

APPLYING FUSION TECHNIQUES TO GRAPHICAL METHODS FOR KNOWLEDGE BASED PROCESSING OF PRODUCT USE INFORMATION

Susanne Dienst, Fazel Ansari Ch., Alexander Holland and Madjid Fathi
University of Siegen, Hoelderlinstr. 3, 57068 Siegen, Germany
Institute of Knowledge Based Systems and Knowledge Management, Siegen, Germany

Keywords: Product Lifecycle Management, Product Use Information, Graphical Methods, Bayesian Networks, Fusion Techniques.

Abstract: In this paper the processing and modelling of product use information raised by graphical methods on the basis of a praxis and application scenario. Product Lifecycle Management (PLM) ensures a uniform data basis for supporting numerous engineering and economic organisational processes along the entire product life cycle – from the first product idea to disposal or recycling of the product respectively. The Product Use Information (PUI) -e.g. condition monitoring data, failures or incidences of maintenance- of many instances of one product type is generated in the product use phase. The processing and modelling of PUI raised by graphical methods like Bayesian Networks. In accordance, the product use knowledge leads back to the product development phase and is used for discovering room for product improvements of the next product generation. Therefore the PUI of the different instances should be aggregated by applying fusion techniques to deduce/achieve generalized product improvements for a product type. As a result this paper reveals a novel approach of applying new feedback mechanism of PLM for product improvements.

1 INTRODUCTION

Today's Product Lifecycle Management (PLM) systems focus on supporting the early phases of the product lifecycle (Holland et. al., 2008b). Downstream phases, such as the product use phase, are currently not, or only rudimentarily flanked and supported. In (Holland et. al., 2008b) a concept for integrating the product use phase into the PLM concept is represented. It highlights the possibility of incorporating the Product Use Information (PUI) of product i , sensor data, environmental parameters, failures and incidences of maintenance from the product use phase into the development of following product generations and propagates the expansion of the conventional product type PLM with regard to the management of product item data, as it occurs within the product use phase (Holland et. al., 2008b). The principal target of the project is to deploy potentials of improvement for the next product generation of production machines. Because, basically, in productions machines objective feedback should be led back (e.g. increase

battery long life time, decrease loudness of the drive belt) from the product use phase into product development phase. Objective feedback refers to the information that should be possibly without subjective meaning (e.g. customer interview). Thus the focus of PUI lays on machine data which can be captured and submitted completely and can be remaining unchanged. The PUI in this paper is, also, captured from production machines e.g. rotation spindles. The advantage is, therefore, to transmit data, e.g. sensor data from customer to manufacturer. The praxis scenario is aimed to process data with Bayesian Networks (BN) and lead back to product development phase of the next product generation (Holland et. al., 2009). Thereby the knowledge is used to locate improvement potentials for the next product generation e.g. raising the quality of a component (motor) of a machine. In this context, a learning algorithm is used to learn BNs from PUI, as formal graphical language for representation and communication of decision scenarios requiring reasoning under uncertainty. Principally, a BN is a probabilistic graphical model

that represents a set of (random) variables and their probabilistic dependencies. Moreover, BNs are directed acyclic graphs whose nodes represent variables, and whose arcs encode the conditional dependencies among the variables (Salini et. al., 2009). The probabilities at the nodes are computed by the Bayesian rule and therefore inferences (as different types of reasoning) are performed by What-If analysis as a learning process (in this paper the learning process is applied on unknown structure and complete data set) i.e. in case of changing sensor data, the possible probability of a defect is recognized as higher or lower. The outcome of such analysis can provide opportunities to calculate the probability based on certain evidences as described in section 3.1. These opportunities are: (a) the probability that a defect appears can become higher or lower and (b) the maintenance will be advanced in order to protect against a machine defect.

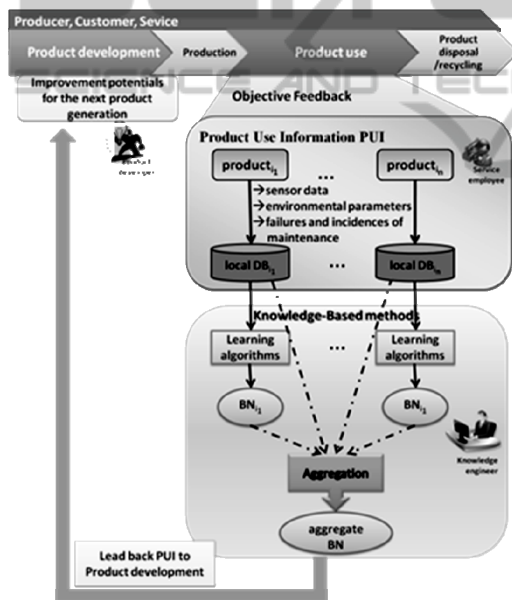


Figure 1: Leading back PUI into product development with BNs.

