REPORT DOCUMENTATION PAGE

AFRL-SR-AR-TR-02-

0355

Public reporting burden for this collection of information is estimated to average 1 hour per response, including the time for reviewing instructions, searching
the collection of information. Send comments regarding this burden estimate or any other aspect of this collection of information, including suggestions
Operations and Reports, 1215 Jefferson Davis Highway, Suite 1204, Arlington, VA 22202-4302, and to the Office of Management and Budget, Paperwork Reduction Project (0704-0188), Washington, DC 20503.

1. AGENCY USE ONLY *(Leave blank)*

2. REPORT DATE

3. REPORT TYPE AND DATES COVERED
01 MAR 97 - 31 DEC 00

4. TITLE AND SUBTITLE
DISTRIBUTED DECISON MAKING AND PLAN RECOGNITION UNDER UNCERTAINTY

5. FUNDING NUMBERS
F49620-97-1-0175

6. AUTHOR(S)
MICHAEL P. WELLMAN

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)
UNIVERSITY OF MICHIGAN
ARTIFICIAL INTELLIGENCE LABORATORY
1101 BEAL AVENUE
ANN ARBOR, MI 48109-2110

8. PERFORMING ORGANIZATION REPORT NUMBER

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)
AFOSR/NM
801 N. Randolph Street Room 732
Arlington, VA 22203-1977

10. SPONSORING/MONITORING AGENCY REPORT NUMBER
F49620-97-1-0175

11. SUPPLEMENTARY NOTES

20020903 046

12a. DISTRIBUTION AVAILABILITY STATEMENT
APPROVED FOR PUBLIC RELEASE, DISTRIBUTION UNLIMITED

13. ABSTRACT *(Maximum 200 words)*

Our new work on abstraction approximations developed techniques for enhancing qualitative and numeric reasoning. First we developed incremental techniques for a task we call tradeoff resolution, in the context of qualitative Bayesian networks [Liu and Wellman 1998]. One approach incrementally marginalizes nodes that contribute to the ambiguous qualitative relationships. Another approach evaluates approximate networks for bounds of probability distributions, and uses these bounds to determine the qualitative relationships in question. This approach is incremental in that the algorithm refines the state spaces of random variables for tighter bounds until the qualitative relationships are determined. Both approaches provide systematic methods for tradeoff resolution at potentially lower computational cost than application of purely numeric methods.

Second, we developed techniques that exploit qualitative probabilistic relationships among variables for computing bounds of conditional probability distributions of interest in Bayesian networks [Liu and Wellman 1998]. Using the signs of qualitative relationships, we define abstraction operations that are guaranteed to bound the distributions of interest in the desired direction. By evaluating incrementally improved approximate networks, our algorithm obtains monotonically tightening bounds that converge to exact distributions. For some classes of utility functions, the tightening bounds monotonically reduce the set of admissible decision alternatives as well.

14. SUBJECT TERMS

15. NUMBER OF PAGES
13

16. PRICE CODE

17. SECURITY CLASSIFICATION OF REPORT

18. SECURITY CLASSIFICATION OF THIS PAGE

19. SECURITY CLASSIFICATION OF ABSTRACT

20. LIMITATION OF ABSTRACT

Standard Form 298 (Rev. 2-89) (EG)
Prescribed by ANSI Std. 239.18
Designed using Perform Pro, WHS/DIOR, Oct 94

AFOSR Final Technical Report

Covering the period 1 Sep 97 to 31 Dec 00

Project: **Distributed Decision Making and Plan Recognition Under Uncertainty**

Grant Number: **F49620-97-0175**

Principal Investigator: **Michael P. Wellman**

University of Michigan
Artificial Intelligence Laboratory
1101 Beal Avenue
Ann Arbor, MI 48109-2110

Overview

The ultimate aim of this project was to identify general principles and develop concrete techniques for distributed decision making in dynamic, uncertain environments. We focused our effort in three areas:

1. Abstraction methods for uncertain reasoning [Liu and Wellman 2002].
2. Market-based models for distributed belief aggregation.
3. New formalisms and algorithms for dynamic plan recognition, extending and generalizing techniques based on probabilistic context-free grammars (PCFGs).

