ARTICLES

STRUCTURING CONDITIONAL RELATIONSHIPS IN INFLUENCE DIAGRAMS

JAMES E. SMITH

Duke University, Durham, North Carolina

SAMUEL HOLTZMAN and JAMES E. MATHESON

Strategic Decisions Group, Menlo Park, California

(Received December 1989; revisions received January, April 1992; accepted July 1992)

An influence diagram is a graphical representation of a decision problem that is at once a formal description of a decision problem that can be treated by computers and a representation that is easily understood by decision makers who may be unskilled in the art of complex probabilistic modeling. The power of an influence diagram, both as an analysis tool and a communication tool, lies in its ability to concisely summarize the structure of a decision problem. However, when confronted with highly asymmetric problems in which particular acts or events lead to very different possibilities, many analysts prefer decision trees to influence diagrams. In this paper, we extend the definition of an influence diagram by introducing a new representation for its conditional probability distributions. This extended influence diagram representation, combining elements of the decision tree and influence diagram representations, allows one to clearly and efficiently represent asymmetric decision problems and provides an attractive alternative to both the decision tree and conventional influence diagram representations.

Influence diagrams were originally developed in the mid-1970s as a description of a decision problem that is "at once both a formal description of the problem that can be treated by computers and a representation easily understood by people in all walks of life and degrees of technical proficiency" (Howard and Matheson 1981; see also Miller et al. 1976). Since their introduction, influence diagrams have become an important tool for professional decision analysts. The power of an influence diagram, both as an analysis tool and a communication tool, lies in its ability to concisely and precisely describe the structure of a decision problem. Decision makers who may be unskilled in the art of complex probabilistic modeling find that influence diagrams provide them with a language to clearly describe their conception of a decision problem (see Owen 1978 and Howard 1988).

The original definition of influence diagrams distinguished three levels of specification for a decision problem: relation, function, and number (Howard and Matheson). In the deterministic case, the *level of relation* indicates that one variable depends in a general way on the others; for example, profit is a function of revenue and cost. The *level of function* specifies the precise function describing this dependence; namely, that profit equals revenue minus cost. Finally, at the *level of number*, we specify numerical values of revenue and cost and hence determine the numerical value of profit.

In the probabilistic case, the level of relation indicates that, given the information available, one variable is probabilistically dependent on certain variables and probabilistically independent of others. For example, we might assert that for a given person, income depends on age and education, and that education depends on age. At the level of function, we describe the form of these dependencies. For instance, if we divide age into 10-year increments, we might assign different distributions on education for each age group under 40 and the same distribution for all age groups over 40. When assessing income given age and a particular educational level, we may wish to assign different distributions for each age group. At the level of number, we specify numerical probabilities for each

Subject classification: Decision analysis, influence diagrams: representing and solving decision problems.
Area of review: Decision Analysis, Bargaining and Negotiation

0030-364X/93/4102-0280$01.25

conditional and unconditional event. Taken together, the three levels implicitly determine a joint probability distribution over all variables.

The success of the influence diagram representation is primarily a result of its ability to describe graphically the structure of a decision problem at the level of relation. The influence diagram representation is compact and, unlike the decision tree representation, clearly indicates the dependence and independence assumptions in the model. The influence diagram representation has also proven valuable for computation. While early efforts showed how an influence diagram could be turned into a decision tree and then evaluated (Howard and Matheson), more recent efforts have focused on evaluating influence diagrams directly (Olmsted 1983, Shachter 1986, 1988). These influence diagram computational methods have achieved substantial efficiencies over "brute force" decision tree methods.

The influence diagram representation has weaknesses as well. In particular, it is commonly believed that influence diagrams are not useful for representing highly asymmetric problems in which particular acts or events lead to very different possibilities (see, for example, Watson and Buede 1987, Phillips 1990, Call and Miller 1990). Phillips (p. 22) writes:

> Certainly there are disadvantages [to using influence diagrams], the most serious being the difficulty influence diagrams have with asymmetrical decision trees. When subsequent events and acts depend on the initial decision, influence diagrams run into difficulties. Often these situations occur when the decision maker wishes to portray cause-and-effect sequences as possible scenarios. These are not times when I would choose to use an influence diagram, and they occur frequently. Look at any textbook on decision analysis and try to find a single symmetrical decision tree.

To represent an asymmetric decision problem as an influence diagram, the problem must be "symmetrized" by adding artificial states and assuming degenerate probability distributions and/or value functions. Unfortunately, these adaptations obscure the structure of the problem and increase the time and space required for solution. In problems like these, many analysts prefer decision trees to influence diagrams.

In this paper, we extend the definition of an influence diagram to describe the structure of a decision problem at the level of function. This extension allows one to represent asymmetric decision problems in influence diagrams without obscuring the structure of the problem or increasing the time and space required for solution. Our primary contribution is a new representation for describing the conditional probability distributions associated with an influence diagram. This representation is easy to express graphically in ways that should be comfortable and familiar to those who have used decision trees.

Our work adds to the existing work in influence diagrams in several ways. The most closely related work is Olmsted's unpublished Ph.D. dissertation (Olmsted 1983) and there is some overlap between our work and his. Although it has not become part of the conventional influence diagram representation (as defined by Shachter 1986, 1988, or Rege and Agogino 1988 and implemented in several software packages), Olmsted (pp. 85–96) discusses the representation of "coalescence within influence diagrams" where "some of the conditional probabilities (in the conditional probability distribution) are known to be identical" and demonstrates how efficiencies in storage and computation can be obtained by recognizing this coalescence. This is one of several types of asymmetries that can be captured in our representation. In addition, unlike the conventional influence diagram representation or Olmsted's representation, we allow the sets of possible outcomes or alternatives to vary in different scenarios; we can represent variables of mixed types (for example, variables whose values are uncertain in some scenarios and deterministic in others); and we can capture and exploit asymmetries due to irrelevant distributions or impossible scenarios. Following Olmsted (and Shachter 1986), we describe influence diagram solution procedures that recognize these asymmetries and provide substantial efficiencies in storage and computation.

Recognizing the problems influence diagrams have with asymmetric decision problems, several researchers have recently (and independently) proposed alternative representations that attempt to combine the strengths of the influence diagram and decision tree representations. We would like to briefly describe two of these representations: Call and Miller (1990) and Fung and Shachter (1990). Call and Miller describe a representation, which they call DPL (for decision programming language), where a decision problem is described using separate decision tree-like and influence diagram-like representations. The influence diagram-like representation is used to describe the probabilistic relationships among variables in the problem and is essentially a conventional influence diagram, except that it does not describe the sequence in which decisions are made and uncertainties resolved. Instead, this decision sequence is captured in the decision tree-like representation. The advantage of this approach, they argue, is that it "allows DPL to take advantage of some of the virtues of each approach, e.g., influence diagrams are best at precisely describing probabilistic relationships, and trees are best at clearly describing the decision sequence" (p. 139). While this approach provides some impressive computational results, we believe that its major weakness is that one must examine both the decision tree-like and influence diagram-like parts to obtain a complete description of the problem. In contrast, both the conventional influence diagram representation (and our extension to it) and the decision tree representation provide a complete description of the decision problem in a single, consistent framework.

Fung and Shachter take a different approach, closer to ours, in defining what they call contingent influence diagrams. Their key innovation is associating each variable with a set of contingencies that describe the scenarios in which the variable is defined and list the variables conditioning the distribution (conditioning parents) in these scenarios. Our representation and the contingent influence diagram representation share some common features, notably the ability to exploit certain forms of coalescence (what we call *collapsed distributions*) and distributions that are irrelevant because of this coalescence (what we call *unspecified distributions*). Beyond these common features, we believe that our representation is more natural and general. For example, we allow the sets of possible outcomes or alternatives for a variable to vary in different scenarios. To do this in a contingent influence diagram, one would have to define many different variables (perhaps with the same name) with different sets of outcomes and mutually exclusive contingencies. It is similarly awkward to represent variables of mixed types in a contingent influence diagram.

The rest of this paper is organized as follows. In the first section, we illustrate some of the difficulties in representing asymmetric decision problems using the conventional influence diagram and decision tree representations by considering the well-known used car buyer problem as an example (Howard 1962). In the second section, we introduce our representation for the conditional probability distributions associated with an influence diagram and illustrate it with distributions from the used car buyer problem. In the third section, we consider the computational implications of our representation. Here, we see the benefits of explicitly describing the structure of a decision problem at the level of function: Structural properties of the conditional distributions are propagated through the solution process and computational efficiencies are obtained by noting the occurrence of special structures in the distributions. In the fourth section, we demonstrate these efficiencies by comparing our representation of the used car buyer problem with the conventional influence diagram and decision tree representations. We conclude by summarizing the strengths and weaknesses of our representation as compared to the conventional influence diagram and decision tree representations as well as the representations proposed by Call and Miller (1990) and Fung and Shachter (1990).

