In ICAPS-06 Workshop on Planning Under Uncertainty and Execution Control for Autonomous Systems

Using Interaction to Compute Better Probability Estimates in Plan Graphs

Daniel Bryce
Dept. of Computer Science & Eng.
Arizona State University
Tempe Arizona 85287–5406
dan.bryce@asu.edu

David E. Smith
Intelligent Systems Division
NASA Ames Research Center
Moffet Field, CA 94035–1000
de2smith@email.arc.nasa.gov

Abstract

Plan graphs are commonly used in planning to help compute heuristic "distance" estimates between states and goals. A few authors have also attempted to use plan graphs in probabilistic planning to compute estimates of the probability that propositions can be achieved and actions can be performed. This is done by propagating probability information forward through the plan graph from the initial conditions through each possible action to the action effects, and hence to the propositions at the next layer of the plan graph. The problem with these calculations is that they make very strong independence assumptions - in particular, they usually assume that the preconditions for each action are independent of each other. This can lead to gross overestimates in probability when the plans for those preconditions interfere with each other. It can also lead to gross underestimates of probability when there is synergy between the plans for two or more preconditions.

In this paper we introduce a notion of the binary *interaction* between two propositions and actions within a plan graph, show how to propagate this information within a plan graph, and show how this improves probability estimates for planning. This notion of interaction can be thought of as a continuous generalization of the notion of mutual exclusion (mutex) often used in plan graphs. At one extreme (interaction $= 0$) two propositions or actions are completely mutex. With interaction $= 1$, two propositions or actions are independent, and with interaction > 1, two propositions or actions are synergistic. Intermediate values can and do occur indicating different degrees to which propositions and action interfere or are synergistic. We compare this approach with another recent approach by Bryce that computes probability estimates using Monte Carlo simulation of possible worlds in plan graphs.

Introduction

Plan graphs are commonly used in planning to help compute heuristic "distance" estimates between states and goals. A few authors have also attempted to use plan graphs in probabilistic planning to compute estimates of the probability that propositions can be achieved and actions can be performed (Blum & Langford 1999; Little, Aberdeen, & Thiébaux 2005). This information can then be used to help guide a probabilistic planner towards the most effective actions for maximizing probability or for achieving the goals with a given probability threshold.

Typically, probability information is given for the propositions in the initial state and is propagated forward through the plan graph, in a manner similar to the propagation of cost and resource estimates in classical planning. The probability of being able to perform an action is taken to be the probability that its preconditions can be achieved, which is usually approximated as the product of the probabilities of the preconditions. The probability of a particular action effect is taken as the product of the action probability and probability of the effect given the action. Finally, the probability of achieving a proposition at the next level of the plan graph is then taken to be either the sum or maximum of the probabilities for the different effects yielding that proposition. As an example, consider one level of a plan graph shown in Figure 1 where we have two actions a and b each with two preconditions and two unconditional effects. As in IPP (Koehler *et al.* 1997) and (Bryce, Kambhampati, & Smith 2006a; 2006b), we have explicitly included an effect layer in between each action layer and the proposition layer at the next level. Suppose that the probabilities for the propositions p, q, and r are $.8$, $.5$, and $.4$ as shown in the figure. The probability that action a is possible would then be the probability of the conjunction $p \wedge q$ which would be $.8(.5) = .4$. Similarly, the probability for action b would be $.5(.4) = .2$. Action a produces effect e with certainty (probability 1), so e simply inherits the probability of $.4$ from a. Action a produces effect f with probability $.5$, so the probability of this effect is $.4(.5) = .2$. In similar fashion, the probabilities of the effects g and h for action b are $.2$ and $.2(.5) = .1$ respectively. At the next level, the propositions s and u only have one contributing effect, so they just inherit the probabilities of those effects. However, the proposition t has two contributing effects, and we could develop a plan that uses them both to increase the chances of achieving t. Assuming that the effects f and g are independent we could compute the probability of the disjunction $f \vee g$ as $.2 + .2 - .2(.2) = .36$.

The problem with these simple estimates is that they assume independence between all pairs of propositions and all pairs of actions in the plan graph. This is frequently a very bad assumption. If two propositions are produced by the same action (e.g. s and t), they are not independent of each other, and computing the probability of the conjunction by taking the product of the individual probabilities can result

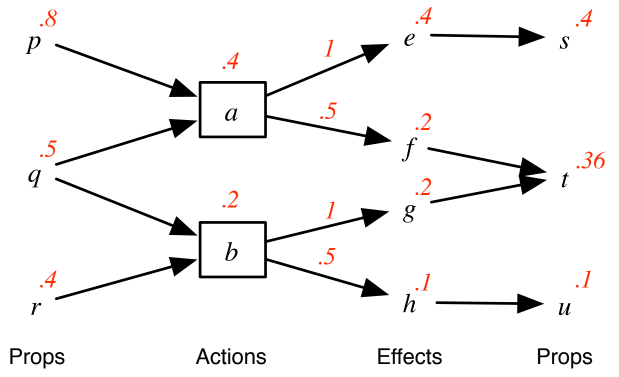

Figure 1: A plan graph level with simple probability calculations made using the independence assumption.

in a significant underestimate. Conversely, if two propositions are mutually exclusive, then the probability of achieving them both is zero, and the product of their probabilities will be a significant overestimate. In our example, we first assumed that the propositions p, q and r were independent of each other when computing the probabilities of actions a and b. Even if this is so, we then proceeded to assume that effects f and g were independent, when computing the probability of proposition t. Clearly this is wrong, since a and b share a precondition.