We describe our main results in each of these threads below. The contributions to abstraction methods and market-based belief aggregation are outlined, whereas the developments in plan recognition are described in greater detail.

It is also worth noting that the project resulted in three PhD theses:

1. State-Space Abstraction Methods for Approximate Evaluation of Bayesian Networks (Chao-Lin Liu, May 1998)
2. Probabilistic Grammars for Plan Recognition (David Pynadath, January 1999)
3. Aggregating Probabilistic Beliefs: Market Mechanisms and Graphical Representations (David M. Pennock, September 1999)

Abstraction Methods for Uncertain Reasoning

Our new work on abstraction approximations developed techniques for enhancing qualitative and numeric reasoning. First, we developed incremental techniques for a task we call *tradeoff resolution*, in the context of qualitative Bayesian networks [Liu and Wellman 1998]. One approach incrementally marginalizes nodes that contribute to the ambiguous qualitative relationships. Another approach evaluates approximate networks for bounds of probability distributions, and uses these bounds to determine the qualitative relationships in question. This approach is incremental in that the algorithm refines the state spaces of random variables for tighter bounds until the qualitative relationships are determined. Both approaches provide systematic methods for tradeoff resolution at potentially lower computational cost than application of purely numeric methods.

Second, we developed techniques that exploit qualitative probabilistic relationships among variables for computing bounds of conditional probability distributions of interest in Bayesian networks [Liu and Wellman 1998]. Using the signs of qualitative relationships, we define abstraction operations that are guaranteed to bound the distributions of interest in the desired direction. By evaluating incrementally improved approximate networks, our algorithm obtains monotonically tightening bounds that converge to exact distributions. For some classes of utility functions, the tightening bounds monotonically reduce the set of admissible decision alternatives as well.

Market-Based Belief Aggregation

Background

The problem of *belief aggregation* is to derive a summary representation of a group's beliefs as a function of the beliefs of its constituent agents. The problem is a classic one in statistics, and it has also been well-studied in decision analysis. Despite the interest it has garnered, the problem has eluded definitive answers, and the plethora of impossibility results and definitional controversies in the literature cast doubt on the prospects for an entirely satisfactory solution.

Nevertheless, as trends toward decentralization in computation continue, aggregation mechanisms are likely to play an increasing role in uncertain reasoning. Software agents representing distinct interests and possessing individual knowledge and information-gathering capabilities will form their own beliefs, and no overarching authority will be technically or computationally able to gather all of the relevant information centrally or enforce any globally consistent consensus. If we wish to gain the benefits of others' knowledge, we need to induce them to provide relevant reports, or perform other actions that will reveal the information we seek.

From this perspective, paramount in the design of a belief aggregation mechanism are the incentives it provides to agents to reveal their private beliefs. Given some behavioral assumptions on the participants, we aim to characterize the aggregation function "computed" by the mechanism, that is, the relationship of the derived summary to the agents' individual beliefs.

In this project, we investigated the behavior of a particular approach to belief aggregation, based on markets in uncertain propositions [Pennock 1999]. The idea is that agents' decisions to trade in such markets will be driven by their beliefs and utility, and therefore the resulting prices in the markets will reflect private information bearing on the likelihood of the propositions. Agents in the market have to back up their stated positions with real money, and so have tangible disincentives to lie as well as positive incentives to participate and to gather all cost-effective relevant information.

One potential application of this approach is to use securities markets as a simple way to disseminate information in a large-scale distributed system. In such systems, relevant information is dispersed widely, and agents do not necessarily know which counterparts know what. By opening securities markets, we provide agents an incentive to (indirectly) reveal their information to the broader system. This can provide a significant benefit to other agents, who can then use this information without incurring the cost of acquiring it.

