

# SEQUENTIAL VALUATION NETWORKS: A NEW GRAPHICAL TECHNIQUE FOR ASYMMETRIC DECISION PROBLEMS

Riza Demirer and Prakash P. Shenoy

University of Kansas, School of Business  
1300 Sunnyside Ave, Summerfield Hall  
Lawrence, KS 66045-7585, USA  
{riza, pshenoy}@ku.edu

**Abstract.** This paper deals with representation and solution of asymmetric decision problems. We describe a new graphical representation called sequential valuation networks, which is a hybrid of Covaliu and Oliver's sequential decision diagrams and Shenoy's asymmetric valuation networks. Sequential valuation networks inherit many of the strengths of sequential decision diagrams and asymmetric valuation networks while overcoming many of their shortcomings. We illustrate our technique by representing and solving a modified version of Covaliu and Oliver's Reactor problem.

## 1 Introduction

The goal of this paper is to propose a new graphical technique for representing and solving asymmetric decision problems. The new graphical representation is called a *sequential valuation network* and it is a hybrid of sequential decision diagrams (SDDs) [3] and asymmetric valuation networks (VNs) [13]. Sequential valuation networks adapt the best features from SDDs and asymmetric VNs and provide a fix to some of the major shortcomings of these techniques as identified by Bielza and Shenoy [1]. The algorithm for solving sequential valuation networks is based on the idea of decomposing a large asymmetric problem into smaller symmetric sub-problems and then using a special case of Shenoy's fusion algorithm to solve the symmetric sub-problems.

In a decision tree representation, a path from the root node to a leaf node is called a *scenario*. A decision problem is said to be *asymmetric* if there exists a decision tree representation such that the number of scenarios is less than the cardinality of the Cartesian product of the state spaces of all chance and decision variables.

There are three types of asymmetry in decision problems—chance, decision, and information. First, the state space of a chance variable may vary depending on the scenario in which it appears. In the extreme, a chance variable may be non-existent in a particular scenario. For example, if a firm decides not to test market a product, we are not concerned about the possible results of test marketing. Second, the state space of a decision variable may depend on the scenario in which it appears. Again, at the

extreme, a decision variable may simply not exist for a given scenario. For example, if we decide not to buy a financial option contract, the decision of exercising the option on the exercise date does not exist. Finally, the information constraints may depend on the scenarios. For example, in diagnosing a disease with two symptoms, the order in which the symptoms are revealed (if at all) may depend on the sequence of the tests ordered by the physician prior to making a diagnosis. A specific example of information asymmetry is described in Section 5. Most of the examples of asymmetric decision problems have focused on chance and decision asymmetry. Information asymmetry has not been widely studied.

Several graphical techniques have been proposed for representing and solving asymmetric decision problems—traditional decision trees [11], combination of influence diagrams (IDs) and decision trees [2], contingent influence diagrams [5], influence diagrams with distribution trees [14], decision graphs within the ID framework [10], asymmetric valuation network representation with indicator valuations [13], sequential decision diagrams [3], configuration networks [7], asymmetric influence diagrams [9], and valuation networks with coarse valuations [8]. Each of these methods has some advantages and disadvantages. For a comparison of decision trees, Smith-Holtzman-Matheson's influence diagrams, Shenoy's asymmetric valuation networks, and Covaliu and Oliver's sequential decision diagrams, see [1].

Covaliu and Oliver's SDD representation [3] is a compact and intuitive way of representing the structure of an asymmetric decision problem. One can think of a SDD as a clustered decision tree in which each variable appears only once (as in influence diagrams and VNs). Also, SDDs model asymmetry without adding dummy states to variables. However, the SDD representation depends on influence diagrams to represent the probability and utility models. Also, preprocessing may be required in order to make the ID representation compatible with the SDD representation so that the formulation table can be constructed. One unresolved problem is that although a SDD and a compatible ID use the same variables, the state spaces of these variables may not be the same. The problem of exponential growth of rows in the formulation table is another major problem of this method. Finally, this method is unable to cope with an arbitrary factorization of the joint utility function. It can only handle either a single undecomposed utility function, or a factorization of the joint utility function into factors where each factor only includes a single variable.

Shenoy's asymmetric VN representation [13] is compact in the sense that the model is linear in the number of variables. It is also flexible regarding the factorization of the joint probability distribution of the random variables in the model—the model works for any multiplicative factorization of the joint probability distribution. However, this representation technique cannot avoid the creation of artificial states that lead to an increased state space for some variables in the model. Some types of asymmetry cannot be captured in the VN representation. Also, the asymmetric structure of a decision problem is not represented at the graphical level, but instead in the details of the indicator valuations.

This paper presents a new graphical representation called a *sequential valuation network* (SVN) that is a hybrid of SDDs and asymmetric VNs. This new graphical method combines the strengths of SDDs and VNs, and avoids the weaknesses of either. We use the graphical ease of SDD representation of the asymmetric structure

of a decision problem, and attach value and probability valuations to variables as in VNs. The resulting SVN representation is able to address many of the shortcomings of VNs and SDDs as follows. The state spaces of the variables do not include artificial states, and all types of asymmetry can be represented. This is true for the Reactor problem and we conjecture that these aspects are true of all asymmetric problems. Most of the asymmetric structure of a decision problem is represented at the graphical level. A SVN does not need a separate graph to represent the uncertainty model. No pre-processing is required to represent a decision problem as a SVN, i.e., it is not necessary to construct a formulation table prior to solving a SVN. Finally, a SVN can easily represent any factorization of the joint utility function.

