Non-linear modeling of a production process by hybrid Bayesian Networks [1]

Rainer Deventer and **Joachim Denzler** and **Heinrich Niemann** [2]

Abstract. This paper shows how non-linear functions can be approximated by hybrid Bayesian networks. The basic idea is to make a piecewise linear approximation with several base points. This approach is applied to an engineering domain and the accuracy is compared to Gibbs sampling. Great accuracy is shown even at non-continuous functions. Due to the general underlying principle, it is possible to adapt this type of network to other domains.

1 INTRODUCTION

Bayesian networks are a flexible mean for modeling the relationship between different variables and several training algorithms are available for them, so that they are used e. g. in expert systems and for the purpose of data mining.

In the work done at our institute Bayesian networks (BN) are used for modeling of production processes. The aim is to model the dependency between input and output parameters and using that model to calculate input parameters that guarantee an optimal output with a maximal probability. This can be done by entering the intended output as evidence and calculating the probabilities of the remaining input nodes. This approach, described in [1], was successfully tested with discrete Bayesian networks for the modeling of injection moulding.

In that paper we present the modeling of hydroforming. The main idea of hydroforming is to form two blanks at the same time with high pressure between them. In contrast to the modeling of injection moulding we use hybrid Bayesian networks, i. e. BNs which consist of both discrete and continuous nodes. The approach taken is to use hybrid BNs to calculate a piecewise linear approximation. This is done by adding a discrete node as predecessor of the continuous one. Then the quantization error is calculated by an additional node which is a successor of both the discrete and the continuous node. That means that the information of the continuous node is still used and not thrown away which happens when only discrete values are used.

Hybrid Bayesian networks are introduced by Lauritzen in [3]. An interesting development is the usage of discrete nodes as successors of continuous nodes, presented in [6]. This type of node is applied for the first model described in this paper.

This article is structured as follows. In the first paragraph a short review about hybrid Bayesian networks and variational approximation is given. After that the engineering domain is presented, followed by the approach for modeling. The article closes with a conclusion in which the most important results are summarized.

[1] This work was funded by the 'Deutsche Forschungsgemeinschaft' (DFG) under grant number SFB 396, projectpart C1

[2] University of Erlangen-Nürnberg, Chair for pattern recognition, Martensstr. 3, D 91058 Erlangen, email: {deventer,denzler,niemann}@informatik.uni-erlangen.de

2 BAYESIAN NETWORKS

BNs are used to model a distribution of random variables $P(X_1, X_2, \cdots X_n)$. (We use P for distributions of discrete random variables and p for continuous ones.) Using the Bayes rule the complete distribution can be written as a product of conditional probabilities. This equation is known as chain-rule, cf. e. g. [8]:

$$P(x_1, x_2, \cdots, x_n) = P(x_1) \cdot \prod_{i=2}^{n} P(x_i | x_{i-1}, \cdots, x_1) \quad (1)$$

where $P(x_1, x_2, \cdots x_n)$ is used as abbreviation for $P(X_1 = x_1, X_2 = x_2, \cdots, X_n = x_n)$. Using conditional independencies between the different random variables the chain rule's right hand side simplifies to

$$P(x_1, x_2, \cdots, x_n) = P(x_1) \cdot \prod_{i=2}^{n} P(x_i | pa(i)) \quad (2)$$

with $pa(i)$ as the instantiations of all variables in $Pa(i) \subseteq \{X_{i-1}, \cdots, X_1\}$, the parents of node X_i. If $Pa(i)$ is a proper subset of $\{X_{i-1}, \cdots, X_1\}$, conditional independencies may be used to speed up calculation of the joint distribution. $X_j \notin Pa(i)$, $j < i$ implies that X_j is conditionally independent from X_i.

Normally a BN is written as a directed acyclic graph $\mathcal{G} = (E, W)$ with $W = \{X_1, X_2, \cdots, X_n\}$ as the set of nodes, and $(X_j, X_i) \in E$, the set of edges, if $X_j \in Pa(i)$. Normally an edge is drawn from X_j to X_i if X_j has an direct influence on X_i. For a more detailed introduction see [8] or [2].

