Knowledge-Based Operations for Graphical Models in Planning

Jörg Gebhardt[1] and Rudolf Kruse[2]

[1] Intelligent Systems Consulting (ISC), Celle, Germany
ISC-Gebhardt@t-online.de
[2] Dept. of Knowledge Processing and Language Engineering (IWS),
Otto-von-Guericke-University of Magdeburg, Magdeburg, Germany

Abstract. In real world applications planners are frequently faced with complex variable dependencies in high dimensional domains. In addition to that, they typically have to start from a very incomplete picture that is expanded only gradually as new information becomes available. In this contribution we deal with probabilistic graphical models, which have successfully been used for handling complex dependency structures and reasoning tasks in the presence of uncertainty. The paper discusses revision and updating operations in order to extend existing approaches in this field, where in most cases a restriction to conditioning and simple propagation algorithms can be observed. Furthermore, it is shown how all these operations can be applied to item planning and the prediction of parts demand in the automotive industry. The new theoretical results, modelling aspects, and their implementation within a software library were delivered by ISC Gebhardt and then involved in an innovative software system realized by Corporate IT for the world-wide item planning and parts demand prediction of the whole Volkswagen Group.

1   Introduction

Complex products like automobiles are usually assembled from a number of prefabricated modules and parts. Many of these components are produced in specialised facilities not necessarily located at the final assembly site. An on-time delivery failure of only one of these components can severely lower production efficiency. In order to efficiently plan the logistical processes, it is essential to give acceptable parts demand estimations at an early stage of planning.

One goal of the project described in this paper was to develop a system which plans parts demand for production sites of the Volkswagen Group. The market strategy of the Volkswagen Group is strongly customer-focused — based on adaptable designs and special emphasis on variety. Consequently, when ordering an automobile, the customer is offered several options of how each feature should be realised. The consequence is a very large number of possible car variants. Since the particular parts required for building an automobile depend on the variant of the car, the overall parts demand can not be successfully estimated from total production numbers alone.

The modelling of domains with such a large number of possible states is very complex. For many practical purposes, modelling problems are simplified by introducing strong restrictions, e.g. fixing the value of some variables, assuming simple functional relations and applying heuristics to eliminate presumably less informative variables. However, as these restrictions can be in conflict with accuracy requirements or flexibility, it is rewarding to look into methods for solving the original task. Since working with complete domains seems to be infeasible, decomposition techniques are a promising approach to this kind of problem. They are applied for instance in graphical models (Lauritzen and Spiegelhalter, 1988; Pearl, 1988; Lauritzen, 1996; Borgelt and Kruse, 2002; Gebhardt, 2000), which rely on marginal and conditional independence relations between variables to achieve a decomposition of distributions. In addition to a compact representation, graphical models allow reasoning on high dimensional spaces to be implemented using operations on lower dimensional subspaces and propagating information over a connecting structure. This results in a considerable efficiency gain.

In this paper we will show how a graphical model, when combined with certain operators, can be applied to flexibly plan parts demand in the automotive industry. We will furthermore demonstrate that such a model offers additional benefits, since it can be used for item planning, and it also provides a useful tool to simulate parts demand and capacity usage in projected market development scenarios.

2   Probabilistic Graphical Models

Graphical Models have often and successfully been applied with regard to probability distributions. The term "graphical model" is derived from an analogy between stochastic independence and node separation in graphs.

Let $V = \{A_1, \ldots, A_n\}$ be a set of random variables. If the underlying distribution fulfils certain criteria (see e.g. Castillo et al., 1997), then it is possible to capture some of the independence relations between the variables in V using a graph $G = (V, E)$.

2.1   Bayesian Networks

In the case of Bayesian networks, G is a directed acyclic graph (DAG). Conditional independence between variables V_i and V_j; $i \neq j$; $V_i, V_j \in V$ given the value of other variables $S \subseteq V$ is expressed by V_i and V_j being d-separated by S in G (Pearl, 1988; Geiger et al., 1990); i.e. there is no sequence of edges (of any directionality) between V_i and V_j such that:

1. every node of that sequence with converging edges is an element of S or has a descendant in S,
2. every other node is not in S.