To solve SVN, we identify different symmetric sub-problems as paths from the source node to the terminal node. Each such path represents a collection of scenarios. Finally, we apply a special case of Shenoy's [12] fusion algorithm for each sub-problem and solve the global asymmetric problem by solving smaller symmetric sub-problems. The strategy of breaking down an asymmetric decision problem into several symmetric sub-problems is also used by [7] and [9].

An outline of the remainder of the paper is as follows. In Section 2, we give a complete statement of a modified version of the Reactor problem of [3], and describe a decision tree representation of it. In Section 3, we represent the same problem using SVN representation and in Section 4, we sketch its solution. Finally, in Section 5, we conclude by summarizing some strengths of our representation as compared to the representations proposed so far.

## 2 The Reactor Problem

An electric utility firm must decide whether to build ( $D_2$ ) a reactor of advanced design ( $a$ ), a reactor of conventional design ( $c$ ), or no reactor ( $n$ ). If the reactor is successful, i.e., there are no accidents, an advanced reactor is more profitable, but it is also riskier. Past experience indicates that a conventional reactor ( $C$ ) has probability 0.980

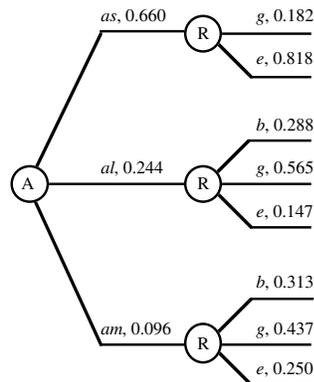


Fig. 1. A Probability Model for A and R in the Reactor Problem

of being successful ( $cs$ ), and a probability 0.020 of a failure ( $cf$ ). On the other hand, an advanced reactor ( $A$ ) has probability 0.660 of being successful ( $as$ ), probability 0.244 of a limited accident ( $al$ ), and probability 0.096 of a major accident ( $am$ ). If the firm builds a conventional reactor, the profits are \$8B if it is a success, and -\$4B if there is a failure. If the firm builds an advanced reactor, the profits are \$12B if it is a success, -\$6B if there is a limited accident, and -\$10B if there is a major accident. The firm's utility function is assumed to be linear in dollars.

Before making the decision to build, the firm has the option to conduct a test ( $D_1 = t$ ) or not



scenarios in this problem. The decision tree is shown using coalescence, i.e., repeating sub-trees are shown only once. With coalescence, the number of endpoints is reduced to 12. Notice that before we can complete the decision tree representation, we need to compute the required probabilities, i.e.  $P(R)$  and  $P(A|R)$ .

### 3 Sequential Valuation Network Representation

In this section, we define a new hybrid representation, which we call a *sequential valuation network*. First we start with some notation.

**Valuation Fragments.** Suppose  $\alpha$  is a utility valuation for  $h$ , i.e.,  $\alpha: \Omega_h \rightarrow \mathbb{R}$ , where  $\Omega_h$  denotes the state space of the variables in  $h$ , and  $\mathbb{R}$  denotes the set of real numbers. We shall refer to  $h$  as the *domain* of  $\alpha$ . Suppose  $g \subseteq h$ , and suppose  $\Gamma \subseteq \Omega_g$ . Then  $\alpha|_\Gamma$  is a function  $\alpha|_\Gamma: \Gamma \times \Omega_{h-g} \rightarrow \mathbb{R}$  such that  $(\alpha|_\Gamma)(x_g, x_{h-g}) = \alpha(x_g, x_{h-g})$  for all  $x_g \in \Gamma$ , and all  $x_{h-g} \in \Omega_{h-g}$ . We call  $\alpha|_\Gamma$  a *restriction* of  $\alpha$  to  $\Gamma$ . We will also refer to  $\alpha|_\Gamma$  as a *fragment* of  $\alpha$ . We will continue to regard the domain of  $\alpha|_\Gamma$  as  $h$ . Notice that  $\alpha|_{\Omega_g} = \alpha$ .

Often,  $\Gamma$  is a singleton subset of  $\Omega_g$ ,  $\Gamma = \{x_g\}$ . In this case, we write  $\alpha|_\Gamma$  as  $\alpha|x_g$ . For example, suppose  $\alpha$  is a valuation for  $\{A, B\}$  where  $\Omega_A = \{a_1, a_2\}$  and  $\Omega_B = \{b_1, b_2, b_3\}$ . Then,  $\alpha$  can be represented as a table as shown in the left hand side of Table 1. The restriction of  $\alpha$  to  $a_1$ ,  $\alpha|_{a_1}$ , is shown in the right hand side of Table 1. In practice, valuation fragments will be specified without specifying the full valuation. In the case of utility valuations, the unspecified values can be regarded as zero utilities (whenever the utility function decomposes additively), and in the case of probability valuations, the unspecified values can be regarded as zero probabilities.

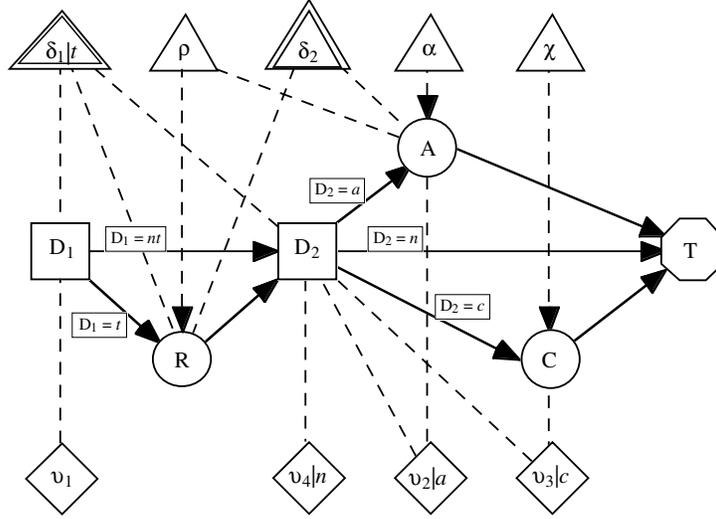
A complete SVN representation of the Reactor problem is given in Figure 3, Table 2, and Table 3. The SVN graph consists of six types of nodes—chance, decision, terminal, indicator, utility and probability. Chance nodes are shown as circles and represent random variables. In the Reactor problem representation, there are three chance nodes,  $R$ ,  $A$ , and  $C$ . Decision nodes are shown as rectangles and represent decision variables. In the Reactor problem representation, there are two decision nodes,  $D_1$  and  $D_2$ . The terminal node is shown as an octagon and is a compact version of the end points of a decision tree. The terminal node is labeled  $T$  in the Reactor

