Automatic Fuzzy Decision Network Transformation

ALEXANDER HOLLAND
Institute of Knowledge Based Systems
University of Siegen
Hölderlinstraße 3, 57068 Siegen
GERMANY
http://www.ws.fb12.uni-siegen.de

MADJID FATHI
Institute of Knowledge Based Systems
University of Siegen
Hölderlinstraße 3, 57068 Siegen
GERMANY
http://www.ws.fb12.uni-siegen.de

Abstract: Among the various types of decision support systems, decision-theoretic models and rule-based systems have gained considerable attraction. Both approaches have advantages and disadvantages. Decision-theoretic models dispose of a sound mathematical basis and comfortable knowledge engineering tools. Rule-based systems provide an efficient execution architecture and represent knowledge in an explicit, intelligible way. In this paper, we consider fuzzy rule-based systems as a special type of condensed decision model. We outline a knowledge compilation scheme which allows one to transform a decision-theoretic model into a fuzzy rule base and, hence, to combine the advantages of both approaches.

Key-Words: Bayesian networks, decision networks, transformation, fuzzy rule base, knowledge representation

1 Introduction

During the recent years, several types of decision support systems have been proposed in literature. Among these systems, *decision-theoretic models* and *rule-based systems* have gained considerable attraction. Expert systems, for instance, are often implemented as rule-based systems, since human expert knowledge can generally be expressed quite well in terms of rules. *Fuzzy rule-based systems* extend classical rule-based systems in a reasonable way, since expert knowledge is usually afflicted with vagueness and imprecision. Decision-theoretic models are based directly on concepts from mathematical (statistical) decision theory. During the recent years, these concepts have been developed further in artificial intelligence. Particularly, the mathematical framework has been extended by graphical models, notably Bayesian networks and decision networks [1, 3, 6]. In fact, the decision-theoretic approach is now recognized as the most important formal foundation of modeling rational behavior. Moreover, decision networks are widely accepted as important tools for both, the design of intelligent agents as well as the realization of (decision-theoretic) expert systems. Our aim is to combine the advantages of both approaches. More precisely, we outline a knowledge compilation scheme which allows one to transform a decision theoretic model into a fuzzy rule base. Thus, we consider fuzzy rule-based systems as a special type of condensed decision model. The idea is to maintain a decision theoretic model "off-line", which appears reasonable from a knowledge engineering point of view. For reasons of efficiency, rule-based systems derived from that model are then used "on-line", e.g. for real-time decision making. We explain this idea in more detail in the following section.

2 Decision Making and Knowledge Transformation

Decision-theoretic models appear advantageous to rule-based systems from a knowledge engineering point of view. Particularly, the former dispose of adequate concepts for the modeling of *uncertainty*. Apart from that, the decision maker's *preferences* are taken into account in an explicit way. Worth mentioning in this connection is the *declarative* character of decision-theoretic models: What the approach requires is only a *description* of the problem and the decision maker's preferences. Given this, the optimal decision behavior is already determined by means of an underlying rationality principle. This way, the declarative approach avoids systematic faults typically made by human experts in rule-based modeling, e.g. caused by an inconsistent handling of uncertainty. The declarative approach also facilitates the adaptation and extension of a model, e.g. the consideration of a new variable. One disadvantage of decision-theoretic methods concerns the issue of complexity: Finding an optimal decision might

become very expensive. In fact, inference in Bayesian networks is known to be NP-hard [1]. Therefore, decision networks are not always suitable for time-critical applications. Rule-based systems are much more efficient in this respect. This is mainly due to the fact that decision knowledge is represented in a very compressed form. For example, so called *condition–action* rules of the form 'IF Condition THEN action' define a kind of *action model* that assigns an action (a decision) to each situation in an immediate way. As opposed to decision-theoretic models, the situation itself is actually not analyzed before a decision is made. Particularly, aspects such as preference and uncertainty do no longer appear explicitly. One possibility to combine the advantages from both approaches is to provide methods for transforming a decision-theoretic model into a rule-based system [4]. Such a transformation can be seen as a special type of *knowledge compilation*, that is the transformation of a formal representation of knowledge into an alternative form which is more suitable for the purpose at hand. Here, we are concerned with the representation of decision knowledge. A (normative) decision-theoretic model is to be *approximated* by means of an efficient action model in the form of a fuzzy rule base. Note that compilation can indeed be *approximate* in the sense that the input–output relation of the original model is not exactly reproduced. In this connection, the idea of *information granulation* plays an important role. Information granulation refers to the partitioning of an object into a set of granules, where each granule is a collection of basic entities drawn together, e.g., by indistinguishability, similarity, proximity, or functionality. More generally, *granular computing* is concerned with the systematic study of abstracting information and of processing information at different levels of abstraction. Information granulation provides a tool for modulating the level of abstraction and, hence, the complexity of a (decision) model. The idea of compiling decision knowledge is not least motivated by human decision behavior. In fact, human experts do have the ability to adapt their decision behaviour in a flexible way, especially to change the level of abstraction. When deciding on the treatment of a patient, for example, a physician will generally weigh the expected advantages and disadvantages in a very careful manner. His reasoning will be guided by background knowledge and experience, and maybe further information will be gathered before the final decision is made. However, the physician will act very differently in an emergency. In such a case, he will avoid lengthy deliberation and rather act according to some proper rules. By doing so, he still employs his background knowledge, but fortunately this knowledge is now quickly accessible in a compiled form.