One obvious way to improve the estimation process would be to propagate and use mutual exclusion information, and assign a probability of zero to actions with mutex preconditions at a given level. However, this only helps with the extreme case where propositions or actions are mutex. It does not help with cases of synergy, or with cases where propositions are not strictly mutex, but it is much “harder” (less probable) to achieve them both.

To attempt to address this problem, we introduce a more general notion which we call *interaction* to capture both positive and negative interactions between pairs of propositions, pairs of actions, and pairs of action effects. In the section that follows, we first give a formal definition of our notion of interaction. We then show how to compute and use interaction information within a plan graph to get better probability estimates. Finally we show some preliminary results, and compare this technique with another recent technique developed by Bryce, Kambhampati, & Smith (2006b).

Definitions and Representation

Action Representation

Following the representation used in (Bryce, Kambhampati, & Smith 2006b), an action a is taken to have:

- an enabling precondition, $\mathrm{Pre}(a)$
- a set of probabilistically weighted outcomes, $\Phi_i(a)$

The enabling precondition $\mathrm{Pre}(a)$ is a conjunction of literals, just as for an action in probabilistic PDDL (PPDDL) (Younes *et al.* 2005; Younes & Littman 2004) or an ordinary classical action in PDDL (McDermott 1998). Each outcome $\Phi_i(a)$ has a weight $w_i(a)$ giving the probability that the outcome is realized, and $\Phi_i(a)$ consists of a conjunction of conditional effects $\phi_{ij}(a)$ of the form:

$$\rho_{ij} \longrightarrow \varepsilon_{ij}$$

where both ρ_{ij} and ε_{ij} are conjunctions of literals. Of course, ρ_{ij} may be empty, in which case ε_{ij} is an unconditional effect. This representation of effects follows the 1ND normal form presented in (Rintanen 2003). The representation in PPDDL (Younes *et al.* 2005; Younes & Littman 2004) is a bit more general since it allows arbitrary nesting of conditional effects and probabilistic outcomes. We have chosen to use the 1ND normal form here because it is a bit easier to work with, and PPDDL can be expanded into this form.

Probability

Before going any further, we need to be clear about what we mean by probabilities attached to propositions and actions in a plan graph. A plan graph provides an optimistic assessment of what propositions and actions are possible. A probability attached to a proposition or action in a plan graph is therefore an indication of the probability that the proposition or action is possible, not the probability that it is true or has actually happened. As such, this probability implicitly refers to some plan. In other words, when we say that $\mathrm{Pr}(p) = .2$ for some proposition in a plan graph we mean that $\mathrm{Pr}(p)$ would be .2 if we executed some particular plan for achieving p – namely the *best* possible plan for achieving p. More precisely, if p is at level k in a plan graph, by $\mathrm{Pr}(p)$ we mean:

$$\max_{k\text{-level plans } \mathcal{P}} \mathrm{Pr}(p|\mathcal{P}) \tag{1}$$

Similarly, when we refer to the probability of an action a at level k in a plan graph, we mean the probability that the action is possible given the best possible plan for achieving its preconditions:

$$\max_{k\text{-level plans } \mathcal{P}} \mathrm{Pr}(\mathrm{Pre}(a)|\mathcal{P}) \tag{2}$$

Of course we can’t possibly compute these probabilities exactly without generating all possible plans for the proposition p or action preconditions $\mathrm{Pre}(a)$. Instead, we are simply trying to estimate these probabilities, and are prepared to make assumptions in order to do so. As we noted in the introduction, a common assumption is to suppose that the probability estimates for different propositions are independent of each other, but this assumption often leads to poor estimates.

Interaction

Formally, we define the interaction between two propositions, two actions, or two effects x and y as:

$$I(x,y) \equiv \frac{\mathrm{Pr}(x \wedge y)}{\mathrm{Pr}(x)\,\mathrm{Pr}(y)} \tag{3}$$

which by Bayes Rule can also be seen as:

$$= \frac{\mathrm{Pr}(x|y)}{\mathrm{Pr}(x)}$$

$$= \frac{\mathrm{Pr}(y|x)}{\mathrm{Pr}(y)}$$

Interaction is a continuous quantity that can range from zero to plus infinity. Essentially, it measures how much more or less probable it is that we can establish x and y together as opposed to if we could establish them independently. It has the following characteristics:

$$\begin{aligned} I(x,y) &= 0 && \text{if } x \text{ and } y \text{ are mutex} \\ &= 1 && \text{if } x \text{ and } y \text{ are independent} \\ &= \tfrac{1}{\Pr(x)} = \tfrac{1}{\Pr(y)} && \text{if } x \text{ and } y \text{ are completely synergistic}^{1} \end{aligned}$$

More generally, $0 < I(x,y) < 1$ means that there is some interference between the best plans for achieving x and y so it is harder (less probable) to achieve them both than to achieve them independently. Similarly, $1 < I(x,y) < 1/\Pr(x)$ means that there is some amount of synergy between plans for achieving x and y, so it is easier (more probable) to achieve them both than to achieve them independently.