problem representation. Indicator valuations are shown as triangles with a double border, probability valuations are shown as triangles with a single border, and utility valuations are shown as diamonds. For further details, see [4].

The structure of the sub-graph is similar to the SDD graphical representation of [3]

**Table 1.** An Example of a Valuation Fragment

$\Omega_{\{A, B\}}$	$\alpha$	$\{a_1\} \times \Omega_B$	$\alpha _{a_1}$
$a_1, b_1$	$\alpha(a_1, b_1)$	$a_1, b_1$	$\alpha(a_1, b_1)$
$a_1, b_2$	$\alpha(a_1, b_2)$	$a_1, b_2$	$\alpha(a_1, b_2)$
$a_1, b_3$	$\alpha(a_1, b_3)$	$a_1, b_3$	$\alpha(a_1, b_3)$
$a_2, b_1$	$\alpha(a_2, b_1)$		
$a_2, b_2$	$\alpha(a_2, b_2)$		
$a_2, b_3$	$\alpha(a_2, b_3)$		



**Fig. 3.** A SVN Graphical Representation of the Reactor Problem

(with minor differences in the terminal node and the annotations associated with the directed edges) and the attached valuations have the same semantics as VNs [13].

In the qualitative part, we first define the state spaces of all chance and decision variables, and then specify the details of the indicator valuations. In the Reactor problem,  $\Omega_{D_1} = \{t, nt\}$ ,  $\Omega_R = \{b, g, e\}$ ,  $\Omega_{D_2} = \{a, c, n\}$ ,  $\Omega_A = \{as, al, am\}$ , and  $\Omega_C = \{cs, cf\}$ . The indicator valuation  $\delta_1|t$  with domain  $\{t\} \times \{R, D_2\}$  is a constraint on the choices available to the decision-maker at  $D_2$ . This constraint can be specified by listing all states in  $\{t\} \times \Omega_{\{R, D_2\}}$  that are allowed. Thus, the states that are allowed by  $\delta_1|t$  are  $\{(t, b, c), (t, b, n), (t, g, a), (t, g, c), (t, g, n), (t, e, a), (t, e, c), (t, e, n)\}$ . Similarly, the indicator valuation  $\delta_2$  with domain  $\{R, A\}$  can be regarded as a constraint on the state space  $\Omega_{\{R, A\}}$ .  $\delta_2$  rules out the state  $(b, as)$  that has zero probability. In this paper, we will regard an indicator valuation as a subset of the state space of its domain. For example,  $\delta_1|t \subset \{t\} \times \Omega_{\{R, D_2\}}$ , and  $\delta_2 \subset \Omega_{\{R, A\}}$ . During the solution phase, the computations in some sub-problems are done on the relevant state space (determined by the valuations that are being processed) constrained by the indicator valuations that are associated with the sub-problem.

In the quantitative part, we specify the numerical details of the probability and utility valuations as given in Tables 2 and 3. The numerical specifications have to be consistent with the graphical and qualitative specifications in the following senses. First, each valuation's domain is specified in the graphical part. For example, the domain of  $\chi$  is  $C$ . Therefore, we have to specify the values of  $\chi$  for each state in  $\Omega_C$ . Second, since the edge from  $\chi$  to  $C$  is directed, this means the probability valuation  $\chi$  is a conditional for  $C$  given the empty set, i.e., the marginal of  $\chi$  for the empty set is a vacuous probability valuation. Third, if we have probability or utility valuations specified on domains for which we have indicator valuations, then it is only necessary

**Table 2.** Utility Valuation Fragments in the Reactor Problem

$\Omega_{D_1}$	$v_1$	$\{a\} \times \Omega_A$	$v_2 a$
$nt$	0	$a, as$	12
$t$	-1	$a, al$	-6
		$a, am$	-10
$\{c\} \times \Omega_C$	$v_3 c$	$\{n\} \subset \Omega_{D_2}$	$v_4 n$
$c, cs$	8	$n$	0
$c, cf$	-4		

**Table 3.** Probability Valuation Fragments in the Reactor Problem

$\Omega_C$	$\chi$	$\Omega_A$	$\alpha$	$\delta_2$	$\rho$
$cs$	0.98	$as$	0.660	$b, al$	0.288
$cf$	0.02	$al$	0.244	$b, am$	0.313
		$am$	0.096	$g, as$	0.182
				$g, al$	0.565
				$g, a$	0.437
				$e, as$	0.818
				$e, al$	0.147
				$e, am$	0.250

$D_2$ ). Utility valuation  $v_4|n$  is only specified for  $D_2 = n$ . Notice that when  $D_2 = n$ , the next node in the SVN is the terminal node  $T$ . Therefore,  $v_4|n$  cannot include either  $A$  or  $C$  in its domain.