3 Inference Steps and Decision Networks

As already mentioned above, decision-theoretic models are *declarative* in nature. The basic ingredients of a model are: (1) A description of the decision maker's environment and the possible actions, including information about the effect of actions on the environment. (2) The decision maker's preferences. (3) The decision maker's beliefs. Apart from that, a decision model includes a principle of rationality. Probably the best-known rationality principle is that of expected utility theory [3, 5, 7]. Here, the environment is characterized by a set of possible world states, the effect of an action depends on the true (but unknown) world state, and the decision maker's preferences and beliefs are modeled, respectively, by means of a utility function and a probability measure. Consider the following setting of Bayesian decision theory: Let A denote the set of possible actions the decision maker can perform in the environment. Θ is the set of possible world states (states of nature), with ξ being a probability measure defined on some σ-field $T \subseteq 2^{\Theta}$: $\xi(\vartheta) = \xi(\{\vartheta\})$ is the (prior) probability (density) of the world state $\vartheta \in \Theta$. The utility of the decision maker's action depends on the world state and is determined by means of a real-valued utility function $U : A \times \Theta \to \mathcal{R}$. That is, $U(a, \vartheta)$ is the utility experienced by the decision maker if he performs action a in world state ϑ. The decision maker cannot observe the world state directly. However,
it has access to some information represented by a random variable

$$X: (\Omega, \mathcal{O}, \nu[\vartheta]) \to (\mathcal{R}^n, \mathcal{B}_n) \tag{1}$$

where $\mathcal{B}_n$ denotes the class of Borel subsets of $\mathcal{R}^n$. The distribution of X depends on the world state: For each $\vartheta \in \Theta$, $X = X_\vartheta$ is a random variable with the distribution $\mu = \mu[\vartheta] = X(\nu[\vartheta])$. Let $\mathcal{D}_X = X(\Omega)$. The decision behavior prescribed by the principle of *maximum expected utility* is determined by some optimal decision function expressed as $\Delta^*: \mathcal{D}_X \to A$. For each observation $x \in \mathcal{D}_X$, this function prescribes an action that maximizes expected utility (whenever such an action exists):

$$\Delta^*(x) \in \text{argmax}_{a \in A} \text{ EU}(a \mid x), \tag{2}$$

where

$$\text{EU}(a \mid x) = \int_{\Theta} U(a, \vartheta) \, d\pi(\vartheta) \tag{3}$$

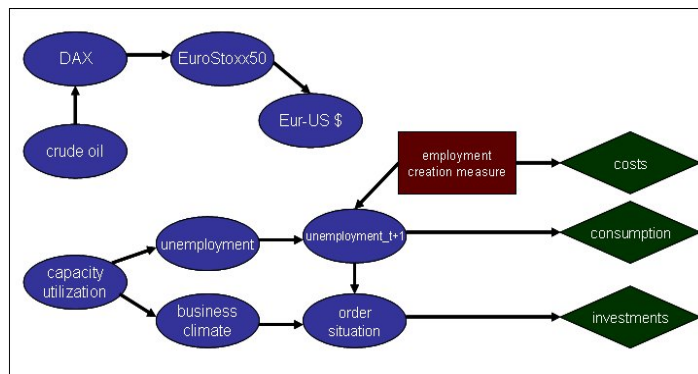is the expected utility of action a.