Probabilistic Bayesian networks are based on the idea that the common probability distribution of several variables can be written as a product of marginal and conditional distributions. Independence relations allow for a simplification of these products. For distributions such a factorisation can be described by a

graph. Any independence map of the original distribution that is also a DAG provides a valid factorisation. If such a graph G is known, it is sufficient to store a conditional distribution for each node attribute given its direct predecessors in G (marginal distribution if there are no predecessors) to represent the complete distribution p_V, i.e.

$$\forall a_1 \in \mathrm{dom}(A_1) : \ldots \forall a_n \in \mathrm{dom}(A_n) :$$
$$p_V \left(\bigwedge_{A_i \in V} A_i = a_i \right) = \prod_{A_i \in V} p \left(A_i = a_i \mid \bigwedge_{(A_j, A_i) \in E} A_j = a_j \right).$$

2.2   Markov Networks

Markov networks are based on similar principles, but rely on undirected graphs and the u-separation criterion instead. Two nodes are considered separated by a set S if all paths connecting the nodes contain an element from S. If G is an independence map of a given distribution, then any separation of two nodes given a set of attributes S corresponds to a conditional independence of the two given values of the attributes in S. As shown by Hammersley and Clifford (1971) a strictly positive probability distribution is factorisable w.r.t. its undirected independence graph, with the factors being nonnegative functions on the maximal cliques $C = \{C_1 \ldots C_m\}$ break in G.

$$\forall a_1 \in \mathrm{dom}(A_1) : \ldots \forall a_n \in \mathrm{dom}(A_n) :$$
$$p_V \left(\bigwedge_{A_i \in V} A_i = a_i \right) = \prod_{C_i \in C} \phi_{C_i} \left(\bigwedge_{A_j \in C_i} A_j = a_j \right).$$

A detailed discussion of this topic, which includes the choice of factor potentials ϕ_{C_i} is given e.g. in Borgelt and Kruse (2002). It is worthy to note that graphical models can also be used in the context of possibility distributions. The product in the probabilistic formulae will then be replaced with the minimum.

3   Analysis of the Planning Problem

The models offered by the Volkswagen Group are typically highly flexible and therefore very rich in variants. In fact many of the assembled cars are unique with respect to the variant represented by them. It should be obvious that under these circumstances a car cannot be described by general model parameters alone. For that reason, model specifications list so called *item variables* $\{F_i : i = 1 \ldots n; \ i, n \in I\!\!N\}$. Their domains $dom(F_i)$ are called *item families*. The item variables refer to various attributes like for example 'exterior colour', 'seat covering', 'door layout' or 'presence of vanity mirror' and serve as placeholders for features of individual vehicles. The elements of the respective domains are called *items*. We will use capital letters to denote item variables and indexed lower case letters for items in the associated family. A variant specification is

Table 1. Vehicle specification

Class: 'Golf'

Item	short back	2.8L 150kW spark	Type alpha	5	no	...
Item family	body variant	engine	radio	door layout	vanity mirror	...

obtained when a model specification is combined with a vector providing exactly one element for each item family (Table 1.)

For the 'Golf' class there are approximately 200 item families—each consisting of at least two, but up to 50 items. The set of possible variants is the product space $\mathrm{dom}(F_1) \times \ldots \times \mathrm{dom}(F_n)$ with a cardinality of more than 2^{200} (10^{60}) elements. Not every combination of items corresponds to a valid variant specification (see Sec. 3.1), and it is certainly not feasible to explicitly specify variant-part lists for all possible combinations.

Apart from that, there is the manufacturing point of view. It focuses on automobiles being assembled from a number or prefabricated components, which in turn may consist of smaller units. Identifying the major components—although useful for many other tasks—does not provide sufficient detail for item planning. However, the introduction of additional structuring layers i.e. 'components of components' leads to a refinement of the descriptions. This way one obtains a tree structure with each leave representing an installation point for alternative parts.

