities. Another problem is the explicit avoidance of discrete nodes as children of continuous parents. There have been attempts to remove this restriction by using sigmoid-functions (Murphy, 1999). This approach was picked up to include it into Lauritzen's mechanism (Lerner *et al.*, 2001), but again, accuracy is limited to the first two moments of the densities.

Dynamic Bayesian networks are an extention to model the evolution of a static Bayesian Network over time. This is done by representing the network at several time slices and connecting the network of time step t with the network in step t+1 via edges pointing from t to t+1.

A rarely considered problem in the context of Bayesian networks is the treatment of nonlinear dependencies between variables. The possibility of approximating the likelihood functions induced by nonlinear dependencies using Gaussian mixtures is offered in (Driver and Morrell, 1995).

A problem in dynamic as well as in large static networks is the increasing complexity of continuous densities, while they are propagated through the network. Sampling methods like particle filters (Murphy, 2002) are often used to overcome this problem. The drawback of these methods is that it is not known a priori how many samples are needed to approximate the densities and no distance between the true and the approximated density can be given. This paper uses a progressive approach (Hanebeck *et al.*, 2003) for approximating the densities by means of Gaussian mixtures. This has the advantage that accuracy (distance between densities and its approximations) can be traded for complexity (number of mixture components) and vice versa.

The remainder of this paper is structured as follows. The next section gives a formulation of the considered problem. Section 3 presents new formulations for hybrid conditional density functions. In section 4 accordingly adapted message representations are shown. The progressive approximation scheme is treated in section 5.

2. PROBLEM FORMULATION

The application of hybrid Bayesian networks requires the simultaneous treatment of continuous and discrete random variables. Hence, a compatible representation for densities in both cases has to be found. This work is concerned with the representation of density functions in hybrid systems and the corresponding hybrid conditional densities used in the likelihood functions. Furthermore, the propagation of densities through the hybrid network will be treated.

Recent works concerning hybrid Bayesian networks use only the first two moments to describe continuous densities. Unfortunately, taking into account only the first two moments yields a drawback, since there exist many densities which have identical first moments. This can be seen in figure 1 for the functions $f_1(x) = N(x, 0, 1)$ and $f_2(x) = \frac{1}{2}N\left(x, -a, \sqrt{1-a^2}\right) + \frac{1}{2}N\left(x, a, \sqrt{1-a^2}\right)$ where

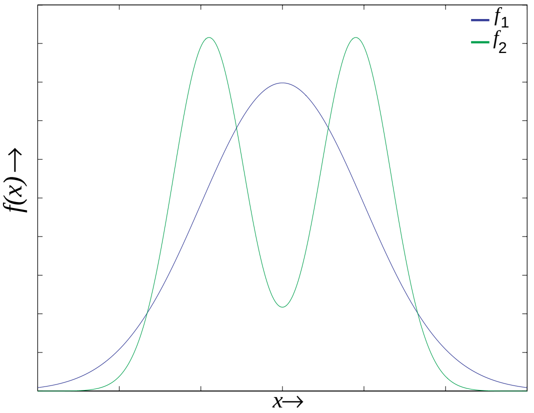

Fig. 1. Two densities $f_1(x) = N(x, 0, 1)$ and $f_2(x) = \frac{1}{2}N\left(x, -a, \sqrt{1-a^2}\right) + \frac{1}{2}N\left(x, a, \sqrt{1-a^2}\right)$ with identical first and second moment. Here for $a = 0.95$.

$$N(x, \mu, \sigma) = \frac{1}{\sqrt{2\pi}\sigma} \exp\left\{-\frac{1}{2}\frac{(x-\mu)^2}{\sigma^2}\right\}$$

is a Gaussian density with mean μ and standard deviation σ. Both densities, f_1, and f_2 have 0 as first and 1 as second moment independent of $a \in [0, 1)$. Obviously, these densities cannot be distinguished by their first two moments. Hence, this work uses a full density representation provided by Gaussian mixtures.