3.1 Bayesian Networks

Bayesian networks are graphical models to represent knowledge under conditions of uncertainty. They has been used in many fields like logistic applications [2], expert systems [3] or classification systems as powerful tools for the knowledge representation and inference under uncertainty. Bayesian networks and the use of such probabilistic models is based on direct acyclic graphs (DAG) with a probability table for each node. The nodes $\mathcal{V}$ in a Bayesian network represent propositional variables in a domain, the edges $\mathcal{E}$ between the nodes represent the dependency relationship among the variables. Each node has a conditional probability table $P(X \mid X_1,...,X_n)$ attached that quantifies the effects that the parents $X_1,...,X_n$ have on the node. We could say that the conditional probabilities encode the strength of dependencies among the variables. For each a conditional probability distribution is defined that specifies the probabilities of $\mathcal{V}$ given the values of the parents of X. A decision maker makes decisions by combining his own knowledge, experience and intuition with that available from other sources. Given a learned network structure [8] like Bayesian networks the decision maker can implement additional information in applying an inference algorithm. We use the learned Bayesian network to calculate new probabilities when particular information is achieved. For instance let A have n states with $P(A) = (x_1, ..., x_n)$ and assume that we get the information e that A can only be in state i or j. This statement expresses that all states except i and j are impossible, so next we can illustrate the probability distribution as $P(A, e) = (0,...,0, x_i, 0,...,0, x_j ,0, ...,0)$. Assume a joint probability table $P(U)$ where $\underline{e}$ is the preceding finding (n-dimensional table of zeros and ones). Using the chain rule for Bayesian networks [5] we can express the following

$$P(U, e) = \Pi \, A \, \epsilon \, U \; P(A \mid parents(A)) \cdot \Pi \, i \, \underline{e}_i \qquad (4).$$

3.2 Decision Networks

Influence diagrams or known as decision network representations can be considered as extensions of Bayesian networks. Apart from chance nodes, a decision network contains two additional types of nodes, namely utility nodes and action nodes. Subsequently, we restrict ourselves to one utility node and one action node. (Several action nodes are needed for sequential decision problems.) The utility node U is associated with a utility function $U : pa(U) \rightarrow \mathcal{R}$, where $pa(U)$ denotes the parents of U (a utility node does not have any children). The action node is associated with the set A of possible actions. The parents of this node are the so-called *evidence variables* or *information variables*: These variables are known to (can be observed by) the decision

maker before acting. Figure 1 shows a decision network representing the german stock exchange and their key influencing factors like unemployment or business climate. This network involves nine random variables. The utility for the costs depends on the employment-creation measure, the utility for the consumption depends on the umemployment in $t+1$ and the utility for the investments depends on the order situation.

Fig. 1. Decision network represents the german stock exchange and their influencing factors

In order to compute the expected utility of an action $a \, e \, A$, the action node is instantiated with that action. Moreover, the evidence variables are instantiated with the corresponding observations. Then, algorithms for Bayesian networks are used in order to compute a probability distribution over the random variables which are parents of the utility node. The expected utility $EU(a)$ can then be derived on the basis of this distribution. After having performed this procedure for all $a \, e \, A$, an optimal action is chosen according to the maximum expected utility criterion.

4 Fuzzy Rule Bases

Consider a set of variables X_ι with domains D_{X_ι} ($1 \le \iota \le n$) and a variable Y with domain $\mathcal{D}_Y$. Moreover, let F_ι be a fuzzy partition of DX_ι , that is a finite set of fuzzy subsets $F \epsilon \mathcal{F}(\mathcal{D}_{X_\iota})$ such that $\Sigma_{F \epsilon F_\iota} F(x) > 0$ for all $x \, \epsilon \mathcal{D}_{X_\iota}$. Likewise, let F be a fuzzy partition of $\mathcal{D}_Y$. A fuzzy rule base $\mathcal{R}$ is a finite set of fuzzy rules of the form "If X_1 is in F_1 and X_2 is in F_2 and . . . and X_n is in F_n then Y is in F", can formally express and written as $<F_1, F_2, . . . , F_n \mid F>$. There are different types of fuzzy inference schemes. Formally, an inference scheme identifies a fuzzy rule base $\mathcal{R}$ with a function

$$\varphi_R : D_{X1} \times . . . D_{Xn} \rightarrow \mathcal{F}(\mathcal{D}_Y), \qquad (5)$$

where $\mathcal{F}(\mathcal{D}_Y)$ is the class of fuzzy subsets of $\mathcal{D}_Y$. If a defuzzification operator $\mathcal{F}(\mathcal{D}_Y) \rightarrow \mathcal{D}_Y$ is applied to

the output of this function, the fuzzy rule base $\mathcal{R}$ induces a function

$$\varphi_R : \mathcal{D}_{X1} \times \ldots \mathcal{D}_{Xn} \to \mathcal{D}_Y . \qquad (6)$$

Here, we do neither stick to a particular inference scheme nor to a special defuzzification operator. The important point to realize is simply the following: Once an inference scheme and a defuzzification operator have been determined, each fuzzy rule base $\mathcal{R}$ can be associated with a function (6).