1. AN ILLUSTRATIVE EXAMPLE: THE USED CAR BUYER

To illustrate some of the problems encountered with asymmetric decision problems in influence diagrams and decision trees, we consider the well-known and highly asymmetric used car buyer problem as an example (Howard 1962). We chose this example (out of many possible examples) because it illustrates many of the difficulties that the conventional influence diagram representation has with asymmetric problems and it is large enough to demonstrate the computational advantages of our representation. In describing the example, we adopt the terminology of Howard and Matheson and refer the reader seeking an introduction to influence diagrams to that paper. Shachter (1986) gives a more formal introduction to influence diagrams, but uses slightly different terminology.

1.1. The Level of Relation

The influence diagram in Figure 1 describes the used car buyer problem at the level of relation. A fellow named Joe is considering buying a used car. Like most used car buyers, he is unsure of the car's condition. For the particular model he is considering, Joe is able to narrow the possible conditions of the car down to either a "peach" or a "lemon," with the uncertainty depending on where the car was manufactured. This uncertainty about the car's condition is represented in

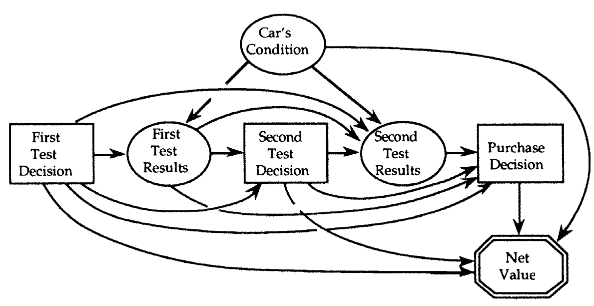

Figure 1. Influence diagram for the used car buyer.

the influence diagram by the chance node labeled Car's Condition.

The used car salesman gives Joe an hour to have the car examined by a mechanic. The mechanic offers to perform three different diagnostic tests on the car that will help Joe determine its condition. Joe's choice of which test to perform, if any, is represented by the decision node labeled First Test Decision. The results of the test are represented by a chance node labeled First Test Results. The arrows into this First Test Results node indicate that the distribution for First Test Results is conditioned on the outcome of First Test Decision and Car's Condition. One of these tests is short and allows time to perform a second related test after reviewing the results of the first. This second testing decision is represented by the Second Test Decision node and its results are represented by the Second Test Results node. The arrows into the Second Test Decision node indicate that, at the time the second test decision is made, Joe knows the results of the first test and remembers which test he chose for First Test Decision. After reviewing all the test results, Joe decides whether to purchase the car; this decision is represented by the Purchase Decision node. The arrows into the Purchase Decision node indicate that at the time of that decision, Joe knows the results of both tests and does not forget which tests he chose to perform. Note that Car's Condition is not known at the time any decision is made.

The value node is a chance node that represents the decision maker's objective function. In this case, the value node, labeled Net Value, represents the value of the car to Joe less what he spends on tests. The dependence of Net Value on the tests performed is indicated by the arrows from the testing decisions to the value node. The arrow from Car's Condition to Net Value indicates that the value of the car depends on the car's condition reflecting the greater repair costs associated with a lemon. The omission of arrows from the test result nodes to the Net Value node indicates that, given the outcomes of its direct predecessors, Net Value does not depend on the test results. The value node is drawn with a double border to indicate that, given the outcomes of its direct predecessors, Net Value is known with certainty (i.e., Net Value is a deterministic function of its direct predecessors).

1.2. The Level of Function

To describe the used car buyer problem in more detail, we need to examine the probability distributions associated with the chance nodes in the diagram and the alternatives associated with decision nodes. These are typically displayed using tables, as shown in Table I. (We propose an alternative representation in this paper.) For example, the principal uncertainty in the problem, Car's Condition, has no conditioning predecessors and the table, shown in part a, specifies the probability assignments for the two possible outcomes, peach and lemon. The alternatives for First Test Decision and Second Test Decision are shown in parts b and c.

The distribution for First Test Results, shown in part d, is conditioned on Car's Condition and First Test Decision. If Joe decides not to perform any tests, he will observe No Result regardless of the car's condition; this is indicated by the degenerate probability distributions given in the first two columns of Table

Table I

Some of the Distributions for the Used Car Buyer Influence Diagram

a. Car's Condition		b. First Test Decision	c. Second Test Decision
Outcome	Probability	Alternatives	Alternatives
Peach	0.8	No Test	No Test
Lemon	0.2	Steering	Differential
		Transmission	
		Fuel & Electrical	

d. First Test Results

First Test Decision	No Test	No Test	Steering	Steering	Transmission	Transmission	Fuel & Electrical	Fuel & Electrical
Car's Condition Outcome	Peach Probability	Lemon Probability	Peach Probability	Lemon Probability	Peach Probability	Lemon Probability	Peach Probability	Lemon Probability
No Result	1.00	1.00	0.00	0.00	0.00	0.00	0.00	0.00
0 Defects	0.00	0.00	0.90	0.40	0.90	0.40	0.80	0.13
1 Defect	0.00	0.00	0.10	0.60	0.10	0.60	0.20	0.53
2 Defects	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.33

I, part d. To consider the distribution for the test results given that a test is performed, we need to explore Joe's understanding of the car's condition in more detail. The lemons were built at a new plant before certain production problems were fixed and are known to have defects in 6 of the car's 10 systems. The peaches were manufactured at an established plant and are known to have defects in 1 of the 10 systems. In both cases, each system is equally likely to be defective. Thus, if Joe tests the steering system and the car is a lemon, the probability of 1 Defect is 0.6. If the car is a peach, the probability of 1 Defect is 0.1. In both cases there is no chance of finding 2 Defects or No Result. These probability assignments are given in the third and fourth columns of part d of Table I. Given Joe's belief that the defects are equally likely to be in any of the 10 systems, the distribution Joe assigns for the test of the transmission is the same as the distribution assigned for the test of the steering system. Thus, the fifth and sixth columns of part d are the same as the third and fourth. Similar reasoning for tests of both the fuel and electrical systems leads to the probability assignments given in the last two columns of the table.

The distribution for First Test Results illustrates some of the problems the conventional influence diagram has with asymmetric probability distributions. The tabular representation requires that asymmetric distributions be "symmetrized" so that the user must define a common set of outcomes for all conditioning scenarios and assign zero probabilities to the many impossible outcomes. While this tabular representation is a complete description of the underlying probability distributions (it contains all of the necessary information), it is inefficient and obscures much of the structure of the distribution. To discover what outcomes are possible in which scenarios, we need to look for the nonzero probabilities in the table. To recognize that the steering and transmission tests lead to identical distributions for First Test Results, we must notice that the fifth and sixth columns of the table are the same as the third and fourth.

We can illustrate another problem with the conventional influence diagram representation by considering the second testing decision. Because of time constraints, Joe has the option to test the differential system only if he chooses to perform the transmission test. This contingent structure poses serious difficulties for the conventional influence diagram representation. In this representation, a single set of alternatives is associated with a decision node, and it is assumed that these alternatives are available in each possible conditioning scenario (see, for example, Shachter 1986). To capture the contingent structure of the second testing decision, we have to design the value function to penalize the unavailable alternatives so that they will not be found to be optimal by the solution algorithm. For example, Joe will not have time to test the differential if he first tests the steering system. To rule out this possible choice, the value function must assign large negative values to all scenarios in which a test of the steering system is followed by a test of the differential system. This limitation of the influence diagram representation has two effects. First, the representation obscures the contingent nature of the decision: To discover that an alternative is impossible, we have to examine the value function and recognize the penalty values. Second, the representation needlessly increases the time and space required to solve the decision problem: The solution algorithm must carry around the penalty values corresponding to the impossible alternatives and then conclude that it is optimal not to choose them.

The second testing decision points to another limitation of the conventional symmetric influence diagram representation. If we examine the distribution for First Test Results, we see that many combinations of test decisions and test results are impossible. For example, if Joe first tests the steering system, it is impossible to observe No Result or 2 Defects. Similarly, if Joe performs no tests, it is impossible to observe 0, 1 or 2 Defects. In the conventional influence diagram representation, the solution algorithm would be asked to compute an optimal policy for Second Test Decision for every combination of scenarios, including the many that are impossible. This is very expensive computationally and, if these impossible scenarios are shown when reporting the optimal strategy for Second Test Decision, the structure of the problem is again obscured.

1.3. A Decision Tree for the Used Car Buyer

Unlike influence diagrams, decision trees have no trouble representing asymmetric decision problems. Figure 2 shows a partial decision tree for this problem; a complete decision tree is given in Howard (1962). Unlike the influence diagram, the decision tree displays only the possible test results, clearly indicates the contingent nature of the second test decision, and does not require computations for impossible scenarios. However, the decision tree representation has its own set of limitations—most of which apply in both symmetric and asymmetric problems.

A fundamental limitation of the tree representation is that it obscures the dependence and independence relationships among variables in the decision problem.