The simultaneous treatment of continuous and discrete variables used in this paper considers two distinct cases, which are shown in figure 2. The nodes in box shape are discrete whereas the continuous nodes have a round outline. For the parent nodes $\mathbf{u}_1, \ldots, \mathbf{u}_m$ and the child nodes $\mathbf{y}_1, \ldots, \mathbf{y}_m$ we assume a partition into continuous ($\mathbf{u}_1, \ldots, \mathbf{u}_n$ or $\mathbf{y}_1, \ldots, \mathbf{y}_n$) and discrete ($\mathbf{u}_{n+1}, \ldots, \mathbf{u}_m$ or $\mathbf{y}_{n+1}, \ldots, \mathbf{y}_m$) variables.

Creating hybrid Bayesian Networks requires hybrid conditional densities to capture the relationship between continuous and discrete variables. These densities describe the probability of a continuous or discrete random variable, depending on the state of a set of mixed parent variables.

Since this new approach is based on message passing, the message schemes known from purely discrete (Pearl, 1988) or purely continuous (Driver and Morrell, 1995) approaches must be extended for the use in hybrid networks. This is due to the fact that messages from continuous variables travel directly to discrete successors and vice versa. Hence, a new representation is needed, allowing the simultaneous treatment of continuous and discrete densities.

Updating the marginal densities in the network requires the multiplication of incoming message densities. Hence, the exact calculation leads to increasingly complex densities. For Gaussian mixtures this means that the number of components grows. To keep

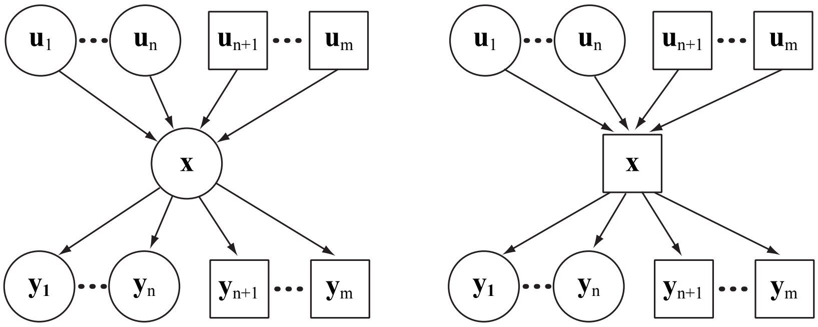

Fig. 2. The simultaneous treatment of continuous and discrete variables requires the consideration of two distinct cases. The nodes in box shape are discrete, whereas the continuous nodes have a round outline. Hence, the left part of this figure shows the continuous and the right part shows the discrete case.

the complexity at a tractable level, the densities are systematically approximated by means of Gaussian mixtures with less components but with respect to a distance measure. This is done by application of the progressive Bayes framework (Hanebeck *et al.*, 2003). A more precise introduction to this approach is given in section 5.

3. HYBRID CONDITIONAL DENSITIES

A hybrid conditional density $f(x|u_1, \ldots, u_m)$ is given as defined in (Schrempf and Hanebeck, 2004) by

$$f(x|u_1, \ldots, u_m) = \sum_{k_{n+1}=1}^{|\mathbf{u}_{n+1}|} \cdots \sum_{k_m=1}^{|\mathbf{u}_m|} \left(\prod_{i=n+1}^{m} \delta(u_i - k_i) \right) f^*(x|u_1, ..., u_n).$$

This formulation contains a single continuous conditional density $f^*(x|u_1, \ldots, u_n)$ for each joint discrete state $(u_{n+1}, \ldots, u_m)$ of $\mathbf{x}$'s discrete predecessors. The asterisk is an abbreviation indicating the dependency on $(k_{n+1}, \ldots, k_m)$. The number of states of a discrete variable $\mathbf{u}_i$ is indicated by $|\mathbf{u}_i|$. $\delta()$ is a Dirac delta function, which is used here to select the appropriate $f^*()$ for each joint discrete state.