Depending on which alternative is chosen, different vehicle characteristics can be obtained. Part selection is therefore based on the abstract vehicle specification, i.e. on the item vector. At each installation point only a subset of item variables is relevant. Using this connection, it is possible to find partial variant specifications *(item combinations)* that reliably indicate whether a component has to be used or not. At the level of whole planning intervals this allows to calculate total parts demand as the product of the relative frequency of these relevant item combinations and the projected total production for that interval. Thus the problem of estimating parts demand is reduced to estimating the frequency of certain relevant item combinations.

3.1   Ensuring Variant Validity

When combining parts, some restrictions have to be considered. For instance, a given transmission t_1 may only work with a specific type of engine e_3. Such relations are represented in a system of technical and marketing rules. For better readability the item variables are assigned unique names, which are used as a synonym for their symbolic designation. Using the item variables T and E ('transmission' and 'engine'), the above example would be represented as:

$$\text{if 'transmission'} = t_1 \text{ then 'engine'} = e_3$$

The antecedence of a rule can be composed from a combination of conditions and it is possible to present several alternatives in the consequence part.

if 'engine' = e_2 and 'auxiliary heater' = h_3 then 'generator' $\in \{g_3, g_4, g_5\}$

Many rules state engineering requirements and are known in advance. Others refer to market observations and are provided by experts (e.g. a vehicle that combines sportive gadgets with a weak motor and automatic gear will not be considered valid, even though technically possible). The rule system covers explicit dependencies between item variables and ensures that only valid variants are considered. Since it already encodes dependence relations between item variables it also provides an important data source for the model generation step.

3.2   Additional Data Sources

In addition to the rule system it is possible to access data on previously produced automobiles. This data provides a large set of examples, but in order to use it for market oriented estimations, it has to be cleared of production-driven influences first. Temporary capacity restrictions, for example, usually only affect some item combinations and lead to their underrepresentation at one time. The converse effect will be observed, when production is back to normal, so that the deferred orders can be processed. In addition to that, the effect of starting times and the production of special models may superpose the statistics. One also has to consider that the rule system, which was valid upon generation of the data, is not necessarily identical to the current one. For that reason the production history data is used only from relatively short intervals known to be free of major disturbances (like e.g. the introduction of a new model design or supply shortages). When intervals are thus carefully selected, the data is likely to be 'sufficiently representative' to quantify variable dependences and can thus provide important additional information. Considering that most of the statistical information obtained from the database would be tedious to state as explicit facts, it is especially useful for initialising planning models.

Finally we want experts to be able to integrate their own observations or predictions into the planning model. Knowledge provided by experts is considered of higher priority than that already represented by the model. In order to deal with possible conflicts it is necessary to provide revision and updating mechanisms.

4   Generation of the Markov Network Model

It was decided to employ a probabilistic Markov network to represent the distribution of item combinations. Probabilities are thus interpreted in terms of estimated relative frequencies for item combinations. But since there are very good predictions for the total production numbers, conversion of facts based on absolute frequency is well possible. In order to create the model itself one still has to find an appropriate decomposition. When generating the model there are two data sources available, namely a rule system $\mathcal{R}$, and the production history.

4.1   Transformation of the Rule System

The dependencies between item variables as expressed in the rule system are relational. While this allows to exclude some item combinations that are inconsistent with the rules, it does not distinguish between the remaining item combinations, even though there may be significant differences in terms of their frequency. Nevertheless the relational information is very helpful in the way that it rules out all item combinations that are inconsistent with the rule system.

In addition to that, each rule scheme (the set of item variables that appear in a given rule) explicitly supplies a set of interacting variables. For our application it is also reasonable to assume that item variables are at least in approximation independent from one another given all other families, if there is no common appearance of them in any rule (unless explicitly stated so, interior colour is expected to be independent of the presence of a trailer hitch). Using the above independence assumption we can compose the relation of 'being consistent with the rule system'. The first step consists in selecting the maximal rule schemes with respect to the subset relation. For the joint domain over the variables in each maximal rule scheme the relation can directly be obtained from the rules.