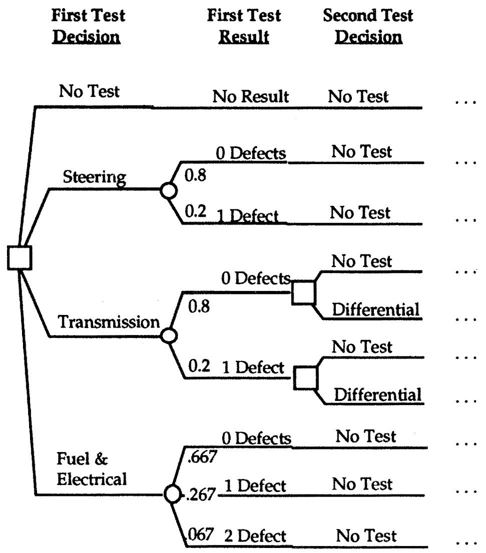

Figure 2. Partial decision tree for the used car buyer.

For example, the influence diagram for the used car buyer problem (Figure 1) clearly indicates that Net Value depends on First Test Decision, Second Test Decision, Purchase Decision and Car's Condition and that, given these, Net Value is independent of First Test Results and Second Test Results. To discover this in the decision tree we would have to notice that all paths through the tree that correspond to a particular combination of values for First Test Decision, Second Test Decision, Purchase Decision, and Car's Condition have the same Net Value regardless of the test results. While this "coalescence" is obvious in the influence diagram, it is very difficult to notice in a decision tree and, consequently, most tree-based solution algorithms have not been able to recognize and exploit it.

Another limitation of the decision tree representation is that the probabilities required in the decision tree may not be the same as those assessed from the decision maker. For example, in the used car buyer problem, it is convenient to assess the probabilities for First Test Results conditioned on Car's Condition (as indicated in the influence diagram of Figure 1). In the decision tree, the order of variables in the tree corresponds to the order in which variables become known to the decision maker. In the used car buyer problem, the outcome of First Test Results is known at the time the Second Test Decision is made, and, similarly, the outcome of Second Test Results is known at the time the Purchase Decision is made. However, the outcome of Car's Condition is not revealed until after the Purchase Decision. To capture this sequence in the decision tree, we must first give marginal probabilities for First Test Results (shown in Figure 2), followed by conditional distributions for Second Test Results given First Test Results, followed by conditional distributions for Car's Condition given both test results. The usual approach for handling these kinds of problems with decision trees is to maintain a separate tree—sometimes called nature's tree—which gives the probabilities in an order convenient for assessment. The probabilities required for the decision tree are then calculated by "flipping" nature's tree using Bayes' rule. Besides the inconvenience of the calculations required to create a decision tree from the assessed distributions, the presence of these two trees sometimes causes confusion between the analyst and decision maker.

2. STRUCTURING CONDITIONAL DISTRIBUTIONS IN INFLUENCE DIAGRAMS

Our goal in this paper is to propose a representation that combines the strengths of the decision tree and influence diagram representations. In particular, we seek a representation that, like an influence diagram, explicitly describes the independence assumptions in the model, and, like a decision tree, can capture and exploit asymmetries in the problem. Our strategy for achieving this goal is to extend the influence diagram representation to include a decision tree-like representation for describing the conditional distributions associated with the influence diagram. This representation for the distributions makes explicit the set of conditionally possible and impossible outcomes or alternatives and makes explicit the relationship between conditioning information and the conditional distributions assigned to each state of information.

2.1. Conditioning Functions and Atomic Distributions

In the conventional influence diagram representation, a conditional probability distribution $P_{X|A,B}(x, a, b)$ is considered to be a function of both the variable whose probability distribution is being described X, and its conditioning variables A and B. To better describe the relationship between states of information and the conditional probability distributions assigned to each particular state of information, we decompose the conditional probability distribution $P_{X|A,B}(x, a, b)$ into a *conditioning function* $C_{X|A,B}(a, b)$ that maps from the set of all possible conditioning scenarios (i.e., outcomes of A and B) to a set of *atomic distributions*, which, in turn, describe the probability distribution assigned in each conditioning scenario. Perhaps the

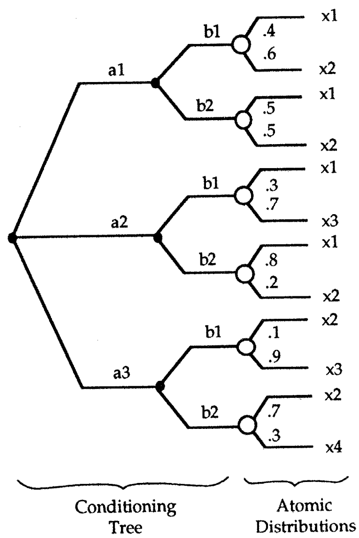

Figure 3. A distribution tree for $P_{X|A,B}(a, b)$.

best way to illustrate this decomposition is to view the probability distribution as a tree, as in Figure 3. We call these *distribution trees*. The atomic distributions are represented by the rightmost nodes and their branches in the distribution tree. The conditioning function is represented by the rest of the tree, which we call the *conditioning tree*. Each path through the conditioning tree represents a conditioning scenario and leads to an atomic distribution that describes the probabilities assigned in that scenario. In our functional notation, the atomic distribution represented by the topmost node in Figure 3 is written $C_{X|A,B}(a1, b1)$, and the probability on the upper branch of that node may be written $[C_{X|A,B}(a1, b1)](x1) = 0.4$.

As a very simple example of a distribution tree from the used car buyer problem, Figure 4a shows the distribution tree for the Car's Condition node. Since the Car's Condition node has no conditioning predecessors, its distribution is represented by a single atomic distribution giving Joe's probability assignments for the two possible outcomes, peach and lemon. Some additional examples of distribution trees are shown in Figure 4 (and in Figures 5–7) and will be discussed shortly.

Unlike a decision tree, a distribution tree makes no assertions about the type of the conditioning variables, dependencies among the conditioning variables, or the probabilities of the events represented in the conditioning tree. The order of variables in the conditioning tree is arbitrary. For example, the conditional distribution represented by the distribution tree of Figure 3 could be drawn with B followed by A. While certain orderings of conditioning variables in the tree may be more natural and may make the structure of the distribution more transparent, the different trees are simply different views of the same conditional probability distribution.

Also, atomic distributions for different conditioning scenarios may have different sets of possible outcomes. For example, in Figure 3, given $A = a1$ and $B = b1$, the two possible outcomes of X are $x1$ and $x2$, while, given $A = a2$ and $B = b1$, the two possible outcomes of X are $x1$ and $x3$. However, for each atomic distribution to make sense as a probability distribution, its set of possible outcomes must be a mutually exclusive partition of the set of possible outcomes of X given that particular conditioning scenario. Similarly, for the collection of atomic distributions to make sense as a conditional probability distribution, the union of these sets of possible outcomes over all conditioning scenarios must be a mutually exclusive partition of the set of possible outcomes of X. This implies that each set of *conditionally possible outcomes* defined by a particular atomic distribution must be a subset of the set of *all outcomes* of X.

Deterministic Atomic Distributions. In particular conditioning scenarios, the outcome of a variable may be known with certainty. The assertion of certainty is a very strong assertion that allows efficient computation as well as concise representation. Accordingly, we distinguish between *deterministic atomic distributions*, those atomic distributions with a single possible

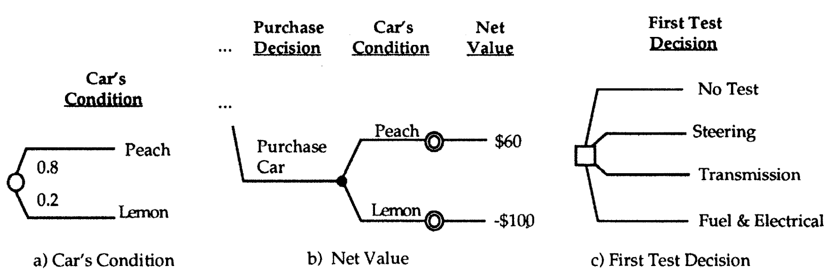

Figure 4. Distributions for Car's Condition, Net Value and First Test Decision.

outcome, and *stochastic atomic distributions*, those atomic distributions with more than one possible outcome. Graphically, deterministic atomic distributions are drawn as double-bordered circles while stochastic atomic distributions are drawn as single-bordered circles. A conditional distribution may contain any mixture of deterministic and stochastic atomic distributions.

While a distribution may contain both deterministic and stochastic atomic distributions, nodes whose conditional distributions contain only deterministic atomic distributions are special and are called *deterministic nodes*. In an influence diagram, deterministic nodes are drawn with a double border, and are distinguished from *stochastic nodes* whose distribution contains at least one stochastic atomic distribution. For example, in the used car buyer problem, the Net Value node is deterministic as Net Value is uniquely determined by the outcomes of its direct predecessors. A partial distribution tree for the Net Value node is shown in Figure 4b.