Opinion Pools

For decades, scientists have grappled with how best to aggregate subjective probabilities from multiple experts in order to form consensus assessments. Many combination functions or *opinion pools* have been proposed, each sensible under particular assumptions or measures. In the market-based approach we explore, the equilibrium prices of securities, paying off contingent on uncertain events, are interpreted as consensus probabilities. We developed a prescriptive model, identifying correspondences between the consensus probabilities derived by the market and by the traditional opinion pools [Pennock and Wellman 1997]. Parameters of the agents' utilities in the market context serve the role of expert weights in opinion pools.

In particular, we showed how securities markets can replicate the "logarithmic opinion pool". A *security* in this context is an asset that pays off $1 contingent on an uncertain proposition *p*. When rational agents buy and sell this security, its equilibrium price will reflect their beliefs about this proposition. The "market price" is thus a summary, or "pool" of these beliefs. If the agents have constant risk aversion, the equilibrium price will behave exactly as the logarithmic opinion pool, that is, a weighted geometric mean of individual agent probabilities. We have further derived conditions under which the equilibrium corresponds to the linear opinion pool, and established related theoretical properties as well. We have also implemented simulations of these security markets, and have demonstrated empirically that the prices converge as expected, even when agents learn from each others' beliefs, as revealed through the market [Pennock and Wellman 2001].

In the extended model, where agents learn from prices, the weights are shown to additionally depend on self-assessed confidence. Even when agents themselves do not learn, we have shown that the market as a whole can adapt over time in the manner of a Bayesian-rational individual, by transferring wealth from inaccurate to accurate agents.

Compact Securities Markets

The securities market is the fundamental theoretical framework in economics and finance for resource allocation under uncertainty. Securities serve both to reallocate risk and to disseminate probabilistic information. *Complete* securities markets—which contain one security for every possible state of nature—support Pareto optimal allocations of risk. Complete markets suffer from the same exponential dependence on the number of underlying events as do joint probability distributions. In this project, we examined whether markets can be structured and "compacted" in the same manner as Bayesian network representations of joint distributions. We were able to show that, if all agents' risk-neutral independencies agree with the independencies encoded in the market structure, then the market is *operationally complete*: risk is still Pareto optimally allocated, yet the number of securities can be exponentially smaller. For collections of agents of a certain type, agreement on Markov independencies is sufficient to admit compact and operationally complete markets [Pennock and Wellman 2000].

Graphical Models of Consensus Belief

Graphical models based on conditional independence support concise encodings of the subjective belief of a single agent. A natural question is whether the *consensus* belief of a *group* of agents can be represented with equal parsimony. We have proved, under relatively mild assumptions, that even if everyone agrees on a common graph topology, no method of combining beliefs can maintain that structure. Even weaker conditions rule out local aggregation within conditional probability tables. On a more positive note, we have shown that if probabilities are combined with the logarithmic opinion pool (LogOP), then commonly held *Markov* independencies are maintained. This suggests a straightforward procedure for constructing a *consensus Markov network.* Based on this idea; we developed an algorithm for computing the LogOP with time complexity comparable to that of exact Bayesian inference [Pennock and Wellman 1999].

Probabilistic Plan Recognition

Pattern and Plan Recognition

The problem of *plan recognition* is to induce the plan of action driving an agent's behavior, based on partial observation of its behavior up to the current time. Deriving the underlying plan can be useful for many purposes—predicting the agent's future behavior, interpreting its past behavior, or generating actions designed to influence the plan itself. Researchers in AI have studied plan recognition for several kinds of tasks, including discourse analysis [Grosz and Sidner 1990], collaborative planning [Huber and Durfee 1993], and adversarial planning [Azarewicz, et al. 1989]. These works have employed a great variety of reasoning techniques, operating on similarly various plan representations and adopting varied assumptions about observability.

The common theme underlying these diverse motivations and approaches is that the object to be induced is a *plan,* and that this plan is the cause of observed behavior. If there is anything special about the task of plan recognition as opposed to recognition in general, it must be due to special properties of plans: how they are constituted, and how they cause the behavior we observe and wish to predict, interpret, and influence.