In terms of PLM the awareness leads back to the producers of the machine to integrate them into product development and to improve the next product generation. In this sense, the rotation spindles, which are used at various places as the feedback of the learnt BNs, can be different. Also the results of only one product whereas general results are used in order to attain an improvement of the next product generation. In this sense, the rotation spindles, which are used at various places as the feedback of the learnt BNs, can be different. Also the results of only one product whereas general

results are used in order to attain an improvement of the next product generation. Therefore the BNs of various products are aggregated to one new BN by means of fusion techniques as shown in Figure 1. Thereby acquiring sufficient products and also enough product data are vitally important. Thus a general possible BN to support the developer improvements for the next product generation can be learnt.

2 BAYESIAN NETWORKS

The use of graph-based or probabilistic models based on directed acyclic graphs apply within the field of artificial intelligence. Such models are known as Bayesian Networks (BN) (Salini et. al., 2009; Koski et. al., 2009). Their development was motivated by the need to model the top-down semantic and bottom-up perceptual combination of evidence in reading. The capability for bi-directional inferences, combined with a rigorous probabilistic foundation, were the reason for the appearance of BNs as a method of choice for reasoning under uncertainty in artificial intelligence and expert systems. A BN can be described as a graphical model for probabilistic relationships among a set of variables.

BNs model the quantitative strength of the connections between variables allowing probabilistic beliefs about them to be updated automatically as new information becomes available. It is therefore a graph in which the following holds:

- A directed acyclic graphs $G = (V, E)$ whose nodes V represent a set of discrete or continuous variables. The variables can be described as propositional variables of interest. Each variable has a set of finite mutually exclusive states. Edges represent conditional dependencies; and unconnected nodes represent variables which are conditionally independent of each other (Cowell et. al., 2007).
- Condensed, a generic entry in the joint probability distribution P is the probability of a conjunction of particular assignments to each variable given by the formula 1:

$$P(V) = \prod_{i=1}^n P(V_i | pa(V_i)) \quad (1)$$

where $pa(V_i)$ is the set of parents of V (Jensen et. al., 2007; Borgolt et. al., 2002). The learning

characteristics (e.g. structure learning) of the BNs are explained in (Holland et. al. 2008a, Neapolitan, 2003). Equation 1 implies certain conditional independent relationships that can be used efficiently to guide a product or knowledge engineer in constructing the network topology.

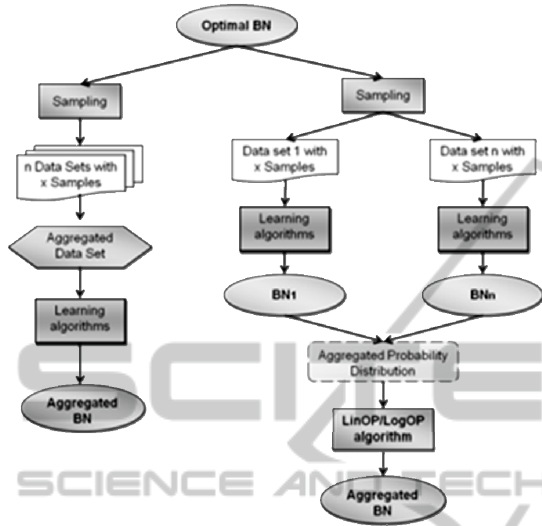


Figure 2: Different fusion techniques for BNs.

3 USING FUSION TECHNIQUES

In product using phase PUI is captured by rotation spindles which are aggregated and led back into product development. Figure 2 illustrates the use of two different fusion techniques within the feedback mechanism of PLM. These two principal approaches are applied in order to aggregate the data sets by:

(1) Merging the data sets directly by using sampling methods, or

(2) Learning BNs, firstly, from the data sets and then merging probability distribution of BNs by applying the Linear Opinion Pool (LinOP)/Logarithmic Opinion Pool (LogOP) algorithms.