Atomic Alternative Sets. In an influence diagram, decision nodes represent those variables whose values are chosen by the decision maker. Having introduced the distinctions of conditioning functions and atomic distributions, we can refine the definition of decision nodes to include variables whose values are chosen by the decision maker in some, but not necessarily all, conditioning scenarios. We can also have different alternatives available in different conditioning scenarios. We describe the set of alternatives available to a decision maker in a particular conditioning scenario with an *atomic alternative set*. A node is considered a decision node if its distribution contains at least one atomic alternative set. One might want to mix atomic distributions and atomic alternative sets in the same conditional distribution if the outcome of some conditioning variable determines whether the decision maker can choose the value of the specified variable. Graphically, atomic alternative sets are drawn like stochastic atomic distributions with squares instead of circles. For example, the First Test Decision node in the used car buyer problem, like the Car's Condition node, has no predecessors and its distribution consists of a single atomic alternative set, as shown in Figure 4c.

2.2. Coalescence

Having introduced the distinction between conditioning functions and atomic distributions, we can explicitly represent the relationship between conditioning scenarios and probability distributions assigned given those scenarios. For example, one may assign the same probability distribution to two different scenarios. To capture this assertion, the two conditioning scenarios are said to *share* the same atomic distribution. Stated formally, two conditioning scenarios ($A = a1$, $B = b1$) and ($A = a2$, $B = b2$) share an atomic distribution if, $C_{X|A,B}(a1, b1) = C_{X|A,B}(a2, b2)$. A conditional distribution is said to be *coalesced* if any of its atomic distributions are shared. (This is essentially a formalization of Olmsted's notion of "coalescence within an influence diagram.") In our view, the sharing of an atomic distribution should be asserted explicitly by the person assigning the probability distribution. We do not consider two atomic distributions with the same outcomes and probabilities to be shared if they are not explicitly identified as shared; the similarity in this case could perhaps be due to limited numerical precision in the specification of probabilities rather than an identity intended by the person assigning the distributions.

Coalescence sometimes occurs in particularly useful patterns. These patterns simplify the description of the probability distribution and allow computational efficiencies. One such pattern occurs in a distribution with two or more conditioning variables when some subset of conditioning scenarios shares a set of atomic distributions so that they may be seen as sharing a distribution tree. Given a distribution for X conditioned on A and B, we say that two conditioning variable outcomes $a1$ and $a2$ *share a subtree* if $C_{X|A,B}(a1, b) = C_{X|A,B}(a2, b)$ for all outcomes b of the variable B. As an example, consider the distribution for the First Test Results node in the used car buyer problem shown in Figure 5. If Joe decides to test one system, he will observe either 0 or 1 Defects. Given

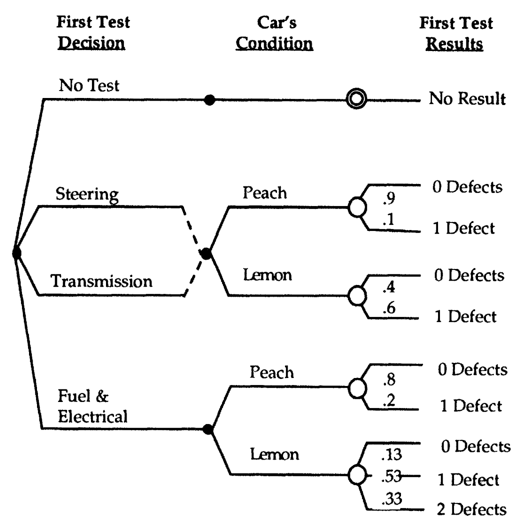

Figure 5. Distribution for First Test Results.

Joe's belief that the defects are equally likely to be in any of the 10 systems, the distribution Joe assigns for the test of the transmission is the same as the distribution assigned for the test of the steering system. This can be captured in the distribution tree by having the two different tests share a subtree, as illustrated in Figure 5.

Another simplifying pattern of sharing occurs when, for some subset of conditioning scenarios, the corresponding conditional probability distributions are assigned independently of the outcome of the rest of the conditioning variables. We say that a distribution for some variable X conditioned on variables A and B is *collapsed* across the variable B given $A = a$ if $C_{X|A,B}(a, b1) = C_{X|A,B}(a, b2)$ for all outcomes $b1$ and $b2$ of B. (More generally, a collapsed distribution may share a subtree rather than a single atomic distribution.) An example of a collapsed distribution is given in Figure 5. If Joe decides not to perform any tests, he will observe no test results regardless of the car's condition; thus the distribution for First Test Results can be collapsed across Car's Condition given No Test. This collapsed distribution is drawn as an asymmetric tree with no label or branches for Car's Condition. This notation indicates that, given that no tests are performed and regardless of whether the car is a peach or a lemon, the outcome No Result certainly occurs.

It is interesting to compare the distribution tree describing the conditional probability distribution for First Test Results (shown in Figure 5) to part d of Table I. The tree representation is more compact than the table and the structure of the distribution—which distributions are assigned in which scenarios and which outcomes are possible in which scenarios—is much clearer in the tree than in the table.

2.3. Clipping

When assessing a conditional probability distribution, one must usually specify an atomic distribution for each conditioning scenario. However, in certain situations, a complete assessment may not be necessary. For example, some of these conditioning scenarios may be impossible (i.e., have zero probability), making any assigned distribution irrelevant. If the conditioning scenario ($A = a$, $B = b$) is impossible, there are no possible outcomes of the variable X, and we say that the atomic distribution $C_{X|A,B}(a, b)$ is *clipped*. When assessing a conditional probability distribution, one may "clip" an atomic distribution to assert that a particular conditioning scenario is impossible. It is important to note that clipping is not an assertion about the distribution of X, but rather an assertion about the distributions of its direct predecessors, A and B. If the distributions for the predecessors of X are specified, a computer program could detect impossible conditioning scenarios and automatically clip the appropriate atomic distributions before the distribution for X is specified.

To illustrate clipping, consider the distribution for the Second Test Decision node of the used car buyer example shown in Figure 6. Here, Joe has the option to test the differential system only if he chooses to test the transmission. There is a great deal of clipping in this distribution, reflecting the many impossible combinations of testing alternatives and results. For example, if No Test is chosen in First Test Decision, the only possible outcome of the test is No Result. The impossibility of observing zero, one, or two defects in the first test given No Test is indicated by the omission of branches corresponding to these outcomes in the conditioning tree. Similarly, the impossibility of observing No Result given that a test was performed is indicated by omitting branches corresponding to that outcome.

Note that both clipped and collapsed distributions are represented as asymmetric distribution trees. A clipped distribution tree is distinguished from a coalesced distribution tree by the labels on the remaining conditioning variable outcomes. The absence of a label in a coalesced distribution tree indicates that the same probability distribution is assigned regardless of the conditioning variable outcome. The labels in a clipped distribution tree indicate that the atomic distributions corresponding to the missing conditioning variable outcomes are clipped. The other distributions for the used car buyer problem (not shown here) provide additional examples of both clipping and coalescence.

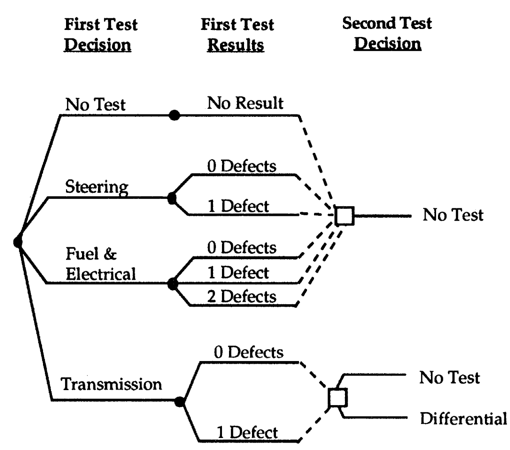

Figure 6. Distributions for Second Test Decision.

2.4. Unspecified Atomic Distributions

It is also unnecessary to specify certain atomic distributions when they are not needed to answer particular questions. For example, consider the example of a hypothetical research and development problem, shown in Figure 7. Here, technical success is defined as overcoming two technical hurdles: crystallization and superconductivity. The arrow from the crystallization node to the superconductivity node indicates that the probabilities assigned to achieving superconductivity depend on whether crystallization is achieved. However, if we are only interested in the probability of technical success, we need not assign probabilities to the success of superconductivity if crystallization is not achieved: The distribution for technical success is collapsed across superconductivity given that crystallization fails. However, if we were interested in computing the unconditional probability of achieving superconductivity, we would need to specify the atomic distribution for superconductivity given that crystallization is not achieved. Thus, whether a distribution is unnecessary depends not only on the structure of the successor distribution, but also on what questions we want to answer with the influence diagram.