5 Transformation Framework

Our point of departure is a decision network with a set $\mathcal{V}$ of variables. Let $\mathcal{E} = \{E_1,..., E_n\} \subseteq \mathcal{V}$ denote the set of evidence (information) variables. Moreover, let $\mathcal{A}$ be the set of actions available to the decision maker. The decision maker's preferences are modeled by means of the utility function $U : \mathcal{V} \times \mathcal{A} \to \mathcal{R}$. Of course, utility will generally not depend on all variables $V \in \mathcal{V}$. Thus, the domain of U might be reduced correspondingly. Note that $S = \mathcal{D}_{E1} \times \mathcal{D}_{E2} \times \ldots \times \mathcal{D}_{En}$ can be considered as the set of potential decision problems or *situations*: Each situation is specified by a vector $(e_1, e_2, \ldots, e_n) \in S$ [9]. Given this information, the decision maker has to choose an appropriate action. Let $\Delta^* : S \to \mathcal{A}$ denote some optimal decision function induced by the decision network. We are now interested in approximating the decision network, that is the optimal decision function, by means of a fuzzy rule base $\mathcal{R}$. In this connection, the evidence variables $E_1, \ldots, E_n$ play the role of the variables $X_1, \ldots, X_n$ in (6), and the action variable A corresponds to the variable Y. Moreover, F_t is a fuzzy partition of $\mathcal{D}_{Et}$, and $\mathcal{F}$ is a fuzzy partition of $\mathcal{A}$. Thus, the function (6) induced by a fuzzy rule base $\mathcal{R}$ can be considered as an approximation

$$\Delta_{\mathcal{R}} : S \to \mathcal{A} \qquad (7)$$

to the optimal decision function. Note that the compiled decision model (7) will generally involve much less variables than the original model. In fact, the function (7) has $n = \text{card}(E)$ arguments, whereas the decision network involves $\text{card}(V)$ variables. Apart from that, the domain of the variables E_t is reduced by passing from $\mathcal{D}_{Et}$ to the fuzzy partition F_t. In fact, the maximal number of rules in a rule base $\mathcal{R}$ is given by

$$\prod_{t=1} \text{card}(\mathcal{F}_t). \qquad (8)$$

A *granulation* G specifies a set of fuzzy partitions $G = \{\mathcal{F}_1, \mathcal{F}_2, \ldots, \mathcal{F}_n, \mathcal{F}\}$, i.e. a fuzzy partition for each evidence variable E_t as well as for the action variable A. Let G denote the class of *allowed* granulations. Moreover, let $\mathcal{M} = \mathcal{M}_G$ denote the class of fuzzy rule bases that can be defined using granulations $G \in \mathcal{G}$. (Note that serveral rule bases can be defined for one granulation G.) The quality of a fuzzy rule base $\mathcal{R}$ as a decision function depends on several factors, notably its approximation quality and its complexity. A natural measure of the approximation quality of a rule base $\mathcal{R}$ is the *expected utility loss*. The expected utility of a decision maker using the decision network for choosing actions is given by

$$\text{EU}(\Delta^*) = \Sigma e \in S \ \text{EU}(\Delta^*(e) \mid e) \cdot \text{Pr}(E = e), \qquad (9)$$

where $\text{Pr}(E = e)$ is the probability of observing the situation e. Now, the expected utility loss experienced by a decision maker choosing actions according to the mapping (7) is given by

$$\alpha(\mathcal{R}) = \text{EU}(\Delta^*) - \text{EU}(\Delta_{\mathcal{R}}). \qquad (10)$$

Next we outline a framework approach for the transformation of decision networks or influence diagrams. The network is represented in an extensional form as the set of situations with optimal decisions i.e. as the pointwise specification of an optimal decision function. The objective, then, is to approximate this function by means of a function $\Delta_{\mathcal{R}}$ such that $\mathcal{R}$ does not exceed an upper complexity bound. Of course, the class S of situations will generally be large. Therefore, it might not be possible to compute $\Delta^*(e)$ for all $e \in S$. Rather, optimal solutions will only be derived for a *sample* $S_0 \subseteq S$:

0. Specify a class $\mathcal{G}$ of granulations and a class $\mathcal{M}$ of rule bases. Let $S_0 = \varnothing$.

1. **Sampling**: Extend the current set S_0 of examples by selecting further decision problems $e \in S$. Employ the decision network in order to derive optimal solutions $\Delta^*(e)$ for these problems.