For efficient reasoning with Markov networks it is desirable that the underlying clique graph has the hypertree property. This can be ensured by graph triangulating (Figure 1c). An algorithm that performs this triangulation is given e.g. in Pearl (1988). However introducing additional edges is done at the cost of losing some more independence information. The maximal cliques in the triangulated independence graph correspond to the nodes of a hypertree (Figure 1d).
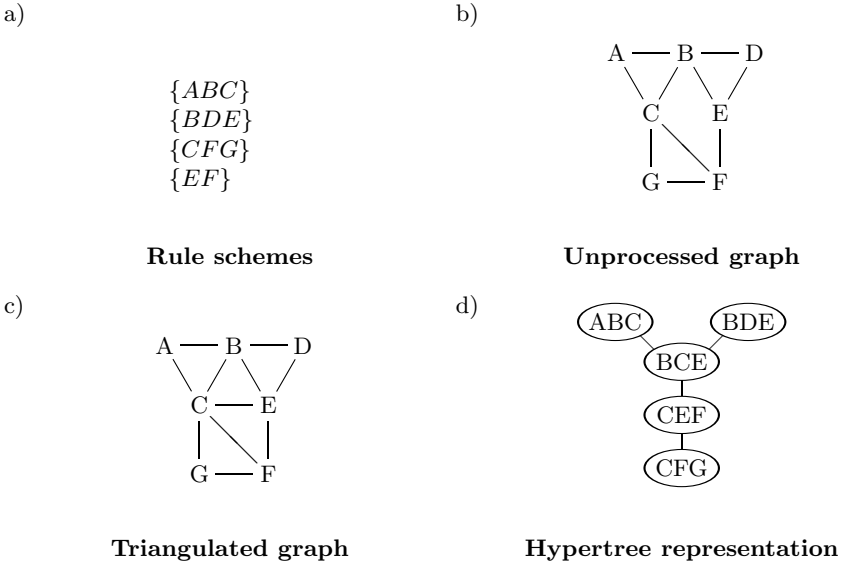

Fig. 1. Transformation into hypertree structure

To complete the model we still need to assign a local distribution (i.e. relation) to each of the nodes.

For those nodes that represent the original maximal cliques in the independence graph they can be obtained from the rules that work with these item variables or a subset of them (see above). Those that use edges introduced in the triangulation process can be computed from them by combining projections, i.e. applying the conditional independence relations that have been removed from the graph when the additional edges were introduced. Since we are dealing with the relational case here this amounts to calculating a join operation.

Although such a representation is useful to distinguish valid vehicle specifications from invalid ones, the relational framework alone cannot supply us with sufficient information to estimate item rates. Therefore it is necessary to investigate a different approach.

4.2   Learning from Historical Data

A different available data source consists of variant descriptions from previously produced vehicles. However, predicting item frequencies from such data relies on the assumption that the underlying distribution does not change all too sudden. In section 3.2 considerations have been provided how to find 'sufficiently representative' data.

Again we can apply a Markov network to capture the distributions using the probabilistic framework this time. One can distinguish between several approaches to learn the structure of probabilistic graphical models from data. Performing an exhaustive search of possible graphs is a very direct approach. Unfortunately this method is extremely costly and infeasible for complex problems like the one given here. Many algorithms are based on dependency analysis (Sprites and Glymour, 1991; Steck, 2000; Verma and Pearl, 1992) or Bayesian statistics, e.g. K2 (Cooper and Herskovits, 1992), K2B (Khalfallah and Mellouli, 1999), CGH (Chickering et al., 1995) and the structural EM algorithm (Friedman, 1998).

Combined algorithms usually use heuristics to guide the search. Algorithms for structure learning in probabilistic graphical models typically consist of a component to generate candidate graphs for the model structure, and a component to evaluate them so that the search can be directed (Khalfallah and Mellouli, 1999; Singh and Valtorta, 1995). However even these methods are still costly and do not guarantee a result that is consistent to the rule system of our application.