To assert that a particular atomic distribution is unnecessary, we leave it *unspecified*. Graphically, unspecified atomic distributions are drawn with question marks replacing the atomic distribution, as in Figure 7. It is also possible to *partially specify* an atomic distribution by defining the set of conditionally possible outcomes given $A = a$, $B = b$ without specifying their probabilities. This is useful when the successor distribution is not collapsed across all possible outcomes of X, but a common atomic distribution is shared by those outcomes of X that are conditionally possible given $A = a$, $B = b$. Note that although both clipped atomic distributions (indicating impossible scenarios) and unspecified atomic distributions (indicating irrelevant distributions) imply that no distribution needs to be specified, because of the different meanings, the two cases are treated differently when solving an influence diagram. (See the discussion of reversing an arrow in the next section.)

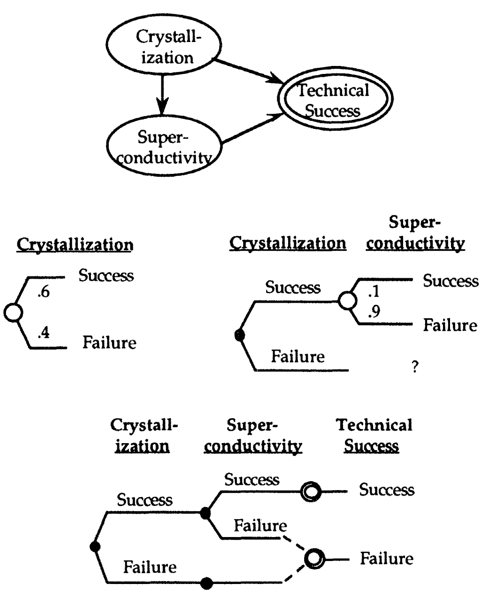

Figure 7. An example of an unspecified atomic distribution.

3. INFLUENCE DIAGRAM TRANSFORMATIONS

As discussed in the Introduction, influence diagrams have proven valuable for solving decision problems (Olmsted 1983, Shachter 1986, 1988). The basic idea underlying influence diagram solution algorithms is to recognize that an influence diagram represents one possible expansion of a joint probability distribution and that we can transform one influence diagram into a different one which represents a different expansion of the same probability distribution. Given an influence diagram in one form, we can repeatedly apply these transformations to convert the original influence diagram into a new one that answers a particular question. For example, we can transform the original used car buyer influence diagram (Figure 1) into another influence diagram that shows how Net Value depends on Purchase Decision and what is known at the time the purchase decision is made. This influence diagram is shown in Figure 8. Algorithms for accomplishing these transformations are described by Shachter (1986, 1988) (see also Rege and Agogino).

Our goal in this section is to show how the structural properties of the conditional distributions can be propagated during the solution process. For example, the clipping and coalescence in the distributions of the original influence diagram of Figure 1 implies clipping and coalescence in the distributions of the derived influence diagram of Figure 8. By recognizing special structures in these conditional distributions, we can achieve computational efficiencies over the conventional symmetric influence diagram procedures. In the next section, we will demonstrate these efficiencies by considering the solution of the used car buyer problem.

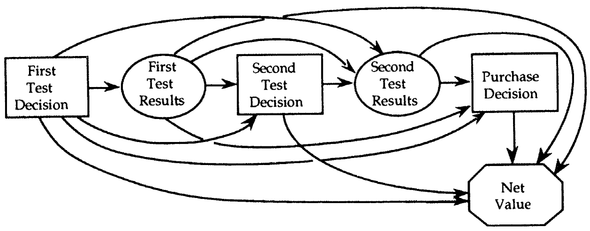

Figure 8. A transformed influence diagram for the used car buyer.

3.1. Basic Influence Diagram Transformations

To describe the effects of the influence diagram transformations on the conditional distributions, we consider four basic transformations: *adding an arrow*, *removing an arrow*, *reversing an arrow*, and *determining a decision node*. Note that these basic transformations differ slightly from those described by Shachter (1986, 1988) (see also Olmsted). For example, Shachter's arrow reversal operation includes the addition of any necessary arrows while our arrow reversal operation assumes that the necessary arrows have already been added. Similarly, his "removing a chance node X into a chance node Y" can be described in terms of our operations as: 1) adding arrows so that X and Y have the same set of direct predecessors, 2) reversing the arrow from X to Y, and 3) simply deleting X. The other basic operations described by Shachter ("removing a decision node" and "deterministic node propagation") can also be defined in terms of the four basic operations discussed here. Thus, his algorithms for solving decision and probabilistic inference problems in influence diagrams can be implemented in terms of the basic transformations described here.

The primary advantage of the basic transformations used here is that they simplify the description of the effects of these operations. Given a particular algorithm for evaluating an influence diagram, it may be more efficient to implement composite transformations that combine several basic transformations. For example, if one is using Shachter's algorithms for evaluating an influence diagram, it would be more efficient to combine the procedures for adding and reversing arrows into a single procedure that implements his form of arrow reversal.

3.1.1. Adding an Arrow

The omission of a conditioning arrow in an influence diagram represents an assertion of conditional independence. Provided a cycle is not created, a new arrow can always be added into a chance node without affecting the diagram's underlying joint distribution. Although adding an unnecessary arrow obscures an assertion of independence in the influence diagram, that assertion is preserved in the structure of the resulting conditional distribution: The existing atomic distributions of the variable at the head of the newly added arrow are shared by all outcomes of the variable's new direct predecessor. The assertion of conditional independence is thus lost at the level of relation, but preserved at the level of function.

For example, suppose that we want to add an arrow from node X to node Y, where node A is a direct predecessor of Y; X's distribution is unaffected by adding an arrow from it to Y. If we let $C_{Y|A}$ denote conditioning function before the transformation and $C_{Y|X,A}$ denote conditioning function after the transformation, then $C_{Y|X,A}(x, a) = C_{Y|A}(a)$ for all a and x. Figure 9 shows the conditional distribution resulting from adding an arrow from First Test Decision to Car's Condition in the used car buyer influence diagram. Note that any clipping or sharing in the original distribution is preserved in the resulting distribution and no new atomic distributions are computed.

3.1.2. Removing an Arrow

Consider a chance variable Y conditioned on a deterministic variable X and on all of X's direct predecessors, as shown on the left side of Figure 10. Given the value of X's direct predecessors, the outcome of X is known with certainty. Therefore, knowing the outcome of X in addition to knowing the outcomes of its direct predecessors gives no additional information about Y's distribution. Thus, the arrow from X to Y can be removed without affecting the underlying joint distribution, as shown on the right side of Figure 10.

A new conditioning function for Y, $C_{Y|A,B}$, is computed with its values depending on the original conditioning function for Y, $C_{Y|X,A,B}$, and the distribution

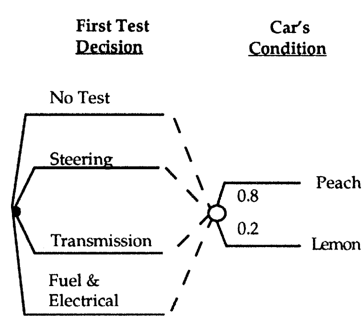

Figure 9. Distribution for Car's Condition after adding an arrow from First Test Decision to Car's Condition.

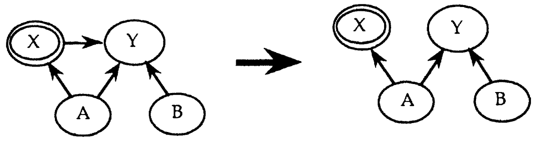

Figure 10. Removing an arrow.

for X. Considering a particular conditioning scenario $A = a$, if the deterministic value of X is specified (i.e., not clipped and not unspecified), then Y's new distribution is obtained by selecting the atomic distribution corresponding to X's deterministic outcome. Formally, if x is the deterministic outcome of X given $A = a$, then $C_{Y|A,B}(a, b) = C_{Y|X,A,B}(x, a, b)$ for all b. Figure 11 gives an example of this case. Note that clipping or coalescence present in the original distribution, $C_{Y|X,A,B}$, is propagated to the new distribution, $C_{Y|A,B}$. The cases in which the atomic distribution $C_{X|A}$ is clipped or unspecified are analogous to the cases of arrow reversal described below. These other cases, like the case described here, do not require computing any new atomic distributions.

3.1.3. Reversing an Arrow

An arrow from a chance node X to a chance node Y means that the distribution for Y is conditioned on the outcome of X. Given a distribution for X (sometimes referred to as the prior distribution) and a distribution for Y conditioned on X (the likelihood distribution), we can use Bayes' rule to reverse this arrow. In the new influence diagram, we have a distribution for X conditioned on the outcome of Y (the posterior distribution) and an unconditional distribution for Y (the preposterior distribution). This arrow reversal operation, first described in Howard and Matheson, plays a key role in solving decision and inference problems using influence diagrams (Shachter 1986, 1988). Here, we extend the arrow reversal operation to handle clipped, coalesced, and unspecified distributions. Although these different cases make the arrow reversal operation somewhat more difficult to describe, we will see that substantial computational efficiencies are gained by recognizing these special cases.