Utility valuations  $v_1$ ,  $v_2|a$ ,  $v_3|c$ , and  $v_4|n$  are additive factors of the joint utility function, and probability valuations  $\chi$ ,  $\alpha$ , and  $\delta$  are multiplicative factors of the joint probability distribution. In the Reactor problem, we have a factorization of the joint probability distribution into conditionals, i.e., a Bayes net model. But this is not a requirement of the sequential valuation network representation. As we will see in the next section, the SVN solution technique will work for any multiplicative factorization of the joint probability distribution.

## 4 Solving a SVN Representation

The main idea of the SVN solution method is to recursively decompose the problem into smaller sub-problems until the sub-problems are symmetric, then to

to specify the values of the valuations for the states permitted by the indicator valuations. For example, probability valuation  $\rho$  has domain  $\{R, A\}$ . Since we have indicator valuation  $\delta_2$  with the same domain, it is sufficient to specify the values of  $\rho$  for the states in  $\delta_2$ . Thus, we can regard  $\rho$  as a valuation fragment. Also, since the edge from  $\rho$  to  $R$  is directed, the values of  $\rho$  have to satisfy the condition  $\rho^{\downarrow A} = \iota_A$  where  $\iota_A$  is the vacuous probability valuation with domain  $\{A\}$ , i.e., a valuation whose values are identically one. Fourth, it is sufficient to specify values of utility or probability valuations for those states that are allowed by the annotations on the edges between variables. For example, consider the utility valuation fragment  $v_2|a$ . The domain of this valuation is  $\{D_2, A\}$ . However, the annotation on the edge from  $D_2$  to  $A$  tells us that all scenarios that include variable  $A$  have  $D_2 = a$ . Therefore, it is sufficient to specify  $v_2$  for all states in  $\{a\} \times \Omega_A$ . Similarly, it is sufficient to specify  $v_3|c$  for  $\{c\} \times \Omega_C$ , and sufficient to specify  $\delta_1|t$  for  $\{t\} \times \Omega_{\{R, D_2\}}$ .

Utility valuation  $v_4|n$  is only specified for  $D_2 = n$ . Notice that when  $D_2 = n$ , the next node in the SVN is the terminal node  $T$ . Therefore,  $v_4|n$  cannot include either  $A$  or  $C$  in its domain.

Utility valuations  $v_1$ ,  $v_2|a$ ,  $v_3|c$ , and  $v_4|n$  are additive factors of the joint utility function, and probability valuations  $\chi$ ,  $\alpha$ , and  $\delta$  are multiplicative factors of the joint probability distribution. In the Reactor problem, we have a factorization of the joint probability distribution into conditionals, i.e., a Bayes net model. But this is not a requirement of the sequential valuation network representation. As we will see in the next section, the SVN solution technique will work for any multiplicative factorization of the joint probability distribution.

solve the symmetric sub-problems, using a special case of the symmetric fusion algorithm [12]. Finally, the solutions to the sub-problems are recursively combined to obtain the solution to the original problem. We begin with some notation.

#### 4.1 Combination

Consider two utility valuations  $\psi_1$  for  $h_1$  and  $\psi_2$  for  $h_2$ . As defined in [12], we combine utility valuations using pointwise addition assuming an additive factorization of the joint utility function. In the SVN method, each sub-problem deals with valuation fragments that are relevant to the sub-problem. We start with defining combination of utility fragments.

**Case 1.** [*Combination of utility fragments*] Suppose  $g_1 \subseteq h_1$ , and  $g_2 \subseteq h_2$ , and consider two utility fragments  $\psi_1|_{\Gamma_1}$  and  $\psi_2|_{\Gamma_2}$  where  $\Gamma_1 \subseteq \Omega_{g_1}$ , and  $\Gamma_2 \subseteq \Omega_{g_2}$ . Let  $\Gamma$  denote  $((\Gamma_1 \times \Omega_{h_1 \cup h_2 - g_1}) \cup (\Gamma_2 \times \Omega_{h_1 \cup h_2 - g_2}))^{\downarrow g_1 \cup g_2}$ . The combination of  $\psi_1|_{\Gamma_1}$  and  $\psi_2|_{\Gamma_2}$ , written as  $(\psi_1|_{\Gamma_1}) \otimes (\psi_2|_{\Gamma_2})$ , is a utility valuation  $\psi$  for  $h_1 \cup h_2$  restricted to  $\Gamma$  given by

$$\begin{aligned} (\psi|_{\Gamma})(\mathbf{y}) &= (\psi_1|_{\Gamma_1})(\mathbf{y}^{\downarrow g_1}, \mathbf{y}^{\downarrow h_1 - g_1}) + (\psi_2|_{\Gamma_2})(\mathbf{y}^{\downarrow g_2}, \mathbf{y}^{\downarrow h_2 - g_2}) \quad \text{if } \mathbf{y}^{\downarrow g_1} \in \Gamma_1 \quad \text{and} \\ &\quad \mathbf{y}^{\downarrow g_2} \in \Gamma_2 \\ &= (\psi_1|_{\Gamma_1})(\mathbf{y}^{\downarrow g_1}, \mathbf{y}^{\downarrow h_1 - g_1}) \quad \text{if } \mathbf{y}^{\downarrow g_1} \in \Gamma_1 \text{ and } \mathbf{y}^{\downarrow g_2} \notin \Gamma_2 \\ &= (\psi_2|_{\Gamma_2})(\mathbf{y}^{\downarrow g_2}, \mathbf{y}^{\downarrow h_2 - g_2}) \quad \text{if } \mathbf{y}^{\downarrow g_1} \notin \Gamma_1 \text{ and } \mathbf{y}^{\downarrow g_2} \in \Gamma_2 \end{aligned}$$

for all  $\mathbf{y} \in \Gamma \times \Omega_{(h_1 \cup h_2) - (g_1 \cup g_2)}$ .