Aggregation is generally defined as the use of techniques that combine data from multiple sources and gather information in order to achieve inferences, which will be more efficient and potentially more accurate than if they are achieved by means of a single source (Klein, 2004). An aggregated BN is a combination of data from two or more BNs, where every BN is an individual data set. Besides, mathematical fusion techniques range from simple methods such as arithmetic or geometric means of probabilities to procedures based on axiomatic approach (Clemen et. al., 1999).

Moreover, sampling is the process of selecting units e.g. product from a population of interest. Hence related results, with respect to the population from which they were chosen, will be fairly generalized. Sampling is a method designed for aggregation of data, and particularly in case of insufficiency of data, samples will be generated based on the existing data. The sampling methods are classified as: The Estimated Posterior Importance Sampling algorithm for Bayesian Networks algorithm (Yuan et. al., 2004), the Adaptive Importance Sampling for Bayesian Networks algorithm (Cheng et. al., 2000), the probabilistic Logic Sampling algorithm (Henrion, 1988), the Backward Sampling algorithm (Fung et. al., 1994), and the Likelihood (Fung et. al., 1994; GeNIe, 2009). Also sampling facilitates the fusion techniques by: (a) learning from available data of a BN taken as the optimal BN, (b) synthesizing of each expert network to a case database using a sampling technique, (c) aggregating the expert case databases, and (d) learning the aggregated BN structure based on the case database determined in section 2 by using a structure learning algorithm. Using sampling methods avoids induced noise by applying an aggregation operator for a common unified probability distribution (Stone, 1961). The LinOP is just a weighted linear combination of the experts' knowledge and thus it is easily understood and calculated as shown by equation 2.

$$P(\mathbf{V}) = \prod_{i=1}^n P(V_i|pa(V_i)) \quad (2)$$

where k is the number of experts, $p_i(\theta)$ represents i 's probability distribution for unknown θ , $p(\theta)$ represents the combined probability distribution and the weights w_1, \dots, w_k sum to one, with $w_1, \dots, w_k \geq 0$ and $\sum w_i = 1$ (Clemen et. al., 1999).

The other similar approach, LogOP, is to use multiplicative averaging as shown by equation 3:

$$P(\mathbf{V}) = \prod_{i=1}^n P(V_i|pa(V_i)) \quad (3)$$

Likewise, definition of variables is the same as definition of variables by LinOP algorithm [10]. As depicted in Figure 2, by using LinOP/LogOP, two versions of the same BN (BN₁...BN_n) with the same graphical structure and different probability are aggregated into a single BN.

In order to evaluate the results of aggregated BNs and compare the fusions techniques with each other a What-If-Analysis by setting evidences needs to be applied. This is explained in section 3.1.

3.1 What-If-Analysis through the use of Evidences

A statement about the certainty of a state of an attribute is called evidence (Russell et. al., 2009). This state will then occur with a probability of 100% and the directed edges determine the causal dependencies and, also, the flow of information in the network. This also means that setting evidences to all nodes within a BN, which are connected to each other, have an effect, and thus spread the probabilities under the given evidence (Lunze, 1995).

On the basis of a BN a What-If analysis is performed through the use of evidences, to show how changing the probability distribution of the nodes is. Thus, in BNs, the dependencies between the measured machine data and the individual components of the spindle are recognized. It can then derive e.g. rules, when the risk of failure of the spindle is particularly high. At high risk customers, normally, will prefer maintaining earlier, in order to prevent an outage. On the other hand, this information is used in the product development phase from manufacturer, to get a higher operational reliability by spindle by the next product generation. This is exemplified in the following.

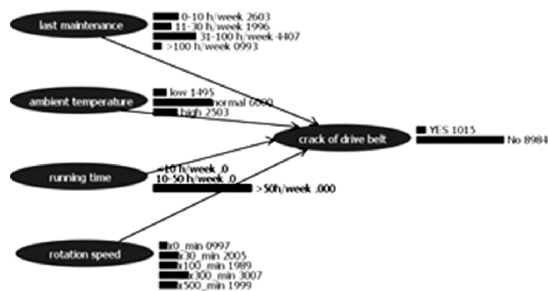


Figure 3: BN with evidence of node “running time”.

