A Knowledge-Based Model Construction approach to Medical Decision Making

Liem Ngo, M.S., Peter Haddawy, Ph.D.
{Liem, Haddawy}@cs.uwm.edu
Department of Electrical Engineering and Computer Science
University of Wisconsin-Milwaukee, Milwaukee, Wisconsin 53201

We present a framework for representing the probabilistic effects of actions and contingent treatment plans. Our language has a well-defined declarative semantics and we have developed an implemented algorithm (named BNG) that generates Bayesian networks (BN) to compute the posterior probabilities of queries. In this paper we address the problem of projecting a contingent treatment plan by automatically constructing a structure of interrelated BNs, which we call a BN-graph, and applying the available propagation procedures on it. To address the optimal plan generation, we base our approach on the observation that normally the target plan space has a well-defined structure. We provide a language to describe plan spaces which resembles a programming language with loops and conditionals. We briefly present the procedures for finding the optimal plan(s) from such specified plan spaces.

INTRODUCTION

For accurate medical diagnosis, prediction and decision making, it is often necessary to model a patient's condition over time. Because there is great uncertainty in clinical medicine, a system for diagnostic or prognostic evaluation must be able to represent and reason with uncertainty. Bayesian networks (BN) are currently the most powerful and popular method for representing and reasoning with probabilistic information. A Bayesian network is a directed acyclic graph in which the nodes represent random variables and the links represent direct influences. The influences are quantified with conditional probabilities in the form of link matrices associated with each node. A link matrix specifies the probabilities of all possible values of a node given all possible combinations of values of its parents.

Although BNs provide a relatively efficient method for representing and reasoning with probabilistic information, the process of computing posterior probabilities (inference) in BNs remains NP-hard [5]. This complexity becomes particularly problematic in large models such as those that arise in modeling temporal processes. We can greatly reduce the size of the network models if we can identify some deterministic information and use it as context to index the probabilistic information.

We propose [1] representing a class of BNs with a knowledge base of probabilistic rules augmented with context constraints. A context constraint is a logical expression that determines the applicability of a probabilistic relation based on some deterministic knowledge. In [1], we provide a declarative semantics for our language and an implemented algorithm (BNG) that generates a temporal BN to compute the posterior probabilities of a query when a set of context information, and a set of evidence are given.

In this paper, we extend the previous work to address the evaluation of medical contingent plans and the generation of optimal plans. In a contingent plan, some actions are performed only when certain conditions occur. It is well-known that, because of the uncertain nature of the outcomes of treatments or tests, medical treatment plans are usually contingent. We represent a contingent plan (CP) as a program written in a simple programming language with conditional and iterative control structures and actions as primitive statements. Our approach assumes that the description of action effects and domain relationship is stored in a knowledge base of context-sensitive probabilistic rules. We evaluate a CP with respect to a knowledge base by constructing a structure of interrelated BNs, which we call a BN-graph, and applying the available probability propagation procedures on it.

The optimal plan generation problem is usually very hard, taking into account the nondeterministic nature of action outcomes and of the uncertain environment. To alleviate that problem, we base our approach on two practical observations. First, plans have structure so that some sequences of actions may be "unreasonable" plans. For example, tests usually preceed treatment. Second, heuristics are usually utilized by decision makers to eliminate the portions of a plan space which

0195—4210/96/$5.00

contain obviously optimal plans. In our approach, the decision maker specifies a "target" plan space by a *plan scheme*. A plan scheme, similar to a CP, is described by a program. The only difference is each primitive statement in a plan scheme is a set of alternative CPs. Each plan scheme characterizes a set of CPs. Our proposal contains the procedures to find the optimal CPs amongst the CPs characterized by a plan scheme.

Unlike decision tree methods where decision makers have to be concerned with branching probabilities, our approach assumes probabilistic knowledge is prespecified in a KB. To find the optimal plans, they need only to concentrate on the logic of the plans, which are specified by plan schemes and the computation of probabilities is performed automatically.

ACUTE DEEP VENOUS THROMBOSIS

To evaluate the applicability of our approach to medical decision making, we construct a model for diagnosis and treatment of acute deep venous thrombosis (DVT) of the lower extremities. Appropriate management of patients with suspected DVT remains an important and complex clinical problem. The clinical findings of DVT do not permit diagnosis with certainty [8, 10]. Unchecked, lower-extremity DVT can progress to pulmonary embolism (PE), a condition that entails significant morbidity and mortality. Anticoagulation therapy for DVT is expensive and carries the risk of severe hemorrhage. Even diagnostic procedures such as venography entail risks such as contrast reaction and iatrogenic DVT. The objective of applying decision analysis to this problem is to determine the optimal strategies for testing and treating a patient suspected of having DVT.

