MCDM 2004, Whistler, B. C. Canada August 6-11, 2004

A BAYESIAN NETWORK BASED FRAMEWORK FOR MULTI-CRITERIA DECISION MAKING

Wiboonsak Watthayu, Yun Peng
Department of Computer Science and Electrical Engineering
University of Maryland, Baltimore County
Baltimore, MD, 21250
{wwatth1, ypeng}@csee.umbc.edu

Keywords: MCDM, Uncertainty, Bayesian networks, influence diagrams

Summary*: Multi-Criteria Decision Making (MCDM) involves the selection of the best actions from a set of alternatives, each of which is evaluated against multiple, and often conflicting, criteria. Most of the existing MCDM methods only focus on decisions under certainty. The criteria were evaluated separately as if they were independent of each other. Complex, often uncertain interactions between criteria, and between criteria and other factors are not modeled in a coherent and systematic manner. To address these issues, we propose in this paper a decision framework based on Bayesian networks (BN) and influence diagram (ID) to structure and manage MCDM problems with explicit modeling of uncertain interactions among entities of interest. In this framework, a decision problem is represented by an ID where each decision node represents the set of alternatives for a decision, a utility node represents the set of objectives (decision maker's preferences), decision criteria and internal or external factors that may affect the criteria are represented by chance nodes. Interdependencies among these nodes are qualitatively modeled by the links in the diagram and quantitatively by conditional probability tables (CPT) associated with each of the chance nodes and the utility node. The joint probability distribution, which is compactly captured by the network structure and CPT, encodes the domain expert's knowledge of interdependency between variables. The decision problem is then treated as an optimization problem: recommend the decision alternative which optimizes the expected utility, given observations of some external factors and preferences made by the decision maker. Various algorithms developed for BN and ID can be employed to automatically solve this problem. The steps that need to be taken to model a MCDM problem as an ID is presented, illustrated with a running example. Other related issues are also discussed. Our preliminary work indicates that this framework is of great potential as a modeling tool to support MCDM decision making in an uncertain environment.*

1. Introduction

The Multi Criteria Decision-Making Method (MCDM) involves "making preference decisions (such as evaluation, prioritization, selection, and so on) over the available alternatives that are characterized by multiple, usually conflicting, criteria" (Hwang & Yoon, 1981). During the past three decades, considerable progress has been made in multiple criteria decision analysis—the modeling and solving of MCDM problems. As electronic commerce becomes increasingly global in recent years, the process of decision-making has become more complicated because there are more alternatives to choose from and more factors to be considered in the process. This requires further advance of decision-making methods for MCDM. The advancement of information technology (IT), which makes globalized e-commerce possible, also provides various technologies for new, and potentially more powerful MCDM methods.

Most existing MCDM methods, when attempting to structure very large and complex decision models, often focus on decisions under certainty. That is, the parameters and their values were known with certainty. Moreover, most of them assume that the relevant criteria are certain and independent of each other. However, in most complex real world decision problems decision criteria and their interrelations are often interdependent in a complex and uncertain way. Moreover, the value of a criterion for a given action can be affected by many factors which are external to the decision system and cannot be controlled by the decision maker. For instance, in a supplier selection problem in a supply chain management system, the price, which is a criterion, can be influenced by external factors such as supply and demand of that product in the market, which are further dependent on other economical and non-economical factors (e.g., interest rate, opening of new markets, labor dispute, natural disaster, terrorist attack, etc.).

This paper focuses on modeling uncertainty in MCDM and presents our ongoing work on developing a framework based on Bayesian network (BN) and its extension influence diagrams (ID), for MCDM.

BN and ID are developed in the artificial intelligence (AI) community as principled formalisms for representing and reasoning under uncertainty in intelligent systems. In a BN, entities of interest (e.g., decision criteria and sub-criteria, factors that influence them) are treated as random variables and represented as nodes in the network, connected by directed arcs indicating probabilistic dependencies between them. The network structure, together with conditional probability tables associated with each node, provide a compact representation of the joint probability distribution of all variables. A suite of algorithms have been developed for probabilistic inference with BN, especially those which, when some variables' values have been observed, compute the posterior probabilities of others. ID extends BN by adding utility nodes and decision nodes, allowing one to perform various decision related tasks, including computing the expected utility, given observations and decision choices, and finding the optimal. BN and ID thus provide a theoretically well-founded and operational basis for modeling MCDM and problem-solving.

The presentation of our work on developing a BN based framework for MCDM in this paper is organized as follows. In Section 2, the background of MCDM and brief critical review of existing MCDM methods are provided. Bayesian networks and influence diagrams are discussed in Section 3. Section 4 outlines the proposed framework, together with a running example. Finally, conclusions and further research are discussed in Section 5.

2. Multi Criteria Decision Making (MCDM)

MCDM refers to making decisions in the presence of multiple criteria. In the essence, a MCDM problem is formed into hierarchy composed of four elements: the *goal*, the *objectives*, the *criteria*, and the *alternatives*. These elements can be presented in a matrix format. Let $A = \{a_1, \ldots, a_m\}$ be a set of decision alternatives and $C = \{c_1, \ldots, c_n\}$ a set of criteria according to which desirability of an action is judged. A decision matrix D is an $m \times n$ matrix, in which element d_{ij} indicates the performance of alternative a_i, evaluated against the decision criterion c_j. It is often assumed that the decision maker has determined the weights of relative importance of the decision criteria, $W = \{w_1, \ldots, w_n\}$ (Zimmermann, 1996). The total score for each alternative is obtained by the following formula:

$$S_i = \Sigma_j w_j d_{ij}$$

When the overall scores are calculated for all the alternatives, the one with the highest score is chosen.