2. **Search**: Derive a fuzzy rule base $R^* \in \mathcal{M}$ which is (approximately) optimal with the expected utility loss replaced by the *empirical utility loss*

$$\alpha(R) = 1/\text{card}(S_0) \ \Sigma e \in S_0 \ \text{EU}(\Delta^*(e)|e) - \text{EU}(\Delta_R(e)|e).$$

The expected utility $\text{EU}(\Delta R(e)|e)$ is derived on the basis of the decision network.

3. **Evaluation**: Evaluate the current rule base, e.g. by means of some kind of cross validation (which requires a further sample). Decide whether the current rule base $R*$ is good enough. If not, proceed with 4, otherwise exit.

4. **Model Adaptation**: Adapt the current class $\mathcal{G}$ of granulations and the class $\mathcal{M}$ of rule bases.

The objective of the above compilation scheme is to find a rule base which guarantees a certain approximation quality with a certain probability.

We have tested the decision network representing the german stock exchange effective including a method *getOptimalRuleBase* to increase the quality of the rule base to fulfil the optimal fuzzy rule base $\mathcal{R}*$ with the following property

$$\mathcal{R}* \in \arg\max_{\mathcal{R} \in \mathcal{M}} Q(\kappa(\mathcal{R}), \alpha(\mathcal{R})) \qquad (11)$$

regarding the complexity measure (8) and expected utility loss (10).

7 Conclusion

We have outlined a general procedure for approximating decision networks by fuzzy rule-based systems. Using fuzzy *IF–THEN* rules for representing decision functions has the following advantages:

- Fuzzy rule bases are an efficient implementation from a computational point of view.
- Fuzzy rule bases can approximate non-linear functional dependencies.
- Information granulation is ideally supported by fuzzy rules.
- Fuzzy models are intelligible (at least if the involved membership functions are restricted to semantically meaningful ones).

Apart from aspects of efficiency, approximating a decision network by a fuzzy rule base is interesting for another reason: A fuzzy rule base can be evaluated for *vague* observations, whereas the decision network requires the evidence variables to be precisely specified. Thus, the compilation of a decision network extends the applicability of the decision model. Currently, we are about to put the different points of the compilation framework into practice. Our goal is to develop a general method for compiling decision networks. Apart from the points mentioned in the paper, we are also investigating some further extensions. For example, in order to further reduce the complexity of the rule base, a kind of feature selection suggests itself: Only the most important evidence variables are selected, whereas the unimportant ones are ignored.

References:

[1] Cooper, G.F.: The Computational Complexity of Probabilistic Inference Using Bayesian Belief Networks. Artificial Intelligence. 42:393–405, 1990.

[2] Holland, A.: A Bayesian Approach to Model Uncertainty in Network Balanced Score cards. Proceedings of the 10th International Conference on Soft Computing (Mendel Conference), Brno, Czech Republic (2004) 134-138.

[3] Russel, S., Norvig, P.: Artificial Intelligence. A Modern Approach. Prentice Hall, New Jersey, 2003.

[4] Borgelt, C.; Kruse, R.: Graphical Models. Methods for Data Analysis and Mining. John Wiley & Sons, New York, 2002.

[5] Pearl, J.: Probabilistic Reasoning in Intelligent Systems. Morgan Kaufmann, San Francisco, 1988.

[6] Jensen, F.V.: Bayesian Networks and Decision Graphs. Statistics for Engineering and Information Science, Springer-Verlag, Berlin Heidelberg New York, 2001.

[7] Brachman, R.J.; Levesque, H.J.: Knowledge Representation and Reasoning. Morgan Kaufman Publishers, San Francisco, California, 2004.

[8] Neapolitan, R.E.: Learning Bayesian Networks. Prentice Hall, 2004.

[9] Holland, A.; Fathi, M.: Concurrent Fusion via Sampling and LinOP-Aggregation. Modeling Decisions for Artificial Intelligence, MDAI 2006, Tarragona, Catalonia, Spain, April 2006 (to appear).