The Figure 3 shows the evidence at the node “running time”, this lies in an interval of [50,168] hours per week (h/week). For the other nodes, the average probability of occurrence is current. Here through setting evidence the probability for a “crack of drive belt” is increased as 4.23% to 10.15%. Consequently, high life of the spindle can be increased by determining the probability of a tear of the belt.

In addition, the BN (Figure 4) shows the combination of two evidences as (1) “last maintenance” in an interval of [0, 10] h, and (2) “running time” in an interval of [50, 168] h / week. The combination setting evidences, in this example, reveals that the relatively high probability can be

reduced for a crack at a high maturity through regular maintenance. Also as illustrated in Figure 5, the fact that the target node “crack of drive belt” evidence has been set, and the probabilities have changed at all nodes, are therefore assumed to be relevant. It is interesting to observe how the probabilities have changed to the node temperature and rotational speed, also where no evidence were placed at the nodes. It shows, for example, that the probability has fallen from 60% to 3.44% that the “ambient temperature” is in the interval [normal].

Therefore the probability of a high temperature in the interval is increased to 76.99%. So it comes to a significant redistribution of output probabilities. These shifts are, however, only so much, if all three evidences - and not just the evidence for the demolition of the belt- will be set.

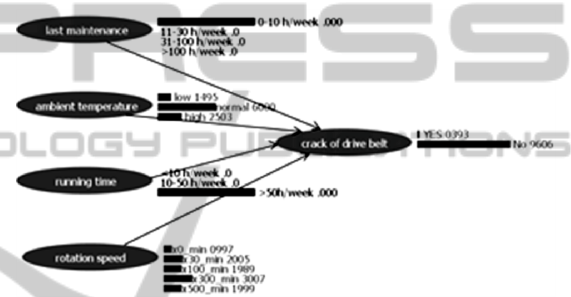


Figure 4: BN with two evidence nodes “running time” and “last maintenance”.

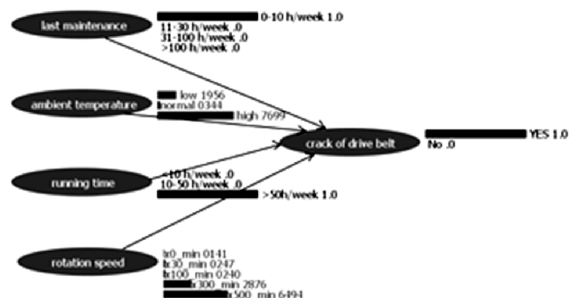


Figure 5: Combination of two evidences with the setting evidence by the child node.

4 KULL BACK-LEIBLER-DIVERGENCE

To merge different BN e.g. BN_1 and BN_2 , it is necessary to determine a measure for the approximation quality. A suitable measure is applying the Kullback-Leibler (KL)-Divergence between BN structures like BN_1 and BN_2 . The KL-Divergence expresses the difference or distance

between two probability distributions (Gaag et. al., 2001). Given the probability distributions p and q where $p = (p_1, p_2, \dots, p_n)$ represents probability distribution $P_{opt}(\theta)$ of optimal BN and $q = (q_1, q_2, \dots, q_n)$ probability distribution $P_{agg}(\theta)$ of aggregated BN. Therefore $KL(P_{opt}, P_{agg})$ is defined in equation 4 as (GeNIe2009):

$$KL(P_{opt}, P_{agg}) = \sum P_{opt}(\theta) \log \frac{P_{opt}(\theta)}{P_{agg}(\theta)} \quad (4)$$

The cross entropy between two probability distributions measures in information theory the average number of bits needed to identify an event from a set of possibilities, if a coding scheme is used based on a given probability distribution q , rather than the true distribution p . The KL divergence values are not negative with $KL(P_{opt}, P_{agg}) = 0$ if and only if $P_{opt} = P_{agg}$, then the probability distribution of aggregate BN is the same as optimal BN (Kullback, 1959; Kuntze, 2007).