The conditional densities $f^*(x|u_1, \ldots, u_n)$ used in this paper are modeled by means of Gaussian mixtures in the continuous case and as sum over Gaussians and Dirac pulses in the case that $\mathbf{x}$ is discrete. This means we have a single Gaussian for each continuous parent variable and another Gaussian or sum of weighted Dirac pulses depending if $\mathbf{x}$ is continuous or discrete. This is

$$f_c^*(x|u_1, \ldots, u_n) = \sum_{j=1}^{M^*} \alpha_j^* N(x, \mu_{x,j}^*, \sigma_j^*) \cdot N(u_1, \mu_{u_1,j}^*, \sigma_{u_1,j}^*) \cdot \ldots \cdot N(u_n, \mu_{u_n,j}^*, \sigma_{u_n,j}^*)$$

in the continuous and

$$f_d^*(x|u_1, \ldots, u_n) = \sum_{j=1}^{M^*} \alpha_j^* \left(\sum_{l_j=1}^{|\mathbf{x}|} p_{l_j}^* \delta(x - l_j) \right) \cdot N(u_1, \mu_{u_1,j}^*, \sigma_{u_1,j}^*) \cdot \ldots \cdot N(u_n, \mu_{u_n,j}^*, \sigma_{u_n,j}^*)$$

in the discrete case.

4. MESSAGES IN A HYBRID NETWORK

Measurements for variables in the network provide information for other (unobserved) variables. This information travels to an arbitrary node $\mathbf{x}$ in the network by means of message densities $\pi_{\mathbf{x}}(u_i)$ from the parent nodes and $\lambda_{\mathbf{y}_j}(x)$ from the child nodes. Continuous parents send their messages as Gaussian mixture densities

$$\pi_{\mathbf{x}c}(u_i) = \sum_{l_i=1}^{M_i} w_{l_i}^{(i)} N(u_i; \mu_{l_i,\pi}^{(i)}, \sigma_{l_i,\pi}^{(i)}) \ ,$$

whereas discrete parents send a sum of weighted Dirac pulses

$$\pi_{\mathbf{x}d}(u_i) = \sum_{l_i=1}^{|\mathbf{u}_i|} p_{l_i}^{(i)} \delta(u_i - l_i) \ .$$

The message from a continuous child $\mathbf{y}_i$ is again a Gaussian mixture

$$\lambda_{\mathbf{y}_i c}(x) = \sum_{l_i=1}^{M_i} w_{l_i}^{(i)} N(x; \mu_{l_i,\lambda}^{(i)}, \sigma_{l_i,\lambda}^{(i)}) \ .$$

In the case that there was no measurement for $\mathbf{y}_i$ or in the part of the network below this node, the message is set to

$$\lambda_{\mathbf{y}_i c}(x) = 1 \ .$$

The message from a discrete child is a sum of weighted Dirac pulses

$$\lambda_{\mathbf{y}_i d}(x) = \sum_{l_i=1}^{|\mathbf{x}|} p_{l_i}^{(i)} \delta(x - l_i)$$

If there is no measurement available from this part of the network, the message of $\mathbf{y}_i$ is set to 1, causing no update for $\mathbf{x}$.

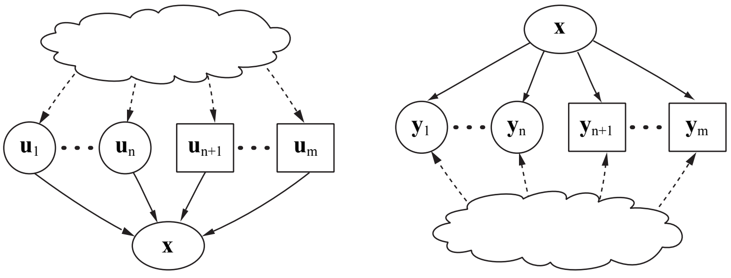

Fig. 3. The probability density over $\mathbf{x}$ is updated according to the information coming from the network above and below $\mathbf{x}$.

4.1 Density Update

Depending on the measured values for observable variables in the network, the probability density over the unobserved variables has to be updated. The probability density over an arbitrary node $\mathbf{x}$ for instance, is updated according to the measurements made in the upper part of the network and the measurements made in the part of the network below $\mathbf{x}$. This is shown in figure 3.