Our approach is based on the fact that we do not need to rely on the production history for learning the model structure. Instead we can make use of the relational model derived from the rule system. Using the structure of the relational model as a basis and combining it with probability distributions estimated from the production history constitutes an efficient way to construct the desired probabilistic model.

Once the hypergraph is selected, it is necessary to find the factor potentials for the Markov network. For this purpose a frequentistic interpretation is assumed, i.e. estimates for the local distributions for each of the maximal cliques are ob-

tained directly from the database. In the probabilistic case there are several choices for the factor potentials because probability mass associated with the overlap of maximal cliques (separator sets) can be assigned in different ways. However for fast propagation it is often useful to store both local distributions for the maximal cliques and the local distributions for the separator sets (junction tree representation). Having copied the model structure from the relational model also provides us with additional knowledge of forbidden combinations. In the probability distributions these item combinations should be assigned a zero probability.

While the model generation based on both rule system and samples is fast, it does not completely rule out inconsistencies. One reason for that is the continuing development of the rule system. The rule system is subject to regular updates in order to allow for changes in marketing programs or composition of the item families themselves.

These problems, including the redistribution of probability mass, can be solved using belief change operations (Gebhardt and Kruse, 1998), which are described in the next section.

5   Planning Operations

A planning model that was generated using the above method, usually does not reflect the whole potential of available knowledge. For instance, experts are often aware of differences between the production history and the particular planning interval the model is meant to be used with. Thus a mechanism to modify the represented distribution is required. In addition to that we have already mentioned possible inconsistencies that arise from the use of different data sources in the learning process itself.

Planning operators have been developed to efficiently handle this kind of problem, so modification of the distribution and restoration of a consistent state can be supported.

5.1   Updating

Let us now consider the situation where previously forbidden item combinations become valid. This can result for instance from changes in the rule system. In this case neither quantitative nor qualitative information on variable interaction can be obtained from the production history. A more complex version of the same problem occurs when subsets of cliques are to be altered while the information in the remaining parts of the network is retained, for instance after the introduction of rules with previously unused schemes (Gebhardt et al., 2003). In both cases it is necessary to provide the probabilistic interaction structure—a task performed with the help of the updating operation.

The updating operation marks these combinations as valid by assigning a positive near zero probability to their respective marginals in the local distributions. Since the replacement value is very small compared to the true item frequencies obtained from the data, the quality of estimation is not affected by this alteration. Now instead of using the same initialisation for all new item

combinations, the proportion of the values is chosen in accordance to an existing combination, i.e. the probabilistic interaction structure is copied from reference item combinations. This also explains why it is not convenient to use zero itself as an initialisation. The positive values are necessary to carry qualitative dependency information. For illustration consider the introduction of a new value t_4 to item family transmission. The planners predict that the new item distributes similarly to the existing item t_3. If they specify t_3 as a reference, the updating operation will complete the local distributions that involve T, such that the marginals for the item combinations that include t_4 are in the same ratio to each other as their respective counterparts with t_3 instead.

Since updating only provides the qualitative aspect of dependency structure, it is usually followed by the subsequent application of the revision operation, which can be used to reassign probability mass to the new item combinations.

5.2   Revision

After the model has been generated, it is further adapted to the requirements of the particular planning interval. The information used at this stage is provided by experts and includes marketing and sales stipulations. It is usually specific to the planning interval. Such additional information can be integrated into the model using the revision operator. The input data consists of predictions or restrictions for installation rates of certain items, item combinations or even sets of either. It also covers the issue of unexpected capacity restrictions, which can be expressed in this form.

Although the new information is frequently in conflict with prior knowledge, i.e. the distribution previously represented in the model, it usually has an important property—namely that it is compatible with the independence relations, which are represented in the model structure.

The revision operation, while preserving the network structure, serves to modify quantitative knowledge in such a way that the revised distribution becomes consistent with the new specialised information. There is usually no unique solution to this task. However, it is desirable to retain as much of the original distribution as possible so the principle of minimal change (Gärdenfors, 1988) should be applied. Given that, a successful revision operation holds a unique result (Gebhardt et al., 2004).