We can distinguish plan recognition from uncertain reasoning in general by noting two special features of plans. First, plans are *structured linguistic objects.* Plan languages considered in AI research range from simple sequences of action tokens to general-purpose programming languages. In either case, the recognizer can and should exploit the structure of plans in inducing them from partial observations of the actions comprising the plan. Another way to say this is that plans are descriptions of action *patterns,* and therefore any general pattern-recognition technique is automatically a plan recognition technique for the class of plans corresponding to the class of patterns associated with the given technique.

The second special feature of plans is that they are *rational constructions*. They are synthesized by a rational agent with some beliefs, preferences, and capabilities, that is, a *mental state*. Knowing the agent's mental state and its rationality properties strongly constrains the possible plans it will construct. (The degree of constraint depends on the power of the rationality theory we adopt.) The rational origin of plans is what distinguishes plan recognition from pattern recognition. If the observations available include evidence bearing on the beliefs, preferences, and capabilities of the agent, then the recognizer should combine this with evidence from the observed actions in reasoning about the entire plan.

Our first step in this project was to elucidate [Pynadath and Wellman 1995] a general Bayesian framework for plan recognition. Our basic approach is similar to that of Charniak and Goldman [Charniak and Goldman 1993], elaborating and departing in some respects, less well-developed in others. We describe the high-level idea below; for a more complete description and some specific developments of the technique see the cited papers.

Our framework is *Bayesian* in that we start from a causal theory of how the agent's mental state causes its plan and executing its plan causes activity, and reason from observed effects to underlying causes. Our recognizer has uncertain *a priori* knowledge about the agent's mental state, the world state, and the world's dynamics, which can be summarized (at least in principle) by a probability distribution. It then makes partial observations about the world, and uses this evidence to induce properties of the agent and its plan.

We begin with a model of the planning agent operating in the world. As it begins planning, the agent has a certain mental state, consisting of its preferences (e.g., goals), beliefs (e.g., about the state of its environment), and capabilities (e.g., available actions). We assume the actual planning process to be some rational procedure for generating the plan that will best satisfy the agent's preferences based on its beliefs, subject to its capabilities. This plan then determines (perhaps with some uncertainty) the actions taken by the agent in the world.

Once we have accounted for the agent's plan-generation process, we need to consider the effects of the plan's execution. In many plan-recognition domains, the external observer finds the agent's actions inaccessible. In such cases, the recognizer observes actions only indirectly, via their effects on the world (which themselves are typically only partially observable). These restricted observations then form the basis of inference.

Thus, observations of the state of the world provide two types of evidence about the plan. First, the world influences the agent's initial mental state, which provides the *context* for plan generation. Second, changes in the world state reflect the effects of the agent's actions, which *result* from executing its plan.

Bayesian Networks for Plan Recognition

To perform plan recognition tasks, we generate a Bayesian network representing the causal planning model and use it to support evidential reasoning from observations to plan hypotheses. The structure of the Bayesian network is based on the framework depicted in Figure 1. That diagram can itself be viewed as a Bayesian network, albeit with rather broad random variables. To make this operational, we replace each component of the model with a subnetwork that captures intermediate structure for the particular problem. The limited connections among the subnetworks reflect the dependency structure of our generic planning model.

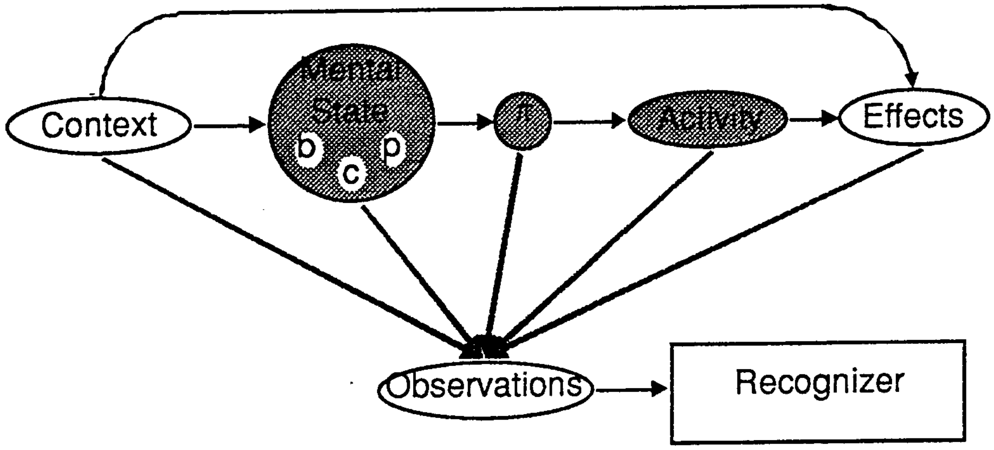

Figure 1: Bayesian plan recognition framework.