Our model is based on data from an article that compared 24 different management strategies [6]. We choose this article because it contains explicit probability and cost data. The test procedures include contrast venography (Veno) and two noninvasive tests: impedance plethysmography (IPG) and real-time ultrasonography (RUS). Treatment, which consisted solely of anticoagulation therapy, include unconditional actions (e.g., Treat all) and conditional actions (e.g., Treat if thigh DVT seen on venography). Although the term "thigh DVT" is not defined in the reference model, we take it to mean thrombosis of the superficial femoral vein, deep femoral vein, and/or common femoral vein.

REPRESENTATION

To represent a planning problem, we must represent the state of the world and the actions available to the planning agent. We used timed predicates to represent both. Time is discrete and represented by nonnegative integers. For example, the predicate $treat(.,.)$ represents the action class "treat". The first argument of an action predicate always represents time. The ground atom $treat(5, john)$ stands for the action "treat, at time 5, a patient named John" and is an *instance* of action *treat*. We describe the state of the world with a set of random variables, which we represent as predicates. For example, the random variable *bleeding* can be represented by a three-position predicate $bleeding(T, X, V)$, where T is the time point, X is the person and V is the value *minor*, *major* or *none*.

We represent the DVT domain with six action predicates: $ipg(T, X)$, $rus(T, X)$, $veno(T, X)$, $Wait7d(T, X)$, $Treat(T, X)$, and $noTreat(T, X)$; and seven random variable predicates. *ipg*, *rus* and *veno* are tests. *Wait7d* is the action 'wait 7 days'. Two random variable predicates represent the outcomes of the tests, while the remainder represent the status of the patient. The predicate $nitResult(T, X, V)$ represents the outcome of either non-invasive test and has values $+$ and $-$. The predicate $venoResult(T, X, V)$, represents the outcome of the venography test and has values 1 (no calf DVT, no thigh DVT), 2 (calf DVT, no thigh DVT), 3 (no calf DVT, thigh DVT), and 4 (calf DVT, thigh DVT). The predicates $calf(T, X, V)$, $thighDVT(T, X, V)$, $dead(T, X, V)$, $pe(T, X, V)$, and $bleeding(T, X, V)$ all have values *yes* and *no*.

We describe action effects and relations among random variables with probabilistic sentences. A **probabilistic sentence (p-sentence)** in our language has the form $P(A_0|A_1, \ldots, A_n) = \alpha \leftarrow B_1, \ldots, B_m, \neg C_1, \ldots, \neg C_k$, where the A_i, B_j and C_r are atoms and α is a number in the $[0, 1]$ interval. The meaning of such a sentence is "in the context that B_j are true, and none of C_k is shown to be true, $P(A_0|A_1, \ldots, A_n) = \alpha$".

We assume that each action takes one unit of time to finish. Figure 1 shows some of the p-sentences modelling the effects of the six possible actions. Notice that every test and treatment includes the condition that the patient is not dead and does not have PE. IPG and RUS only test for thigh DVT and differ only in their specificity. The description of venography includes two sentences describing its possible side-effects. The probability of a fatal contrast reaction is .0001 and the probability of venography-induced thigh DVT is .01. The action *Wait7d* models the endogenous change of the status of the patient.

CONTINGENT PLANS

We represent a CP using a programming language in which the primitive statements are actions and there are only two control structures: sequential and conditional (by using the CASE construct).

$P(nitResult(T+1,X,+)|$ $thighDVT(T,X,+), dead(T,X,no), pe(T,X,no)) = .95 \leftarrow ipg(T,X);$
$P(nitResult(T+1,X,-)|$ $thighDVT(T,X,-), dead(T,X,no), pe(T,X,no)) = .90 \leftarrow ipg(T,X);$

$P(venoResult(T+1,X,1)|$ $thighDVT(T,X,+), calfDVT(T,X,+), dead(T,X,no), pe(T,X,no)) = 0.002 \leftarrow veno(T,X)$
$P(venoResult(T+1,X,2)|$ $thighDVT(T,X,+), calfDVT(T,X,+), dead(T,X,no), pe(T,X,no)) = 0.018 \leftarrow veno(T,X)$
$P(venoResult(T+1,X,3)|$ $thighDVT(T,X,+), calfDVT(T,X,+), dead(T,X,no), pe(T,X,no)) = 0.098 \leftarrow veno(T,X)$
$P(venoResult(T+1,X,4)|$ $thighDVT(T,X,+), calfDVT(T,X,+), dead(T,X,no), pe(T,X,no)) = 0.882 \leftarrow veno(T,X)$
$P(dead(T+1,X,yes)|$ $dead(T,X,no), pe(T,X,no)) = .0001 \leftarrow veno(T,X)$
$P(thighDVT(T+1,X,yes)|$ $thighDVT(T,X,no), dead(T,X,no), pe(T,X,no)) = .01 \leftarrow veno(T,X)$