2.1 Hybrid Bayesian networks

In the beginning of the development of BNs only networks with discrete nodes are used. That means that discretization is needed for all continuous variables. Additionally a great number of parameters is required to describe exactly a BN with discrete nodes. If only continuous nodes are regarded, it is possible to use a Gaussian network instead, where normal distributions are associated with every random variable, whose mean is calculated as linear combination of its predecessor's values. I. e. the distribution p of a random variable X with parents Y is

$$p(x|\vec{y}) = \mathcal{N}(\mu_{X_0} + \vec{w}_X \vec{y}, \sigma_X) \quad (3)$$

with $\mathcal{N}$ as the one-dimensional normal distribution. μ_{X_0} is the normal distribution's mean, when $\vec{y} = 0$, $\vec{w}_X$ is the weight vector between X and Y. Of course, it is possible to regard X also as a multidimensional random variable, but for the purpose of the article it is

sufficient to use a one-dimensional distribution. If not only continuous variables are used or if non-linearities are required, these needs are met by hybrid BNs as described in [3],[7],[5] and [4].

The set of nodes of a hybrid BN contains both discrete and continuous nodes. Discrete nodes having only discrete predecessors are handled as usual. I.e. each node X_i stores the conditional probabilities $P(X_i|Pa(i))$ in a table which is used for calculation of joint and marginal distributions. Major changes are made for continuous nodes having both discrete and continuous predecessors. As in Gaussian networks the values of continuous nodes are still assumed to be normal distributed, but this time a mixture of normal distribution is used with p_k, the probability of the parents having configuration k, as the mixing coefficients. As defined in [2] a configuration for a set of nodes W is a set of states with exactly one state for each variable. Therefore, there are different means $\mu_{X_0}[k]$, weights $w_X[k]$ and standard deviations $\sigma_X[k]$ for every possible configuration. The distribution of node X given that the continuous parent Y has value $\vec{y}$ and configuration k for the discrete parents is

$$p(x|\vec{y},k) = p_k \mathcal{N}(\mu_{X_0}[k] + \vec{w}_X[k]\,\vec{y}, \sigma_X[k]) \tag{4}$$

If a continuous node has no discrete parents there is only one possible configuration and the equation is reduced to the pure Gaussian case. It remains the problem whether discrete nodes are allowed to have continuous parents. Some authors, e.g. Lauritzen [3] [4] and Olesen [7] assume, that there are no such nodes allowed, which simplifies training of hybrid BNs. At the moment there are two main approaches discussed e.g. in [6] to deal with continuous predecessors of discrete nodes. A short introduction to the problem is given in the next section.

2.2 Variational approximation

When a discrete node is used as successor of a continuous node a function is needed to calculate the probability of the different states. When only the two states 0 and 1 are used, the logistic-function can be used to calculate the probability that a random variable has state 1, depending on its parents Y.

$$P(X = 1|\vec{y}) = \frac{1}{1+\exp(-\eta)} \tag{5}$$

$$\eta = a + \vec{w}_X\,\vec{y} \tag{6}$$

The length of $\vec{w}_X$ describes the steepness of the slope and the parameter a describes the location of the boundary between the two different states. The logistic function is implemented in BUGS [3] and the BN-Toolbox [4], the software packages we used for modeling. If more than two states are used, the logistic function can be expanded to the softmax-function.

$$P(X = i|\vec{y}) = \frac{\exp(a_i + \vec{w}_i\,\vec{y})}{\sum_j \exp(a_j + \vec{w}_j\vec{y})} \tag{7}$$

For more details see [6]. We decided not to use softmax nodes, because they require observable predecessors. As this requirement is not met by our models we circumvent their usage by linear approximation. That means, that a discrete node is used as predecessor of a continuous node, whose distribution is coupled to the continuous node. Additionally a continuous node is inserted with the intention, to calculate the quantization error. This idea is discussed in more detail in section 4.2.