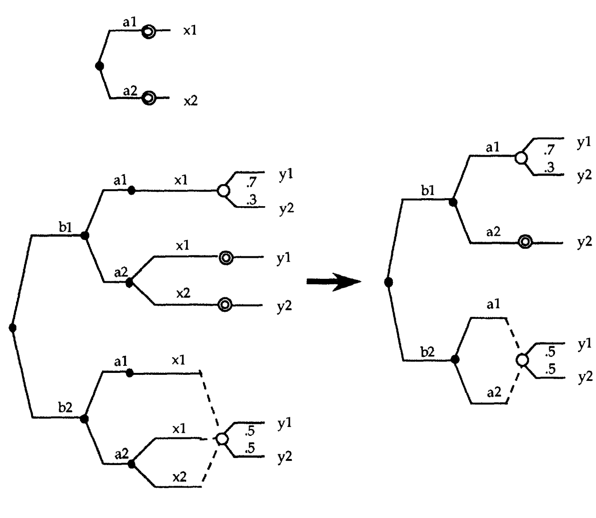

Figure 11. The effect of removing an arrow on the conditional distribution of Y.

Consider the influence diagram in Figure 12 as the prototypical instance of an arrow reversal. The distributions for X and Y must be conditioned on the same states of information; this means that X and Y may not have any direct predecessors that are not also direct predecessors of both nodes, except X, which is a direct predecessor of Y but not of itself. Both X and Y may have any number of successors provided that none of the successors of X are predecessors of Y. If the latter condition does not hold, reversing the arrow would introduce a cycle, and the influence diagram would no longer correspond to an expansion of a joint probability distribution.

Arrow reversal is carried out by considering each conditioning scenario (i.e., each outcome of A) separately. The effects of arrow reversal depend on the structure of the prior and likelihood distributions for the particular outcome of A. Table II describes the different special structures and indicates which of eight possible cases corresponds to that structure. The columns of Table II distinguish the four different types of atomic distributions occurring in the prior (i.e., X's) distribution: deterministic, clipped, unspecified, and stochastic. The rows of the table distinguish among different special structures in the likelihood (i.e., Y's) distribution. In each case, we can write formulas for the conditioning functions of the posterior and preposterior distributions, $C_{X|Y,A}$ and $C_{Y|A}$, from the prior and likelihood distributions, $C_{X|A}$ and $C_{Y|X,A}$.

In Case 1, the prior atomic distribution $C_{X|A}(a)$ is deterministic. Because given $A = a$, X's outcome is known with certainty, conditioning on Y provides no additional information about the outcome of X. Thus, in the posterior distribution, $C_{X|Y,A}$, X's deterministic

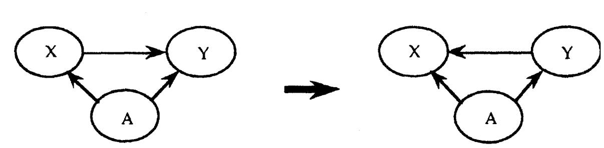

Figure 12. Prototypical arrow reversal.

Table II
Cases for Arrow Reversal

Structure of Likelihood Distribution	Type of Prior Atomic Distribution: $C_{X\mid A}(a)$			
	Deterministic	Clipped	Unspecified	Stochastic
Collapsed Distribution	Case 1	Case 2	Case 3	Case 5
Some Relevant Clipping	Case 1	Case 2	Case 4	Case 6
Some Relevant Distribution Unspecified	Case 1	Case 2	Case 4	Case 7
Full Distribution	Case 1	Case 2	Case 4	Case 8

outcome is shared across all conditionally possible outcomes of Y. As in the basic transformation of removing the arrow from X to Y, we compute $C_{Y\mid A}(a)$ by selecting the atomic distribution corresponding to $C_{X\mid A}(a)$'s deterministic outcome in $C_{Y\mid X,A}$.

In Case 2, the prior distribution $C_{X\mid A}(a)$ is clipped, indicating that the conditioning scenario $A = a$ is impossible. The arrow reversal operation propagates that assertion by clipping both the preposterior $C_{Y\mid A}(a)$ and the posterior $C_{X\mid Y,A}(y, a)$ for all y.

In Cases 3 and 4, the prior distribution $C_{X\mid A}(a)$ is unspecified. If the likelihood distribution $C_{Y\mid X,A}$ shares one atomic distribution for all possible outcomes of X given $A = a$ (Case 3), as in the hypothetical research and development problem discussed in the previous section, then the preposterior distribution $C_{Y\mid A}(a)$ must be that shared atomic distribution. If we do not have this sharing, we cannot compute the preposterior, and it becomes unspecified (Case 4). In either case, the posterior distribution $C_{X\mid Y,A}(a)$ remains unspecified because we cannot compute it without specifying a prior distribution.

The logic behind Cases 5, 6, and 7 is similar to the previous three cases. Here, the prior distribution $C_{X\mid A}(a)$ is a stochastic atomic distribution. If, given $A = a$, the likelihood distribution $C_{Y\mid X,A}(x, a)$ shares one atomic distribution for all conditionally possible outcomes of X (Case 5), then the preposterior distribution $C_{Y\mid A}(a)$ must be that shared atomic distribution. Since, in this case, the likelihood distribution is collapsed, indicating that information about X tells us nothing about Y (given $A = a$, the posterior distribution $C_{X\mid Y,A}(y, a)$ must be equal to the prior $C_{X\mid A}(a)$ for all conditionally possible outcomes of Y. In Cases 6 and 7, some relevant distribution is either clipped or unspecified, and the resulting posterior and preposterior distributions, $C_{X\mid Y,A}(y, a)$ and $C_{Y\mid A}(a)$, must also be clipped or unspecified.

Case 8 of arrow reversal corresponds to the general case described in Howard and Matheson, where Bayes' rule is used to compute the posterior and preposterior atomic distributions. Note that of the eight cases described, this is the only one that requires computing new atomic distributions. In the seven other cases, the structural properties of the prior and likelihood distributions allow us to avoid computing new atomic distributions.

As an example of arrow reversal, Figure 13 shows the distributions for Car's Condition and First Test Results after reversing the arrow between them. Note that before the arrow can be reversed, we must add an arrow from First Test Decision to Car's Condition: the resulting prior distribution for Car's Condition was shown in Figure 9. The likelihood distribution for First Test Results is shown in Figure 5. The conditioning scenario corresponding to No Test for the First Test Decision is an example of Case 5: The likelihood distribution is collapsed across Car's Condition and, consequently, the posterior distribution for Car's Condition is equal to the prior distribution. The other distributions are examples of the general case (Case 8) in which Bayes' rule is used to compute the revised atomic distributions. Note the deterministic atomic distribution in the posterior distribution for Car's Condition: Since a peach has exactly one defect, if we observe two defects the car is certainly a lemon.

The computations of arrow reversal can be simplified further by recognizing sharing in the joint distribution for X and Y. Suppose that two conditioning scenarios, $A = a1$ and $A = a2$, share the same joint probability distribution for X and Y: $C_{X\mid A}(a1) = C_{X\mid A}(a2)$ and $C_{Y\mid X,A}(x, a1) = C_{Y\mid X,A}(x, a2)$ for all possible outcomes of $C_{X\mid A}(a1)$.

Since arrow reversal converts from one expansion of the joint distribution to the other, two conditioning scenarios that share the same joint distribution before arrow reversal should also share the same joint distribution after arrow reversal. Thus, if we compute the posterior and preposterior distributions for one conditioning scenario, say $a1$, the computed distributions can be shared with the other conditioning scenario $a2$. Recognizing this joint coalescence avoids redundant computing of new atomic distributions and

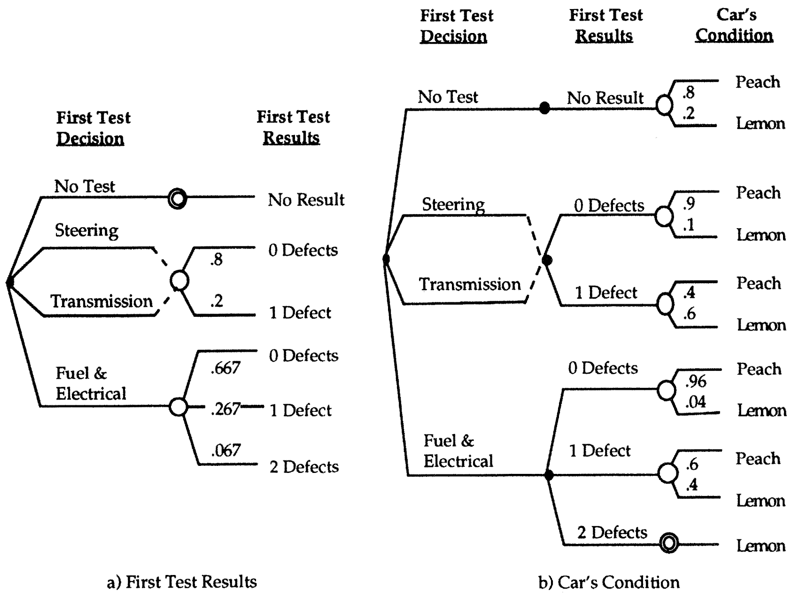

Figure 13. Distributions after reversing the arrow from Car's Condition to First Test Results.

preserves the joint coalescence captured in the original distributions. An example of joint coalescence is shown in Figure 13. In the distributions for Car's Condition and First Test Results before the arrow reversal (shown in Figures 9 and 5), the scenarios corresponding to steering and transmission share the same joint distribution. These two conditioning scenarios also share the same joint distribution after arrow reversal (Figure 13).