Instead of computing and keeping mutex information in the plan graph, we will compute interaction information between all pairs of propositions and all pairs of actions at each level. It is worthwhile noting that for a pair of propositions or actions x and y we could instead choose to directly store the probability $\Pr(x \wedge y)$, or either of the two conditional probabilities $\Pr(x|y)$ or $\Pr(y|x)$ instead of the interaction $I(x,y)$. This is because these quantities are essentially equivalent - from our definition of interaction and Bayes Rule any of these quantities can be computed from any other. We have chosen to introduce the notion of interaction and store this quantity because:

1. it is symmetric, unlike the conditional values.
2. we only need to store it for cases where it is not one - i.e. the propositions/actions are not independent.
3. it can be easily interpreted and understood in terms of the intuitive concepts of mutex, independence, and synergy.

Computing Probability and Interaction

To compute probability and interaction information in a plan graph, we begin at the initial state (level 0) and propagate information forward through the plan graph to subsequent levels (just as with construction and propagation in ordinary classical plan graphs). In the subsections that follow, we give the details of how to do this beginning with the initial proposition layer and working forward to actions, then effects, and finally to the next proposition layer.

Computing Action Probabilities

Suppose that we have the probabilities and interaction information for propositions at a given level of the plan graph. How do we use this information to compute probabilities and interaction information for the subsequent action layer? First consider an individual action a with preconditions $\{x_1, \ldots, x_n\}$. The probability that the action can be executed is the probability that all the preconditions can be achieved:

$$\begin{aligned} \Pr(a) &= \Pr(x_1 \wedge \ldots \wedge x_n) \\ &= \Pr(x_1)\Pr(x_2|x_1)\ldots\Pr(x_n|x_1 \ldots x_{n-1}) \end{aligned} \quad (4)$$

If the propositions x_i are all independent this is just the usual product of the individual probabilities of the preconditions. However, if they are not independent then we need the conditional probabilities, $\Pr(x_i|x_1 \ldots x_{i-1})$. Since we have pairwise interaction information we can readily compute the first of these terms:

$$\Pr(x_2|x_1) = I(x_1,x_2)\Pr(x_2)$$

However, to compute the higher order terms (i.e. $i > 2$) we must make an approximation. Applying Bayes Rule we get:

$$\Pr(x_i|x_1 \ldots x_{i-1}) = \frac{\Pr(x_1 \wedge \ldots \wedge x_{i-1}|x_i)\Pr(x_i)}{\Pr(x_1 \wedge \ldots \wedge x_{i-1})}$$

If we make the assumption that $x_1 \ldots x_{i-1}$ are independent for purposes of this computation we get:

$$\Pr(x_i|x_1 \ldots x_{i-1}) = \frac{\Pr(x_1|x_i)\ldots\Pr(x_{i-1}|x_i)\Pr(x_i)}{\Pr(x_1)\ldots\Pr(x_{i-1})}$$

Applying our analogue of Bayes Rule again $i-1$ times, we get:

$$\begin{aligned} \Pr(x_i|x_1 \ldots x_{i-1}) &= \Pr(x_i)\frac{\Pr(x_i|x_1)}{\Pr(x_i)}\cdots\frac{\Pr(x_i|x_{i-1})}{\Pr(x_i)} \\ &= \Pr(x_i)I(x_i,x_1)\ldots I(x_i,x_{i-1}) \\ &= \Pr(x_i)\prod_{j=1\ldots i-1} I(x_i,x_j) \end{aligned}$$

Returning to the calculation of:

$$\begin{aligned} \Pr(a) &= \Pr(x_1 \wedge \ldots \wedge x_n) \\ &= \Pr(x_1)\Pr(x_2|x_1)\ldots\Pr(x_n|x_1 \ldots x_n - 1) \end{aligned}$$

if we plug in the above expression for the $\Pr(x_i|x_1 \ldots x_{i-1})$ we get

$$\begin{aligned} \Pr(a) &= \Pr(x_1 \wedge \ldots \wedge x_n) \\ &= \prod_{i=1\ldots n}\left[\Pr(x_i)\prod_{j=1\ldots i-1} I(x_i,x_j)\right] \end{aligned} \quad (5)$$

Several properties of this approximation are worth noting:

1. the above expression is easy to compute and does not depend on the order of the propositions.
2. If the x_i are independent, the $I(x_i,x_j)$ are 1 and the above simplifies to the product of the individual probabilities.
3. If any x_i and x_j are mutex then $I(x_i,x_j) = 0$ and the above expression becomes zero. If the $I(x_i,x_j)$ are positive but less than one then the probability of the conjunction is less than the product of the probabilities of the individual elements.
4. If the $I(x_i,x_j)$ are greater than one, there is synergy between the conjuncts. The probability of the conjunction is greater than the product of the probabilities of the individual conjuncts, but less than or equal to the minimum of those probabilities.

[1] x cannot occur without y, and vice versa, which means that their probabilities must be the same.