The framework as described above is of course very general. We have explored particular instances of the approach, specifically looking at the issue of modeling context in the domain of traffic monitoring [Pynadath and Wellman 1995]. In subsequent work, we investigated more deeply the problem of modeling the planning process. In doing so, we need to adopt particular assumptions about the plans generated, and determine an effective recognition strategy.

Probabilistic Context-Free Grammars

Our approach has been to treat plan generation as a structured stochastic process, and recognition as the task of answering queries about events in the generation of particular observations. Our first deep study adopted the generative model of *probabilistic context-free grammars* (PCFGs), a well-studied and commonly applied model for pattern recognition [Charniak and Goldman 1993; Wetherell 1980]. Interpreting a string of observations generated from a grammar is known as *parsing*, and the general recognition problem can be cast in terms of queries about the parse. For PCFGs, efficient algorithms have been developed for several useful types of queries (i.e., calculating the probability of a given string, or finding the most likely parse). However, for other queries potentially useful in plan recognition, only brute-force enumeration is available.

To extend this approach, we have shown [Pynadath and Wellman 1996; Pynadath and Wellman 1998] how to construct a Bayesian network to represent the distribution of parse trees induced by a given PCFG. The network structure mirrors that of the chart in a standard parser, and is generated using a similar dynamic-programming approach. By augmentations of the network, we can relax the context-free restriction of the grammar in a controlled way, admitting important context-sensitivities without invalidating the inferences drawn by the recognizer.

This method generalizes the class of queries that can be answered in several ways:

(1) allowing missing tokens in a sentence or sentence fragment,

(2) supporting queries about intermediate structure, such as the presence of particular nonterminals, and

(3) flexible conditioning on a variety of types of evidence.

We direct the reader to our published work for discussion of the technical details of our algorithm. In these documents, we present an algorithm for constructing Bayesian networks from PCFGs, and show how queries or patterns of queries on the network correspond to interesting queries on PCFGs.

Air Force Plan Recognition Application

The generalized pattern-recognition procedure is potentially applicable to a wide range of Air Force problems involving interpreting uncertain or incomplete observations. One example comprises problems of plan recognition, where the aim is to interpret or predict the actions of an observed agent (friend or foe), based on uncertain observations of its action thus far.

One of the more common representations for planning structures used in plan-recognition research is an action decomposition hierarchy, sometimes called an *event tree* [Kautz and Allen 1986]. Event trees and other variants of hierarchies map easily to context-free grammars, and indeed the parsing approach to recognition has previously been proposed (for the deterministic case) by Vilain [Vilain 1990]. By extending the event-tree model to include probabilities, we provide a basis for distinguishing among equally possible but unequally plausible explanations of the observations. As Charniak and Goldman [Charniak and Goldman 1993] (among others) have argued, this is a critical requirement for any useful plan recognition algorithm.

In air-combat scenarios, for example, we can model the behavior of a fighter plane to allow tracking and prediction of its actions. The probabilistic event tree could include information about possible specializations of its general mission (e.g. fly to target, intercept enemy plane), as well as decompositions of plans into subplans (e.g. employ weapons, evade, chase) or observable actions (e.g. start turning, stop turning, maintain current heading). An example event tree for an air-to-air combat scenario (borrowed from Tambe and Rosenbloom [Tambe and Rosenbloom 1995]) is presented in Figure 2.