The density over an arbitrary variable $\mathbf{x}$ in the network depending on the the measured values is calculated as

$$\begin{aligned} f(x) &= \alpha f(x|\hat{u}_1, \ldots, \hat{u}_m) f(\hat{y}_1, \ldots, \hat{y}_m|x) \\ &= \alpha\, \pi(x)\, \lambda(x) \ , \end{aligned} \tag{1}$$

where α is a normalization constant. $\pi(x)$ and $\lambda(x)$ are introduced as abbreviations.

The information from the upper part of the net can be written as

$$\pi(x) = \int\limits_{-\infty}^{\infty} \cdots \int\limits_{-\infty}^{\infty} f(x|u_1, \ldots, u_m) \cdot \prod_{j=1}^{m} \pi_{\mathbf{x}}(u_j)\, \mathrm{d}u_1 \cdots \mathrm{d}u_m \ .$$

This calculates the marginal density over $\mathbf{x}$ given the message provided by every predecessor, weighted by the likelihood of $\mathbf{x}$.

Inserting the definitions from above and simplifying the formula yields

$$\begin{aligned} \pi(x) = \sum_{k_{n+1}=1}^{|\mathbf{u}_{n+1}|} \cdots \sum_{k_m=1}^{|\mathbf{u}_m|} \left(\prod_{i=n+1}^{m} \Pr(\mathbf{u}_i = k_i) \right) \\ \cdot \int\limits_{-\infty}^{\infty} \cdots \int\limits_{-\infty}^{\infty} f^*(x|u_1, \ldots, u_n) \cdot \\ \prod_{j=1}^{n} \pi_{\mathbf{x}c}(u_j)\, \mathrm{d}u_1 \cdots \mathrm{d}u_n \ . \end{aligned} \tag{2}$$

$\Pr(\mathbf{u}_i = k_i)$ is the probability, that variable $\mathbf{u}_i$ is in state k_i. (2) is equal for both continuous and discrete variables $\mathbf{x}$. To make the distinction between continuous or discrete $\mathbf{x}$, $f^*(x|u_1, \ldots, u_n)$ has to be chosen accordingly.

Hence, for a continuous $\mathbf{x}$ we receive the message

$$\pi_c(x) = \sum_{k_{n+1}=1}^{|\mathbf{u}_{n+1}|} \cdots \sum_{k_m=1}^{|\mathbf{u}_m|} \sum_{t=1}^{M^*} \gamma_t^* N(x; \mu_{x,t}^*, \sigma_t^*) \ ,$$

which is a Gaussian mixture density with the weights

$$\begin{aligned} \gamma_t^* = \alpha_t^* \left(\prod_{i=n+1}^{m} \Pr(\mathbf{u}_i = k_i) \right) \cdot \\ \prod_{j=1}^{n} \sum_{l_j=1}^{M_j} w_{l_j}^{(j)} N_{u_j} \left(\mu_t^*; \mu_{l_j,\pi}^{(j)}, \sigma_t^* + \sigma_{l_j,\pi}^{(j)} \right) \end{aligned} \tag{3}$$

The term $N_{u_j}(\mu_t^*; \mu_{l_j,\pi}^{(j)}, \sigma_t^* + \sigma_{l_j,\pi}^{(j)})$ describes a Gaussian density over u_j with mean $\mu_{l_j,\pi}^{(j)}$ and variance $\sigma_t^* + \sigma_{l_j,\pi}^{(j)}$ evaluated at μ_t^*.

In the case that $\mathbf{x}$ is discrete, the message from the upper part of the net is

$$\pi_d(x) = \sum_{k_{n+1}=1}^{|\mathbf{u}_{n+1}|} \cdots \sum_{k_m=1}^{|\mathbf{u}_m|} \sum_{t=1}^{M^*} \gamma_t^* \left(\sum_{h_t=1}^{|\mathbf{x}|} p_{h_t}^* \delta(x - h_t) \right)$$

with the same weights γ_t^* as in (3). This message is a sum of weighted Dirac pulses.