$P(bleeding(T+1,X,major)|$ $dead(T,X,no), pe(T,X,no)) = .05 \leftarrow treat(T,X)$
$P(bleeding(T+1,X,minor)|$ $dead(T,X,no), pe(T,X,no)) = .05 \leftarrow treat(T,X)$
$P(bleeding(T+1,X,none)|$ $dead(T,X,no), pe(T,X,no)) = .9 \leftarrow treat(T,X)$
$P(dead(T+1,X,yes)|$ $bleeding(T+1,X,major), dead(T,X,no), pe(T,X,no)) = .05 \leftarrow treat(T,X)$

$P(thighDVT(T+1,X,+)|$ $calfDVT(T,X,+), thigh(T,X,-), dead(T,X,no), pe(T,X,no)) = .20 \leftarrow wait7d(T,X)$
$P(pe(T+1,X,yes)|$ $calfDVT(T,X,-), thigh(T,X,+), dead(T,X,no), pe(T,X,no)) = .25 \leftarrow wait7d(T,X)$

Figure 1: A portion of the action model for DVT domain.

The conditions of CASE can refer to the values of random variables in previous time slices. We always assume that in a CASE construct the different branching conditions are mutually exclusive.

In the following CP, the performance of *treat* is contingent on the result of the previous *veno* test.

veno;
CASE *result IS*
\+ : *treat*;
ENDCASE;

In a CP, we call a sequence of consecutive generic actions in the plan which does not contain the CASE or ENDCASE keywords a (sequential) plan fragment. A CP can be represented by a graph of its maximal plan fragments as shown in Figure 2. In the figure, the names starting with F denote maximal plan fragments. In the graph representation, each horizontal bar represents a maximal plan fragment and is annotated with the corresponding name of the fragment. The lines connecting the horizontal bars represent the diverging (corresponding to the CASE keyword) and converging (corresponding to the ENDCASE keyword) links. The diverging links are annotated with the corresponding conditions in the program.

The Goal of Plan Projection

We are interested in evaluating the probability of some random variables, which are called *goal random variables*, at the end of the performance of a CP. Notice that such a plan has several branching possibilities, each with a specific time length and a specific probability of occurrence. For example, one branch of the CP in Figure 2 is $\langle F0, F1, F11, F6\rangle$. Suppose the CP P has n possible branches $\mathcal{F}_i, i = 1, \ldots, n$, the probability of occurrence of branch $\mathcal{F}_i$ is $Pr(\mathcal{F}_i)$, the length (duration) of branch $\mathcal{F}_i$ is n_i and we want to evaluate the probability that an (atemporal) random variable X achieving the value x. The desired probability is given by the following formula:

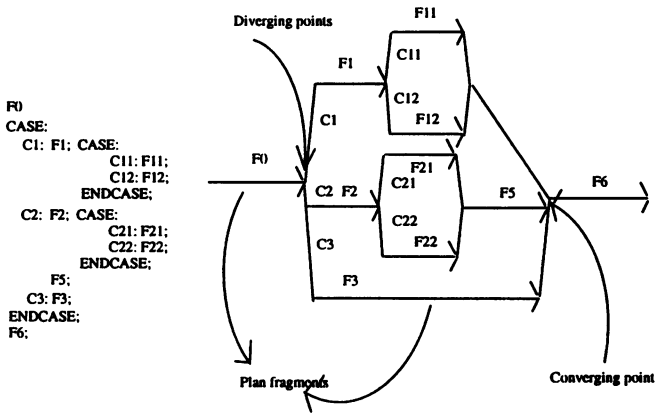

Figure 2: The graph model of a contingent plan.

$$Pr(X = x|P) = \sum_{i=1}^{n} (Pr(A_i|\mathcal{F}_i) \times Pr(\mathcal{F}_i))$$

where A_i is the ground p-atom in our language representing the fact that the random variable X achieves the value x at time n_i and $Pr(A_i|\mathcal{F}_i)$ is the probability of A_i when $\mathcal{F}_i$ is actually performed.