**Case 2.** [*Combination of a utility fragment and a probability fragment*] Suppose  $g_1 \subseteq h_1$ , and  $g_2 \subseteq h_2$ , and consider utility fragment  $\psi_1|_{\Gamma_1}$  and probability fragment  $\psi_2|_{\Gamma_2}$  where  $\Gamma_1 \subseteq \Omega_{g_1}$ , and  $\Gamma_2 \subseteq \Omega_{g_2}$ . Let  $\Gamma$  denote  $((\Gamma_1 \times \Omega_{h_1 \cup h_2 - g_1}) \cap (\Gamma_2 \times \Omega_{h_1 \cup h_2 - g_2}))^{\downarrow g_1 \cup g_2}$ . The combination of  $\psi_1|_{\Gamma_1}$  and  $\psi_2|_{\Gamma_2}$ , written as  $(\psi_1|_{\Gamma_1}) \otimes (\psi_2|_{\Gamma_2})$ , is a utility valuation  $\psi$  for  $h_1 \cup h_2$  restricted to  $\Gamma$  given by:

$$(\psi|_{\Gamma})(\mathbf{y}) = (\psi_1|_{\Gamma_1})(\mathbf{y}^{\downarrow g_1}, \mathbf{y}^{\downarrow h_1 - g_1}) (\psi_2|_{\Gamma_2})(\mathbf{y}^{\downarrow g_2}, \mathbf{y}^{\downarrow h_2 - g_2}) \quad \text{if } \mathbf{y}^{\downarrow g_1} \in \Gamma_1 \text{ and } \mathbf{y}^{\downarrow g_2} \in \Gamma_2$$

and 0 otherwise, for all  $\mathbf{y} \in \Gamma \times \Omega_{(h_1 \cup h_2) - (g_1 \cup g_2)}$ .

**Case 3.** [*Combination of probability fragments*] Suppose  $g_1 \subseteq h_1$ , and  $g_2 \subseteq h_2$ , and consider probability fragments  $\psi_1|_{\Gamma_1}$  and  $\psi_2|_{\Gamma_2}$  for  $h_1$  and  $h_2$ , respectively, where  $\Gamma_1 \subseteq \Omega_{g_1}$ , and  $\Gamma_2 \subseteq \Omega_{g_2}$ . Let  $\Gamma$  denote  $((\Gamma_1 \times \Omega_{h_1 \cup h_2 - g_1}) \cap (\Gamma_2 \times \Omega_{h_1 \cup h_2 - g_2}))^{\downarrow g_1 \cup g_2}$ . The combination of  $\psi_1|_{\Gamma_1}$  and  $\psi_2|_{\Gamma_2}$ , written as  $(\psi_1|_{\Gamma_1}) \otimes (\psi_2|_{\Gamma_2})$ , is a probability valuation  $\psi$  for  $h_1 \cup h_2$  restricted to  $\Gamma$  given by

$$(\psi|_{\Gamma})(\mathbf{y}) = (\psi_1|_{\Gamma_1})(\mathbf{y}^{\downarrow g_1}, \mathbf{y}^{\downarrow h_1 - g_1}) (\psi_2|_{\Gamma_2})(\mathbf{y}^{\downarrow g_2}, \mathbf{y}^{\downarrow h_2 - g_2}) \quad \text{if } \mathbf{y}^{\downarrow g_1} \in \Gamma_1 \text{ and } \mathbf{y}^{\downarrow g_2} \in \Gamma_2$$

and 0 otherwise for all  $\mathbf{y} \in \Gamma \times \Omega_{(h_1 \cup h_2) - (g_1 \cup g_2)}$ . The reactor problem described in this paper does not require this case of combination.

Note that, the combination of two utility valuations is a utility valuation; the combination of two probability valuations is a probability valuation; and the combination of a utility and a probability valuation is a utility valuation.

As for the marginalization and division operations, the SVN method uses the same marginalization and division operations as defined in [12]. For further details, see [4].

## 4.2 Tagging

The recursive algorithm of solving lower level sub-problems and sending the results to an upper level sub-problem requires the use of a concept that we call *tagging*. Suppose  $\psi$  is a utility valuation with domain  $h$ , and suppose  $X \notin h$ . Tagging  $\psi$  by  $X = x$  is denoted by  $\psi \otimes (\iota_X|x)$ , where  $\iota_X|x$  is the vacuous utility valuation with domain  $\{X\}$  restricted to  $X = x$ . A vacuous utility valuation is a valuation that is identically zero. This operation basically extends the domain of  $\psi$  from  $h$  to  $h \cup \{X\}$  without changing the values of  $\psi$ .

## 4.3 The Fusion Algorithm

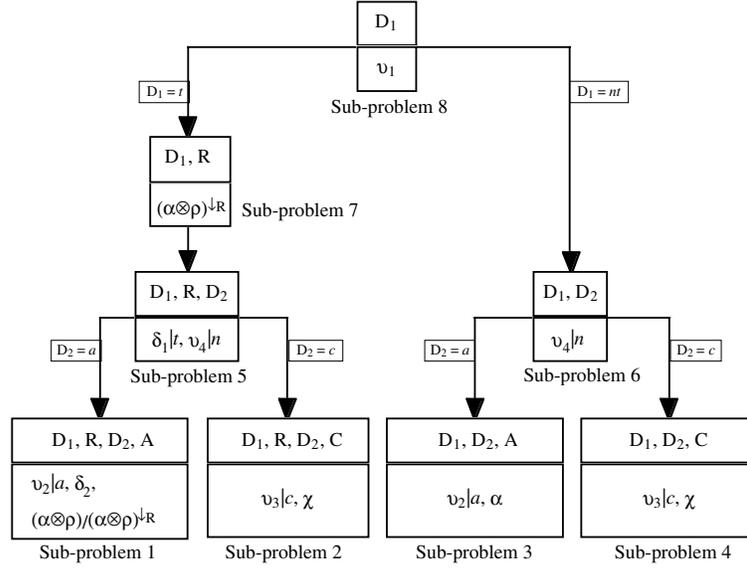
The details of the fusion algorithm are given in [12]. In the context of sequential valuation networks, the fusion algorithm is the same as rollback in decision trees. Fusion with respect to decision variables is similar to the “folding back” operation in decision trees [11] and fusion with respect to chance variables is similar to the “averaging out” operation in decision trees [11]. Further details of the fusion algorithm for sequential valuation networks are found in [4].