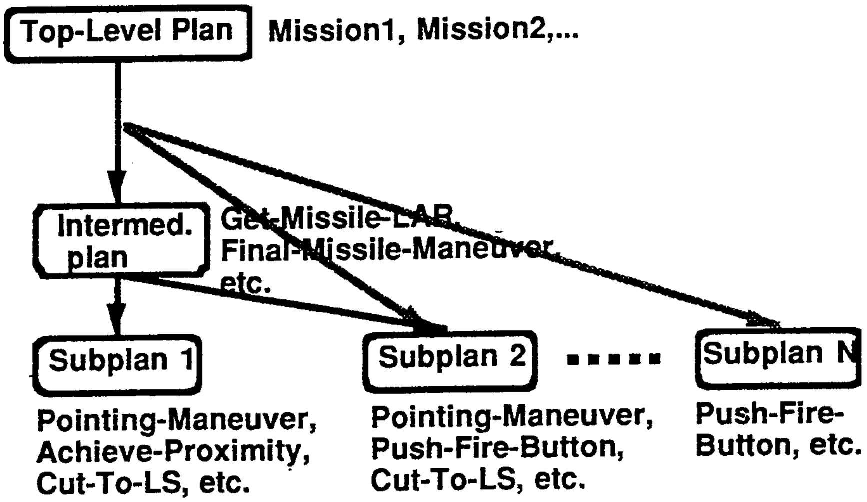

Figure 2: An example event tree from air combat domain.

We can then translate this event tree representation into a probabilistic grammar whose rules correspond to the plan specializations and decompositions. The algorithm mentioned above can use this grammar to generate a Bayesian network corresponding to the probability distribution over the possible behavior of the tracked plane. This network would support a wide variety of useful queries, using the traditional methods of evidence propagation to compute the relevant probabilities. In the air-combat example, a pilot may wish to determine whether a nearby enemy plane is about to launch a missile, or is merely flying to another target. The Bayesian network can provide the probability of either subplan, conditioned on whatever behavior has been observed so far. These probabilities, along with the different implications of the two cases, can aid the pilot in choosing the correct course of action.

Probabilistic State-Dependent Grammars

One drawback of using PCFGs for plan recognition (as well as most other extant approaches), is that they require maintaining the entire history of observations as context for subsequent plan recognition queries. Whereas this is unavoidable in general, it may well be possible to employ graphical modeling techniques to exploit whatever independence exists to support practical inference.

Our investigations culminated in the development of *probabilistic state-dependent grammars* (PSDGs), a hybrid representation based on PCFGs augmented with dependencies of production probabilities on underlying states [Pynadath and Wellman 2000]. The state evolution is defined by transition probabilities, represented in dynamic Bayesian networks (DBNs) [Kjærulff 1992]. Although any PSDG can be represented as a PCFG, the PSDG representation may be exponentially more compact when the state space is highly structured.

Example: Highway Traffic

Consider the following PSDG, representing a simplified model of driving plans.

0) Drive → Stay Drive $(p_0(q) = \ldots)$
1) Drive → Left Drive $(p_1(q) = \{0 \text{ if } \mathrm{Lane}(q) = \text{left-lane} \ldots)$
2) Drive → Right Drive $(p_2(q))$
3) Drive → Pass Drive $(p_3(q))$
4) Drive → Exit $(p_4(q))$
5) Drive → Left Right $(p_5(q))$
6) Drive → Right Left $(p_6(q))$

The state includes the observable features of the driver's position and speed, as well as the position and speeds of other cars on the highway. The state also includes aspects of the driver's mental state, such as the agent's preferences about driving speed, distance from other cars, intended exit, etc. We can explore the generation of the parse tree of Figure 3 (corresponding to one possible instance of the agent's plan generation and execution) to illustrate the interactions between the plan and state models. The pictures across the bottom of the diagram represents the observable portion of state at that point of the parse tree. The darker rectangle (blue if reading in color) is the driver whose planning process we are trying to recognize. The lighter rectangles (green) are the other cars on the highway that the driver of interest must consider when planning.