The message from the lower part of the net is written as

$$\lambda(x) = \prod_{j=1}^{m} \lambda_{\mathbf{y}_j}(x) \ ,$$

which is a product of the single messages coming from every child node of $\mathbf{x}$. In the case that $\mathbf{x}$ is continuous, this is again a mixture of Gaussians according to

$$\lambda_c(x) = \sum_{l_0=1}^{M_0} w'_{l_0} N(x; \mu_{l_0,\lambda}, \sigma_{l_0,\lambda})$$

with $w'_{l_0} = \prod_{j=1}^{m} w_{l_j}^{(j)}$. In the discrete case we have a product over sums of weighted Dirac pulses

$$\lambda_d(x) = \prod_{j=1}^{m} \sum_{l_j=1}^{|\mathbf{x}|} p_{l_j}^{(j)} \delta(x - l_j) \ .$$

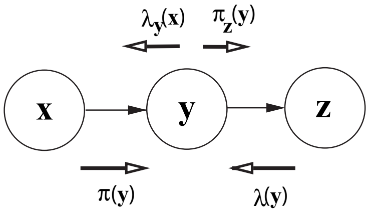

Fig. 4. Messages flow into $\mathbf{y}$ to update its density. Then $\mathbf{y}$ sends back messages to the network.

The density function for a continuous or discrete $\mathbf{x}$ can now be obtained by multiplying the appropriate π- and λ-messages.

4.2 *Calculating the Messages to be sent by an Updated Node*

Every node receiving messages from its neighbor sends out messages to the other neighbors as well. It sends π-messages to its children and λ-messages to its parents as depicted in figure 4. These messages are calculated according to (Schrempf and Hanebeck, 2004) and result in Gaussian mixtures for continuous and Dirac mixtures for discrete variables.

The π-Messages

The message $\pi_{\mathbf{y}_i}(x)$ that a node $\mathbf{x}$ sends to its i-th successor is calculated as

$$\begin{aligned}\pi_{\mathbf{y}_i}(x) &= f(x|\hat{u}_1, \ldots, \hat{u}_m, \hat{y}_1, \ldots, \hat{y}_{i-1}, \hat{y}_{i+1}, \ldots, \hat{y}_m) \\ &= \alpha\pi(x)\lambda(x)|_{\lambda_{\mathbf{y}_i(x)=1}} .\end{aligned}$$

This means that all messages excluding the one coming from $\mathbf{y}_i$ are passed ahead. Hence, this message can be calculated as shown in section 4.1 under the assumption $\lambda_{\mathbf{y}_i}(x) = 1$.

The λ-Messages

The calculation of the λ-messages is a little more tricky since these messages travel against the direction of the modeled dependency $f(x|u_i)$. Depending on the continuous or discrete identity of the parent variable, the message sent by $\mathbf{x}$ is a Gaussian mixture density or a sum of weighted Dirac pulses as shown in table 1. The main information of these messages is carried by their weight vectors $\psi^*_{j,i}$ and $\eta^i_{l_i}$ which are calculated in different ways for continuous or discrete $\mathbf{x}$.

Table 1. The λ-messages from $\mathbf{x}$ to its parent differs for continuous or discrete $\mathbf{u}_i$.

$\mathbf{u}_i$	$\lambda_{\mathbf{x}}(u_i)$
cont.	$\sum_{l_{n+1}=1}^{\lvert\mathbf{u}_{n+1}\rvert} \cdots \sum_{l_m=1}^{\lvert\mathbf{u}_m\rvert} \sum_{j=1}^{M^*} \psi^*_{j,i} N(u_i, \mu^*_{u_i,j}, \sigma^*_j)$
disc.	$\sum_{l_i=1}^{\lvert\mathbf{u}_i\rvert} \delta(u_i - l_i) \cdot \eta^i_{l_i}$

4.3 *Boundary Conditions*

If $\mathbf{x}$ is a root node for which no measurement is available, its π-message is set to be the prior density for that node. This is a Gaussian mixture density for a continuous $\mathbf{x}$ and a sum of weighted Dirac pulses for a discrete $\mathbf{x}$.

If $\mathbf{x}$ is a leaf node that has not been observed so far, its λ-message is set to 1. Hence the density for this node is calculated as $f(x) = \pi(x)$.

