Decision Support Systems 49 (2010) 354–364

Contents lists available at ScienceDirect

Decision Support Systems

journal homepage: www.elsevier.com/locate/dss

Modeling challenges with influence diagrams: Constructing probability and utility models

C. Bielza [a], M. Gómez [b,*], P.P. Shenoy [c]

[a] Dept. de Inteligencia Artificial, Universidad Politécnica de Madrid, Boadilla del Monte, Madrid 28660, Spain
[b] Dept. of Computer Science and Artificial Intelligence, University of Granada, Granada 18071, Spain
[c] School of Business, University of Kansas, Lawrence, KS 66045, USA

ARTICLE INFO

Article history:
Received 27 May 2009
Received in revised form 28 March 2010
Accepted 4 April 2010
Available online 9 April 2010

Keywords:
Decision-making under uncertainty
Influence diagrams
Bayesian networks
Probabilistic graphical models
Generalized additive independence networks
Ceteris paribus networks
Utility ceteris paribus networks
Expected utility networks
Utility diagrams

ABSTRACT

Influence diagrams have become a popular tool for representing and solving complex decision-making problems under uncertainty. In this paper, we focus on the task of building probability models from expert knowledge, and also on the challenging and less known task of constructing utility models in influence diagrams. Our goal is to review the state of the art and list some challenges. Similarly to probability models, which are embedded in influence diagrams as a Bayesian network, preferential/utility independence conditions can be used to factor the joint utility function into small factors and reduce the number of parameters needed to fully define the joint function. A number of graphical models have been recently proposed to factor the joint utility function, including the generalized additive independence networks, ceteris paribus networks, utility ceteris paribus networks, expected utility networks, and utility diagrams. Similarly to probability models, utility models can also be engineered from a domain expert or induced from data.

1. Introduction

Decision-making problems based on uncertain information are composed of four different elements: (1) a sequence of decisions to be made; (2) a set of uncertain variables described by a probability model; (3) decision maker's preferences for the possible outcomes described by a utility model; and (4) some information constraints on what uncertainties can and cannot be observed before a decision has to be made. All of these elements can be graphically represented by influence diagrams (IDs), see [48]. Nowadays, IDs have become a popular and standard modeling tool for decision-making problems. As pointed out in a recent special issue of the journal *Decision Analysis* devoted to IDs, these models "command a unique position in the history of graphical models" [77].

IDs are directed acyclic graphs with three types of nodes: (1) decision nodes (rectangular) representing decisions to be made; (2) chance nodes (oval or elliptical) representing uncertainties modeled by probability distributions; and (3) value nodes (diamond-shaped) without children (direct successors), representing the (expected) utilities that model decision-maker's preferences. The arcs have different meanings depending upon which node they are directed to: the arcs to chance nodes or the value nodes indicate probabilistic dependence and functional dependence, respectively, while the arcs pointing at a decision node indicate the information known at the time of making that decision. The former are called *conditional* arcs while the latter are called *informational* arcs. Informational arcs are related to the information constraints mentioned above.

Therefore we can distinguish two levels in an ID: qualitative and quantitative. The qualitative (or graphical) level has a requirement: there must be a directed path comprising all decision nodes. This ensures the definition of a temporal sequence (total order) of decisions and it is called *sequencing constraint*. As a consequence, IDs have the "*no-forgetting*" property: the decision maker remembers the past observations and decisions. At the quantitative level, an ID specifies the domains of all decision and chance nodes. A conditional probability table is attached to each chance node consisting of conditional probability distributions, one for each state of its parents (direct predecessors). The utility functions (real-valued functions) quantify the decision maker's preferences for outcomes and will be attached to value nodes. They are defined over the states of the value node's parents. If several value nodes are present, then each represents an additive factor of the joint utility function.

Fig. 1 shows an example of the graphical part of an ID. D_1 and D_2 are decision nodes; A, C and R are chance nodes; and υ_1, υ_2, and υ_3 are value nodes. υ_1 is a function of the states of D_1, υ_2 is a function of the

* Corresponding author.
E-mail addresses: mcbielza@fi.upm.es (C. Bielza), mgomez@decsai.ugr.es (M. Gómez), pshenoy@ku.edu (P.P. Shenoy).

0167-9236/$ – see front matter © 2010 Elsevier B.V. All rights reserved.
doi:10.1016/j.dss.2010.04.003

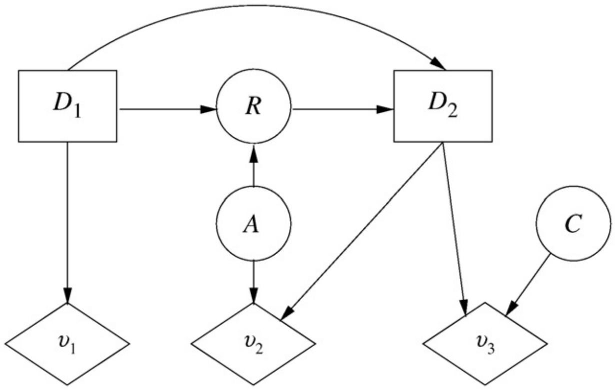

Fig. 1. An influence diagram.

states of D_2 and A, and υ_3 is a function of the states of D_2 and C. The joint utility function is the pointwise sum of υ_1, υ_2 and υ_3. As in a Bayesian network, the arcs directed to chance nodes like R mean that the conditional probability attached to R is given by $P(R|D_1, A)$. Finally, since there are no informational arcs directed to D_1, nothing is known when a decision at D_1 has to be made. The informational arcs (D_1, D_2) and (R, D_2) directed to D_2 mean that at the time a decision at D_2 has to be made, we know the outcome of R and the decision made at D_1. The informational arc (D_1, D_2) is also called a no-forgetting arc, and it can be deduced from the fact that there is a directed path from D_1 to D_2.