## 4.4 Decomposition of the Problem

Starting from the SVN graphical representation, we decompose the decision problem into symmetric sub-problems. The symmetric sub-problems are identified by enumerating all distinct directed paths and sub-paths from the source node to the terminal node in the SVN graphical representation.

**Variables.** We start with the root node, say  $S$ . Next we identify all directed arcs in the SVN that lead out of the source node  $S$ . For each directed arc, say to variable  $X$ , we create a new sub-problem consisting of variables  $S$  and  $X$  on the path from the source node to variable  $X$ . We retain the annotation on the edges. We recursively proceed in this manner until all paths and sub-paths have been enumerated. Notice that the terminal node is not a variable and we do not include it in any sub-problem. The resulting directed tree is called a “decomposition tree.” Figure 4 shows the decomposition tree that is constructed for the reactor problem.

**Utility and Indicator Valuations.** We start at the root node, say  $S$ , of the decomposition tree with the set of all utility valuation fragments included in the SVN representation. All valuation fragments whose domains are included in the set of variables associated with the sub-problem are associated with this sub-problem. The valuations that are not are passed on to the child sub-problems suitably decomposed



**Fig. 4.** The Decomposition Tree for the Reactor Problem

as per the annotation on the edges leading to the child sub-problems. This is recursively repeated.

In the Reactor problem, we start with utility and indicator valuations  $v_1$ ,  $v_2|a$ ,  $v_3|c$ ,  $v_4|n$ ,  $\delta_1|t$ , and  $\delta_2$ . Valuation  $v_1$  with domain  $\{D_1\}$  is associated with sub-problem 8. Of the remaining valuations, only  $\delta_1|t$  has  $D_1$  in its domain. Since there is no fragment of  $\delta_1|t$  that has  $D_1 = nt$ , Sub-problem 7 receives valuations  $v_2|a$ ,  $v_3|c$ ,  $v_4|n$ ,  $\delta_1|t$ , and  $\delta_2$ . Sub-problem 6 receives valuations  $v_2|a$ ,  $v_3|c$ ,  $v_4|n$ , and  $\delta_2$ .

This process of associating utility and indicator valuations with sub-problems continues recursively as above. The resulting distribution of utility and indicator valuations in the sub-problems is shown in Figure 4.

**Probability Valuations.** We start by assuming that we have a factorization of the joint probability distribution for all chance variables in the problem. In the reactor problem, for example, the joint probability distribution  $\tau$  for  $\{C, A, R\}$  is given by  $\tau = \chi \otimes \alpha \otimes \rho$ .

We recursively compute the probability valuation associated with a leaf sub-problem that ends with a chance variable, say  $C_m$ , as follows. Let  $\Gamma = \{C_1, \dots, C_m\}$  denote the chance variables on a path from the source node to the leaf node whose last variable is  $C_m$ , and let  $P = \{\pi_1, \dots, \pi_k\}$  denote the set of probability potentials with domains  $h_1, \dots, h_k$ , respectively, such that  $(\pi_1 \otimes \dots \otimes \pi_k)^{\downarrow \Gamma}$  is the joint distribution for the chance variables in  $\Gamma$ . The probability valuation associated with the leaf sub-problem whose last variable is  $C_m$  is given by  $\pi^{\downarrow \Gamma} / \pi^{\downarrow \Gamma - \{C_m\}}$ , where  $\pi = \otimes \{\pi_j | C_m \in h_j\}$ . Furthermore, the set of probability valuations associated with the set of chance variables  $\Gamma - \{C_m\}$  is  $\cup \{\pi_j | C_m \in h_j\} \cup \{\pi^{\downarrow \Gamma - \{C_m\}}\}$ , i.e.,

$(\otimes\{\pi_j | C_m \notin h_j\} \otimes \pi^{\downarrow \Gamma - \{C_m\}})^{\downarrow \Gamma - \{C_m\}}$  is the joint distribution for the chance variables in  $\Gamma - \{C_m\}$ . Thus, we can recursively compute the probability valuations associated with the other sub-problems whose last variable is a chance node. It follows from Lauritzen and Spiegelhalter [1988] that  $\pi^{\downarrow \Gamma} / \pi^{\downarrow \Gamma - \{C_m\}}$  is the conditional probability distribution for  $C_m$  given the variables in  $\Gamma - \{C_m\}$ . For further details on how the sub-problems are populated with indicator, utility, and probability valuations, see [4].

#### 4.5 Solving the Sub-Problems

We start with solving the leaf sub-problems. After solving a sub-problem (as per the definition of fusion stated in [12]), we pass the resulting utility valuation fragment to its parent sub-problem and delete the sub-problem. In passing the utility valuation fragment to the parent sub-problem, if the domain of the utility valuation fragment does not include any variables in the parent sub-problem, we tag the utility valuation with the value of the last variable in the parent sub-problem that is in the annotation. We recursively continue this procedure until all sub-problems are solved.