In our framework, utility functions can be specified as arbitrary procedures which accept some goal random variables as inputs. For the DVT example, we use the utility function suggested in [6] whose input parameters are the *dead*, *pe* and *bleeding* status of the patient after a plan is performed.

Bayesian Network-Graphs

Given a set of goal random variables and a plan fragment F, we can use BNG [2] to construct a

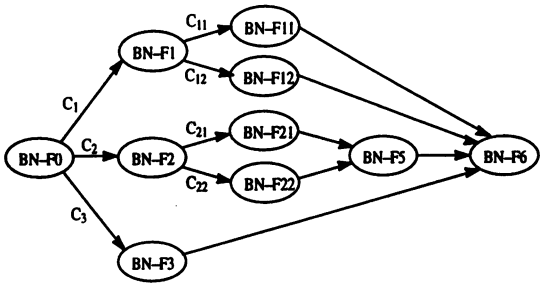

Figure 3: A BN-graph model of a contingent plan.

BN for evaluating the posterior probability of the goal random variables after F is performed.

A CP can be represented as a graph of plan fragments (see Figure 2). In order to evaluate the whole CP, we connect BNs for evaluating the plan fragments into a Bayesian network-graph (BN-graph) (see Figure 3). In the Figure 3, each $BN\text{-}F_i, i > 0$, is the BN of F_i.

Let X be the set of all random variables in a BN-graph G and x be one value assignment of X. Then, by the mutual exclusive property of the CASE construct, there is one and only one maximal directed path b in G such that the conjunction of the conditions on b is consistent with x. We define the probability of x induced by G as $P_G(x) = P_b(x)$, where P_b is the probability function induced by the BN formed from b.

Constructing a BN-graph to Evaluate a contingent plan

We use a backward chaining process to construct a BN-graph to evaluate the effect of a CP on a set of goal random variables. In the backward chaining process, the random variables at the starting state of the BNs of all plan fragments departing from one (diverging or converging) point are used as goal random variables in building the BN of the plan fragment(s) immediately preceding that point. Figure 4 shows two BNs constructed by our procedure for two plan fragments. The purpose is constructing the BNs relevant to the evaluation of the final goal random variables *bleeding*, *pe* and *dead* after performing the example CP of the previous section. The process starts with the final goal random variables and the action $treat(t+1)$. The random variables at the starting state of the BN in Figure 4.(a) and the observable random variable $venoResult(t+1)$ become the goal random variables for evaluating the plan fragment $\langle veno(t) \rangle$ (Figure 4.(b)). Figure 4.(c) shows the BN-graph. The simplest way to evaluate this BN-graph is considering two cases: $VenoResult(1)$ is + and $VenoResult(1)$ is -. In each case, we have one simple BN and we can apply any of the available propagation algorithms on it.

Notice that we do not generate the entire BN-graph but only the portion relevant to a set of goal random variables. This strategy makes the procedure more efficient. We have shown the soundness and completeness properties of our procedure under certain conditions.

KBMC FOR MEDICAL DECISION MAKING

Many planning systems assume that the space is simply the infinite space defined by all possible sequences of actions from some set [7]. But for many practical planning problems, constraints can be placed on the space of possible plans. For example, in the DVT domain all reasonable plans consist of some sequence of tests followed by treatment. We specify the constraints on the plan space by using programming language control constructs to specify how CPs may be combined.

We represent plan space by using *plan schemes*. A plan scheme is a program in which each primitive statement is a set of alternative CPs, and is called a *plan step*. We assume that there is a special *empty CP* (containing no action) called *noact*. In the next example, each alternative CP is simply a one action plan. The control structures are: sequential, conditional (by using the CASE construct) and iterative. There are two iterative constructs:

DO body UNTIL condition;

DO body UNTIL condition OR AT MOST n;

where *body* is a sub-plan scheme expressed in the same language, *condition* is a condition and n is an integer constant. Their semantics is the same as that of the normal loop constructs in programming languages.

One of the plan schemes investigated by the original DVT study [6] can be described by the following code.

```
DO
  {ipg, rus};
  {wait7d, noact}
UNTIL result is + OR AT MOST 2 times
CASE
  result = + : {veno}
       CASE
            result = + : {treat}
            OTHERWISE : {notreat}
       ENDCASE
  OTHERWISE : {notreat}
ENDCASE
```

Note that if actions have probabilistic effects, iterative loops without AT MOST qualifiers may never achieve their termination condition. However, Ngo et al. [3] show that decision-theoretic criteria can be used to confine the search for optimal plans to bounded spaces.