While these properties are certainly desirable, and match our intuitions, it is reasonable to ask how good the approximation in Equation 5 is in other cases. As it turns out, for a conjunction with n terms, Equation 5 turns out to be exact if only about n of the possible n^2 $I(x_i, x_j)$ are not equal to 1. More precisely:

Theorem 1 *Consider the undirected graph consisting of a node for each conjunct x_i, and an edge between x_i and x_j whenever x_i and x_j are not independent ($I(x_i, x_j)$ is not equal to 1). If this graph has no cycles, then Equation 5 is exact.*

As an example, consider the simple case of:

$$\Pr(a \wedge b \wedge c) = \Pr(a)\Pr(b|a)\Pr(c|ba)$$

Our graph consists of the three nodes a, b and c, and zero to three edges depending on the C's. If b and c are independent, there are only two edges in the graph, and no cycle, so the theorem states that Equation 5 is exact. To see this, with b and c independent the above expansion becomes:

$$\begin{aligned}\Pr(a \wedge b \wedge c) &= \Pr(a)\Pr(b|a)\Pr(c|a)\\ &= \Pr(a)\Pr(b)I(a,b)\Pr(c)I(a,c)\end{aligned}$$

Which is the approximation in Equation 5, since $I(b, c) = 1$

More generally, the proof of this theorem relies on the fact that a graph without cycles can be represented as a tree:

Proof: Suppose we have a conjunction $x_1 \wedge \ldots \wedge x_n$ that obeys the conditions of the theorem. Since the graph has no cycles, it can be arranged as a tree. Without loss of generality, assume the conjuncts are in the same order as a depth first traversal of that tree.

In general, we know that:

$$\Pr(x_1 \wedge \ldots \wedge x_n) = \prod_{i=1,\ldots,n} \Pr(x_i|x_1 \ldots x_{i-1})$$

But since the conjuncts are ordered according to a depth first traversal of the tree, each conjunct x_i has only one predecessor $x_j = x_{par(i)}$ (its parent in the tree) for which $I(x_i, x_j)$ is not one. As a result,:

$$\begin{aligned}\Pr(x_i|x_1 \ldots x_{i-1}) &= \Pr(x_i|x_{par(i)})\\ &= \Pr(x_i)I(x_i|x_{par(i)})\end{aligned}$$

This means that:

$$\Pr(x_1 \wedge \ldots \wedge x_n) = \prod_{i=1,\ldots,n} \Pr(x_i)I(x_i, x_{par(i)})$$

But since $I(x_i, x_j) = 1$ for all $j < i$ and $j \neq par(i)$ there is no harm in adding these terms and we get:

$$\begin{aligned}\Pr(a) &= \Pr(x_1 \wedge \ldots \wedge x_n)\\ &= \prod_{i=1\ldots n}\left[\Pr(x_i)\prod_{j=1\ldots i-1} I(x_i, x_j)\right]\end{aligned}$$

which is Equation 5. ■

Computing Interaction Between Actions

As with propositions, the probability that we can execute two actions, a and b, may be more or less than the product of their individual probabilities. If the actions are mutually exclusive (in the classical sense) then the probability that we can execute them both is zero. Otherwise, it is the probability that we can establish the union of the preconditions for the two actions.

$$\begin{aligned}\Pr(a \wedge b) &= 0 && a \text{ and } b \text{ mutex}\\ &= \Pr\left(\bigwedge(\text{Pre}(a) \cup \text{Pre}(b))\right) && \text{otherwise}\end{aligned}$$

Using Equation 5 we can compute the probability of the conjunction $\Pr\left(\bigwedge(\text{Pre}(a) \cup \text{Pre}(b))\right)$. By our definition of interaction, Equation 3, we can then compute the interaction between two actions a and b.

As an example, consider the plan graph in Figure 1 again.

$$\begin{aligned}\Pr(a \wedge b) &= \Pr\left(\bigwedge(\text{Pre}(a) \cup \text{Pre}(b))\right)\\ &= \Pr(p \wedge q \wedge r)\\ &= \Pr(p)\Pr(q)\Pr(r)I(p,q)I(q,r)I(p,r)\\ &= .8(.5)(.4) = .16\end{aligned}$$

assuming that the interactions between the preconditions are all one. The interaction between a and b is therefore:

$$I(a,b) = \frac{\Pr(a \wedge b)}{\Pr(a)\Pr(b)} = \frac{.16}{.4(.2)} = 2$$

Computing Effect Probabilities and Interaction

Given the tools we have developed so far, it is relatively straightforward to compute the probability of an individual action effect. Let Φ_i be an outcome of action a with weight w_i, and let $\phi_{ij} = \rho_{ij} \rightarrow \varepsilon_{ij}$ be a conditional effect in Φ_i. If the effect is unconditional – that is the antecedent ρ_{ij} is empty – then:

$$\Pr(\varepsilon_{ij}) = w_i \Pr(a)$$

However, if the antecedent ρ_{ij} is not empty, there is the possibility of interaction (positive or negative) between the preconditions of a and the antecedent ρ_{ij}. As a result, to do the computation right we have to compute the probability of the conjunction of the preconditions and the antecedent:

$$\Pr(\varepsilon_{ij}) = w_i \Pr\left(\bigwedge(\text{Pre}(a) \cup \rho_{ij})\right)$$

For convenience, we will refer to the weight w_i associated with an effect ε_{ij} as $w(\varepsilon_{ij})$. We will also refer to the union of the action preconditions and the antecedent ρ_{ij} for an effect ε_{ij} as simply the *condition* of ε_{ij} and denote it $\text{Cond}(\varepsilon_{ij})$. For an effect ε, the above expression then becomes simply:

$$\Pr(\varepsilon) = w(\varepsilon)\Pr\left(\bigwedge \text{Cond}(\varepsilon))\right)$$

As with actions, we can compute the probability of the conjunction of $\text{Cond}(\varepsilon)$ using the approximation in Equation 5.

We can also compute the interaction between two different effects just as we did with actions. For two effects, e and f we have:

$$\Pr(e \wedge f) = w(e)w(f)\Pr\left(\bigwedge \text{Cond}(e) \cup \text{Cond}(f)\right) \quad (6)$$

As before, the probability of the conjunction of Cond$(e) \cup$ Cond(f) using the approximation in Equation 5. By our definition of interaction, Equation 3, we can then compute the interaction between the two effects e and f.

