

# Solving influence diagrams: Exact algorithms

Ross D. Shachter  
Management Science and Engineering  
Stanford University  
Stanford, CA 94305, USA  
shachter@stanford.edu

Debarun Bhattacharjya  
Business Analytics and Math Sciences  
IBM T. J. Watson Research Center  
Yorktown Heights, NY 10598, USA  
debarunb@us.ibm.com

July 1, 2010

## Abstract

Influence diagrams were developed as a graphical representation for formulating a decision analysis model, facilitating communication between the decision maker and the analyst [1]. We show several approaches for evaluating influence diagrams, determining the optimal strategy for the decision maker and the value or certain equivalent of the decision situation when that optimal strategy is applied. They can be converted into decision trees, evaluated directly as influence diagrams, or converted into another graphical model, a rooted cluster tree, for analysis.

Influence diagrams [1] are popular graphical models in decision analysis for representing, solving and analyzing a single decision maker's decision situation. Many of the important relationships among uncertainties, decisions and values can be captured in the structure of these models, explicitly revealing irrelevance and the timing of observations. The phrase "solving influence diagrams" refers to computing the optimal strategy and the value at the optimal strategy, based on the norms of decision analysis. In this article, we highlight some of the most popular exact methods for solving influence diagrams.

Influence diagrams were initially conceived as tools primarily to be used in the formulation stage to assess the beliefs of a decision maker. However, research has shown that they are also efficient computational tools for the evaluation stage, and not merely graphical depictions for communicating ideas between the decision maker and the analyst [2][3]. There have been many developments in solution and analysis since then [4][5][6][7][8][9][10][11][12][13].

Methods for the solution of decision analysis problems have mainly been based on three graphical representations: decision trees, influence diagrams, and cluster trees. Decision trees are commonly taught in a first course in decision analysis. They are able to capture the asymmetries in real-world decision problems, and are easy to understand since they capture the informational order of variables in decision situations. On the other hand, influence diagrams capture the conditional independencies in the model and are able to represent the model as conveniently assessed by the decision maker. Cluster trees are a solution representation derived from influence diagrams and will be presented in Section 4. In the next section we introduce the terminology and notation for influence diagrams used in this article.

## 1 INFLUENCE DIAGRAMS

### 1.1 Node and arc semantics

To describe a decision situation, a decision maker defines a number of *variables*. When a variable is *uncertain*, it represents a feature in the decision situation that the decision maker has no control over, and its possibilities are the *states* of the uncertainty. When a variable is a *decision*, the decision maker can choose to pursue any of its *alternatives*. In this article, we will deal with decision situations where there are a finite number of states or alternatives. We allow uncertainties to be *observed* to take specific states for certain and these observations are known as *evidence*.

An influence diagram is a directed graphical model for reasoning under uncertainty with nodes corresponding to the variables. Variables are denoted by upper-case letters ( $X$ ) and their states by lower-case letters ( $x$ ). A bold-faced letter indicates a set of variables ( $\mathbf{X}$ ). We will refer interchangeably to a node in the diagram and the corresponding variable. If there is an arc directed from node  $X$  to node  $Y$ ,  $X$  is known as a *parent* of  $Y$ , and  $Y$  is a *child* of  $X$ . A node along with its parents forms the *family* for that node. The parents for node  $X$  are denoted  $\mathbf{Pa}(X)$ , and the family is  $X\mathbf{Pa}(X)$ . If there is a directed path from node  $X$  to node  $Y$ , we say that  $X$  is an *ancestor* of  $Y$ , and that  $Y$  is a *descendent* of  $X$ . Influence diagrams are *acyclic* graphs, meaning that there is no node  $X$  in the graph that is its own ancestor.

Influence diagrams have three kinds of nodes: decision, *uncertain* (or *chance*), and *value* nodes, which together represent choices, beliefs, and preferences. Decision nodes are drawn as rectangles, uncertain nodes as ovals, and value nodes as rounded rectangles (or polygons). Decision nodes correspond to decision variables that are under the control of the decision maker and uncertain and value nodes correspond to variables that are not.

The semantics of an arc in an influence diagram depends on the type of node it is directed toward. Arcs directed into an uncertain or value node  $X$  are *conditional arcs*, and the distribution assessed for  $X$  is conditioned on its parents,  $P\{X|\mathbf{Pa}(X)\}$ . For value variables, we assume that this distribution can

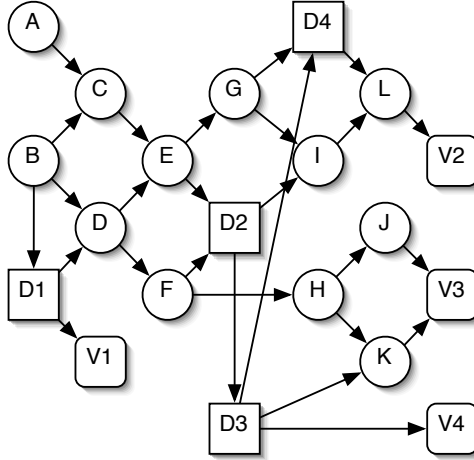


Figure 1: Influence diagram example from Jensen, Jensen and Dittmer [7].

be represented as a deterministic function of the parent variables. *Informational arcs* directed toward decision nodes indicate those variables that will be observed before the decision is made. An influence diagram is *completely specified* if states (or alternatives) are assigned for each node, as well as conditional probability distributions for all uncertain nodes.

We will demonstrate some solution algorithms with the help of an influence diagram example from Jensen, Jensen and Dittmer [7], shown in Figure 1. This influence diagram has 4 decision nodes  $D1$  through  $D4$ , four value nodes  $V1$  through  $V4$  and twelve uncertain nodes  $A$  through  $L$ .