Evaluating an ID means computing a strategy with the maximum expected utility. This strategy consists of a policy for each decision node. A policy for decision node D_i is a function δ_{D_i} that associates each state of D_i's parents with a state d_j of D_i, that results in the maximum expected utility:

$$\delta_{D_i} : x_{pa(D_i)} \rightarrow d_j \quad (1)$$

The evaluation algorithms take advantage of the independencies among the ID variables. The dependencies and independencies appear naturally during the construction of the model and are represented by arcs and absence of arcs respectively. The absence of an arc among two variables represents their mutual independence. Therefore the removal of a link in order to simplify the model may lead to a wrong picture of the decision problem under examination. As it happens while building any model, a tradeoff between simplicity and expressivity is needed.

Olmsted [70] described a method to solve IDs. Shachter [83] published the first ID evaluation algorithm. After that, several algorithms based on variable elimination strategies or on clique-trees approaches may be now used to solve IDs [22,50,62,85,86,94]. Computational issues related to ID evaluation are beyond the scope of this paper. Some critical difficulties and their solutions are discussed and exemplified in [4,37], where a large ID, called *IctNeo*, models neonatal jaundice management for an important public hospital in Madrid.

IDs have an enormous potential as a tool for modeling uncertain knowledge. The process of building an ID itself provides a deep understanding of the problem, and ID outputs are remarkably valuable. Given a specific configuration of variables, an ID yields the *best course of action*. But ID responses are not limited to providing optimal strategies for the decision-making problem. *Inferred posterior distributions* may be employed to generate diagnosis outputs (probabilities of each cause). IDs may also automatically generate *explanations* of their proposals as a way to justify their reasoning [30].

The domain expert may formulate a more difficult *query*, without specifying all the variables required to determine the optimal decision, leading to imprecise responses that should be refined if we want the decision maker to be satisfied [29]. Reasoning in the reverse direction, assuming that the final results of the decisions are known, the ID can be used to generate *probabilistic profiles* that fit these final results (answering questions like “which kind of patients receive this specific treatment?”). Also, the computation of the *expected value of information* have shown to play a vital role in assessing the different sources of uncertainty [84].

The aforementioned special issue of *Decision Analysis* devoted to IDs is a sign of the lively interest in IDs. Boutilier [10] discusses the profound impact that IDs have had on artificial intelligence. As a professional decision analyst, Buede [15] reports on the value of IDs for tackling challenging real decision problems and considers IDs almost as indispensable as a laptop computer. Pearl [77] recognizes the significant relevance of IDs but he underscores some limitations. First, due to their initial conception with emphasis on subjective assessment of parameters, econometricians and social scientists continued using traditional *path diagrams* where parameters were inferred from the data itself. Second, artificial intelligence researchers, with little interaction with decision analysis researchers at that time (early 1980s), established conditional independence semantics through the d-separation criterion developing competitive computational tools. Thus, although IDs are informal precursors to Bayesian networks, the former had a milder influence on automated reasoning research than the latter. Finally, Pauker and Wong [75] consider that IDs have disseminated slowly in the medical literature ([74] and [66] are two papers analyzing the use of IDs for structuring medical decision problems), compared to the dominating model of decision trees, the reasons remaining unclear.

In a separate paper, we concentrate on the qualitative graphical structure of a decision problem including information constraints [5]. Here, we concentrate on the construction of a utility model and review some lesser known issues about constructing probability models. In constructing a probability model, we need to identify the relevant chance variables, the qualitative structure of conditional independencies between the chance variables, and the quantitative parameters of the joint probability distribution of all chance variables that respects the conditional independence relations among the variables. This part of an ID is also called a Bayesian network (BN). When we have a large set of variables, constructing a BN model of the uncertainties can be a challenge.

One way to construct a BN model is by knowledge engineering using a domain expert. The domain expert can identify the relevant uncertainties, the structure of conditional independencies among the variables, and finally the numerical parameters of the joint distribution. To facilitate the knowledge engineering, we describe the SRI protocol developed by the Decision Analysis group at Stanford University. We also describe some methods for reducing the number of parameters needed to fully describe a joint probability distribution. If the conditional distribution of a binary chance variables has n parents, say with 2 states each, then the number of parameters needed is 2^n. However, if there are no interactions among the n parents, we can reduce the number of parameters of the conditional distribution to $o(n)$. We describe some techniques such as divorcing parents and noisy-OR models that have been proposed in the literature.

Another way to induce a BN model is from data. In the last two decades, there has been an explosion of techniques in the machine learning community to learn BN models from data and these techniques are rather well-known and will not be reviewed here. In practice, a combination of expert knowledge and data are used to construct a BN model.

Construction of a utility model is as challenging as constructing a probability model, if not more. Again, this can be done with the help of a domain expert or from a data set, assuming one is available. The task consists of describing the objectives in terms of a hierarchy of sub-objectives, defining a measurement scale for each sub-objective, and