[3] Available at http://www.mrc-bsu.cam.ac.uk/bugs/

[4] The BN-Toolbox is an expansion of matlab and freely available at http://www.cs.berkeley.edu/∼murphyk/Bayes/bnt.html

3 HYDROFORMING

In hydroforming sheet metal is formed by high pressure. Up to now only hydroforming of tubes is used in large scale production. At the Chair of Manufacturing Technology (LFT) at our University the usage of sheet metal pairs is examined which has the additional advantage that two blanks are formed at the same time.

Hydroforming begins with the closing of the press. Afterwards the clamp forces F pressing the blanks on top of each other are increased until the selected clamp force of 200, 300, 400 or 500 kN is reached.

After that the phase called preforming starts. Fluid is pressed between the two blanks and the pressure increases linearly. During that process the volume V of fluid between the blanks, the pressure D and the clamp forces F are recorded. The increasing pressure between the blanks causes a movement of the blanks into the form. Thus the clamp forces has to be selected so that a movement of the blanks is still possible, but that leaks are avoided, at least at the beginning of the process. At the end the clamp forces are too low to prevent leaks. In the data presented in figure 2 this effect can be noticed at the end when the steepness of the curve decreases. Three of the curves shown on that figure represent the data used for training, and the forth is the prediction made by the hybrid BN. First leaks occur when the volume V is about 450 dl. Please note that there is some scattering within the data, so that it is impossible to find a perfect fitting model. We didn't use the complete data for training, to ensure an equal number of data before and after the occurrence of leaks. So the time distance between two data points at the beginning is not equal to the time distance at the end.

The compression is reduced when a predetermined volume is reached. Further increasing the amount of fluid between the blanks would have no effect, because more and more hydroforming medium would escape. Then the pressure is released. This is to avoid to interrupt the calibration process and has technical reasons. This phase is not of interest, the data are not used for any model.

Before hydro calibration starts the clamp forces are increased to a large value, so that any movement of the blanks is avoided. After that the volume is increased a second time. Due to remaining hydroforming fluid between the blanks, the value of V at the beginning is not zero, but equal to the value where preforming stops. Even if the clamp forces used for hydrocalibrating are equal, it is still important to keep the clamp forces used for preforming in mind, because different clamp forces during preforming results in a different amount of metal moved inside the form, which results in a different behavior during hydrocalibration.

At the beginning of calibration shown at figure 3, there is a steep pressure increase until the preforming-pressure is approximately reached. This phase could be modeled by assuming an exponential increase [5], but since such models cannot be realized with BNs we did not pursue that approach.

Calibration is done to make sure that the edges of the workpiece are formed correctly. So it makes sense to use a maximal volume, respectively pressure. But there is of course a maximal threshold. If the pressure exceeds this threshold the blanks burst, the workpiece is damaged. This is the steep decrease of the pressure at the end. Further increasing the volume after burst has no effect on the pressure. Therefore, a good model helps to predict this point, such that a maximal possible pressure can be used at the one hand and bursting is avoided at the other hand. The BN-models are discussed and compared with the results in BUGS, where Gibbs Sampling is used for inference, in section 4.3.

[5] It was tested using a BUGS model

4 MODELING

4.1 Preforming

In the model shown in figure 1 a logistic node Sw is used to model the decision whether leaks in the system occur or not. Beside the observable nodes D, F and V there are two hidden nodes Sw and ΔV. The first hidden node named ΔV is successor of the volume V and the clamp forces F. Since the volume, where the first leaks occur depends on the clamp-forces F, this node can be used to calculate the difference between the current volume and the volume where leaks occur. This difference is used to trigger the logistic node Sw (switch) which has the task to switch between the two lines used to model the relation between volume V and pressure D shown in figure 2. To