During the past three decades, along with the rapid development of information technologies, considerable progress has been made in multiple criteria decision analysis—the modeling and solving of MCDM problems, and many MCDM methods have been developed (see Hwang & Yoon, 1981 and Steward, 1992 for a survey). However, some of the existing methods have been criticized as ad hoc and, to certain degree, unjustified on theoretical and/or empirical grounds. These methods can be divided into three categories: multiple attribute utility theory (MAUT), outranking methods, and interactive methods. The first category, MAUT, consists of methods that aggregate different points of view into a utility function, which must subsequently be optimized. When attempting to optimize the unified utility function, often a strict order over the set of all possible decisions is defined based on some assumptions. The problem for methods in this category is that interdependence of attributes is not modeled. Since all criteria are considered independent of each other, a single attribute utility function must be defined for each criterion. The second category, the outranking method, aims at building a relation, called an *outranking relation*, among the set of alternatives. Given two alternatives A_i and A_j , A_i outranks A_j, if given all that is known about the two alternatives, there are enough arguments to decide that A_i is at least as good as A_j. The goal of outranking methods is to find all alternatives that dominate other alternatives while they cannot be dominated by any other alternative. The weak points of methods in this category are twofold: (1) to find the best alternatives, the criteria weights are always assumed to be measured on some scale, which is not practical in real world decision problems; (2) a complete ranking of alternatives may not be achieved, often only a partial prioritization of alternatives is computed. The best these methods can achieve is to reduce the number of alternatives to be considered. The final category, the interactive method, involves incrementally narrowing down the set of possible decisions by interactive techniques (i.e., after each "round" the decision maker is asked to input additional information). This method allows the decision-maker to actively engage in expressing requirements and decision criteria, and choosing among available alternatives in a sensible way. Although this method does not offer "optimality", it does prevent most of the inconsistencies commonly associated with such decision-making.

Existing methods for MCDM reviewed in this section are popular because of their ease of implementation and intuitiveness. However, they are not applicable when the problem becomes large and complex, especially in uncertainty situations because they always make the assumptions that relevant criteria are well defined (e.g., for a given action *a* it is obvious how to compute *f*(*a*) for a given criteria *f*), and certain (e.g., for a given action *a* and criteria *g* the value *g*(*a*) is deterministic rather than stochastic) (Fenton, 2000). In addition, in real world decision problems, uncertainty can be caused by incomplete or noisy information, the conflict among criteria, the uncertainties in subjective judgment, and different preferences among different decision makers and so on. Better decision making can be achieved if the uncertain interrelations among these decision elements can be explicitly modeled and reasoned with rather than ignored by some unrealistic assumptions or summarized out by subjective weighting schemas or heuristic rules as did in many of the existing MCDM methods.

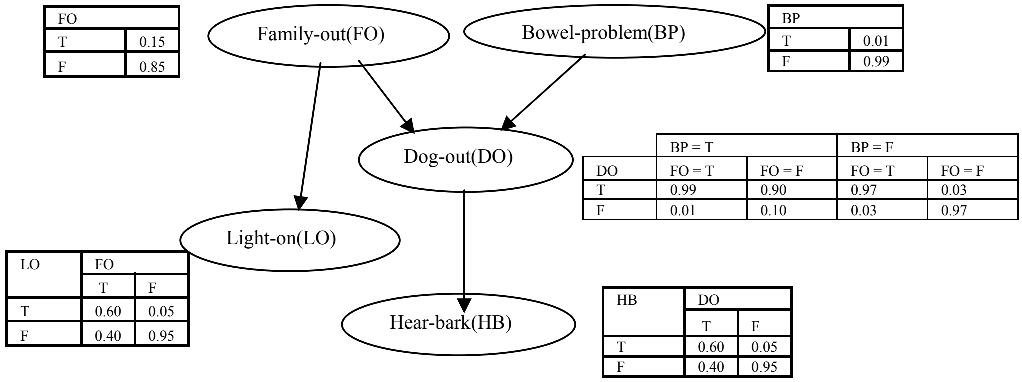
3. Bayesian Networks and Influence Diagrams