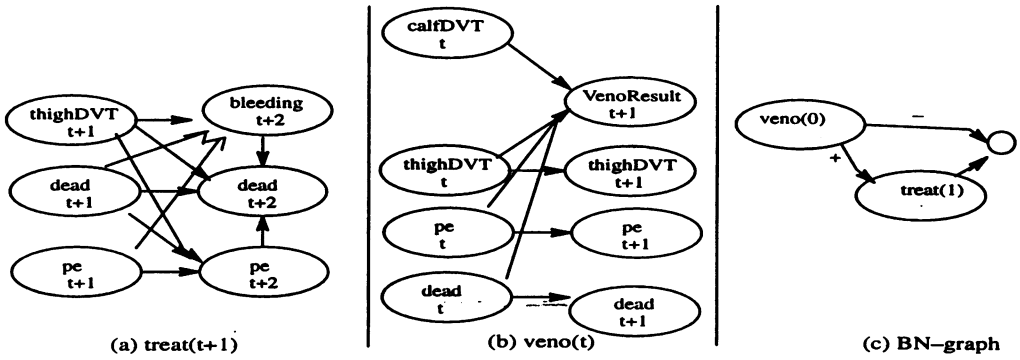

Figure 4: (a) The BN for evaluating the plan fragment $\langle treat(t+1)\rangle$ with respect to goal random variables $bleeding(t+2), pe(t+2), dead(t+2)$; (b) The BN for evaluating the plan fragment $\langle veno(t)\rangle$ with respect to goal random variables $thighDVT(t+1), VenoResult(t+1), pe(t+1), dead(t+1)$; (c) The BN-graph for evaluating the example contingent plan.

To find the optimal CP(s) from a given plan scheme, we construct a decision tree. The evaluation of the decision tree is performed by the well-known *average-out-and-fold-back* method [9]. The branching probabilities are obtained from evaluating dynamically constructed BN-graphs. Details of the procedures are provided in [4].

DISCUSSION

We have presented a theoretically well-founded method for constructing temporal Bayesian network-graphs for the evaluation of CPs. The presence of a formal semantics for the representation language is necessary in order to prove the correctness of the network generation algorithm. Such proofs are important for the high-stakes decision making problems encountered in medicine. Our technique is capable of selecting that portion of a probability model that is relevant to a particular inference problem by using context information and by pruning the generated network. The naturalness of the encoding of the Deep Venous Thrombosis domain shows that the representation is relatively easy to use. The networks generated to solve the example problems illustrate the potential computational savings of the technique.

We also proposed a framework for optimal treatment plans generation. The framework possesses attractive features. It allows the specification of plan structure. By maintaining separate action models, the decision maker can concentrate on the plan logic. In [4] we show that our framework is more expressive than Influence Diagrams.

Acknowledgements

This work was partially supported by NSF grant #IRI-9509165 (PH), by a University of Wisconsin-Milwaukee Graduate School Fellowship (LN) and by a Sun Microsystems AEG award (PH).

References

1. L. Ngo, P. Haddawy and J. Helwig. A Theoretical Framework for Temporal Probability Model Construction with Application to Plan Projection. UAI 1995.

2. P. Haddawy, J. Helwig, L. Ngo and R. Krieger. Clinical Simulation using Context-Sensitive Temporal Probabilistic Models. SCAMC 1995.

3. L. Ngo and P. Haddawy. Representing Loops for Decision-Theoretic Planning. AAAI 1995 Spring Symposium on Extended Theories of Actions.

4. L. Ngo and P. Haddawy. A KBMC Approach to Decision-Theoretic Planning with Conditionals and Loops. Technical Report 1996.

5. G.F. Cooper. The computational complexity of probabilistic inference using bayesian belief networks. *Artificial Intelligence*, 42(2-3):393–405, 1990.

6. Hillner BE, Philbrick JT, Becker DM. Optimal management of suspected lower-extremity deep vein thrombosis: an evaluation with cost assessment of 24 management strategies. *Arch Intern Med*, 152:165–175, 1992.

7. N. Kushmerick, S. Hanks, and D. Weld. An algorithm for probabilistic planning. Technical Report 93-06-03, Dept. of Computer Science and Engineering, University of Washington, June 1993. *Artificial Intelligence.*

8. Landefeld S, McGuire E, Cohen AM. Clinical findings associated with acute proximal deep vein thrombosis: a basis for qualifying clinical judgment. *Am J Med*, 88:382–388, 1990.

9. H. Raiffa. *Decision Analysis.* Addison-Wesley, Reading,MA, 1968.

10. Weinmann EE, Salzman EW. Deep-vein thrombosis. *N Engl J Med*, 331:1630–1641, 1994.