Exact measurements $\mathbf{x} = \hat{x}$ are represented by $\lambda(x) = \delta(x - \hat{x}) = N(x, \hat{x}, 0)$. This implies $f(x) = \hat{x}$.

5. PROGRESSIVE APPROXIMATION

To tackle the problem of the increasing number of mixture components described in section 2 we reapproximate the mixtures resulting from the product of mixtures. The approach described in (Hanebeck *et al.*, 2003) provides a framework for approximating arbitrary densities by means of a Gaussian mixture density. It estimates an optimal set of parameters $\boldsymbol{\eta}$ consisting of mixture weights w, means μ and variances σ for all components in the mixture

$$\boldsymbol{\eta} = \left[w^{(1)},\, \mu^{(1)},\, \sigma^{(1)},\, \cdots\, w^{(L)},\, \mu^{(L)},\, \sigma^{(L)}\right]^T .$$

This vector of parameters is approximated optimally in the sense of minimizing a distance measure between the exact and the approximate density. In our case, the exact density is a product of two Gaussian mixtures. To achieve this, instead of approximating the exact density directly, the approach starts with a tractable density that can be approximate with no error. This density is continuously transformed into the exact density via intermediate densities. To realize this progression, a parameter γ is introduced into the exact density which varies from 0 to 1. While γ approaches 1, the parameter vector $\boldsymbol{\eta}$ is adjusted infinitesimally, to keep the distance between the parameterized and the approximated density at the minimum. Since the approach starts at the global minimum, it is guaranteed that the global minimum is reached for $\gamma = 1$.

As a measure of deviation between the true density and its approximation, we use the squared integral distance

$$G(\boldsymbol{\eta}, \gamma) = \frac{1}{2}\int_{\mathbb{R}} \left(\tilde{f}(x,\gamma) - f(x,\boldsymbol{\eta})\right)^2 \mathrm{d}x \ ,$$

where $\tilde{f}(x,\gamma)$ is the parameterized true density and $f(x,\boldsymbol{\eta})$ is the approximation density.

For $\gamma = 0$, the parameterized density can easily be approximated by a Gaussian mixture or is already given as Gaussian mixture. Since for this work the product of two arbitrary Gaussian mixture densities $f_1(x)$ and $f_2(x)$ has to be approximated by means of a Gaussian mixture, we choose the parameterization $\tilde{f}(x,\gamma) = f_1(x) \cdot f_2(x,\gamma)$ with

$$f_2(x,\gamma) = \sum_{i=1}^{L} w_i \exp\left(\frac{1}{2}\frac{(x-\mu_i)^2}{\frac{1+\epsilon}{\gamma+\epsilon}\sigma_i}\right) \ ,$$

where ϵ is a small constant. Hence, for $\gamma = 0$, $f_2(x)$ is close to 1 for all x and $\tilde{f}(x,\gamma) = f_1(x)$. For $\gamma = 1$, the desired function $\tilde{f}(x,\gamma) = f_1(x) \cdot f_2(x)$ is approached.

To obtain a progression from $\gamma = 0$ to $\gamma = 1$ while keeping the distance at the minimum we first take the partial derivative of the distance measure $G(\boldsymbol{\eta},\gamma)$ with respect to the parameter vector $\boldsymbol{\eta}$ and setting the result to zero

$$\frac{\partial G}{\partial \boldsymbol{\eta}} \overset{!}{=} 0 \ .$$

The partial derivative with respect to the progression parameter γ yields a system of explicit ordinary first-order differential equations

$$\mathbf{b}(\boldsymbol{\eta},\gamma) = \mathbf{P}(\boldsymbol{\eta})\dot{\boldsymbol{\eta}}$$

with coefficients given by

$$\mathbf{b}(\boldsymbol{\eta},\gamma) = \int_{\mathbb{R}} \frac{\partial \tilde{f}(x,\gamma)}{\partial \gamma} \frac{\partial f(x,\boldsymbol{\eta})}{\partial \boldsymbol{\eta}} \,\mathrm{d}x$$

and

$$\mathbf{P}(\boldsymbol{\eta}) = \int_{\mathbb{R}} \left(\frac{\partial f(x,\boldsymbol{\eta})}{\partial \boldsymbol{\eta}}\right) \left(\frac{\partial f(x,\boldsymbol{\eta})}{\partial \boldsymbol{\eta}}\right)^T \mathrm{d}x \ .$$

This system of explicit ordinary first-order differential equations can be solved on the interval $0 \ldots 1$ by any appropriate solver, e.g. Euler or Runge-Kutta.