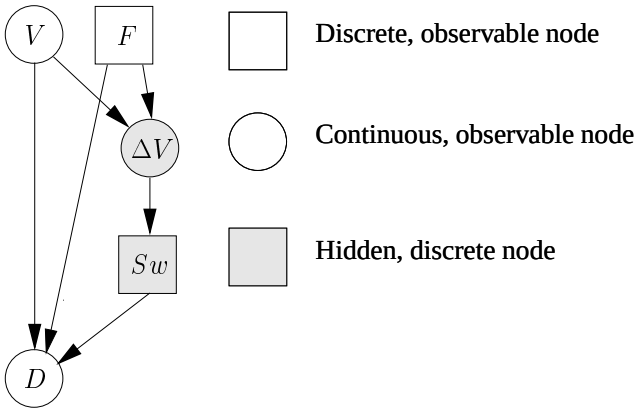

Figure 1. Model for preforming with logistic node

control the learning behavior $n-1$ values are entered as evidence, where n stands for the length of the data vector containing values for F, V and D. Then the marginal distribution of the remaining random variable $X \in \{F, V, D\}$ is computed whose mean, respectively its maximal probable state, is taken as estimation $\hat{X}$ of X and compared with the exact value of X. Then the relative error is calculated.

$$RelError(X) = \frac{|\hat{X} - X|}{X} \tag{8}$$

E. g. V and F are given as evidence, and the marginal distribution of D is calculated and compared to the original value. A very good result is obtained by the hybrid BN, the pressure D is predicted with an relative error of 1.95% (see table 1). Figure 2 shows a comparison

Table 1. Accuracy of the preforming models.

Used Model	Predicted Random Variable	Rel. error BUGS	Rel. error BN-Toolbox
Logistic node	Pressure	10.24%	1.95%

of the training data and the predicted values of the hybrid BN. At the beginning, where there is only a low dispersion in the data, there is nearly no difference between the predicted values and the data. After first leaks occur the dispersion in the training data increases. Regarding the predicted data a change in the slope can be observed, which means that the switch is working as intended. Additionally the predicted data are between the three datasets of observed data, so that the relative error gets a minimal value.

Besides the pure approximation, BNs have the ability to calculate the most probable input parameters for given output parameters without changing or re-learning the parameters or the structure of the net, i. e. they are able to compute a kind of inverse function. For the preforming model the clamp forces F are predicted with an relative error of 7,93% when given V and D and the relative error of predicting V is 3.76% given D and F.

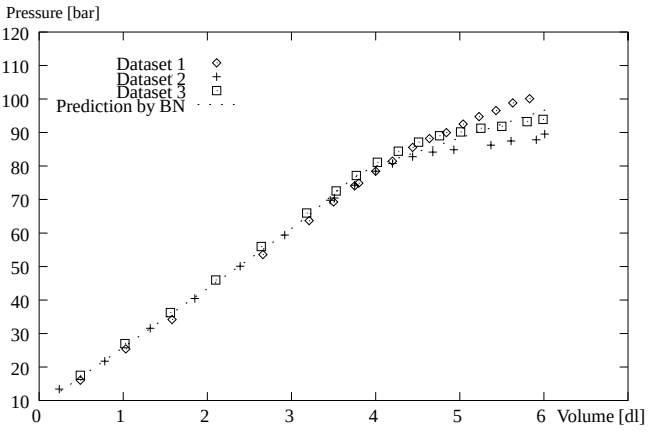

Figure 2. Results for modeling of preforming (200kN)

On of the great advantages of BNs is that the values can be interpreted easily. That means that relationships between process parameters can be evaluated using a Bayesian network. For the discussed model it gets clear, that there is a linear dependency between the clamp force F and the steepness m if Sw has state 0, i. e. $m \approx const \cdot F$.