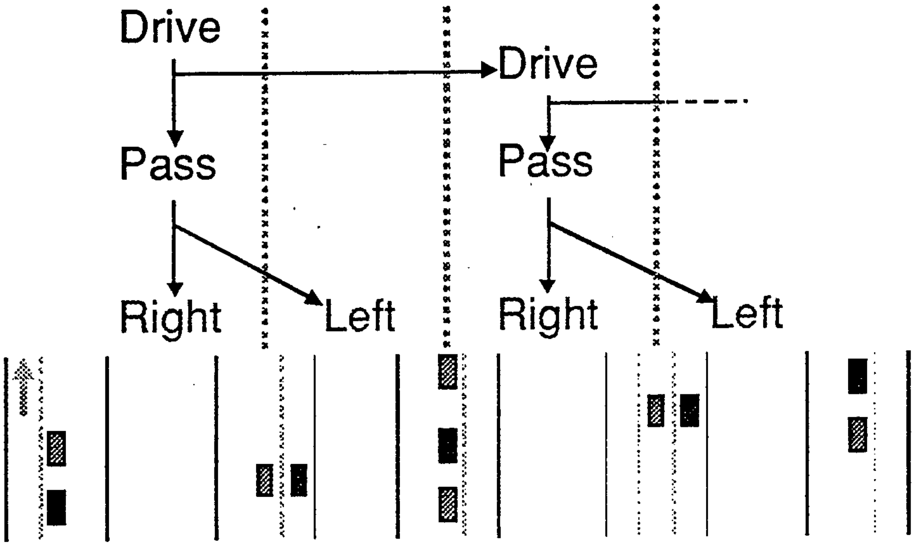

Figure 3: A parse tree for the simple highway example.

In this scenario, the driver passes two cars, both times on the right. The PSDG formalism makes it easy to specify how the driver's decision to pass, and on which side, may depend on situational factors. This is accomplished by conditioning the production probabilities in the grammar on the underlying state. For example, the decision to pass might be based on desired speed and speed of the car ahead, whereas choice of side might depend on the presence of cars in other lanes.

Our implemented traffic PSDG has 14 nonterminal symbols (plans), 7 terminal symbols (actions), and 15 state features (with the mean state space size being 431 elements). Three of these state features correspond to aspects of the driver's mental state (preferred speed, intended exit, aggressiveness); the rest of the state features are completely observable. There are a total of 40 productions with a mean length of two symbols. We also implemented a PSDG representation for an air combat domain based on an existing specification using SOAR productions [Laird, et al. 1987].

PSDG Inference

Although we can perform inference on a given PSDG with a finite state space by generating the corresponding PCFG and using PCFG inference algorithms, the explosion in the size of the symbol space can lead to prohibitive costs. In addition, existing PCFG algorithms cannot handle most plan-recognition queries.

We can potentially perform inference by generating a DBN representation of a PSDG distribution. The definition of the PSDG language model supports an automatic DBN generation algorithm. The resulting DBN supports queries over the symbol, production, and state random variables. Unfortunately, the complexity of DBN inference is likely to be impractical for most PSDGs, where the belief state must represent the entire joint distribution over all possible combinations of state and parse tree branches. For instance, for the complete PSDG representation of the traffic domain, the DBN belief state would have more than 10^{25} entries.

Instead, we have designed and implemented inference algorithms that exploit the particular structure of the PSDG model to answer a set of queries more restricted than that provided by DBNs. These algorithms use a compact belief state to answer queries based on observations of the state variables. At time t, the recognizer observes some or perhaps all of the features of the state, Q^t. Based on this evidence, the algorithm computes posterior probabilities over the individual state elements, as well as posterior probabilities over the possible plans and productions that the agent executed at time $t-1$. The algorithm then computes the posterior probabilities over the plans and productions that the agent will select at time t, as well as updating the recognizer's belief state.[1]

---

[1] A pseudocode description of the algorithm is available online at http://www.isi.edu/~pynadath/Research/PSDG. Both the pseudocode and proofs of correctness are presented in the dissertation [Pynadath 1999].

References

Azarewicz, J., Fala, G. and Heithecker, C., Template-based multi-agent plan recognition for tactical situation assessment. in *Sixth Conference on Artificial Intelligence Applications*, (1989), 247-254.