## 1.2 Maximizing expected utility

The criterion for decision making is represented by the value nodes. In this article, we assume that there may be multiple value nodes in the influence diagram, which are summed to obtain the total value. Influence diagrams with multiple additive or multiplicative value nodes were introduced and solved by Tatman and Shachter [14]. We assume that the value nodes characterize the value of the attributes in terms of a single numeraire, which we assume is dollars, and that there is a von Neumann-Morgenstern utility function  $u(\cdot)$  applied to their total [15]. The utility function  $u(\cdot)$  is typically assumed to be strictly increasing and continuously differentiable. The *certain equivalent (CE)* of an uncertain  $\mathbf{V}$  represents the certain payment that the decision maker finds indifferent to  $\mathbf{V}$ , and is given by  $u^{-1}(E[u(\sum_{V \in \mathbf{V}} V)])$ . The most common utility functions are *linear*,  $u(v) = av + b$ , and *exponential* functions of the sum of the values,  $u(v) = -ae^{-v/\rho} + b$ , where  $a > 0$  and  $\rho > 0$ , both of which allow us to express the *value of information* [16][17] exactly in closed form. For simplicity in this

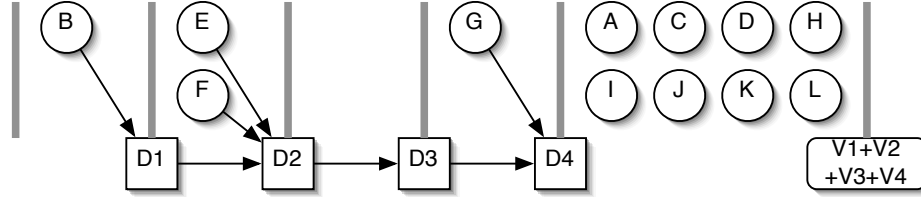


Figure 2: Decision windows for the influence diagram from Figure 1.

article, we will assume a linear utility function. Note that the linear utility function corresponds to a risk-neutral decision maker, and in this situation the certain equivalent is equal to the maximum expected total value.

Thus the general problem we describe is to determine the decisions  $\mathbf{D}$  that maximize the expected sum of values when the variables  $\mathbf{E}$  have already been observed and the parents of each decision  $D$  are observed before it is made. In other words, the goal is to find the *optimal strategy*, given that evidence  $\mathbf{E} = \mathbf{e}$  has been observed. Therefore, the solution of an influence diagram should reveal the optimal strategy and the certain equivalent of the decision situation, informing the decision maker about what to do in her decision situation, and what the decision situation is worth to her.

### 1.3 Other standard assumptions and requirements

It is inconsistent to make observations that tell us anything about the decisions we are yet to make, so we require that evidence variables  $\mathbf{E}$  not be *responsive* in any way to  $\mathbf{D}$  [18]. This is enforced by not allowing the nodes in  $\mathbf{E}$  to be descendants of the nodes in  $\mathbf{D}$ .

We also require that there be a total ordering of the decisions. In the case of the diagram in Figure 1, the decisions are made in the order  $D_1$ ,  $D_2$ ,  $D_3$ , and  $D_4$ . We assume *no forgetting*, that all earlier observations and decisions are observed before any later decisions are made. No forgetting is a standard assumption for influence diagrams, and the no forgetting arcs are usually not drawn in the diagram when they can be inferred implicitly. For example, since there is a directed path from  $D_1$  to  $D_2$  there does not need to be an explicit arc from  $D_1$  to  $D_2$ , but there should be directed arcs from  $D_2$  to  $D_3$  and  $D_3$  to  $D_4$ . That way, it is clear that  $F$  is observed before  $D_3$ .

No forgetting can also be understood using the notion of *decision windows* [19]. The total ordering of the decisions in the no forgetting ordering imposes a partial order on the other variables. Figure 2 shows decision windows for our influence diagram example, that indicate what variables are known at the time of each decision. The windows are one-way in the sense that future decisions can observe anything from earlier windows but nothing from the future. From Figure 2, note that  $B$  is observed before  $D_1$  and therefore is known at the time all

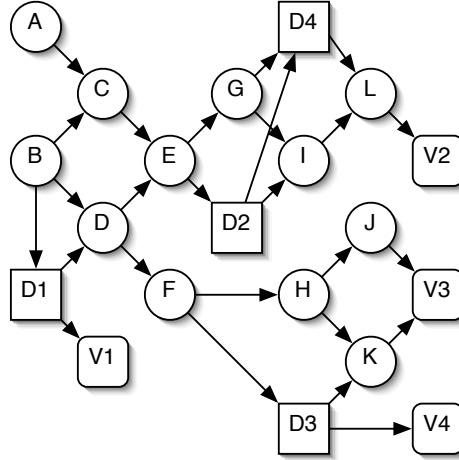


Figure 3: Requisite observations for the influence diagram from Figure 1.

decisions are made. Any evidence would also be in this window. Uncertainties  $A, C, D, H, I, J, K, L$  are not known at the time of any decision, hence they are placed in the decision window after the last decision  $D4$ .

We assume that the influence diagram is a *single component*, i.e. there is at least one path between any two nodes if we ignore the direction of the arcs. Otherwise we can solve each component independently. We also assume that there are no nodes which can be simply removed. For example, some non-responsive variables might become relevant if observed.

#### 1.4 Requisite observations

We make a quick note about *requisite* observations [5][20][21]. Evaluating an influence diagram entails finding the optimal alternative for every decision under the given observations available for the decision. However, based on the structure of the graph, only some of the observations are needed for a particular decision and there can be significant efficiency gains by recognizing which of the available observations are *requisite* for each of the decisions. The requisite observations for each decision can be computed, in reverse order, by considering which observations are relevant to the value descendants of the decision [22][20][21].

Therefore, a pre-processing step can be carried out on the influence diagram under consideration. In this pre-processing step, if an observation is not requisite for a decision, the arc from that observation into the decision is removed. Figure 3 presents the influence diagram example where pre-processing has been carried out and shows the requisite observations for every decision. Note that although Figure 3 has more arcs than Figure 1, it actually reflects a significant

reduction in graph complexity since the no forgetting arcs were hidden in Figure 1. For instance,  $B$  is available at the time of all decisions, but it is not requisite for any decision other than  $D1$ , hence there are no arcs from  $B$  to  $D2$ ,  $D3$  or  $D4$  in Figure 3.