Consider the decomposition of the Reactor problem into the eight sub-problems as shown in Figure 4. Consider Sub-problem 1 consisting of valuation fragments  $v_2|a$ ,  $(\rho \otimes \alpha) / (\rho \otimes \alpha)^{\downarrow R}$ , and  $\delta_2$ . We fuse the valuation fragments with respect to A using the definition of fusion from [12].

$$\text{Fus}_A\{v_2|a, (\alpha \otimes \rho) / (\alpha \otimes \rho)^{\downarrow R}\} = \{[v_2|a \otimes (\alpha \otimes \rho) / (\alpha \otimes \rho)^{\downarrow R}]^{-A}\} = \{v_5|a\}.$$

The resulting utility valuation  $v_5|a$  is sent to parent Sub-problem 5. Since  $v_5|a$  includes  $D_2$  in its domain, there is no need for tagging. All computations are done on relevant state spaces as constrained by indicator valuation  $\delta_2$ . The details of the computation are shown in Table 4. The solutions to the remainder of the sub-problems are given in [4].

## 5 Summary and Conclusions

The main goal of this paper is to propose a new representation and solution technique for asymmetric decision problems.

The advantages of SVNs over SDDs are as follows. SVNs do not require a separate influence diagram to represent the uncertainty model. SVNs can represent a more general uncertainty model than SDDs, which like influence diagrams assume a Bayes net model of uncertainties. All asymmetries can be represented in SVNs. This is not true for SDDs. For example, in the Reactor problem, the impossibility of  $R = b$  when  $A = as$  is not represented in a SDD representation of the problem. SVNs do not require a separate formulation table representation as in SDDs. Finally, SVNs can handle any factorization of the joint utility function whereas SDDs as currently described can only be used with either an undecomposed joint utility function or with a factorization of the joint utility function into singleton factors.

**Table 4.** The Details of Solving Sub-problem 1

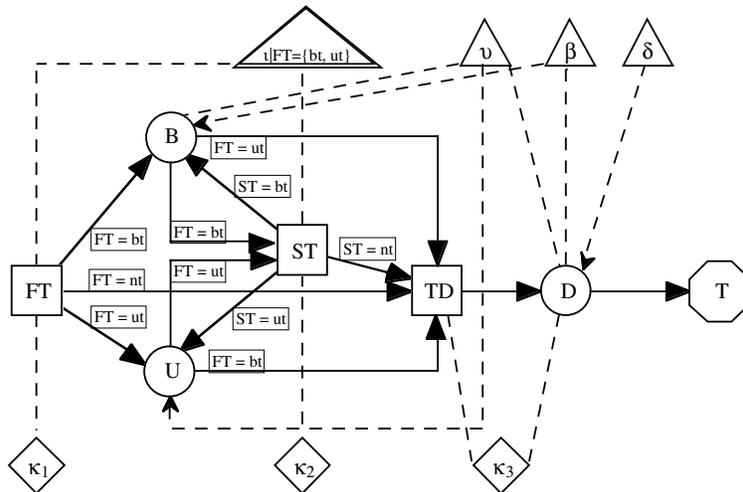
$\{a\} \times \delta_2$	$v_2 a$	$(\alpha \otimes \rho) / (\alpha \otimes \rho)^{\downarrow R}$	$v_2 a \otimes (\alpha \otimes \rho) / (\alpha \otimes \rho)^{\downarrow R} = \varphi$	$\varphi^{-A} = v_5 a$
$a, b, al$	-6	0.700	-4.200	-7.200
$a, b, am$	-10	0.300	-3.000	
$a, g, as$	12	0.400	4.800	0.649
$a, g, al$	-6	0.460	-2.760	
$a, g, am$	-10	0.140	-1.400	
$a, e, as$	12	0.900	10.800	10.043
$a, e, al$	-6	0.060	-0.360	
$a, e, am$	-10	0.040	-0.400	

The advantages of SVNs over VNs are as follows. SVNs represent most of the asymmetry at the graphical level (some asymmetry is represented in the details of the indicator valuations) whereas in the case of VNs, all asymmetry is represented in the details of the indicator valuations. The state spaces of chance and decision nodes in SVNs do not include dummy states. All types of asymmetry can be represented in SVNs whereas VNs cannot represent some types of asymmetry. Finally, the modeling of probability distributions in SVNs is as intuitive as in influence diagrams (assuming we are given a Bayes net model for the joint probability distribution).

One main advantage of the SVN technique is that we do not need to introduce dummy states for chance or decision variables. To see why this is important, we will describe a simple example called Diabetes diagnosis. Consider a physician who is trying to diagnose whether or not a patient is suffering from Diabetes. Diabetes has two symptoms, glucose in urine, and glucose in blood. Assume we have a Bayes net model for the three variables—Diabetes ( $D$ ), glucose in blood ( $B$ ) and glucose in urine ( $U$ )—in which the joint distribution for the three variables  $P(D, B, U)$  factors into three conditionals,  $P(D)$ ,  $P(B | D)$ , and  $P(U | D, B)$ . Furthermore, assume that  $D$  has two states,  $d$  for Diabetes is present, and  $\sim d$  for Diabetes is absent,  $U$  has two states,  $u$  for elevated glucose levels in urine, and  $\sim u$  for normal glucose level in urine, and  $B$  has two states,  $b$  for elevated glucose levels in blood, and  $\sim b$  for normal glucose level in blood. The physician first decides,  $FT$  (first test), whether to order a urine test ( $ut$ ) or a blood test ( $bt$ ) or no test ( $nt$ ). After the physician has made this decision and observed the results (if any), she next has to decide whether or not to order a second test ( $ST$ ). The choices available for the second test decision depend on the decision made at  $FT$ . If  $FT = bt$ , then the choices for  $ST$  are either  $ut$  or  $nt$ . If  $FT = ut$ , then the choices for  $ST$  are either  $bt$  or  $nt$ . Finally, after the physician has observed the results of the second test (if any), she then has to decide whether to treat the patient for Diabetes or not. As described so far, the problem has three chance variables,  $D$ ,  $U$ ,  $B$ , and three decision variables  $FT$  (first test),  $ST$  (second test), and  $TD$  (treat for Diabetes). Using the SVN technique, one can represent this problem easily without introducing any more variables or any dummy states. A SVN graphical representation