3.1 Bayesian Networks

Bayesian networks (BN), are widely used for knowledge representation and reasoning under uncertainty in intelligent systems (Pearl 1988, Russell & Norvig 1995). In a general form, the structure of a BN is a directed acyclic graph (*DAG*) in which nodes correspond to random variables of interest and directed arcs represent *direct* causal or influential relation between nodes. The uncertainty of the interdependence of the variables is represented locally by the conditional probability table (*CPT*) $\Pr(x_i \mid \pi_i)$ associated with each node x_i, where π_i is the parent set of x_i. An independence assumption is also made with BN that x_i, given its parents π_i, is independent of any other variables except its descendents. The graphical structure of BN allows an unambiguous representation of interdependency between variables. This, together with the independence assumption, leads to one of the most important features of BN: the joint probability distribution of $X = (x_i, \ldots, x_n)$ can be factored out as a product of the conditional distributions in the network,

$$\Pr(X = x) = \prod_{i=1}^{n} \Pr(x_i \mid \pi_i).$$

Figure 1 below gives an example BN, "Family Out", where *Family-out* and *Bowel-problem* are direct causes for dog going out, and *Dog-out* directly affects whether we hear barking or not. Once we know whether the dog is out or not, the probability of our hearing barking has nothing to do with Family-out and Bowel-problem because their influence on hearing barking is blocked by the instantiation of *Dog-out* = *T*. Each node in the network has a conditional probability table (CPT). Each column in the table contains the conditional probabilities of all possible values of that variable, given a possible combination of values of its parent nodes.



FO	
T	0.15
F	0.85

BP	
T	0.01
F	0.99

DO	BP = T		BP = F	
	FO = T	FO = F	FO = T	FO = F
T	0.99	0.90	0.97	0.03
F	0.01	0.10	0.03	0.97

LO	FO	
	T	F
T	0.60	0.05
F	0.40	0.95

HB	DO	
	T	F
T	0.60	0.05
F	0.40	0.95

Figure 1 The “Family out” Bayesian Network (Charniak, 1991). Various probabilistic queries can be answered. For example, P(HB), the prior probability of hearing barks is computed as 0.148, and the posterior probability P(FO=T|HB=T $\wedge$ LO=F) = 0.689.

With the joint probability distribution, BNs can support in theory any probabilistic inference in the joint space. Moreover, although it has been proven that inference in BN with general *DAG* structure is *NP*-hard (Cooper, 1990), probabilistic inference algorithms that are more efficient than the brute force use of the gigantic joint probability table have been developed by exploring the interdependency captured by the network structure. Most important among them are algorithms for computing posterior probabilities $\Pr(x_i \mid e)$, where *e* denotes evidence, the observed values for some variables. These class of algorithms include “belief propagation” (Pearl, 1988) and “Junction tree” (Shafer, 1996, Jensen, 1990, and Madsen, 1998) for exact solutions, and various statistical sampling techniques (e.g., Markov Chain Monte Carlo sampling) for approximate solutions with extremely large BN (see Castillo, 1997 for a detailed explanation of the most commonly used BN inference algorithms).

3.2 Influence Diagram

Influence diagrams for solving decision problems extend BN with two additional types of nodes, namely *decision nodes* and *utility nodes*. Nodes for the random variables in the BN are called *chance nodes* in ID. A decision node defines the action alternatives considered by the user. Every decision node has a finite number of alternatives standing for the actions that the decision maker can take to achieve the desired outcome. A decision node is connected to those chance nodes whose probability distributions are directed affected by the decision. A utility node is a random variable whose value is the utility of the outcome. Like other random variables, a utility node holds a table of utility values for all value configurations of its parent nodes. In an influence diagram, the value of each decision variable is not determined probabilistically by its predecessors, but rather is imposed from the outside by the decision maker to meet some optimization objective (Jensen, 1995).

In an ID, let $A = \{a_1, a_2, \ldots, a_n\}$ be a set of mutually excusive actions, and *H* the set of determining variables. A utility table *U(A,H)* is need to yield the utility for each configuration of action and determining variable in order to assess the actions in *A*. The problem is solved by calculating the action that maximizes the expected utility:

$$EU(a) = \prod_H U(a,H)P(H \mid a)\ ,$$

where *U*(*a,H*) are the entries of the utility table in the value node *U*. The conditional probability *P*(*H*|*a*) is can be computed from CPT of the variable $h_i \in H$, given the action *a* is fired.

Figure 2 below represents an ID about weather and a decision to carry an umbrella (Jensen, 1995). *Forecast* and *Weather* are chance nodes containing the probabilistic information about the weather and forecast. *Satisfaction* is a utility or value node. *Umbrella* is a decision node. The objective is to maximize expected Satisfaction by appropriately selecting values of Umbrella for each possible forecast. In addition to probabilities, the values of Satisfaction for each combination of Umbrella and Weather are also given.

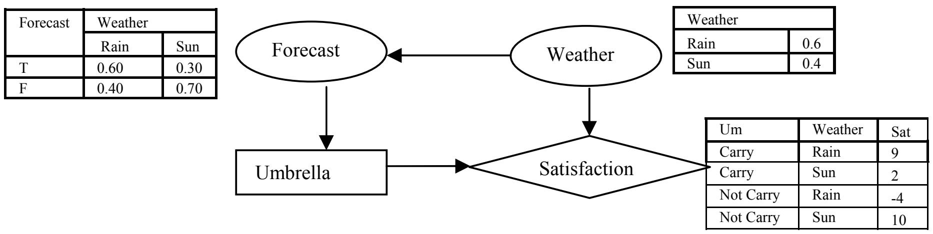


Figure 2 An example of influence diagram (Jensen, 1995)

The algorithm for evaluating an ID goes as follows (Russell and Norvig 1995): (1) set the evidence variables for the current state, (2) for each possible value of the decision node, set the decision node to that value; (3) calculate the posterior probabilities for the parent nodes of the utility node using a standard probabilistic inference algorithm, and (4) calculate the resulting utility function for the action and return the action with the highest utility.