4.1 Evaluation with KL-Divergence

In the evaluation, it is important to compare the results of various techniques which are described in section 3. In the sampling methods only the data sets are aggregated as visualized in Figure 6. In contrast, LinOP/LogOP algorithms are based on aggregating of BNs. Therefore in order to obtain an aggregated BN and to assess and evaluate, there are always two BNs used with the same number of generated test data for aggregation in all techniques e.g. in Figure 7 the first column of the table shows the number of samples as the integral of generated test data for BN1 and BN2 while the 50% of test data belongs to BN1 and the rest 50% to BN2. Finally these BNs are merged to obtain the aggregated BN (see Figure 2). In this context, the use of sampling algorithms is generated out of optimum BN e.g. for the number of samples for BN₁ and BN₂. These are then aggregated, and hence the aggregated BN is learned in Waikato Environment for Knowledge Analysis (WEKA). There are generated multiple aggregated BN from a different number of samples to find out how the KL-Divergence develops.

In Figure 6 each of the plotted curves shows one identical small fluctuation in all five sampling algorithms. Also the KL-Divergence decreases with increasing number of samples, and approaches from 50,000 samples to the value 0. So the probability distribution of the aggregated BN is close to that of the optimal BN. The Logic and Likelihood Sampling

supplying the test performed on the average the best results.

Number of Samples	LOGIC	EPIS	BACKWARD	Likelihood	AI5
100	0.1108467	0.1098341	0.1958142	0.2099882	0.1254329
1000	0.0299468	0.0356456	0.0342437	0.0301255	0.0289953
5000	0.0171346	0.0185293	0.0170797	0.0161554	0.0251337
10000	0.0110487	0.0159397	0.0134541	0.010739	0.0097018
25000	0.0090399	0.0090396	0.0090381	0.0092983	0.0086413
50000	0.0024825	0.0025745	0.0022891	0.0026563	0.0023455
100000	0.0017368	0.0016496	0.0018547	0.0015112	0.0019018

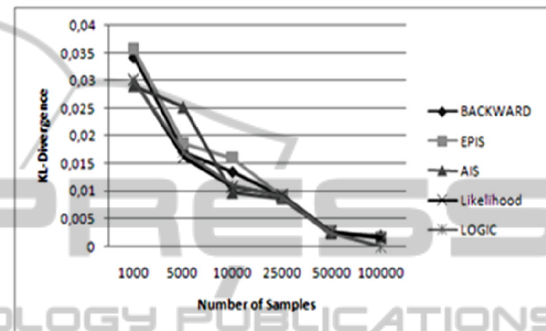


Figure 6: KL-Divergence by the different Sampling methods.

From a number of 50,000 samples, the curves are close to 0, and thus a sufficient number is given to learn a general BN too. This means that the BN is then aggregated as good as the optimal BN to derive general statements or e.g. a What-If analysis be performed to lead back PUI for the development of next product generation. The improvement of the KL-Divergence is so marginal of 50,000 samples to 100,000, that to use fusion techniques is no longer reasonable.

To determine the KL-Divergence for LinOP/LogOP, two BNs from test data should be learned and therefore probability distribution of the BNs to aggregate is obtained. These are then merged with both the LinOP/LogOP algorithms to obtain the aggregated BN. As Figure 7 shows, based on the curves clearly the level of KL-Divergence of the two methods is very close. Also this comparison shows that the curve of the aggregated BN with increasing number of data sets dramatically tends to 0. This means that the BN, formed by the two fusion algorithms, is more and more close to the optimal BN. Due to the fact that the BN used to aggregate are the same, the both curves are also similar. As the example shows, both methods are equally well suited for aggregating BNs, while LinOP tends slightly better.

Furthermore, the Figure 8 shows comparison of

Logic Sampling and LinOP/LogOP. Besides the black curve represents the Logic Sampling method and the grey curve depicts the LinOP/LogOP algorithms. The curve of the Sampling method may begin at a higher KL-Divergence, crosses the LinOP/LogOP curve already between 1000 and 5000 data sets, and then runs below the LinOP/LogOP curve. The two curves meet by a number of 100,000 data sets and remain, from that time, on a similar course. From the shape of the curves can be concluded that sampling methods is faster and more improvement achieved than in the LinOP/LogOP algorithms.

Number of Samples	LogOp Divergence	LinOp Divergence
1000	0.0211658	0.020003
5000	0.022001	0.021141
10000	0.016052	0.015908
25000	0.013741	0.013151
50000	0.0119892	0.010777
100000	0.0027748	0.001864
150000	0.0018008	0.001995

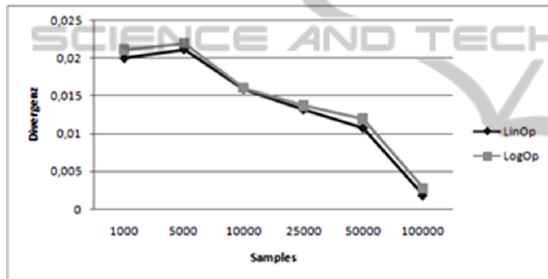


Figure 7: KL-Divergence by LinOP /LogOP algorithms.