## 2 SOLVING INFLUENCE DIAGRAMS: DECISION TREES

One way to solve an influence diagram is to convert it into a decision tree [1] and then solve the tree. To be able to do this, the influence diagram must be a *decision tree network*, where all the ancestors of any decision node are parents of that node. Equivalently, if we call an arc between uncertain nodes *non-sequential* if the parent is in a later decision window than the child, a decision tree network is an influence diagram without non-sequential arcs.

Any influence diagram satisfying our assumptions can be transformed into a decision tree network through a sequence of *arc reversals* [19]. Arc reversal applies Bayes' rule to reverse the topological order of uncertain nodes [2][3]. An arc from uncertain node  $X$  to uncertain node  $Y$  with parent sets  $\mathbf{A}, \mathbf{B}$  and  $\mathbf{C}$  can be reversed, as shown in Figure 5a, unless there is another directed path from  $X$  to  $Y$ ; if there were, then reversing the arc would create a directed cycle. When the arc  $(X, Y)$  can be reversed, the joint distribution of  $X, Y$  conditioned on  $A, B, C$  can be factored two ways:

$$\begin{aligned} P\{X, Y | \mathbf{A}, \mathbf{B}, \mathbf{C}\} &= P\{X | \mathbf{A}, \mathbf{B}\} P\{Y | X, \mathbf{B}, \mathbf{C}\} \\ &= P\{X | Y, \mathbf{A}, \mathbf{B}, \mathbf{C}\} P\{Y | \mathbf{A}, \mathbf{B}, \mathbf{C}\}. \end{aligned}$$

The new distribution for  $Y$  given  $A, B, C$  is obtained by marginalizing  $X$  from the joint,

$$\begin{aligned} P\{Y | \mathbf{A}, \mathbf{B}, \mathbf{C}\} &= \sum_x P\{X = x, Y | \mathbf{A}, \mathbf{B}, \mathbf{C}\} \\ &= \sum_x P\{X = x | \mathbf{A}, \mathbf{B}\} P\{Y | X = x, \mathbf{B}, \mathbf{C}\}, \end{aligned}$$

and it is used to obtain the new distribution for  $X$  given  $Y, A, B, C$ :

$$P\{X | Y, \mathbf{A}, \mathbf{B}, \mathbf{C}\} = \frac{P\{X, Y | \mathbf{A}, \mathbf{B}, \mathbf{C}\}}{P\{Y | \mathbf{A}, \mathbf{B}, \mathbf{C}\}} = \frac{P\{X | \mathbf{A}, \mathbf{B}\} P\{Y | X, \mathbf{B}, \mathbf{C}\}}{P\{Y | \mathbf{A}, \mathbf{B}, \mathbf{C}\}}.$$

The influence diagram shown in Figure 1 is not a decision tree network, because the arcs  $(C, E)$ ,  $(D, E)$ , and  $(D, F)$  are non-sequential. We can obtain a decision tree network by reversing those arcs. Note that in the process non-sequential arcs would be created from  $A$  to  $E$  and  $F$  that would also need to be reversed. We simplify matters by reversing  $(A, C)$  before reversing the non-sequential arcs. Figure 4 presents the decision tree network for the example, after the four arc reversals. Note that in Figure 1,  $D$  is an ancestor but not a

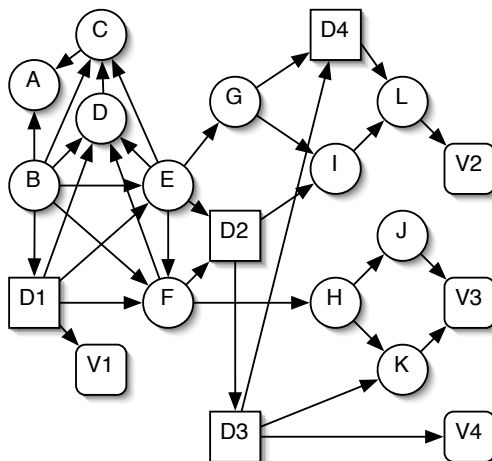


Figure 4: Decision tree network for the influence diagram from Figure 1.

parent of decisions nodes  $D2$ ,  $D3$ , and  $D4$ , while in Figure 4, it is no longer an ancestor of any decision nodes.

Decision trees are easy to understand but they become intractable for decision situations involving a large number of variables since they are exponential in the number of nodes. Decision trees do have some advantage in that they may be able to better represent asymmetric decision situations such as the used car buyer problem [23]. We briefly discuss asymmetry later in the article. Larger decision situations may be more amenable for representation and solution through other means. In the following sections, we review techniques that are able to exploit the conditional independence of the influence diagram, and therefore reduce the complexity of the solution.

### 3 SOLVING INFLUENCE DIAGRAMS: MODIFICATIONS TO THE DIAGRAM

A *variable elimination algorithm* directly evaluates the original influence diagram through a sequence of node elimination steps [2][3][14][24]. The algorithm iteratively removes nodes from the diagram until all that remains is a value node and any evidence nodes, representing the optimal value of the influence diagram and the observations already made. The optimal strategy is determined in terms of an optimal policy for each decision before that decision is eliminated.

The algorithm recognizes a variety of conditions under which nodes can be eliminated, and at least one node can be removed in any proper influence diagram until all that remains are value and evidence nodes. For example, *barren nodes* are decision or unobserved uncertain nodes that have no children.