4. Proposed Framework

In general, the decision basis for a MCDM framework is composed of the components that represent states, relationships, alternatives, preferences, and interrelations between them. The framework we developed allows the representation of a decision basis capturing the uncertain interrelations using BN and ID and provides mechanisms for decision-making based on this representation.

First, we start with a discussion on how BN could be used to model the MCDM problem. When applying the standard terminology of MCDM from Section 2 (Vincke, 1992), we need to identify all variables involved, which include the set of possible actions (set of alternatives), the set of criteria, and the set of constraints (properties of the criteria), and other factors that affect them. Therefore, our MCDM framework will be composed of three types of node (chance node, utility node, and decision node), as depicted in the MCDM hierarchy in Figure 3 below.

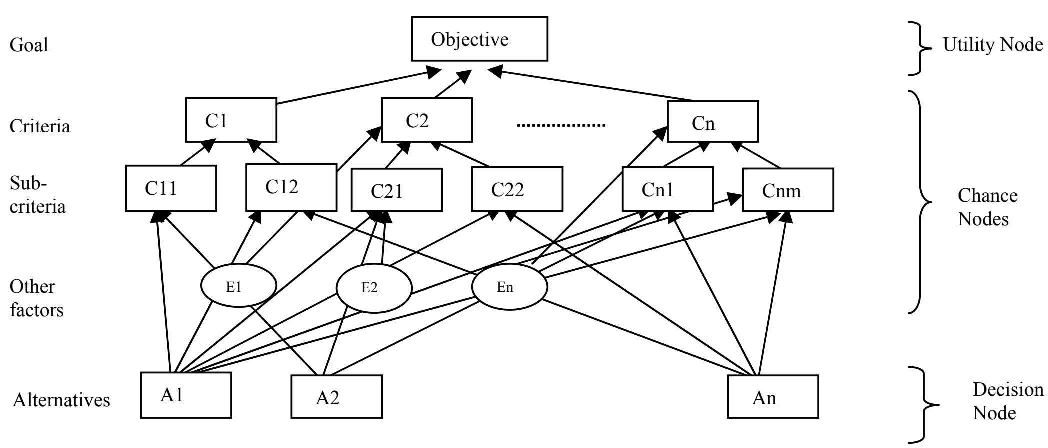


Figure 3 MCDM Hierarchy

As mentioned earlier, the *decision nodes* represent the set of alternatives $\{A_i, \ldots, A_n\}$, the *utility node* represents the set of objectives (decision making's preferences) to be optimized, and the *chance nodes* include the set of criteria and sub-criteria $\{C_1, \ldots, C_n, C_{11}, ..., C_{nm}\}$. These criteria may interact with each other. They may also be affected by various factors $\{E_1, \ldots, E_n\}$, either internal (those you may be able to control) or external (those you cannot control), which are also modeled as chance nodes. Interdependencies among these nodes are represented as directed arcs.

A simple made-up MCDM problem is created for preliminary emprical evaluation of the basic ideas of the frmaework, it will be used for illustration purpose throughout this paper. A wide variety of BN software programs are available (Murphy, 2002), among them we have chosen Netica software (Norsys, 1998) for demonstration in this paper. The example problem is briefly decribed below.

Example: *Mr. A is trying to decide what time to leave from his home to catch the 8 a.m. flight to NYC and what mode of transportation to use with reasonable cost/comfort. The modes of transport include car, taxi, bus, and train. For simplicity, choices of leave time are grouped into one hour intervals between 4a.m. to 9a.m. Decision criteria include cost, comfort, safety, and the waiting time at the airport. Each of these criteria is influenced by some other factors, including both internal factors (e.g., number of luggage to carry), and external ones (e.g., weather condition, roadwork, and train problems).*

4.1 Building a Network

4.1.1 Identifying uncertain criteria, factors, and their causal relations

This step involves identifying the variables of interest and determining the interdependencies between them. Variables judged to be direct causes of another variable are connected with directed arcs following a causal ordering. The result of this step is the lay out of the network structure (DAG). Extensive interaction with decision maker and other domain experts is imperative to ensure a good network structure. We are actively working on developing procedures and rules that may help the modeler to identify relevant variables and their causal associations. For our example problem, we have manually identified 21 variables, including 2 decision nodes and one utility node (see Figure 4). All nodes can be classified into objectives, actions, criteria, constraints, and factors as shown in Table 1.

	Daily Traveling Problem
Objectives	To get to BWI in time to catch the NYC flight with reasonable cost/comfort/waiting-time
Decision problem	To determine the most suitable mode of transport and start-time
The set of possible actions	The set of pairs of action alternatives in the form of (A, B) where A is the transport type (car, bus, taxi, train) and B is a start time (4-5a.m., 5-6a.m., 6-7a.m., 7-8a.m., 8-9a.m.)
Criteria	Cost, comfort, waiting time, traffic safety, arrival time, and adjust journey time
Constraints (properties of criteria that he specifies as desirable)	Waiting time > 15 minutes (otherwise he misses the flight) Cost < $50
External factors (variables you cannot control, but which can influence the value of criteria for a given action)	Roadwork, Train problem, Weather
Internal factors (variables you may be able to control and which can influence the value of criteria for a given action)	Number of luggage, flight schedule, congestion delay, train delay.