The framework further provides a method for structural adaptation by means of splitting and removing mixture components, which we omit here for the sake of brevity.

6. CONCLUSIONS

This paper presented a new approach for modeling and evaluating hybrid Bayesian networks, that allows arbitrary combinations of continuous and discrete variables while making no restrictions on their ancestry. This is achieved by a unified notation for the corresponding likelihood functions. Even nonlinear dependencies between variables can be modeled by following this approach.

Due to the application of full densities as messages instead of only their first two moments, the proposed approach gains accuracy. The application of full densities is essential since it is not possible to reconstruct an arbitrary density solely from its first two moments. Furthermore, a drawback of sample based approaches is overcome. We can give the distance between the exact and the approximate density according to a measure of deviation.

Since the exact evaluation of such networks is computationally not tractable, the proposed approximation scheme offers a good tradeoff between accuracy and complexity.

One drawback of the proposed approach is the limitation to singly connected graphs. One possible way to overcome this is to find a method to do clustering in the graph. Lauritzen uses clustering techniques when building junction trees, but this results in collapsing the densities to the first two moments. To preserve the accuracy of the approach presented here, the persistence of the full density structures must be guaranteed.

The presented approach has already been used in the context of intention recognition in the robotics domain and showed good results. Especially the freedom in modeling is of great value.

REFERENCES

Driver, E. and D. Morrell (1995). Implementation of Continuous Bayesian Networks Using Sums of Weighted Gaussians. In: *Proceedings of the* 11^{th} *Conference on Uncertainty in Artificial Intelligence* (Besnard and Hanks, Eds.). Montreal, Quebec, Canada. pp. 134–140.

Hanebeck, U. D., K. Briechle and A. Rauh (2003). Progressive Bayes: A New Framework for Nonlinear State Estimation. In: *Proceedings of SPIE*. Vol. 5099. Orlando, Florida. pp. 256–267. AeroSense Symposium.

Lauritzen, Steffen L. (1992). Propagation of Probabilities, Means and Variances in Mixed Graphical Association Models. *Journal of the American Statistical Association* **87**(420), 1098–1108.

Lauritzen, Steffen L. and Nanny Wermuth (1989). Graphical Models for Associations between Variables, some of which are Qualitative and some Quantitative. *The Annals of Statistics* **17**(1), 31–57.

Lerner, Uri, Eran Segal and Daphne Koller (2001). Exact Inference in Networks with Discrete Children of Continuous Parents. In: *Proceedings of the* 17^{th} *Conference on Uncertainty in Artificial Intelligence*. Seattle, Washington, USA. pp. 319–238.

Murphy, Kevin Patrick (1999). A Variational Approximation for Bayesian Networks with Discrete and Continuous Latent Variables. In: *Proceedings of the* 15^{th} *Conference on Uncertainty in Artificial Intelligence* (K. Blackmond-Laskey and H. Prade, Eds.). Stockholm, Sweden. pp. 457–466.

Murphy, Kevin Patrick (2002). Dynamic Bayesian Networks. http://www.ai.mit.edu/∼murphyk. To appear in Probabilistic Graphical Models (M. Jordan ed.).

Olesen, Kristian G. (1993). Causal Probabilistic Networks with Both Discrete and Continuous Variables. *IEEE Transactions on Pattern Analysis and Machine Intelligence* **15**(3), 275–279.

Pearl, Judea (1988). *Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference*. Morgan Kaufmann.

Schrempf, O. C. and U. D. Hanebeck (2004). A New Approach for Hybrid Bayesian Networks Using Full Densities. In: *Proceedings of* 6^{th} *Workshop on Computer Science and Information Technologies, CSIT 2004*. Budapest, Hungary.