As an example, consider the two unconditional effects e and h from Figure 1. Since both these effects are unconditional, Cond(e) and Cond(h) are just the preconditions of a and b respectively. As a result:

$$\begin{aligned}\Pr(e \wedge h) &= w(e)w(h) \Pr\left(\bigwedge \text{Cond}(e) \cup \text{Cond}(h)\right)\\ &= w(e)w(h) \Pr(p \wedge q \wedge r)\\ &= 1(.5)(.8)(.5)(.4)\\ &= .08\end{aligned}$$

since p, q and r were assumed to be independent. Using this, we get:

$$I(e,h) = \frac{\Pr(e \wedge h)}{\Pr(e)\Pr(h)} = \frac{.08}{.4(.1)} = 2$$

Intuitively, the fact that $I(e,h) > 1)$ means that there is some degree of synergy between the effects e and h. In other words, establishing them both is not as hard as might be expected from their individual probabilities. This is because the actions for achieving them share a precondition.

Note that Equation 6 for $\Pr(e \wedge f)$ applies whether the effects e and f are from the same or different actions. In the case where they are effects of the same action, there will be overlap of the action preconditions between Cond(e) and Cond(f). However, the antecedents of the conditional effects may be quite different, and there can be interaction (positive or negative) between literals in those antecedents, which will be captured by the probability calculation in Equation 6.

Computing Proposition Probabilities

Computing the probability for a proposition is complicated by the fact that there may be many actions with effects that produce the proposition, and we are not limited to using only one such action or effect. For example, if two action effects e and f both produce proposition p with probability .5, then we may be able to increase our chances of achieving p by performing both of them. However, whether or not this is a good idea depends upon the interaction between the two effects. If the effects are independent or synergistic, then it is advantageous. If the two effects are completely mutex ($I(e,f) = 0$), then it is not a good idea. If there is some degree of mutual exclusion between the actions (i.e. $0 < I(e,f) < 1$) then the decision depends on the specific probability and interaction numbers.

Suppose we choose a particular set of effects $E = \{e_1, \ldots, e_k\}$ that produce a particular proposition p. Intuitively, it would seem that the probability that one of these effects would yield p is:

$$\Pr(e_1 \vee \ldots \vee e_k)$$

Unfortunately, this isn't quite right. By choosing a particular set of effects to try to achieve p, we are committing to (trying to) establish the conditions for all of those effects, which means establishing both the action preconditions and the antecedents of each of the conditional effects. There may be interaction between those conditions (positive or negative) that increases or decreases our chances for each of the effects. The above expression essentially assumes that all of the effects are independent of each other.

In this case, the correct expression for $\Pr(p)$ using a set of effects E is both complicated and difficult to compute. Essentially we have to consider the probability table of all possible assignments to the conditions for the effects E, and multiply the probability of each assignment by the probability that the effects enabled by that assignment will produce p. Let $\mathcal{T}(E)$ be the set of all possible $2^{|\text{Cond}(E)|}$ truth assignments to the conditions in Cond(E). Formally we get:

$$\Pr(p_E) = \sum_{\tau \in \mathcal{T}(E)} \Pr(\tau)\Pr(p|\tau) \tag{7}$$

where $\Pr(p_E)$ refers to the probability of p given that we are using the effects E to achieve p.

As an example, consider the calculation of the probability for the proposition t in Figure 1 assuming that we are using the two unconditional effects f and g from actions a and b. The set of conditions for these effects is just the union of the preconditions for a and b which is $\{p, q, r\}$. There are eight possible truth assignments to this set, but only three of them permit at least one of the actions:

$$\begin{aligned} p \wedge q \wedge \neg r \quad & \text{permits } f \text{ but not } g\\ \neg p \wedge q \wedge r \quad & \text{permits } g \text{ but not } f\\ p \wedge q \wedge r \quad & \text{permits both } f \text{ and } g\end{aligned}$$

The probabilities for these truth assignments are:

$$\begin{aligned}\Pr(p \wedge q \wedge \neg r) &= .8(.5)(.6) = .24\\ \Pr(\neg p \wedge q \wedge r) &= .2(.5)(.4) = .04\\ \Pr(p \wedge q \wedge r) &= .8(.5)(.4) = .16\end{aligned}$$

The probability for t using both actions is therefore:

$$\Pr(t) = .24(.5) + .04(1) + .16(.5 + 1 - .5(1)) = .32$$

This calculation was fairly simple because we were only dealing with three propositions p, q and r and they were independent. More generally, however, an expression like $\Pr(p \wedge q \wedge \neg r)$ is problematic when r is not independent of the other two propositions, since we do not have interaction information for the negated proposition. There are a number of approximations that one can use to compute such probabilities. For our purposes, we assume that two propositions are independent if interaction information is not available. Thus, in this case we make the assumption that:

$$\Pr(p \wedge q \wedge \neg r) = \Pr(p \wedge q)\Pr(\neg r)$$

We now return to the problem of computing the probability for a proposition p. In theory we could consider each possible subset E' of effects E that match the proposition p and compute the maximum:

$$\max_{E' \subseteq E} \Pr(p_{E'}) \tag{8}$$

and use Equation 7 to expand and compute $\Pr(p_{E'})$. Unfortunately, when there are many effects that can produce a proposition this maximization is likely to be quite expensive, because 1) we would need to consider all possible subsets of the set of effects, and 2) in Equation 7 we would have to consider all possible truth assignments to the conditions for each set of effects. As a result, some approximation is in order. One possibility is a greedy approach that adds effects one at a time, as long as they still increase the probability. More precisely:

1. Let E be the set of effects matching p
2. let E_0 be the empty set of effects, let $P_0 = 0$
3. let e be an effect in E not already in E_{i-1}, and let $P^* = \Pr(p_{e \cup E_{i-1}})$. If

   e maximizes P^*

   and

   $P^* > P_{i-1}$

   then set

   $E_i = e \cup E_{i-1}$
   $P_i = P^*$

Using this procedure the final set P_i will be a lower bound on:

$$\max_{E' \subseteq E} \Pr(p_{E'})$$

Even this approximation is somewhat expensive to compute, because it requires repeated computation of $\Pr(p_{E'})$ at each stage using equation 7. A different approximation that avoids much of this computation is to construct all maximal subsets E' of the effects in E such that there is no pair of effects e and f in E' with $I(e, f) < 1$ (no interference). We then compute or estimate $\Pr(p_{E'})$ for each such subset and choose the maximum. This approximation has the advantage that we must only calculate $\Pr(p_{E'})$ for a relatively small number of sets.

Computing Interaction Between Propositions

Finally, we consider the probability for a pair of propositions p and q which will allow us to compute the interaction between the propositions. As with a single proposition, this calculation is complicated because we want to find the best possible set of effects for establishing the conjunction. If we let E be the set of effects matching proposition p, and F be the set of effects matching proposition q, then what we are after is:

$$\Pr(p \wedge q) \;=\; \max_{\substack{E' \subseteq E \\ F' \subseteq F}} \Pr(p_{E'} \wedge q_{F'}) \tag{9}$$

In order to compute $\Pr(p_{E'} \wedge q_{F'}$ we must again resort to considering all possible truth assignments for the union of the conditions for E' and F' as we did in Equation 7:

$$\Pr(p_{E'} \wedge q_{F'}) = \sum_{\tau \in \mathcal{T}(E' \cup F')} \Pr(\tau) \Pr(p \wedge q | \tau) \tag{10}$$

Returning to our example, consider the calculation of the probability for the pair of propositions s and t in Figure 1. Proposition s has only one supporting effect e, but t has two supporting effects. For illustration, assume that we are using both the effects f and g in order to increase the probability of t. The set of conditions for all three effects $\{e, f, g\}$ is just the union of the preconditions for a and b which is $\{p, q, r\}$. Again there are eight possible truth assignments to this set, but only two of them permit the effect e:

$p \wedge q \wedge \neg r$ permits e and f but not g
$p \wedge q \wedge r$ permits e, f, and g

As before, the probabilities for these truth assignments are:

$$\Pr(p \wedge q \wedge \neg r) \;=\; .8(.5)(.6) = .24$$
$$\Pr(p \wedge q \wedge r) \;=\; .8(.5)(.4) = .16$$

The probability for s and t using effects e, f, and g is therefore:

$$\Pr(g) = .24(.5) + .16(.5 + 1 - .5(1)) = .28$$

For our example, the maximization in equation 9 is trivial because the effects f and g do not interfere. As a result, using both will yield higher probability and we get the final result that $\Pr(s \wedge t) = .28$

More generally this maximization could be costly to compute, since it involves computing a complex expression for all subsets of effects in E and F. To approximate this, we could use either the greedy strategy developed in the previous section, or the strategy of finding maximal non-interfering effect subsets.

Given $\Pr(s \wedge t)$ and the individual probabilities $\Pr(s)$ and $\Pr(t)$ we can compute $I(s, t)$ from the definition in Equation 3. For our example, we get

$$I(a, b) = \frac{\Pr(s \wedge t)}{\Pr(s) \Pr(t)} = \frac{.28}{.4(.36)} \approx 2.19$$

Thus we see that there is synergy between s and t, as we would expect, since action a can produce them both.

Using Probability Estimates

Probability estimates in a plan graph can be useful for guiding both progression and regression planners. Consider a regression planner, such as that discussed in (Bryce, Kambhampati, & Smith 2006a). Such a planner works backwards from the goals. At any given stage there is a partial plan (plan suffix) along with a set of open conditions that still need to be achieved. Using the plan graph probability and interaction estimates, the planner can estimate the probability of achieving the conjunction of the open conditions. Given the current plan suffix the planner can then compute an estimate of the probability that the goals will be achieved. If that probability is too low, the planner can abandon the candidate plan and pursue others that appear more promising.

Once the planner has chosen to pursue a candidate plan, it must then choose the open condition to work on. Here the probability estimates can guide the planner to work on the most difficult open condition. Once the open condition is chosen, probability estimates are useful for guiding the planner towards the best set of actions for achieving the open condition.

For a progression planner probability estimates can be used in a similar fashion: 1) to estimate the probability that a given state will lead to the goals, and 2) to choose the action most likely to lead to the goals. The disadvantage in progression is that one must recompute (or update) the planning graph for each newly generated state.

A number of current planning systems compute relaxed plans, and use these as distance estimates to guide planning search. In this case, the probability estimates we have developed are useful for extracting better relaxed plans. Basically, the probability estimates are used in the same way as in regression search, to help choose the most appropriate actions in the greedy construction of the relaxed plan. This is the approach we have taken in our preliminary implementation.