Table 1 Classification of each variable for daily travelling problem (Fenton, 2000)

The uncertain criteria are criteria that depend stochastically on other factors. For example, adjusted journey time for a specific choice of action will vary according to weather condition or if there are train problems or roadwork, and the amount of delays that may occur is also considered as a random variables, whose distribution is dependent on the time of travel (e.g., rush hour or not).

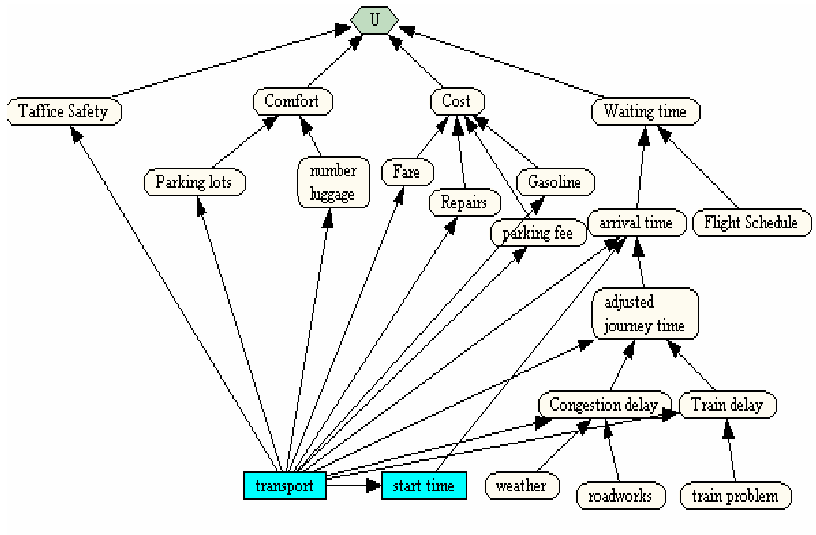


Figure 4 Influence diagram for daily travelling problem

Figure 4 shows the example of this framework. The two decision nodes (transport, start-time) are represented by the rectangle shape. Their domains are all possible alternatives. The arc going from "transport" to "start-time" is the only sequence arc in this ID. In this case, the first decision, the transport mode, also affects the starting-time decision. The criteria, sub-criteria and other factors are chance node, represented by the oval shape. In this example, waiting time, cost, and comfort, and traffic safety are four primary criteria. Each of these criteria has sub-criteria, and factors that may affect them. Here, the criteria are an effect of sub-criteria (e.g., the arrival time affects the waiting time), factors (e.g., the flight schedule affects the waiting time), and some external factors (e.g., weather and road works affect the congestion delay). The utility node, *U*, represented by a diamond shape, is a utility function that measures the degree the performance goals is achieved for each alternative. The utility function can be normalized, meaning that the most desirable state has a utility of 1, and the least desirable has a utility of 0.

4.1.2 Generating Conditional Probability Table (CPT)

Once the network structure is completed, the probabilities are entered into the network in the form of the CPT for each chance node in the network. These tables can be constructed by either subjective estimate from the experts or learned from case data. For variables with no parents, this task is simple; its CPT is reduced to its prior probability. The probability tables of variables with parents are more complicated, as these tables can easily grow to large number of entries (exponential to the number of parent nodes). The small tables can be filled directly. Techniques exist to ease the construction of large tables under certain conditions. Also, a large table may be break into smaller ones by introduction of additional nodes (called "hidden nodes" (Neapolitan, 2004). In this example network, all of the tables are small enough to be populated by hand, one of which (for variable "cost") is shown in Figure 5 below.

repair	gas	parkingfee	fare	high	low
cheap	true	true	cheap	25.000	75.000
cheap	true	true	expensive	35.000	65.000
cheap	true	false	cheap	30.000	70.000
cheap	true	false	expensive	25.000	75.000
cheap	false	true	cheap	10.000	90.000
cheap	false	true	expensive	5.000	95.000
cheap	false	false	cheap	0.000	100.00
cheap	false	false	expensive	10.000	90.000
expensive	true	true	cheap	80.000	20.000
expensive	true	true	expensive	100.00	0.000
expensive	true	false	cheap	75.000	25.000
expensive	true	false	expensive	90.000	10.000
expensive	false	true	cheap	65.000	35.000
expensive	false	true	expensive	70.000	30.000
expensive	false	false	cheap	60.000	40.000
expensive	false	false	expensive	65.000	35.000

Figure 5 The CPT for node "cost"

4.1.3 Utility Function

The use of the maximum expected utility assumes easy access to the decision maker's utilities for all relevant outcomes. The utility values can be obtained by consulting experts and from the preference of decision makers. Generally, the decision maker has to assign the preference value for utility value by manual calculation. In this example, the utility function is assumed to obtain from the preference of decision makers. Therefore, we have to assign the utility function for each combination of the criteria. Then these values are normalized at this stage to be in the range from 0 to 1. If the utilities could be normalized (bring the utilities to a 0-1 scale through linear transformations), the problem of solving IDs could be reduced to the problem of inference in BN. And then the utility value to normalize it in the range from 0-1 could be used. Therefore, we normalized over all sate values in this example by