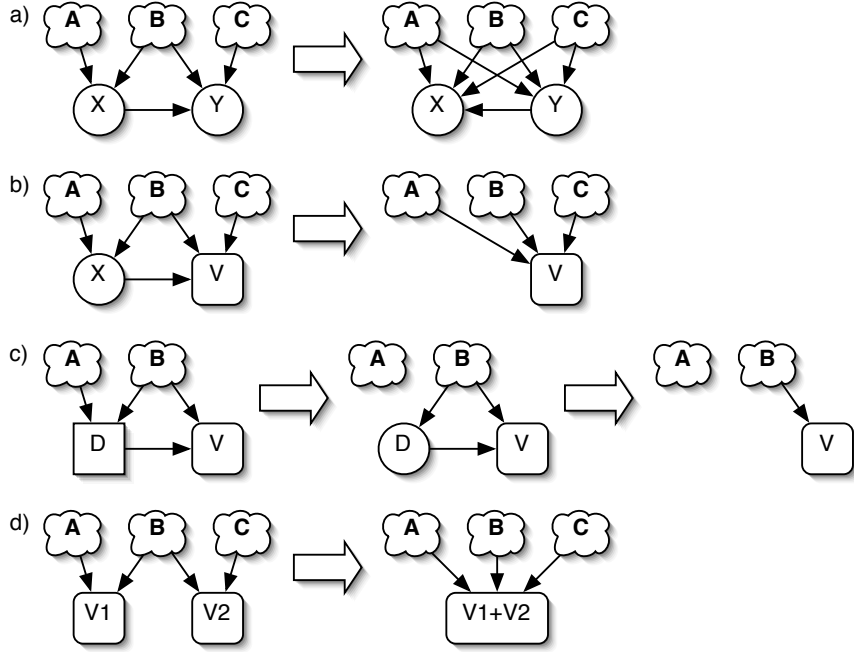


Figure 5: The four graphical transformations to the influence diagram.

These variables have no effect upon the value and can be simply removed from the diagram. These might be present in the original diagram or created in the process of variable elimination.

There are four other graphical transformations needed: arc reversal between uncertain nodes, discussed in the previous section, and a removal operation for each type of node: decision, uncertain, and value.

When an uncertain node has only one child, and that child is a value node, *uncertain node removal* eliminates the node from diagram by taking expectation, and the value node inherits its parents as shown in Figure 5b,

$$E[V|\mathbf{A}, \mathbf{B}, \mathbf{C}] = \sum_x P\{X = x|\mathbf{A}, \mathbf{B}\}E[V|X = x, \mathbf{B}, \mathbf{C}].$$

When a decision node has only one child, that child is a value node, and all other parents of the value node are also parents of the decision node, *optimal policy determination* replaces the decision node with a (deterministic) uncertain one, representing the optimal policy for each possible combination of the other parents of the value node as shown in Figure 5c,

$$E[V|\mathbf{B} = \mathbf{b}] = \max_d E[V|D = d, \mathbf{B} = \mathbf{b}],$$

for all possible  $\mathbf{b}$ . That policy node can then be removed, but the optimal policy should be saved.

When there are multiple value nodes, *value node removal* replaces two value nodes by one value node equal to their sum with the union of their parents, as shown in Figure 5d,

$$E[V_1 + V_2 | \mathbf{A}, \mathbf{B}, \mathbf{C}] = E[V_1 | \mathbf{A}, \mathbf{B}] + E[V_2 | \mathbf{B}, \mathbf{C}].$$

To exploit the decomposition of the value, this operation should be delayed as long as possible, ideally until the parents of one of the value nodes are all parents of the other, or when both value nodes are descendants of the latest decision.

Given these operations, an influence diagram with no directed cycles and ordered decisions with no forgetting can be solved by the following algorithm [3][14]. First, if any of the value nodes has a child, turn it into an uncertain node and give it a value child equal to itself.

Repeat, until all that remains is a single value node and the evidence nodes:

1. If there is a barren node, remove it;
2. Otherwise, if the latest decision node has exactly one child, a value node, and all of the other parents of the value are observed before the decision, then determine its optimal policy and remove it;
3. Otherwise, if there is an uncertain node that is not observed before the latest decision and has [at most] one value child then remove it, after reversing any arcs to its non-value children in graph order;
4. Otherwise, a value node needs to be removed, preferably a descendant of the latest decision or when its parents are all parents of another value node.

Let us demonstrate the algorithm with the help of value reduction steps shown in Figure 6. Figure 3 presents the original influence diagram with the requisite observations for all decisions. Figure 6a shows the diagram after the removal of five unobserved uncertain nodes,  $L$ ,  $I$ ,  $J$ ,  $K$ , and  $H$ . The optimal policy for  $D4$  can now be determined and Figure 6b shows the diagram after the decision node  $D4$  has been removed and the value nodes  $V3$  and  $V4$  have been summed so that the policy for  $D3$  can be determined. Uncertain node  $G$  and decision node  $D3$  have been removed in Figure 6c. Decision node  $D2$  and uncertain node  $F$  have been removed in Figure 6d. Uncertain nodes  $E$ ,  $A$ , and  $C$  have been removed in Figure 6e. Value nodes  $V2$  and  $V3 + V4$  have been summed in Figure 6f so that uncertain node  $D$  can be removed in Figure 6g. The two value nodes are summed in Figure 6h so that decision node  $D1$  can be removed in Figure 6i. Finally, uncertain node  $B$  is removed in Figure 6j, so that only the total value remains.