3.1.4. Determining a Decision Node

Given an influence diagram with decision nodes (and a unique value node), we can replace a decision node with a chance node that represents its optimal policy without affecting the optimal joint probability distribution in any way. The basic transformation of determining a decision node performs this replacement for a decision that is explicitly represented as a choice among alternative value lotteries. In calling this operation *determining* a decision node, we intend to invoke two meanings of the term *to determine*: "to come to a decision" and "to fix the position of." In the first sense of *determine*, the basic transformation decides upon the optimal policy. In the second sense of *determine*, the basic transformation commits the decision maker to the optimal policy in the context of the given influence diagram.

To determine decision node *D*, *D* must be a direct predecessor of a unique value node *V*, and every other direct predecessor of *V* must also be a direct predecessor of *D*. If this is the case, the choice represented by the decision node amounts to a choice among value lotteries in each possible conditioning scenario. Figure 14 shows the prototypical example of determining a decision node in an influence diagram. Note that the decision node *D* may have direct predecessors that are not also direct predecessors of the value node *V*; node *A* in Figure 14 is an example. This allows the possibility that the decision maker may know the outcome of *A* and the set of alternatives at decision *D* may be affected by the outcome of *A*, even though, given knowledge of *B*, knowledge of *A* gives no additional information about *V*.

Determining a decision node converts the decision node *D* into a chance node. If the distribution for the decision node originally contains only atomic alternative sets and, possibly, deterministic atomic distributions, *D* becomes a deterministic node. If original distribution contains any stochastic atomic distributions, *D* becomes a stochastic chance node as a result of this operation. The value node *V* is unaffected by determining *D*.

In general, to determine a decision node, we

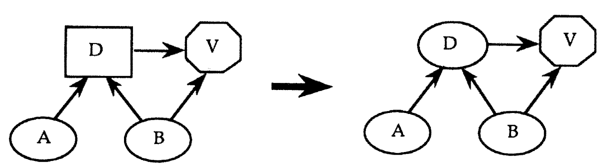

Figure 14. Determining a decision node.

examine each conditioning scenario that corresponds to an atomic alternative set and select an alternative corresponding to the best lottery in that case. (Here, best means the lottery with maximum expected value or expected utility.) As with arrow reversal, special structures in the conditional distributions are treated as special cases.

4. COMPUTATIONAL EFFICIENCY: SOLVING THE USED CAR BUYER PROBLEM

To illustrate the use of these basic influence diagram transformations and the efficiencies provided by the asymmetric influence diagram representation, we solve the used car buyer problem using Shachter's (1986) algorithm. By solving the influence diagram, we mean computing optimal policies for all decision nodes and computing the unconditional probability distribution for Net Value given by following the optimal policy (the value lottery).

4.1. Measures of Efficiency

We will compare the conventional "symmetric" influence diagram representation with our "asymmetric" representation using two measures of efficiency: solution time and storage space. To measure solution time, we will examine the time required to perform particular steps in the solution algorithm. To measure storage space, we will examine the sizes of the distributions in the influence diagram at different points in the solution procedure. We will consider three different measures of the size of a distribution. Because each basic transformation requires iterating over the conditioning scenarios and the different possible outcomes, these measures provide a good explanation of the time involved in the solution process and the advantages of the asymmetric representation.

The Number of Conditioning Scenarios. In the symmetric representation, the number of conditioning scenarios for a node's distribution is equal to the product of the number of outcomes of each direct predecessor of that node. For example, in the distribution of Car's Condition given First Test Decision and First Test Results shown in Figure 13b, the symmetric representation would be counted as having a total of 16 conditioning scenarios given by the product of the four alternatives in First Test Decision and the four outcomes of First Test Results. If a node has no predecessors, we say that its distribution has a single conditioning scenario.

In the asymmetric representation, the number of conditioning scenarios corresponds to the number given in the symmetric representation less those that are clipped. For example, the distribution for Car's Condition given First Test Results and First Test Decision would have 8 conditioning scenarios (16 in the symmetric representation minus 8 impossible combinations of tests and test results). The difference between the number of conditioning scenarios in the symmetric representation and asymmetric representations highlights the value of recognizing clipping.

The Number of Atomic Distributions. In the symmetric representation, the number of atomic distributions is equal to the number of conditioning scenarios, as each conditioning scenario must have an assigned distribution. Thus, the symmetric distribution for Car's Condition given First Test Decision and First Test Results would have 16 atomic distributions.

In the asymmetric representation, the total number of atomic distributions may be less than the number of conditioning scenarios as some atomic distributions may be shared by several conditioning scenarios. The asymmetric distribution for Car's Condition given First Test Decision and First Test Results has six atomic distributions, as shown in Figure 13b. The difference between the number of atomic distributions in the symmetric and asymmetric representations (16 and 6 in the example) shows the cumulative benefit of recognizing both clipping and coalescence. The difference between the number of conditioning secnarios and the number of atomic distributions in the asymmetric representation (eight and six in the example) highlights the incremental benefit of recognizing coalescence.

The Number of Outcome-Probability Pairs. In the symmetric representation, the total number of outcome-probability pairs is equal to the product of the number of atomic distributions times the number of possible outcomes. The symmetric distribution for Car's Condition given First Test Decision and First Test Results has 32 outcome-probability pairs (16 atomic distributions times 2 outcomes). The total number of outcome-probability pairs for a deterministic node is equal to the number of atomic distributions, because one need only store a single outcome for each scenario.

In the asymmetric representation, different atomic distributions may have different sets of possible outcomes and the number of outcome-probability pairs is computed by direct counting. Each deterministic atomic distribution is counted as one outcome-probability pair. The asymmetric distribution for Car's Condition given First Test Decision and First Test Results has 11 outcome-probability pairs. The difference in the total number of outcome-probability pairs in the two representations (32 and 11 in the example) reflects the cumulative benefit of recognizing clipping and coalescence, and of allowing different atomic distributions to have different sets of possible outcomes. The incremental benefit of allowing different sets of possible outcomes can be measured by comparing the ratio of the number of outcome-probability pairs over the number of atomic distributions in the symmetric and asymmetric cases (32/16 and 11/6 in the example).

4.2. Results

Table III compares the efficiencies of the symmetric and asymmetric representations of the used car buyer influence diagram. The rows of the table correspond to different points of the solution algorithm. Each row lists the total number of conditioning scenarios, atomic distributions, and outcome-probability pairs for the symmetric and asymmetric representations of the used car buyer influence diagram. The first row of Table III shows the sizes of distributions in the initial influence diagram for the used car buyer (shown in Figure 1). Comparing the total number of conditioning scenarios for the symmetric and asymmetric representations (234 versus 79), we see that there is a fair amount of clipping in the problem. Comparing the number of atomic distributions with the number of conditioning scenarios in the asymmetric representation (40 versus 79), we see that there is also a fair amount of coalescence in the problem. Comparing the total number of outcome-probability pairs for the two representations (598 versus 55), we see that the asymmetric representation is considerably more compact than the conventional symmetric representation. Examining the ratios of the number of outcome-probability pairs over the number of atomic distributions for the two representations (598/234 = 2.56 versus 55/40 = 1.38), we see that there is some incremental benefit in allowing different sets of possible outcomes for different atomic distributions.

The differences between the symmetric and asymmetric influence diagram representations become even more pronounced in the intermediate stages of the solution procedure. In particular, the symmetric influence diagram becomes very large after removing the Car's Condition node into the Net Value node. The influence diagram at this point is shown in Figure 8. Here, the value node has 25 different possible values and 6 direct predecessors, resulting in a total of 288 conditioning scenarios in the symmetric representation and a total of 7,200 outcome-probability pairs.