The operation itself starts by modifying a single marginal distribution. Using the iterative proportional fitting method, first the local clique and ultimately the whole network is adapted to the new information. Since revision relies on the qualitative dependency structure already present, one can construct cases where revision is not possible. In such cases an updating operation is required before revision can be applied. In addition to that the supplied information can be contradictory in itself. Such situations are sometimes difficult to recognise. Criteria that entail a successful revision and proves for the maximum preservation of previous knowledge have been provided in Gebhardt et al. (2004). Gebhardt (2001) deals with the problem of inconsistent information and how the revision operator itself can help dealing with it.

Depending on circumstances human experts may want to specify their knowledge in different ways. Sometimes it is more convenient to give an estimation of future item frequency in absolute numbers, while at a different occasion it might be preferable to specify item rates or a relative increase. With the help of some readily available data and the information which is already represented in the network before revision takes place, such inputs can be transformed to item rates. From the operator's point of view this can be very useful.

As an example for a specification using item rates experts might predict a rise of the popularity of a recently introduced navigation system and set the relative frequency of this respective item from 20% to 30%. Sometimes the stipulations are embedded in a context as in "The frequency of air conditioning for Golfs with all wheel drive in France will increase by 10%". In such cases the statements can be transformed and amount to a changing the ratio of the rates for the combination of all items in the statement (air conditioning present, all wheel drive, France) to the rates of that, which only includes the items from the context (all wheel drive, France).

5.3   Focussing

While revision and updating are essential operations for building and maintaining a distribution model, it is a much more common activity to apply the model for the exploration of the represented knowledge and its implications with respect to user decisions.

Typically users would want to concentrate on those aspects of the represented knowledge that fall into their domain of expertise. Moreover, when predicting parts demand from the model, one is only interested in estimated rates for particular item combinations (see Sec. 3). Such activities require a focussing operation. It is achieved by performing evidence-driven conditioning on a subset of variables and distributing the information through the network. The well-known variable instantiation can be seen as a special case of focussing where all probability is assigned to exactly one value per input variable.

As with revision, context dependent statements can be obtained by returning conditional probabilities. Furthermore, item combinations with compatible variable schemes can be grouped at the user interface providing access to aggregated probabilities.

Apart from predicting parts demand, focussing is often employed for market analyses and simulation. By analysing which items are frequently combined by customers, experts can tailor special offers for different customer groups. To support planning of buffer capacities, it is necessary to deal with the eventuality of temporal logistic restrictions. Such events would entail changes in short term production planning so that the consumption of the concerned parts is reduced. This in turn affects the overall usage of other parts. The model can be used to simulate scenarios defined by different sets of frame conditions, to test adapted production strategies and to assess the usage of all parts.

6   Application

The results obtained in this paper have contributed to the development of the planning system EPL (EigenschaftsPLanung, item planning). It was initiated in 2001 by Corporate IT, Sales, and Logistics of the Volkswagen Group. The aim was to establish for all trademarks a common item planning system that reflects the presented modelling approach based on Markov networks.

System design and most of the implementation work of EPL is currently done by Corporate IT. The mathematical modelling, theoretical problem solving, and the development of efficient algorithms, extended by the implementation of a new software library called MARNEJ (MARkov NEtworks in Java) for the representation and the presented functionalities on Markov networks have been entirely provided by ISC Gebhardt.

Since 2004 the system EPL is being rolled out to all trademarks of the Volkswagen group and step by step replaces the previously used planning systems. In order to promote acceptance and to help operators adapt to the new software and its additional capabilities, the user interface has been changed gradually. In parallel planners have been introduced to the new functionality, so that EPL can be applied efficiently.

In the final configuration the system will have 6 to 8 Hewlett Packard Machines running Linux with 4 AMD Opteron 64-Bit CPUs and 16 GB of main memory each.

With the new software, the increasing planning quality, based on the many innovative features and the appropriateness of the chosen model of knowledge representation, as well as a considerable reduction of calculation time turned out to be essential prerequisites for advanced item planning and calculation of parts demand in the presence of structured products with an extreme number of possible variants.