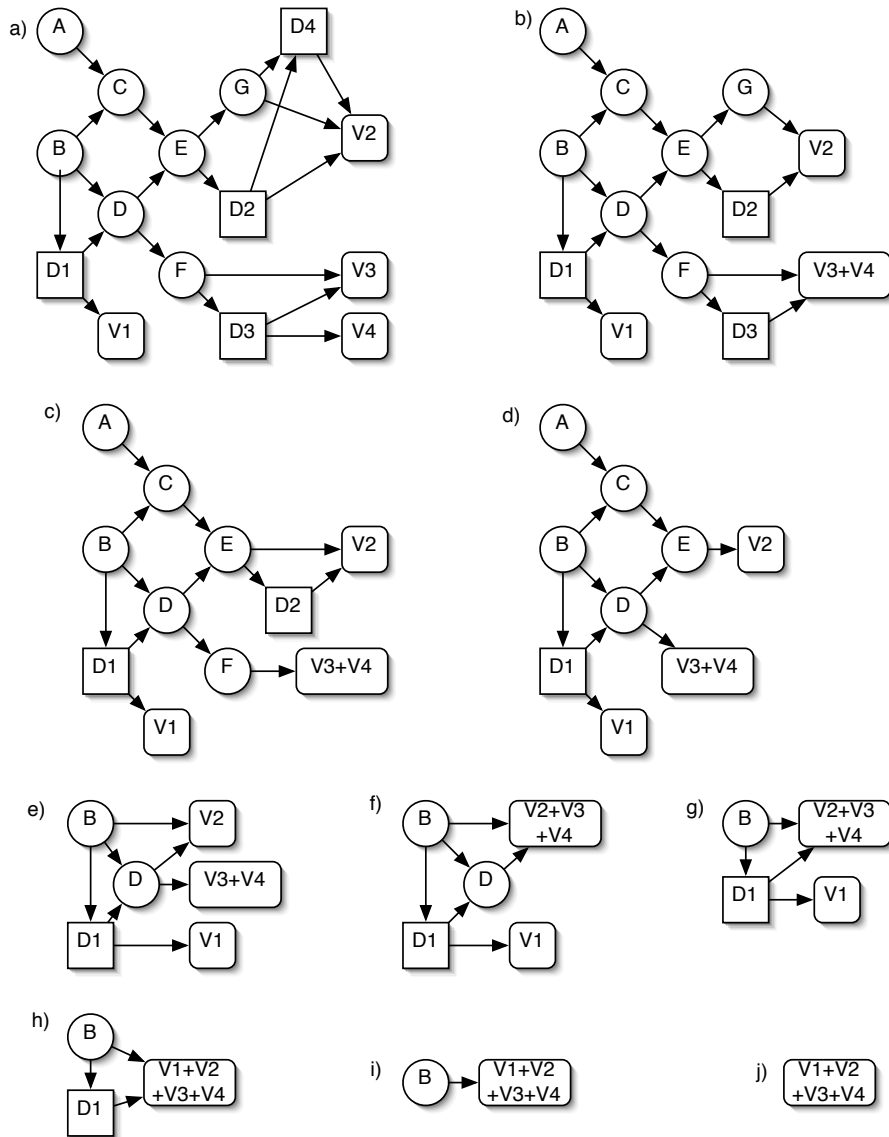


Figure 6: Node reduction steps for the influence diagram from Figure 1.

## 4 SOLVING INFLUENCE DIAGRAMS: CLUSTER TREES

Much of the popularity of belief networks arises from the improvement in computational power through the development of efficient algorithms, many of which



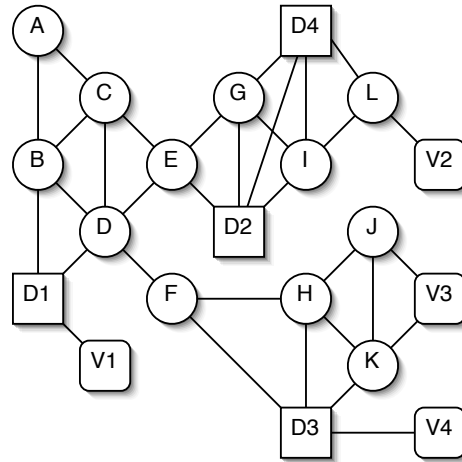


Figure 8: Undirected graph for the moralized diagram from Figure 7.

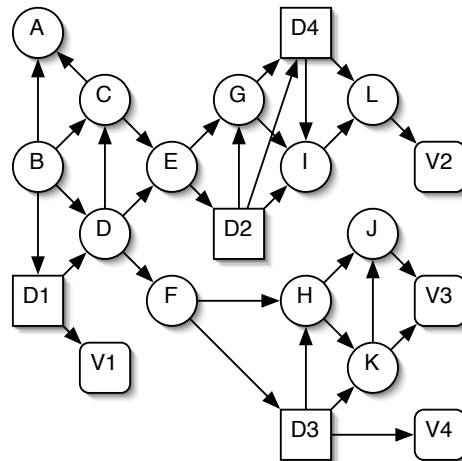


Figure 9: Directed chordal graph for the influence diagram from Figure 1.

any two nodes with a common child [12]. Unlike the moralizing edges, these new arcs can themselves create the need for additional arcs. When we add them, we use the elimination order to determine the direction of the arcs. In this example, because the graph in Figure 8 is chordal and this elimination order is “perfect”, we have no new arcs to add. However, if  $C$  were eliminated before  $A$ , then there would be an arc from  $A$  to  $C$  and we would have needed to add an arc from  $D$  to  $A$  since both of them would be parents of  $C$ . The best elimination order is

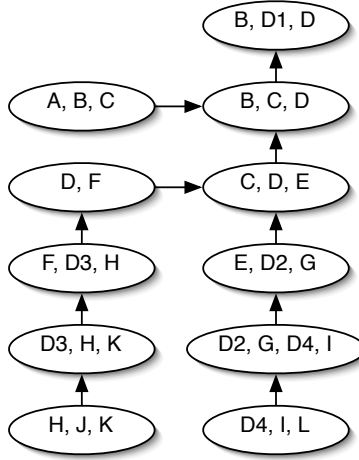


Figure 10: Rooted cluster tree for the influence diagram from Figure 1.

one that requires no extra arcs in this diagram.

We are now prepared to construct a *rooted cluster tree* [6] (also called a *strong junction tree* [7]), a directed tree with each node corresponding to a *cluster* or set of variables, and satisfying four important properties: (1) there is one root cluster with no child clusters, and all other clusters have exactly one child; (2) there are arcs in the directed chordal graph between all of the variables in each cluster; (3) if the same variable appears in two clusters, it appears in every cluster in between them in the tree; and (4) if one cluster follows another in the tree, then the variables that are only in the parent cluster will be eliminated before any of the variables in the child cluster.

There can be many different rooted cluster trees for the influence diagram shown in Figure 1, depending on the elimination order, and even with the same elimination order, but here is an algorithm to construct the rooted cluster tree shown in Figure 10, from a directed chordal graph such as Figure 9.