Results

We have developed a preliminary implementation of the technique presented above. Interaction and probability information is computed using the above methods. This information is then used to guide construction of a relaxed plan, which is used to guide the POND heuristic search planner (Bryce, Kambhampati, & Smith 2006a) in a manner similar to that described in (Bryce, Kambhampati, & Smith 2006b). The planner is implemented in C and uses several existing technologies. It employs the PPDDL parser (Younes & Littman 2004) for input, and the IPP planning graph construction code (Koehler *et al.* 1997). Because the implementation and debugging is still not complete, we have so far only tested the ideas on the small domains Sandcastle-67 and Slippery gripper. Figures 2 and 3 show some early results for time, plan length, and node expansion for the sandcastle-67 and slippery Gripper domains respectively. The plots compare 4 different planners:

- CPlan (Hyafil & Bacchus 2004)
- McLug-16 (Bryce, Kambhampati, & Smith 2006b), the POND planner using Monte Carlo Simulation on plan graphs
- pr-rp, relaxed plan construction using simple plan graph probability information computed using independence assumptions
- int-rp, relaxed plan construction using probability and interaction information.

The other two entries (pr-rp-mx and corr-rp-mx) represent variants that are not fully debugged and should therefore be regarded as suspect.

Generally, performance of the four methods is similar on these simple domains. Plans are somewhat longer for pr-rp and corr-rp because the objective for these planners is to maximize probability rather than minimize the number of actions. There is some indication that corr-rp is showing less growth in time and number of node expansions as the probability threshold becomes high, but additional experiments are needed to confirm this and examine this behavior more closely.

Related Work

A number of authors have made use of plan graphs to try to speed up probabilistic planning. Boutilier, Brafman and Geib 1998 examine the impact of reachability analysis and n-ary mutex relationships on the size of the state space for MDPs. PGraphplan (Blum & Langford 1999) also makes use of a plan graph in probabilistic planning, primarily for the purpose of reachability analysis. In recent work, Little and Thiebaux (Little & Thiébaux 2006) also use a plan graph for reachability analysis, but introduce more powerful mutual exclusion reasoning for handling concurrent probabilistic planning problems. Probapop (Onder, Whelan, & Li 2006) uses a relaxed planing graph to compute distance estimates and guide a partial order planner in solving conformant probabilistic planning problems. None of these systems attempt to compute probability estimates in a planning graph.

Prottle (Little, Aberdeen, & Thiébaux 2005) computes lower bound probability estimates of reaching the goal from a given state by back-propagating probability on the plan graph. Given a progression search state, Prottle identifies the relevant propositions in the plan graph and takes the product of their back-propagated probabilities. In taking the product of the probabilities, Prottle assumes full independence between subgoals, leading to a weak lower bound on goal probability. In comparison with our technique, we propagate probability forward using interaction instead of assuming full independence.

Discussion and Conclusions

We have introduced a continuous generalization of the notion of mutex, which we call *interaction*. We showed how such a notion could be used to improve the computation of probability estimates within a plan graph. Our implementation of this technique is still preliminary and it is much too early to draw any significant conclusions about the practicality or efficacy of these computations for problems of any size. In addition to finishing our implementation and doing more significant testing, there are a number of issues that we wish to explore:

Interaction vs Relaxed Plans The approach of keeping interaction information is different from the method of using a relaxed plan to estimate probability in an important way: relaxed plans are constructed greedily, so a relaxed plan to achieve $p \wedge q$ would normally choose the best way to achieve p and the best way to achieve q independently. This will not always lead to the best plan for achieving the conjunction. Interaction information can be used to guide (relaxed) plan selection and would presumably give better relaxed plans. This is the approach we have taken in our preliminary implementation. Of course there is always a trade-off between heuristic quality and computation time, and this is something we intend to investigate further.

Admissibility Although probability estimates computed using interaction information should be more informative, they are not admissible. The primary reason for this is that keeping only binary interaction information, and approximating the probability of a conjunction using only binary interaction information can both underestimate and overestimate the probability of the conjunction. Note, however, that the usual approach of estimating probability by assuming independence is also not admissible for the same reason. Similarly, relaxed plans do not provide an admissible heuristic -

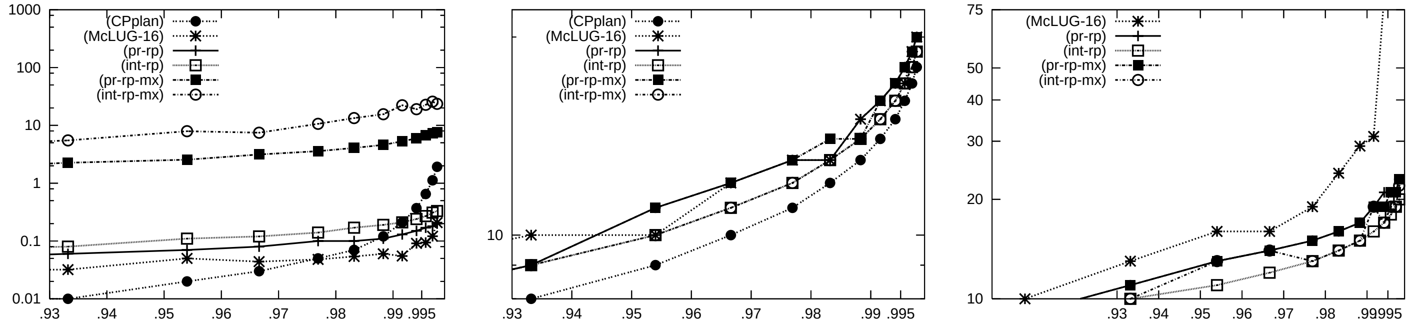

Figure 2: *Run times (s), Plan lengths, and Expanded Nodes vs. probability threshold for sandcastle-67*