References

C. Borgelt and R. Kruse. *Graphical Models—Methods for Data Analysis and Mining.* J. Wiley & Sons, Chichester, 2002.

E. Castillo, J.M. Guitérrez, and A.S. Hadi. *Expert Systems and Probabilistic Network Models.* Springer-Verlag, New York, 1997.

D.M. Chickering, D. Geiger, and D.Heckerman. Learning Bayesian networks from data. *Machine Learning*, 20(3):197–243, 1995.

G.F. Cooper and E. Herskovits. A Bayesian method for the induction of probabilistic networks from data. *Machine Learning*, 9:309–347, 1992.

N. Friedman. The Bayesian structural EM algorithm. In *Proc. of the 14th Conference on Uncertainty in AI*, pages 129–138, 1998.

P. Gärdenfors. *Knowledge in the Flux—Modeling the Dynamics of Epistemic States.* MIT press, Cambridge, MA, 1988.

J. Gebhardt. The revision operator and the treatment of inconsistent stipulations of item rates. Project EPL: Internal Report 9. ISC Gebhardt and Volkswagen Group, GOB-11, 2001.

J. Gebhardt. Learning from data: Possibilistic graphical models. In D. M. Gabbay and P. Smets, editors, *Handbook of Defeasible Reasoning and Uncertainty Management Systems*, volume 4: Abductive Reasoning and Learning, pages 314–389. Kluwer Academic Publishers, Dordrecht, 2000.

J. Gebhardt and R. Kruse. Parallel combination of information sources. In D. M. Gabbay and P. Smets, editors, *Handbook of Defeasible Reasoning and Uncertainty Management Systems*, volume 3: Belief Change, pages 393–439. Kluwer Academic Publishers, Dordrecht, 1998.

J. Gebhardt, H. Detmer, and A.L. Madsen. Predicting parts demand in the automotive industry – an application of probabilistic graphical models. In *Proc. Int. Joint Conf. on Uncertainty in Artificial Intelligence (UAI'03, Acapulco, Mexico), Bayesian Modelling Applications Workshop*, 2003.

J. Gebhardt, C. Borgelt, and R. Kruse. Knowledge revision in markov networks. *Mathware and Soft Computing*, 11(2-3):93–107, 2004.

D. Geiger, T.S. Verma, and J. Pearl. Identifying independence in Bayesian networks. *Networks*, 20:507–534, 1990.

J.M. Hammersley and P.E. Clifford. Markov fields on finite graphs and lattices. Cited in Isham (1981), 1971.

V. Isham. An introduction to spatial point processes and markov random fields. *Int. Statistical Review*, 49:21–43, 1981.

F. Khalfallah and K. Mellouli. Optimized algorithm for learning Bayesian networks from data. In *Proc. 5th European Conference on Symbolic and Quantitative Approaches to Reasoning with Uncertainty (ECSQUARU'99)*, pages 221–232, 1999.

S. L. Lauritzen and D. J. Spiegelhalter. Local computations with probabilities on graphical structures and their application to expert systems. *Journal of the Royal Statistical Society, Series B*, 2(50):157–224, 1988.

S.L. Lauritzen. *Graphical Models*. Oxford University Press, 1996.

J. Pearl. *Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference*. Morgan Kaufman, San Mateo, USA, 1988. (2nd edition 1992).

M. Singh and M. Valtorta. Construction of Bayesian network structures from data: Brief survey and efficient algorithm. *Int. Journal of Approximate Reasoning*, 12: 111–131, 1995.

P. Sprites and C. Glymour. An algorithm for fast recovery of sparse causal graphs. *Social Science Computing Review*, 9(1):62–72, 1991.

H. Steck. On the use of skeletons when learning Bayesian networks. In *Proc. of the 16th Conference on Uncertainty in AI*, pages 558–565, 2000.

T. Verma and J. Pearl. An algorithm for deciding whether a set of observed independencies has a causal explanation. In *Proc. 8th Conference on Uncertainty in AI*, pages 323–330, 1992.