Using this formula, $U_i = 1 - \dfrac{X_{\max} - X_i}{X_{\max} - X_{\min}}$ (Blank, 1980)

For example, TrafficSafe, Comfort, Cost and Waiting time are assigned to some value in range 0-10 based on the preference of decision makers i.e. if Cost = "high", Comfort = "good", TrafficSafe = "safe", and waiting time = "Time15mins", then the suitable value for this state can be "5". Supposing the maximal utility value is 10 and the minimal utility value is 0. Therefore, the normalized utility value would be 1 – (10-5)/(10-0) = 0.5". The utility value used in the example is shown in Figure 6.

cost	comfort	TrafficSafe	waiting	U
high	good	safe	Time15mins	0.5
high	good	safe	Time30mins	0.65
high	good	safe	Time60mins	0.4
high	good	safe	Time90mins	0.35
high	good	safe	Time120mins	0.2
high	good	safe	GT120mins	0.15
high	good	safe	MissFlight	0
high	good	unsafe	Time15mins	0.4
high	good	unsafe	Time30mins	0.45
high	good	unsafe	Time60mins	0.3
high	good	unsafe	Time90mins	0.25
high	good	unsafe	Time120mins	0.15
high	good	unsafe	GT120mins	0.1
high	good	unsafe	MissFlight	0
high	good	notsure	Time15mins	0.45
high	good	notsure	Time30mins	0.5
high	good	notsure	Time60mins	0.45
high	good	notsure	Time90mins	0.35
high	good	notsure	Time120mins	0.3
high	good	notsure	GT120mins	0.2
high	good	notsure	MissFlight	0
high	fair	safe	Time15mins	0.35
high	fair	safe	Time30mins	0.4
high	fair	safe	Time60mins	0.3
high	fair	safe	Time90mins	0.2
high	fair	safe	Time120mins	0.15

Figure 6 The Example of Utility Function

4.2 Updating Beliefs and Making Decisions

Once a BN/ID is constructed, it can be used to make various inferences about the variables in the model. Root node (those without parents) is often variables for external factors, whose values may be observed and cannot be changes. When no observation is made, they are characterized by their prior probabilities. The first inference we can do with the BN/ID framework is to compute the prior distributions of other variables and the prior distribution of the utility, based on the priors of the roots and CPT of other nodes. Figure 7 below shows such computation result for our example problem. It turns out that in the transport decision node, *train* has the highest prior expected utility value (0.64114) when no observation is made.

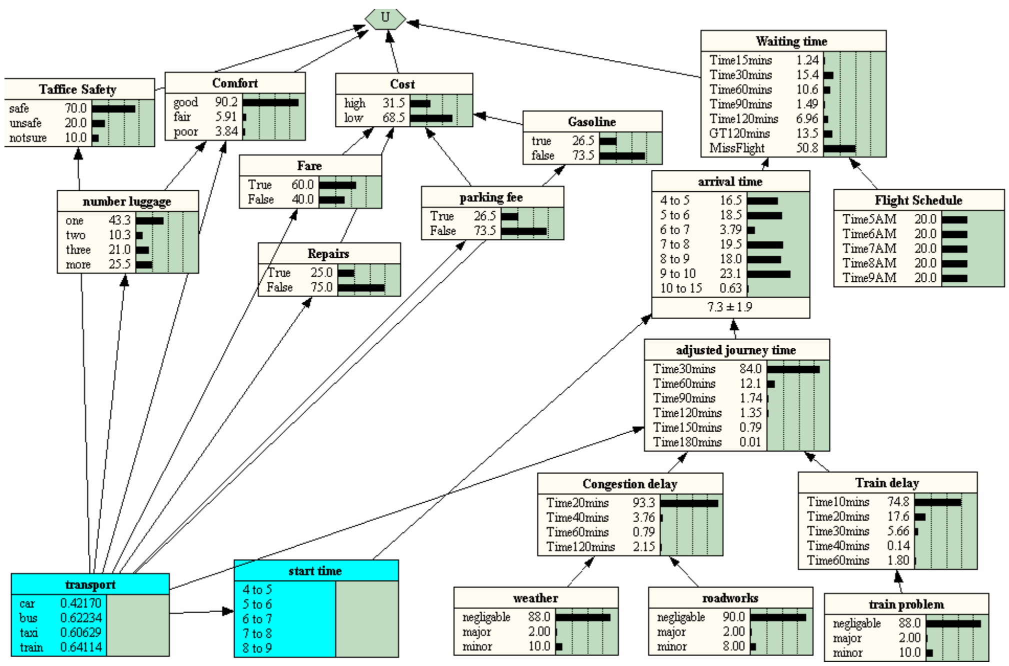


Figure 7 The prior distributions of variables in the example network.