Table 2. Steepness for different clamp forces

Clamp forces	Switch Sw	Steen. BUGS	Steepn. BN-Toolbox
200 kN	0	17.5	17.6
	1	8.21	8.25
300 kN	0	18.88	18.87
	1	4.82	2.49
400 kN	0	20.19	20.37
	1	1.73	15.08
500 kN	0	21.32	21.14
	1	11.27	10.57

Additionally the values a and w of the logistic node Sw can be used to predict where first leaks occur by calculating where the probability that the switch Sw has state 1 is 0.5. This makes sense because there is a very steep change between the two states.

$$P(Sw = 1|a, w, V) = \frac{1}{1 + \exp(-\eta(a, w, V))} = 0.5 \tag{9}$$

$$\Rightarrow \eta(a, w, V) = a + w \cdot V = 0 \tag{10}$$

$$\Rightarrow V = -\frac{a}{w} \tag{11}$$

The results of this calculation are shown in the table 3.

Table 3. First occurrence of leaks during preforming

Clamp forces	Leaks BUGS	Leaks BN-Toolbox
200 kN	4.53 dl	3.66 dl
300 kN	4.45 dl	4.02 dl
400 kN	6.89 dl	3.83 dl
500 kN	5.00 dl	4.85 dl

Here differences between the two models occur. Since the maximal volume reached during preforming for a clamp force of F = 400 kN is 5.3 dl a meaningful result must be between 0 and 5.3 dl. Larger values mean that only one line is used for modeling. That has happened for the BUGS model and is also an explanation for the different steepness in table 2 for F = 400 kN and Sw = 1.

4.2 Linear approximation

For modeling hydrocalibrating two lines are insufficient. Using more than one logistic node failed due to convergence problems in the training process. So we tried a model which makes a discretization to learn nonlinearities and uses the information stored in the continuous nodes to make a linear approximation between discrete points.

In this section we explain how a linear approximation $\hat{f}(x)$ of $f(x)$ between several base points x_k can be calculated, e. g. a linear approximation for the pressure D given the volume V. This is done by adding a discrete node X_d, e. g. V_d, for the selection of a suitable base point. Beside that a node Qe that calculates the error made by approximating X with X_d is needed. (See figure 4).

Now we will describe possible weights for the calculation of an linear approximation, but it should be clear that these are not the only ones. The comparison between learnt and suggested weights is presented in section 4.4. The formula to get an estimation $\hat{f}(x)$ of $f(x)$ is

$$\hat{f}(x) = f(x_k) + f'(x_k) \cdot (x - x_k) \tag{12}$$

where f' is the first derivative of f. This formula is compared to the calculation of the mean of a continuous node Y in a Bayesian network with Z as its continuous parent, e.g. the calculation of D depending on V:

$$\mu_Y[k] = \mu_{Y_0}[k] + w_Y[k] \cdot z \tag{13}$$

Setting $\mu_{Y_0}[k] = f(x_k)$ and the weight $w_Y[k] = f'(x_k)$, in our example $\mu_{D_0}[k] = D(V_k)$ and $w_D[k] = D'(V_k)$, it can be seen that a Bayesian network is able to calculate a linear approximation given the difference $x - x_k$. The difference $x_k - x$ between the selected base point x_k and the current value of X is calculated by a continuous, hidden node Qe which is a successor of X and a hidden, discrete node X_d, used to select the correct base point. Setting the mean $\mu_{Qe_0}[k] = x_k$ and the weight between X and $Qe = -1$ supplies the correct result. The selection of the correct base point has to be insured by setting $\mu_{X_0}[k] = x_k$. This results in

$$\mu_{Qe}[k] = \mu_{Qe_0}[k] + w_{Qe}[k] \cdot x = x_k - x \tag{14}$$

The wrong sign has to be compensated by changing the sign of the weight w_Y to $w_Y[k] = -f'(x_k)$ assuming that Y is the node representing $f(x)$. Of course piecewise linear approximation is not limited to only one dimension, due to the fact that Taylor progression works also in a multidimensional space. But the reader should keep in mind that the number of parameters to be learnt for the output node is proportional to the product of the number of base-points selected for the input variables.