is shown in Figure 5. In this figure, the indicator valuation fragment  $\iota|_{FT = \{bt, ut\}}$  represents a constraint on  $ST$  as described above, the utility valuations  $\kappa_1$ ,  $\kappa_2$ , and  $\kappa_3$  represents a factorization of the total cost of diagnosing and treating the patient for Diabetes, and the probability valuations  $\delta = P(D)$ ,  $\beta = P(B | D)$ , and  $\nu = P(U | B, D)$  represent a factorization of the joint probability distribution into conditionals specified by the Bayes net model. Notice that the SVN graphical representation has several directed cycles. However, these directed cycles are disallowed by the annotations on the directed edges and the indicator valuation  $\iota$ , which forbids, e.g.,  $FT = bt$ ,  $ST = bt$ , and also  $FT = ut$ ,  $ST = ut$ .

Representing this problem using Smith-Holtzman-Matheson's asymmetric influence diagrams [14] or Shenoy's asymmetric valuation networks [13] is possible but only after either introducing additional variables or introducing dummy states for the existing variables. This is because if one uses the existing variables, the modeling of information constraints would depend on the  $FT$  decision. If  $FT = bt$ , then the true state of  $B$  is revealed prior to making the  $ST$  decision, and the true state of  $U$  is unknown when the  $ST$  decision is made. However if  $FT = ut$ , then the true state of  $U$  is known prior to making the  $ST$  decision and the true state of  $B$  is unknown when the  $ST$  decision is made. We call this aspect of the decision problem *information asymmetry*. Using either traditional influence diagrams or valuation networks, it is not possible to model this information asymmetry without either introducing additional variables or introducing dummy states for existing variables. In either of these cases, the modeling will need to adapt the Bayes net to a model that includes additional variables or dummy states or both. We leave the details of representing the Diabetes diagnosis problem using either influence diagrams or valuation networks or some other technique to the ingenuity of the reader.



**Fig. 5.** A SVN Representation of the Diabetes Diagnosis Problem

## Acknowledgements

The paper has benefited from comments and discussions with Concha Bielza, Finn Jensen, Thomas Nielsen, Zvi Covaliu, Liping Liu, and Kelli Wikoff.

## References

1. Bielza, C. and P. P. Shenoy (1999), "A comparison of graphical techniques for asymmetric decision problems," *Management Science*, **45**(11), 1552–1569.
2. Call, H. J. and W. A. Miller (1990), "A comparison of approaches and implementations for automating decision analysis," *Reliability Engineering and System Safety*, **30**, 115–162.
3. Covaliu, Z. and R. M. Oliver (1995), "Representation and solution of decision problems using sequential decision diagrams," *Management Science*, **41**(12), 1860–1881.
4. Demirer, R. and P. P. Shenoy (2001), "Sequential Valuation Networks for Asymmetric Decision Problems," Working Paper No. 286, University of Kansas School of Business, Lawrence, KS. Available by anonymous ftp from <ftp://ftp.bs.school.ku.edu/home/pshenoy/wp286.pdf>
5. Fung, R. M. and R. D. Shachter (1990), "Contingent influence diagrams," Working Paper Department of Engineering-Economic Systems, Stanford University, Stanford, CA.
6. Lauritzen, S. L. and D. J. Spiegelhalter (1988), "Local computations with probabilities on graphical structures and their application to expert systems" (with discussion), *Journal of Royal Statistical Society, Series B*, **50**(2), 157–224.
7. Liu, L. and P. P. Shenoy (1995), "A decomposition method for asymmetric decision problems," in *Proceedings of the 1995 Decision Sciences Institute Annual Meeting*, **2**, 589–591, Boston, MA.
8. Liu, L. and P. P. Shenoy (2000), "Representing asymmetric decision problems using coarse valuations," Working Paper No. 287, University of Kansas School of Business, Summerfield Hall, Lawrence, KS.
9. Nielsen, T. D. and F. V. Jensen (2000), "Representing and solving asymmetric decision problems," in C. Boutilier and M. Goldszmidt (eds.), *Uncertainty in Artificial Intelligence: Proceedings of the Sixteenth Conference*, 416–425, Morgan Kaufmann, San Francisco, CA.
10. Qi, R., L. Zhang and D. Poole (1994), "Solving asymmetric decision problems with influence diagrams," in R. L. Mantaras and D. Poole (eds.), *Uncertainty in Artificial Intelligence: Proceedings of the Tenth Conference*, 491–497, Morgan Kaufmann, San Francisco, CA.
11. Raiffa, H. (1968), *Decision Analysis: Introductory Lectures on Choices under Uncertainty*, Addison-Wesley, Reading, MA.
12. Shenoy, P. P. (1992), "Valuation-based systems for Bayesian decision analysis," *Operations Research*, **40**(3), 463–484.
13. Shenoy, P. P. (2000), "Valuation network representation and solution of asymmetric decision problems," *European Journal of Operational Research*, **121**(3), 2000, 579–608.
14. Smith, J. E., S. Holtzman and J. E. Matheson (1993), "Structuring conditional relationships in influence diagrams," *Operations Research*, **41**(2), 280–297.