1. Select the variable having no parents,  $X$ , mark all other non-value variables as *unselected*, set the *current* cluster to be  $\{X\}$ , and make it the root cluster.
2. While there is at least one unselected variable  $X$ , with all of its parents comprising the current cluster, select the latest such  $X$  in the elimination order and add it to the current cluster.
3. If there are no more unselected variables, we are done. Otherwise, select the unselected variable  $X$  latest in the elimination order, creating a new current cluster with its family,  $X\text{Pa}(X)$ . Make this current cluster parent to one of the clusters containing all of  $X$ 's parents; we suggest that the

child cluster should be chosen as close to the root cluster as possible. Now go to step 2.

In our example, variables  $B$ ,  $D1$ , and  $D$ , would be selected to form the root cluster. Variable  $C$  is selected next and the cluster  $\{B, C, D\}$  is created as a parent of cluster  $\{B, D1, D\}$ . Variable  $A$  is selected next and the cluster  $\{A, B, C\}$  is created as a parent of  $\{B, C, D\}$ . This proceeds until all of the variables, except the value variables, have been included in clusters, and those clusters have been added to the tree. Note that for our purposes, we do not need to have the value variables in clusters.

The first step in solving the influence diagram with the cluster tree is to associate each uncertain variable  $X$  with a cluster in the cluster tree that contains the family of  $X$ ,  $X\mathbf{Pa}(X)$ . (For value variables, the family does not need to include the value variable itself, just its parents.) The addition of moralizing edges guarantees that there will be at least one such cluster for each uncertain variable. In the cluster tree shown in Figure 10, the variable  $B$  could be associated with any of the three clusters containing  $B$ , but  $C$  must be associated with  $\{A, B, C\}$  because that is the only cluster containing its family in Figure 1.

For each cluster we compute a *utility potential* as the product of all of the conditional probabilities (and evidence likelihoods) of the uncertain variables associated with that cluster. For example, if  $B$  is associated with cluster  $\{B, C, D\}$  then its utility potential would be  $P\{B\}$ ; otherwise, if no variables were associated with the cluster its utility potential would be 1. Cluster  $\{H, J, K\}$  is associated with variables,  $J$ ,  $K$ , and  $V3$ , so its utility potential would be  $P\{J|H, K\}P\{K|H\}E[V3|J, K]$ .

For each cluster we also compute a *probability potential*, similar to the utility potential except that it does not include expectation factors for value variables. For example, the probability potential for  $\{B, C, D\}$  would be the same as its utility potential, but the probability potential for  $\{H, J, K\}$  would be  $P\{J|H, K\}P\{K|H\}$ .

We then visit each cluster in graph order. We multiply the cluster potentials by any corresponding potentials we received from parent clusters. We then eliminate, in order, those variables not present in its child cluster. For uncertain variables, we sum over all states of the variable in both potentials. For decision variables, we need to maximize over the utility potential for each state of the requisite variables, all of which will be in that cluster, and select the corresponding element from the probability potential. (For value variables, there is no need to do anything.) The resulting potentials should be a function of variables in the child cluster and they should be passed along to it.

At the end, in the root cluster, we have a single number left in each potential. The probability potential equals the probability of the evidence and the utility potential divided by the probability potential equals the maximal expected value. The optimal strategy is represented by the maximization choices we made for the decision variables.

For example, in cluster  $\{H, J, K\}$  we sum the potentials over the states of  $J$  and pass along the new potentials, as a function of  $H, K$  to cluster  $\{D3, H, K\}$ .

We multiply that cluster’s own potentials,  $P\{K|D3, H\}E[V4|D3]$ , by the passed potential, sum over the states of  $K$ , and pass the resulting potential as a function of  $D3, H$  to cluster  $\{F, D3, H\}$ . We multiply that cluster’s own potential,  $P\{H|F\}$ , by the passed potential, sum over the states of  $H$ , and maximize over  $D3$  given  $F$ , saving the optimal choices. We pass the resulting potential as a function of  $F$  to cluster  $\{D, F\}$ . We continue processing each cluster this way until the root cluster’s potentials are reduced to a single number each.

The cluster tree method uses the conditional independence in the diagram and organizes the computations so that summation and maximization operations occur in the appropriate order. We provided a brief overview of solving influence diagrams with cluster trees, leaving out many of the details. Shachter and Peot [6], Jensen et al [7], Shachter [21], and Jensen and Nielsen [28] are some references for further reading on cluster tree construction and evaluation with message passing.

## 5 OTHER RELATED TOPICS

The literature on influence diagrams is vast and varied; here we provide a quick overview of other related topics.

### 5.1 Asymmetry

The methods we have described for solving influence diagrams exploit the *global structure* of the influence diagram, i.e. the topology of the graph, and in particular through conditional independence and separable values. The complexity of the influence diagram at the level of global structure is studied in terms of the number of nodes and the *treewidth*, which is a measure of the connectivity of the network. On the other hand, *local structure* refers to the specific numbers used for building the graphical model. Here we discuss the issue of local structure in influence diagrams.

Real world decision situations often exhibit a significant amount of *asymmetry*, i.e decision situations where some possible combinations of variables have no meaning. In an influence diagram representation, this typically means that the full rectangular table does not need to be evaluated. Asymmetry in the decision situation is represented through the numbers in the model, and mainly in the form of: *determinism*, or the presence of zeros in conditional probability distributions, and *context specific independence* [29], which refers to independence between variables given a particular instantiation of some other variables. In most real-world decision situations, there is a tremendous opportunity to also exploit the specific numbers in the model so that evaluation and analysis can be made even more efficient. Previous research has shown the potential benefits from tapping local structure for improving evaluation efficiency [30][31].

Influence diagrams face some difficulty in representing asymmetric decision situations. Some of the shortcomings of influence diagrams for representing asymmetric situations are documented in Smith et al [31], Bielza and Shenoy

[32], and Shenoy [33]. Several researchers have worked on trying to extend and augment the influence diagram representation to capture asymmetry directly, see for instance Covaliu and Oliver [34], Shenoy [35], and Jensen et al [36]. Another direction of research has been to focus on identifying ways to exploit the asymmetry while solving the influence diagram. This has been actively pursued in the belief network literature, with the help of graphical representations such as arithmetic circuits [37][38]. Decision circuits have been developed to perform efficient evaluation of influence diagrams [9][12], building on the advances in arithmetic circuits for belief network inference. The benefit of the decision circuit framework is that compact circuits are compiled in an offline phase so that subsequent online operations for evaluation and analysis can be more efficient. This can be particularly useful in situations for real-time decision making and in decision systems [39], i.e. for decision situations under uncertainty that recur frequently and involve considerable automation.