If we observe the values of some of the observable variables (e.g., weather), the corresponding variables in the network are instantiated (and clamped) to the observed values. The influence of these observations is propagated throughout the network to every other node, causing them to update their beliefs to become the posterior probabilities, given observations. In other words, the joint distribution of the variables changes each time we learn new information about the observable variables. The belief update happens in the same way when values of variables representing internal factors is selected (e.g., number of luggage = 3). Figure 8 shows the network after evidence has been entered and the probabilities updated. For example, Mr. A wants to take a flight of departure time 8 a.m. with three pieces of luggage and the lowest cost. Therefore, the flight schedule node is instantiated to “Time8AM”, the number of luggage node to “three” and the cost node to “low”, respectively. The solution for the most optimal way to achieve Mr. A’s objective turns out to be traveling by *car* with the expected utility value of 0.9180 as shown in Figure 8.

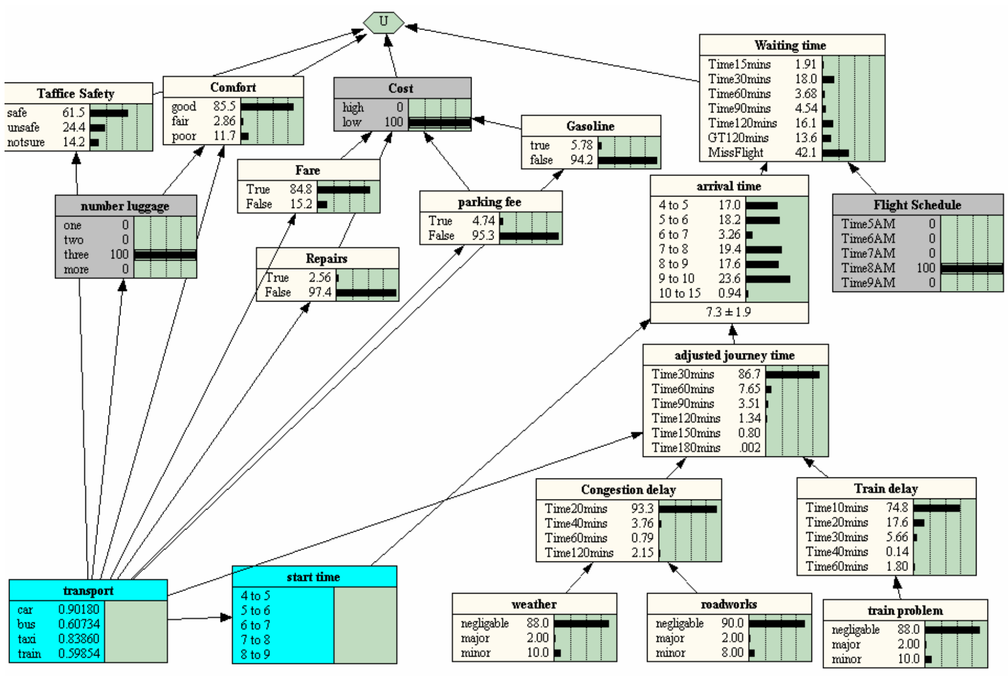


Figure 8 the optimal solution by entering evidence for daily traveling problem

In other case, if Mr. A has three luggage with same flight schedule (8 a.m.). Supposing he also made observations of three external factors (major problems for weather, roadwork, and train), *Taxi* is chosen as it has the highest expected utility value of 0.76607, as shown in Figure 9.