Number of Samples	LogOp/LinOP Divergence	Logic Divergence
1000	0.0211658	0.0299468
5000	0.022001	0.0171346
10000	0.016052	0.0110487
25000	0.013741	0.0090399
50000	0.0119892	0.0024825
100000	0.0027748	0.0017368

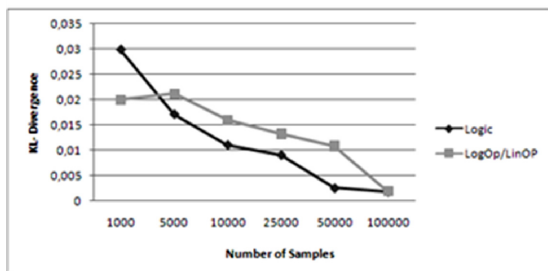


Figure 8: KL-Divergence of the Logic sampling and LinOP/LogOP.

5 CONCLUSIONS/ OUTLOOK

In this paper the knowledge-based processing of PUI of the PLM is explained and described particularly with a graph-based model, to aggregate and lead back this information to product development. From the real product rotation spindle as praxis scenario, sensor data and environmental parameters are measured and stored as PUI. The processing of PUI is carried out by the Bayesian Networks. In this context, the PUI is collected from multiple instances of a product type. The aim is to improve the quality of the next product generation and not only a product instance. Therefore the data must be aggregated to deduce generalized information, thus, it is indispensable to apply fusion algorithms. For this purpose an extension of the BN is made, and an aggregated BN is created. It is also possible to merge PUI of the individual spindles directly, and then learn from an aggregated BN, or each product instance is learned by a BN and then the entire instances are aggregated. Besides, using the KL-Divergence for evaluating the various fusion techniques is shown that both possibilities are likely to create aggregated BNs. Also it is pointed out that within small number of samples it is advantageous to apply sampling. In this sense by using the LinOP/LogOP algorithm the graphical structure of the BN which should be aggregate always must be the same. However for attaining an optimal BN, the experimental results of the rotation spindle imply that nearly 50,000 of data sets the WEKA threshold value is achieved. For data sets that are less than 50,000 samples the graph of the learned BN is not as the optimal BN and because of this the merged BN cannot be optimal. From a number of greater than 100,000 samples the gain is so low that no further aggregation with LinOP/LogOP is reasonable.

Within this process, some questions are open e.g. which sources of information are available and how they can be integrated? The existing data is mainly sensor data which are measured in the environment of the rotation spindle. Furthermore, the proper description of a defect and frequency of replacing of the rotation spindle components is not fulfilled. Also the prospective research trend of applying knowledge-based processing of PUI is to provide lead back to product development. This is enabled by applying quality management systems and policies for modification of know-how through processes, standardization of best practices within production, and identifying customer requirements and expectation by defining of proper measures.

ACKNOWLEDGEMENTS

Within the project “PLM Management Extension through Knowledge-Based Product Use Information Feedback into Product Development“ (WiRPro), KBS staff is currently working on a solution to integrate information from the product use phase into the product development phase. The authors of this paper have made significant contributions to that development.

We express our sincere thanks to the Deutsche Forschungsgemeinschaft (DFG) for financing this research and to our project partner of the University of Bochum, Chair of Information Technology in Mechanical Engineering (ITM).