## 5.2 Other related graphical models

In the previous section we mentioned a few graphical models for specifically dealing with the issue of asymmetry. There are various other graphical models closely related to the influence diagram literature.

There are other graphical models for representing and solving decision problems, such as game trees [40] and valuation networks [41]. The fusion algorithm for solving valuation networks is closely related to the cluster tree methods described here. There have also been several extensions to influence diagrams that relax some of the assumptions, such as limited memory influence diagrams (LIMIDs) [42], unconstrained influence diagrams [43], etc. Many of these graphical models are harder to solve than standard influence diagrams and like the graphical models for representing asymmetry, are rather complex. There is also a considerable amount of literature on influence diagrams with continuous uncertain variables, starting with Shachter and Kenley [44]. Another direction of research has been on extending influence diagrams to problems with multiple decision makers [45][42][46].

## 5.3 Sensitivity analysis and value of information

The phrase *sensitivity analysis* refers, in general, to understanding how the output for a system varies as a result of changes in the system's inputs. For influence diagrams, one may be concerned about how the optimal solution and the value change with respect to a change in the parameters, i.e. the probabilities and the utilities, or a change in the informational assumptions of the problem. Sensitivity analysis has been an essential aspect of decision analysis throughout the field's development. Sensitivity analysis aids in identifying the model's critical elements, forming the basis for iterative refinement of the model, and can also be used after the analysis for defending a particular strategy to the decision maker [47].

Sensitivity analysis is an essential technique to support influence diagram modeling and it provides valuable insight about the critical assessments. Sensitivity analysis for influence diagrams has been studied in Nielsen and Jensen [10] and Bhattacharjya and Shachter [11][13]. Decision circuits are particularly suitable for performing sensitivity analysis on influence diagrams since partial derivatives are easy to compute with compiled circuits, and therefore analysis on the model can be performed conveniently within this framework.

One of the most powerful sensitivity analysis techniques of decision analysis is the computation of *value of information* (or clairvoyance), the difference in value obtained by changing the decisions by which some of the uncertainties are observed [16][17]. The value of information for a particular uncertainty specifies the maximum that the decision maker should be willing to pay to observe the uncertainty before making a particular decision. Value of information can be computed on the cluster tree used to solve the original problem [21] or in other ways [48] [11].

## References

- [1] Howard, R, Matheson, J. 1984. Influence diagrams. Pp 3–23 in Howard, R. and J. Matheson, eds. *The Principles and Applications of Decision Analysis*, Vol II. Strategic Decisions Group; pp. 721-762.
- [2] Olmsted, S. On solving influence diagrams. Ph.D dissertation. 1983. Stanford University, CA, USA.
- [3] Shachter, R. Evaluating influence diagrams. *Operations Research*. 1986 Nov-Dec; 34: 871–882.
- [4] Cooper, G. 1988. A method for using belief networks as influence diagrams. *Proceedings of the Fourth Conference on Uncertainty in Artificial Intelligence*, Minneapolis, MN. Pp 55–63.
- [5] Shachter, R. Probabilistic inference and influence diagrams. *Operations Research*. 1988 July-Aug; 36: 589–605.
- [6] Shachter, R, Peot, M. 1992. Decision making using probabilistic inference methods. *Proceedings of the Eighth Conference on Uncertainty in Artificial Intelligence*, Stanford, CA. Pp 276–283.
- [7] Jensen, F, Jensen, FV, Dittmer, S. 1994. From influence diagrams to junction trees. *Proceedings of the Tenth Conference on Uncertainty in Artificial Intelligence*, Seattle, WA. Pp 367–373.
- [8] Madsen, A, Jensen, FV. 1998. Lazy propagation in junction trees. *Proceedings of the Fourteenth Conference on Uncertainty in Artificial Intelligence*, Madison, WI. Pp 362–369.

- [9] Bhattacharjya, D, Shachter, R. 2007. Evaluating influence diagrams with decision circuits. Proceedings of the Twenty Third Conference on Uncertainty in Artificial Intelligence, Vancouver, Canada. Pp 9–16.
- [10] Nielsen, T, Jensen, FV. Sensitivity analysis in influence diagrams. IEEE Transactions on Systems, Man, and Cybernetics - Part A: Systems and Humans. 2003 March; 33(1): 223–234.
- [11] Bhattacharjya, D, Shachter, R. 2008. Sensitivity analysis in decision circuits. Proceedings of the Twenty Fourth Conference on Uncertainty in Artificial Intelligence, Helsinki, Finland. Pp 34–42.
- [12] Bhattacharjya, D, Shachter, R. 2010. Dynamic programming in influence diagrams with decision circuits. Proceedings of the Twenty Sixth Conference on Uncertainty in Artificial Intelligence. Catalina Island, CA.
- [13] Bhattacharjya, D, Shachter, R. 2010. Three new sensitivity analysis methods for influence diagrams. Proceedings of the Twenty Sixth Conference on Uncertainty in Artificial Intelligence. Catalina Island, CA.
- [14] Tatman, J, Shachter, R. Dynamic programming and influence diagrams. IEEE Transactions on Systems, Man and Cybernetics. 1990 20(2): 365–379.
- [15] von Neumann, J, Morgenstern, O. Theory of Games and Economic Behavior, 2nd edition. 1947. Princeton University Press, Princeton, NJ.
- [16] Howard, R. Information value theory. IEEE Transactions on Systems Science and Cybernetics. 1966 2:22–26.
- [17] Raiffa, H. Decision Analysis: Introductory Lectures on Choices under Uncertainty. 1968. Addison-Wesley.
- [18] Heckerman, D, Shachter, R. 1995. A definition and graphical representation for causality. Proceedings of the Eleventh Conference on Uncertainty in Artificial Intelligence, Montreal, Canada. Pp 262–273.
- [19] Shachter, R. An ordered examination of influence diagrams. Networks. 1990 20: 535–563.
- [20] Nielsen, T, Jensen, FV. 1999. Well-defined decision scenarios. Proceedings of the Fifteenth Conference on Uncertainty in Artificial Intelligence, Stockholm, Sweden. Pp 502–511.
- [21] Shachter, R. 1999. Efficient value of information computation. Proceedings of the Fifteenth Conference on Uncertainty in Artificial Intelligence, Stockholm, Sweden. Pp 594–601.