4.3 Hydrocalibrating

For the modeling of hydrocalibrating we used linear approximation. The special problem is that the data we got stems from proving the stability of the blanks. That means that the volume is increased until one blank bursts. So the curve for hydrocalibration in figure 3 has an non-continuous point. We tried to model hydrocalibrating using linear approximation with 6 different base points. The accuracy, presented in table 4, is below the models for preforming. There are two reasons for that behavior. Firstly it is not possible to predict the steep slope, where the blanks bursts, due to dispersion in the data. This causes a great relative error when the hybrid BN predicts a high value and the exact value is very low. The second is the method used for prediction. To calculate the prediction for the pressure the volume was entered as evidence. Then, after passing the distribute and collect-messages, the marginal distribution of the pressure node is calculated. That means that not only one, but all six states for the discrete volume are taken into account, of course weighted according to their probability. The normal distribution for the six states of V are shown in figure 3, multiplied for optical reasons with a factor 25. Shortly before or after the burst there are two different distributions having a high probability. One is predicting a high value, the other is predicting a low one. So it is not possible to model a real non-continuous point, but the hybrid BN shows an acceptable performance.

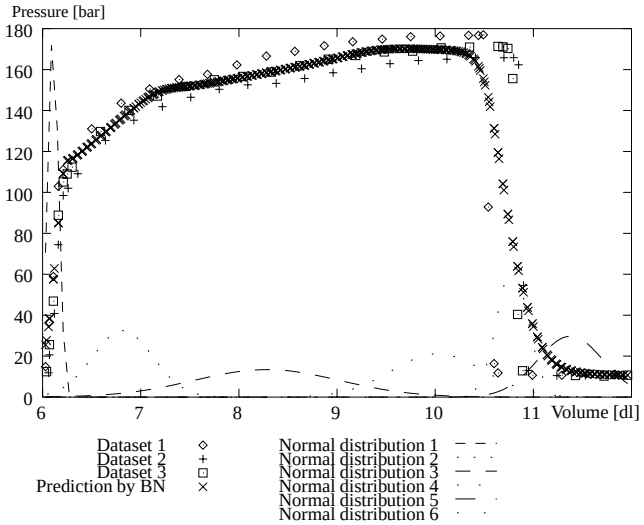

Figure 3. Modeling hydrocalibrating (200kN) with BNs

Table 4. Accuracy of the calibration models.

Used Clamp force	Predicted Random Variable	Rel. error BUGS	Rel. error BN-Toolbox
200 kN	Pressure	31.19%	28.6%
300 kN	Pressure	26.87%	23.38%

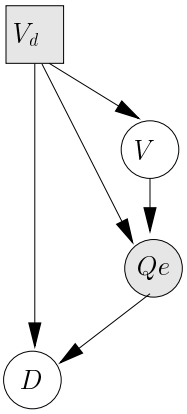

Figure 4. Model for modeling the calibration process

4.4 Analysis of the weight

This section shows which means and weights are learnt by the Bayesian network. Comparing the results in table 5 with the suggested values shows great differences. According to section 4.2 $\mu_{Qe_0}[k]$ should be equal to the base point used and thus $\mu_{Qe_0}[k]$ = $\mu_{V_0}[k]$. Additionally the node Qe should calculate the difference between the used base point $\mu_{Qe_0}[k]$ and the current value of V and thus get the value -1. Regarding the following equations explains

Table 5. Weights in the model for linear approximation

Var.		1	2	3	4	5	6
V	μ	6.106	6.68	8.28	10.03	10.79	11.38
	σ	0.056	0.31	0.74	0.47	0.16	0.34
D	μ	22.18	99.2	149.6	170.1	7.448	10.42
	w	-1.012	1.47	-0.968	-0.425	1.226	0.075
Qe	μ	3235.4	-161	68.1	-25.5	816.9	48.4
	w	-533.93	27.2	-9.33	2.68	-73.52	-3.89

why the learnt network is able to make exact predictions, even if the selected values are different from the suggested ones. According to section 2.1 the mean for P depending on Qe respectively the mean for Qe depending on V is:

$$\mu_D = \mu_{D_0} + w_D \cdot Qe \quad (15)$$

$$\mu_{Qe} = \mu_{Qe_0} + w_{Qe} \cdot V \quad (16)$$

Thus, dropping the difference between μ_{Qe} and Qe, that is identifying the expectation with the actual value, the mean for D can be calculated as

$$\mu_D = \mu_{D_0} + w_D \cdot (\mu_{Qe_0} + w_{Qe} \cdot V) \quad (17)$$

$$= \mu_{D_0} + w_D \cdot \mu_{Qe_0} + w_D \cdot w_{Qe} \cdot V \quad (18)$$

Of course, there is no single representation for μ_D, our suggested solution with $\mu_{D_0} = D(V_0), w_D = -D'(V_0), \mu_{Qe_0} = V_0$ and $w_{Qe} = -1$ is only one possible. Using the data from table 5 equation 17 shows another possible solution.

$$\mu_D = 22.18 - 1.012 \cdot (3235.4 - 533.93 \cdot V)$$

$$= -3245.2 + 539.3 \cdot V.$$

To bring the equation above into a form, so that $D(V_0)$ and the slope at this point can be identified we introduce $V_0 = 6.106$ which results in $\mu_D = 45.2 + 540 \cdot (V - V_0)$.

Similar calculations can be done for the other base points. The results show how the weights of a hybrid Bayesian network can be interpreted in the sense of calculating a linear approximation at multiple points. It cannot be expected that the values μ_{D_0} and w_D represent exactly the functional value and the value of the first derivative due to multiple degrees of freedom.

5 CONCLUSION

In the previous sections it was shown that BNs are suitable means for the modeling of technical processes. On one side they are able to make exact predictions and on the other side their interpretation is easy, so BNs can be used to gain additional insight in the technical process. To model nonlinearities there are two possibilities, either logistic nodes or linear regression. The latter has the advantage that this kind of model is available in all Bayesian network tools being able to deal with hybrid BNs and that no special training algorithms are necessary, which speeds up the training. At the moment we have not tested whether the training process of linear approximation takes advantage from a suitable initialization which might come from classical algorithms calculating a linear approximation.

ACKNOWLEDGEMENTS

Thanks to the Chair of Manufacturing Technology for provision of the data and the reviewers for valuable discussion which helps to improve the paper.

REFERENCES

[1] Rainer Deventer, Bernhard Kahles, and Heinrich Niemann. Process modelling using bayesian networks. Book of abstracts, Workshop on Conditional Independence Structures and Graphical Models, 1999. http://www.utia.cas.cz/user_data/matus/toronto'99/workshop.html.

[2] Finn V. Jensen, *An introduction to Bayesian networks*, UCL Press, 1996.

[3] Steffen L. Lauritzen, 'Poropagation of probabilities, means, and variances in mixed graphical association models', *Journal of the American Statistical Association*, Vol. 87(420), 1098 – 1108, (December 1992).

[4] Steffen L. Lauritzen and Frank Jensen, 'Stable local computation with conditional gaussian distributions', Technical Report R-99-2014, Aalborg University, Department of Mathematical Sciences, (September 1999). http://www.math.auc.dk/∼steffen/publications.html.

[5] Kevin P. Murphy, 'Inference and learning in hybrid bayesian networks', Technical report, University of California, Computer Science Division (EECS), (January 1998).

[6] Kevin P. Murphy, 'A variational approximation for bayesian networks with discrete and continuous latent variables', in *Proceedings of the Fifteenth Conference on Uncertainty in Artificial Intelligence*, eds., Kathryn Blackmond Laskey and Henri Prade, San Francisco, (1999). Morgan Kaufmann Publishers Inc. http://http.cs.berkeley.edu/∼murphyk/.

[7] K. G. Olesen, 'Causal probabilistic networks with both discrete and continuous variables', *IEEE Transaction on Pattern Analysis and Machine Intelligence, 15, vol 3*, (1993).

[8] J. Pearl, *Probabilistic reasoning in intelligent systems*, Morgan Kaufmann Publishers, Inc., 1988.