Charniak, Eugene and Goldman, Robert P. A Bayesian model of plan recognition. *Artificial Intelligence, 64*. 53-79, 1993.

Grosz, B. J. and Sidner, C. L. Plans for discourse. in Cohen, P.R., Morgan, J. and Pollack, M.E. eds. *Intentions in Communication*, MIT Press, 1990, 417-444.

Huber, Marcus H. and Durfee, Edmund H., Observational uncertainty in plan recognition among interacting robots. in *IJCAI Workshop on Dynamically Interacting Robots*, (Chambery, France, 1993), 68-75.

Kautz, Henry A. and Allen, James F., Generalized plan recognition. in *National Conference on Artificial Intelligence*, (Philadelphia, 1986), 32-37.

Kjærulff, Uffe, A computational scheme for reasoning in dynamic probabilistic networks. in *Eighth Conference on Uncertainty in Artificial Intelligence*, (Standord, CA, 1992), 121-129.

Laird, John E., Newell, Allen and Rosenbloom, Paul S. SOAR: An architecture for general intelligence. *Artificial Intelligence*, 33. 1-64, 1987.

Liu, Chao-Lin and Wellman, Michael P. Evaluation of Bayesian networks with flexible state-space abstraction methods. *International Journal of Approximate Reasoning, 30*. 1-39, 2002.

Liu, Chao-Lin and Wellman, Michael P., Incremental tradeoff resolution in qualitative probabilistic networks. in *Fourteenth Conference on Uncertainty in Artificial Intelligence*, (Madison, WI, 1998), 338-345.

Liu, Chao-Lin and Wellman, Michael P., Using qualitative relationships for bounding probability distributions. in *Fourteenth Conference on Uncertainty in Artificial Intelligence*, (Madison, WI, 1998), 346-353.

Pennock, David M. and Wellman, Michael P., Compact securities markets for Pareto optimal reallocation of risk. in *Sixteenth Conference on Uncertainty in Artificial Intelligence*, (Stanford, CA, 2000), 481-488.

Pennock, David M. and Wellman, Michael P., Graphical representations of consensus belief. in *Fifteenth Conference on Uncertainty in Artificial Intelligence*, (Stockholm, 1999), 531-540.

Pennock, David M. and Wellman, Michael P. A market framework for pooling opinions, NEC Research and University of Michigan, 2001.

Pennock, David M. and Wellman, Michael P., Representing aggregate belief through the competitive equilibrium of a securities market. in *Thirteenth Conference on Uncertainty in Artificial Intelligence,* (Providence, RI, 1997), 392-400.

Pennock, David Myer. Aggregating Probabilistic Beliefs: Market Mechanisms and Graphical Representations, University of Michigan, 1999.

Pynadath, David V. Probabilistic Grammars for Plan Recognition, University of Michigan, 1999.

Pynadath, David V. and Wellman, Michael P., Accounting for context in plan recognition, with application to traffic monitoring. in *Eleventh Conference on Uncertainty in Artificial Intelligence,* (Montréal, 1995), 472-481.

Pynadath, David V. and Wellman, Michael P., Generalized queries on probabilistic context-free grammars. in *National Conference on Artificial Intelligence,* (Portland, OR, 1996), AAAI Press.

Pynadath, David V. and Wellman, Michael P. Generalized queries on probabilistic context-free grammars. *IEEE Transactions on Pattern Analysis and Machine Intelligence, 20*. 65-77, 1998.

Pynadath, David V. and Wellman, Michael P., Probabilistic state-dependent grammars for plan recognition. in *Sixteenth Conference on Uncertainty in Artificial Intelligence,* (Stanford, CA, 2000), 507-514.

Tambe, Milind and Rosenbloom, Paul S., RESC: An approach for real-time, dynamic agent tracking. in *Fourteenth International Joint Conference on Artificial Intelligence,* (Montreal, 1995), 103-110.

Vilain, Marc, Getting serious about parsing plans. in *National Conference on Artificial Intelligence,* (Boston, 1990), 190-197.

Wetherell, C. S. Probabilistic languages: A review and some open questions. *Computing Surveys,* 12 (4). 361-379, 1980.