- [22] Shachter, R. 1998. Bayes-ball: The rational pastime (for determining irrelevance and requisite information in belief networks and influence diagrams). Proceedings of the Fourteenth Conference on Uncertainty in Artificial Intelligence, Madison, WI. Pp 480–487.
- [23] Howard, R. The used car buyer. Pp 689–718 in Howard, R. and J. Matheson, eds. Readings on the Principles and Applications of Decision Analysis, Vol. II. 1962. Strategic Decisions Group, Menlo Park, CA.
- [24] Shachter, R. Model building with belief networks and influence diagrams. Pp 177–201 in Edwards, W, Miles, R. and D. von Winterfeldt, eds. Advances in Decision Analysis: From Foundations to Applications. 2007. Cambridge University Press.
- [25] Lauritzen, S, Spiegelhalter, D. Local computations with probabilities on graphical structures and their application to expert systems. Journal of the Royal Statistical Society, Series B. 1988 50 (2): 157–224.
- [26] Jensen, FV, Lauritzen, S, Olesen, KG. Bayesian updating in causal probabilistic networks by local computation. Computational Statistics Quarterly. 1990 4: 269–282.
- [27] Shenoy, P, Shafer, G. 1990. Axioms for probability and belief-function propagation. Shachter, R, Levitt T, Lemmer J, et al., editors. Uncertainty in Artificial Intelligence 4. Amsterdam: North-Holland. Pp 169–198.
- [28] Jensen, FV, Nielsen, TD. Bayesian Networks and Decision Graphs, 2nd edition. 2007. Springer Verlag, New York.
- [29] Boutilier, C, Friedman, F, Goldszmidt, M, Koller, D. 1996. Context specific independence in Bayesian networks. Proceedings of the Twelfth Conference on Uncertainty in Artificial Intelligence, Portland, OR. Pp 115-123.
- [30] Gomez, M, Cano, A. Applying numerical trees to evaluate asymmetric decision problems. Pp 196–207 in Nielsen, T. and N. Zhang, eds. Symbolic and Quantitative Approaches to Reasoning with Uncertainty, Lecture Notes in Artificial Intelligence 2711. 2003. Springer-Verlag.
- [31] Smith, J, Holtzman, S, Matheson, J. Structuring conditional relationships in influence diagrams. Operations Research. 1993 41(2): 280-297.
- [32] Bielza, C, Shenoy, P. A comparison of graphical techniques for asymmetric decision problems. Management Science. 1999 45(11): 1552–1569.

- [33] Shenoy, P. A comparison of graphical techniques for decision analysis. *European Journal of Operational Research*. 1994 78(1): 1–21.
- [34] Covaliu, Z, Oliver, R. Representation and solution of decision problems using sequential decision diagrams. *Management Science*. 1995 41(12): 1860–1881.
- [35] Shenoy, P. Valuation network representation and solution of asymmetric decision problems. *European Journal of Operational Research*. 2000 121(3): 579–608.
- [36] Jensen, FV, Nielsen, TD, Shenoy, P. Sequential influence diagrams: A unified asymmetry framework. *International Journal of Approximate Reasoning*. 2006 42(1): 101–118.
- [37] Darwiche, A. 2000. A differential approach to inference in Bayesian networks. *Proceedings of the Sixteenth Conference on Uncertainty in Artificial Intelligence*. Pp 123–132.
- [38] Darwiche, A. A differential approach to inference in Bayesian networks, *Journal of the ACM*. 2003 50(3): 280–305.
- [39] Holtzman, S. *Intelligent Decision Systems*. 1989. Addison-Wesley, Reading, MA.
- [40] Shenoy, P. Game trees for decision analysis, *Theory and Decision*. 1998 Apr; 44(2): 149–171.
- [41] Shenoy, P. Valuation based systems for Bayesian decision analysis. *Operations Research*. 1992 May-June; 40: 463–484.
- [42] Lauritzen, S, Nilsson, D. Representing and solving decision problems with limited information. *Management Science*. 2004 47: 1235–1251.  
Minneapolis, MN 6. Stanford, CA 7. Seattle, WA 8, 20. Madison, WI 9. Vancouver, Canada 11. Helsinki, Finland 16. Montreal, Canada 18, 19. Stockholm, Sweden 27. Portland, OR 41.
- [43] Jensen, FV, Vomlelova, M. 2002. Unconstrained influence diagrams. *Proceedings of the Eighteenth Conference on Uncertainty in Artificial Intelligence*, Edmonton, Canada. Pp 234–241.
- [44] Shachter, R, Kenley, C. Gaussian influence diagrams, *Management Science*. 1989 May; 35: 589–605.
- [45] Koller, D, Milch, B. Multi-agent influence diagrams for representing and solving games. *Games and Economic Behavior*. 2003 45: 181–221.
- [46] Detwarasiti, A, Shachter, R. Influence diagrams for team decision analysis. *Decision Analysis*. 2005 2(4): 207–228.

- [47] Howard, R. The evolution of decision analysis. In Howard, R. and J. Matheson, eds. *The Principles and Applications of Decision Analysis*. 1983. Strategic Decisions Group, Menlo Park, CA.
- [48] Matheson, J. Using influence diagrams to value information and control. Pp 25–63 in Oliver, R.M. and J.Q. Smith, eds. *Influence Diagrams, Belief Networks and Decision Analysis*. 1990. John Wiley and Sons Ltd., New York.