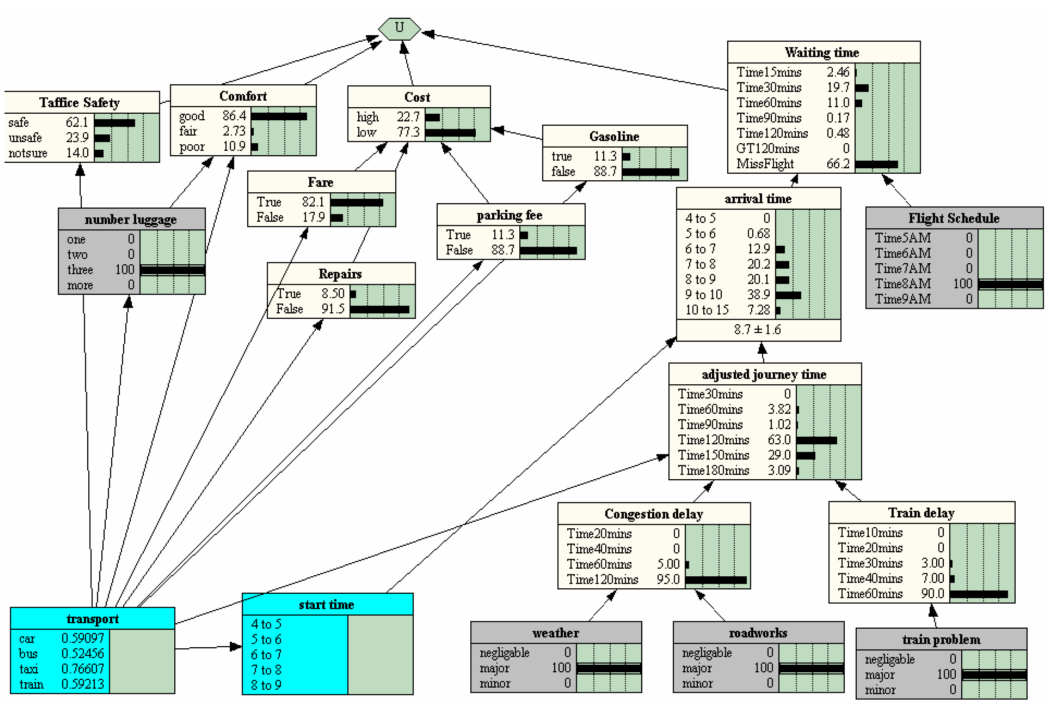


Figure 9 the optimal solution by entering evidence for daily traveling problem

5. Conclusions

In this paper we presented our ongoing work on developing a computational framework for a BN based decision support system for MCDM problem in uncertain environment. The hypothesis in this paper is that Bayesian networks, if applied to MCDM problems, may improve the effectiveness and accuracy of problem solving. The proposed MCDM framework provides a new technique for structuring MCDM problems. The advantages of our BN based MCDM framework stem from the fact that the Bayesian network allows us to explicitly model the probabilistic interdependency between all elements involved in a MCDM problem (the decisions, criteria, utility, and other factors). Various BN inference algorithms are readily available and can be directly used for typical decision tasks such as recommending the optimal solution. The simple example of travel planning shows the applicability and efficiency of this methodology.

Work presented here is only the first step of our effort toward a comprehensive solution to this very complex problem. Several issues need to be addressed in order to transform our framework from conceptual to one that is workable in real world situations. First, we need to develop a set of procedures and rule that can interactively guide the user step by step to identify the variables and their interdependencies in designing and refining the network model. Similar tools are also needed for constructing CPT. This is especially important for constructing the utility table. The utility values we used in this paper were hand picked, which would be prohibitively time consuming when dealing with large, complex problems. Focus of research on this issue will be different approaches to reduce the table size, using techniques such as decomposing tables with introduction of hidden nodes, and hybrid approach that combines deterministic, functional, and probabilistic means. We will also investigate using machine learning techniques for CPT construction. Secondly, we will investigate how to incorporate BN sensitivity analysis algorithms into our framework to check whether the best decision is sensitive to small changes in the assigned probabilities and utilities value, and more importantly, what one shall do when it is determined the solution is too sensitive to some probability values in such a highly interactive system. The third issue concerns the computational complexity of BN inference, which has been shown to be NP-hard (e.g., it may in worst case take time exponential to the network size, and thus computationally intractable for MCDM problems involving large number of elements). A promising approach is to combine the techniques of autonomous software agents and multi-sectioned BN to decompose a large BN into smaller subnets, each of which is managed by an agent (Xiang 2002). We also plan to experiment with various approximate inference algorithms to see if they provide acceptable accuracy.

References

Hwang, C.L., Yoon, K., (1981) *Multiple Attribute Decision Making: Methods and Applications*, Springer-Verlag, Berlin.

Pearl, J., (1988) *Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference*. Morgan Kaufmann Publishers, ISBN 1-55860-479-0.

Stewart T. J., (1992) “A critical survey on the status of multiple criteria decision making theory and practice.” OMEGA International.

Russell, S., and Norvig, P., (1995) *Artificial Intelligence: A modern approach*. Prentice Hall, ISBN 0-13-103805-2.

Cooper, G.F., (1990) “The computational complexity of probabilistic inference using Bayesian belief networks”, Artificial Intelligence, 42, p393-405.

Jensen, Finn V., (1995) “Cautious Propagation in Bayesian Networks.” Proceedings of the Eleventh Confernece on Uncertainty in AI (UAI-95), Morgan Kaufman, San Mateo, CA, 323-328.

Vincke, P., (1992), Multi-*criteria Decision-Aid*. Wiley, Chichester.

Chan, H. and Darwiche, A., (2002), “When do Numbers Really Matter?” Journal of AI Research, 17.

Norsys (1998) *Netica Application*, Norsys Software Corp., http://www.norsys.com, 1998.

Zimmermann, H.-J., (1996), Fuzzy Set Theory and Its Applications, Kluwer Academic Publishers, Third Revised Edition, Boston, MA, USA.

Fenton, N. and Neil, M., (2000) "Making Decisions: Using Bayesian Nets and MCDA," Computer Science Department, Queen Mary and Westfield College, London.

Shafer, G., (1996) *Probabilistic Expert Systems*. CBMS-NFS regional Conference Series in Applied Mathematics; SIAM.

Jensen, Finn V., Olesen, K. G., and Anderson, S. K., (1990) "An Algebra of Bayesian Belief Universe for Knowledge-based System": In Networks, 20:637-660.

Madson, A. L. and Jensen, F. V., (1998) "Lazy propagation in junction trees", in proceedings of the 14th Conference on uncertainty in Artificial Intelligence, 1998.

Castillo, E., Gutierrez, J. M., and Hadi, A. S., (1997) *Expert Systems and Probabilistic Network Models*: Springer, 1997.

Murphy, K. P., (2002) "Software Packages for Graphical Models: Bayesian Networks",

http://www.cs.berkeley.edu/~murphyk/Bayes/bnsoft.html, 31 March 2004.

Neapolitan, R. E., (2004) *Learning Bayesian Networks*, Pearson Prentice Hall, 2004.

Xiang, Y., *Probabilistic Reasoning in Multiagent Systems: A Graphical Model Approach*, Cambridge University Press, 2002.

Blank, L. *Statistical procedures for engineering, management, and science*. McGraw Hill, New York, 1980.