Table III
Efficiency Results for Solving the Used Car Buyer Problem

	Size of Influence Diagram After Operation							
	Number of Conditioning Scenarios		Number of Atomic Distributions		Number of Outcome-Probability Pairs		Time to Perform Operation (Seconds)	
Operation	Symmetric	Asymmetric	Symmetric	Asymmetric	Symmetric	Asymmetric	Symmetric	Asymmetric
0. Initial Sizes	234	79	234	40	598	55	—	—
1. Reverse Arrow from Car's Condition to First Test Results	245	82	245	43	612	60	1.0	0.6
2. Reverse Arrow from Car's Condition to Second Test Results	261	77	261	44	612	63	7.2	1.6
3. Remove Car's Condition into Net Value	437	71	437	39	7,636	70	131.4	5.8
4. Remove Purchase Decision into Net Value	245	47	245	24	2,644	43	28.9	3.7
5. Remove Second Test Results into Net Value	149	35	149	19	948	37	20.7	1.2
6. Remove Second Test Decision into Net Value	133	33	133	17	532	32	3.9	0.8
7. Remove First Test Result into Net Value	117	25	117	10	216	25	3.1	0.7
8. Remove First Test Decision into Net Value	114	22	114	7	138	10	0.4	0.3
					Total Elapsed Time		196.7	14.8

However, many of these conditioning scenarios are impossible (all but 36) and no more than 2 of the 25 possible values are conditionally possible in any particular conditioning scenario. As a result, the asymmetric influence diagram is much more compact at this point: The symmetric representation has a total of 7,636 outcome-probability pairs versus a total of 70 outcome-probability pairs in the asymmetric representation.

As the algorithm continues and more nodes are removed, the differences between the two representations become less pronounced. In the final influence diagram, the only distributions remaining are those required to store the optimal policies for the decision nodes and the value lottery. Even here, we see some advantages to the asymmetric representation as both the optimal policies and the value lottery are represented more compactly (138 outcome-probability pairs versus 10).

In terms of solution time, the asymmetric representation is also substantially more efficient than its symmetric counterpart. Table III shows the time required to perform each step in solving the used car buyer problem. (All computations were performed using the same Macintosh II computer.) The asymmetric version of the used car buyer was solved in 14.8 seconds; an equivalent symmetric representation of the same problem was solved in 196.7 seconds. As expected, the amount of time required to perform a particular step of the solution algorithm is closely related to the sizes of the distributions involved. The exact solution times are, of course, highly dependent on the particular software and hardware implementation of the solution algorithm as well as the choice of solution algorithm.

4.3. Comparison With Decision Trees

It is interesting to compare both influence diagram representations to the original decision tree for the used car buyer given in Howard (1962). Not counting "nature's tree," the tree contains a total of 97 outcome-probability pairs. Here, we have counted each branch in the tree and each endpoint value, analogous to a deterministic atomic distribution, as one outcome-probability pair. This number is somewhat higher than the total number of outcome-probability pairs in our asymmetric influence diagram (55), reflecting some duplication of probabilities and values in the decision tree that is avoided in the asymmetric influence diagram. The decision tree is much more compact than the symmetric influence diagram which has a total of 598 outcome-probability pairs.

Note that Howard's tree for the used car buyer is a relatively sophisticated decision tree. It exploits coalescence in the problem by omitting branches of the tree that are duplicated elsewhere and is abbreviated in that, unlike the partial tree shown in Figure 2, it omits deterministic outcomes like No Test Results. A less sophisticated decision tree that recognizes the asymmetries in the problem but is not coalesced or abbreviated would have a total of 182 outcome-probability pairs. A fully symmetric decision tree for the used car buyer problem would have a total of 1,588 outcome-probability pairs. The reason the decision trees (even the sophisticated version) are less compact than the asymmetric influence diagram can be traced back to their inability to capture the fact that Net Value is conditionally independent of the test results given the other variables in the problem. As a result, identical probabilities and values are duplicated in several places.

5. CONCLUDING REMARKS

In this paper, we have extended the definition of an influence diagram to describe the structure of a decision problem at the level of function. This extension allows one to represent asymmetric decision problems in influence diagrams without obscuring the structure of the problem or increasing the time and space required for solution. Comparing our asymmetric influence diagram representation with the conventional symmetric influence diagram representation, we find that the savings in the time and space required for solution are substantial: The time to solve the used car buyer problem is reduced by a factor of more than 10, and the space required for solution (as measured by the maximum total number of outcome-probability pairs) is reduced by a factor of more than 100. While these exact results are specific to the used car buyer problem, we believe that the asymmetric influence diagram representation effectively dominates the conventional symmetric influence diagram representation. The two representations are essentially equivalent for perfectly symmetric problems, and the asymmetric influence diagram representation is both clearer and more efficient for asymmetric problems.

Comparing asymmetric influence diagrams with decision trees is more complex. In general, the efficiency of the influence diagram representation, as compared with the decision tree representation, depends on the amount of independence among the variables in the problem. In problems with no independence (even symmetric problems), the influence diagram representation has no real computational advantages over the decision tree. Thus, given an asymmetric decision problem with little or no independence, the decision tree representation may be the most natural and efficient representation. In problems with both independence and asymmetries, the asymmetric influence diagram representation—capturing asymmetries like a decision tree and independence like an influence diagram—provides an attractive alternative to both the decision tree and conventional symmetric influence diagram representations.

Although we have not yet done a detailed performance analysis, we would like to offer a few observations about the strengths and weaknesses of our representation as compared to those described by Call and Miller (1990) and Fung and Shachter (1990). Our representation and Fung and Shachter's are similar in that both extend the influence diagram representation by incorporating features of decision trees. One implication of this influence diagram-based strategy is that both representations capture asymmetries "locally" in the distributions (or contingencies) for particular variables. In contrast, Call and Miller use separate influence diagram-like and decision tree-like representations in DPL to describe each decision problem. An advantage of the DPL approach is that "global" asymmetries can be captured easily in the decision tree-like representation. For example, one can label some part of the tree and, at some other point in the tree, use a "goto" statement (they call it "perform") to indicate that the remainder of the tree is exactly the same as the labeled part. In our and Shachter and Fung's influence diagram representations, such a "global" statement would be captured in the distributions (or contingencies) for the particular variables involved; if there are many variables involved this "local" representation may become awkward and inefficient. The disadvantage of the DPL approach is that one must examine both the decision tree-like and influence diagram-like representations to obtain a complete description of the problem. In contrast, the influence diagram representations both provide a complete description of the problem in a single, consistent framework.

ACKNOWLEDGMENT

This paper is based on work done in 1986–1988 while the first author was at Strategic Decisions Group (SDG). The authors express their gratitude to their colleagues at SDG, particularly Léonard Bertrand, for their many helpful comments on this research and the various drafts of this paper. We are also very grateful for the helpful comments provided by the anonymous referees and the area editor.

REFERENCES

CALL, H. J., and W. A. MILLER 1990. A Comparison of Approaches and Implementations for Automating Decision Analysis. *Reliab. Engin. and Syst. Safety* **30,** 115–162.

FUNG, R. M., and R. D. SHACHTER 1990. Contingent Influence Diagrams. Working Paper, Department of Engineering-Economic Systems, Stanford University, Stanford, Calif.

HOWARD, R. A. 1962. The Used Car Buyer. In *Readings on the Principles and Applications of Decision Analysis, Vol. II,* R. A. Howard and J. E. Matheson (eds.). Strategic Decisions Group, Menlo Park, Calif., 689–718.

HOWARD, R. A. 1988. Progress and Promise in Decision Analysis. *Mgmt. Sci.* **34,** 679–695.

HOWARD, R. A., and J. E. MATHESON 1981. Influence Diagrams. In *Readings on the Principles and Applications of Decision Analysis, Vol. II,* R. A. Howard and J. E. Matheson (eds.). Strategic Decisions Group, Menlo Park, California, 719–762.

MILLER, A. C., M. W. MERKHOFER, R. A. HOWARD, J. E. MATHESON and T. R. RICE 1976. Development of Automated Computer Aids for Decision Analysis. Technical Report 3309, SRI International, Menlo Park, Calif.

OLMSTED, S. M. 1983. Representing and Solving Decision Problems. Ph.D. Dissertation, Department of Engineering-Economic Systems, Stanford University, Stanford, Calif.

OWEN, D. L. 1978. The Use of Influence Diagrams in Structuring Complex Decision Problems. In *Readings on the Principles and Applications of Decision Analysis, Vol. II,* R. A. Howard and J. E. Matheson (eds.). Strategic Decisions Group, Menlo Park, Calif., 765–772.

PHILLIPS, L. D. 1990. Discussion of From Influence to Relevance to Knowledge by R. A. Howard. In *Influence Diagrams, Belief Nets and Decision Analysis,* R. M. Oliver and J. Q. Smith (eds.). John Wiley, New York, 22–23.

REGE, A., and A. M. AGOGINO 1988. Topological Framework for Representing and Solving Probabilistic Inference Problems in Expert Systems. *IEEE Trans. Syst. Man and Cyber.* **18,** 402–414.

SHACHTER, R. D. 1986. Evaluating Influence Diagrams. *Opns. Res.* **34,** 871–882.

SHACHTER, R. D. 1988. Probabilistic Inference and Influence Diagrams. *Opns. Res.* **36,** 589–604.

WATSON, S. R., and D. M. BUEDE 1987. *Decision Synthesis.* Cambridge University Press, Cambridge, U.K.