REFERENCES

- Holland, A.; Fathi, M.; Abramovici, M.; Neubach, M., 2008b. Enhancing a PLM System in Regard to the Integrated Management of Product Item and Product Type Data. *In: Proceedings of the 2008 IEEE International Conference on Systems, Man, and Cybernetics (IEEE SMC 2008), 12.-15.10.2008, Singapore, ISBN: 1-4244-2384-2.*
- Holland, A.; Fathi, M.; Abramovici, M.; Neubach, M., 2009. Knowledge-Based Feedback of Product Use Information into Product Development. *In: International Conference on Engineering Design (ICED'09), Stanford University, Stanford, CA, USA*
- Salini S., Kenett R. S., 2009. Bayesian Networks of Customer Satisfaction Survey Data. *In: Journal of Applied Statistics, Volume 36, Issue 11 November 2009, pages 1177 – 1189.*
- Koski, T., Noble, J. M., 2009. *Bayesian Networks – An Introduction.* John Wiley & Sons, Ltd. The Atrium, Southern Gate, Chichester, West Sussex, PO19 8SQ, United Kingdom.
- Cowell R. G., Dawid P., Lauritzen S. L., Spiegelhalter D.J., 2007. *Probabilistic Networks and Expert Systems.* Springer, New York, USA.
- Jensen, F.V., Nielsen, T., 2007. *Bayesian Networks and Decision Graphs.* Statistics for Engineering and Information Science, Springer-Verlag, Berlin Heidelberg New York, 2nd Edition.
- Borgelt, C.; Kruse, R., 2002. *Graphical Models. Methods for Data Analysis and Mining.* John Wiley & Sons, West Sussex, United Kingdom.
- Holland, A.; Fathi, M.; Abramovici, M.; Neubach, M., 2008a. Competing Fusion for Bayesian Applications. *In Proceedings of the 12th Intl. Conference on Information Processing and Management of Uncertainty in Knowledge-Based Systems (IPMU 2008), Malaga, Spain.*
- Klein L.: Sensor and Data Fusion, 2004. *A Tool for Information Assessment and Decision Making.* SPIE – The Society of Photo-Optical Instrumentation Engineers, Beellingham, Washington.
- Clemen R.T., Winkler R. L., 1999. Combining Probability Distributions from Experts in Risk Analysis. *In: Risk Analysis 19(2): 187–203.*
- Yuan C., Druzdzel M., 2004. A Comparison on the Effectiveness of Two Heuristics for Importance Sampling. *In: Second European Workshop on Probabilistic Graphical Models (PGM-04), Leiden, Netherlands.*
- Cheng J., Druzdzel M., 2000. BN-AIS: An Adaptive Importance Sampling Algorithm for Evidential Reasoning in Large Bayesian Networks. *In: Journal of Artificial Intelligence Research, 13:155-188.*
- Henrion M., 1988. Propagating Uncertainty in Bayesian Networks by Probabilistic Logic Sampling. *In: Uncertainty in Artificial Intelligence 2, pages 149-163, New York, N.Y. Elsevier Science Publishing Company, Inc..*
- Fung R., Favero B. del, 1994. Backward Simulation in Bayesian Networks. *In: 10th Annual Conference on Uncertainty in Artificial Intelligence (UAI-94), pages 227-234, San Mateo, CA. Morgan Kaufmann Publishers, Inc..*
- Fung R., Chang K.-C., 1989. Weighing and Integrating Evidence for Stochastic Simulation in Bayesian Networks. *In: M. Henrion, R.D. Shachter, L.N. Kanal, and J.F. Lemmer, editors, Uncertainty in Artificial Intelligence 5, pages 209-219, New York, N. Y. Elsevier Science Publishing Company, Inc..*
- Decision Systems Laboratory DSL, GeNie Guide, University of Pittsburgh, USA (2009) <http://genie.sis.pitt.edu/> (last visit: 25.01.2010)
- Stone M., 1961. *The Opinion Pool.* The Annals of Mathematical Statistics, 32(4):1339-1342.
- Russell S. J., Norvig P., 2009. *Artificial Intelligence: A Modern Approach.* Prentice Hall, USA, 3rd edition.
- Lunze J., 1995. *Künstliche Intelligenz für Ingenieure – Band 2: Technische Anwendungen.* R. Oldenburger Verlag GmbH, München.
- Gaag L.C. van der, Renooij S., 2001: On the Evaluation of Probabilistic Networks. *In: 8th Conference on Artificial Intelligence in Medicine.* Lecture Notes in Computer Science, volume 2101, pages 457-461, Springer-Verlag, Berlin Heidelberg New York.
- Kullback S., 1959. *Information Theory and Statistics.* Wiley.
- Kuntze D., 2007. *Untersuchung von Clustering- Algorithmen für die Kullback-Leibler-Divergenz.* Fakultät für Elektrotechnik, Informatik und Mathematik der Universität Paderborn.
- Neapolitan, R. E. 2003. *Learning Bayesian Networks.* Prentice Hall, USA.