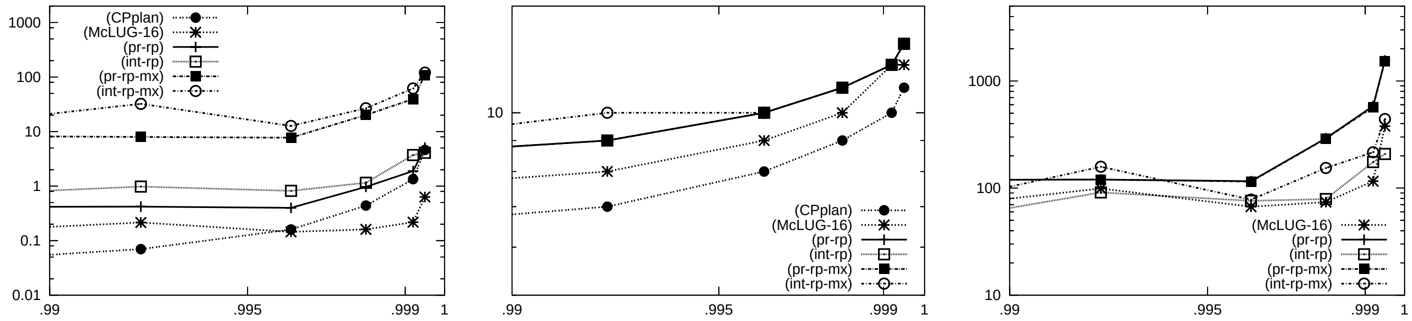

Figure 3: *Run times (s), Plan lengths, and Expanded Nodes vs. probability threshold for slippery gripper*

they can underestimate probability because the relaxed plan may not take full advantage of synergy between actions in the domain. It is possible to construct an admissible heuristic for probability by taking:

- the probability of a conjunction to be the minimum probability of the conjuncts,
- the probability of a proposition as the sum of all the probabilities of the producing effects.

However, this heuristic is very weak and not likely to be very effective. It is not yet clear whether we can construct a stronger admissible heuristic using interaction.

Interaction in the Initial State The mechanism we have described easily admits the use of interaction information between propositions in the initial state. That information would be treated in the same was as at any other level in the plan graph. Thus, if the initial state has $\Pr(p \wedge q) = .5$ and $\Pr(\neg p \wedge \neg q) = .5$ we could represent this as $\Pr(p) = \Pr(q) = \Pr(\neg p) = \Pr(\neg q) = .5$ and $I(p, q) = I(\neg p, \neg q) = \frac{.5}{.5 .5} = 2$. The limitation of this approach is that binary interaction can only approximate joint probability information for conjunctions larger than two.

Bayesian Networks There are a number of similarities between techniques we have used here, and methods used in Bayesian Networks. We speculate that the calculation of probability information for individual actions and pairs of actions could be modeled using a simple Bayes net with nodes for the preconditions and actions, arcs between the preconditions and corresponding actions and arcs between pairs of preconditions that are dependent (interaction not equal to one). These later arcs would be labeled with the conditional probability corresponding to the interaction. It would be necessary to structure the network carefully to avoid cycles among the preconditions. The more complex calculations for propositions would require influence diagrams with choice nodes for each of the establishing effects. There doesn't seem to be any particular advantage to doing this, however. Solution of this influence diagram would require investigating all possible sets of the decisions, which corresponds to the unwieldy maximization over all subsets of establishing effects. It also seems unlikely that approximate techniques for solving influence diagrams would help - they would still likely require the investigation of all possible action choices, and would produce an approximate joint probability distribution. Instead, the approach that we take does not attempt to compute this joint probability distribution, and works on one Plan Graph level at a time.

Cost Computation in Classical Planning The idea that we have explored here – a continuous generalization of mutex – is not strictly limited to probabilistic planning. A similar notion of the "interaction" between two proposiitons or two actions could be used in classical planning to improve plan graph estimates of cost or resource usage. To do this we could define "interaction" as:

$$I(x, y) = Cost(x \wedge y) - (Cost(x) + Cost(y))$$

$$
\begin{aligned}
&= Cost(y|x) - Cost(y)\\
&= Cost(x|y) - Cost(x)
\end{aligned}
$$

For this definition, positive interaction means that there is some conflict between two propositions, actions or effects, and that it is more expensive to achieve the conjunction than to achieve them separately. An interaction of plus infinity corresponds to mutex. Negative interaction corresponds to synergy between the propositions, meaning that achieving them together is easier than achieving them independently. An interaction of zero corresponds to independence. Essentially, this can be seen as the negative logarithm of the definition for probabilistic interaction given in Equation 3.

The computation of cost interaction for actions, effects and propositions is very similar to what we have described above. The primary difference is that the computations for propositions are significantly simpler because there is no need to maximize over all subsets of possible effects that give rise to a proposition. Although we have worked out the equations and propagation rules for this notion of interaction, we have not yet implemented or tested this idea. We intend to investigate this in the near future.

Acknowledgements Thanks to Ronen Brafman, Nicolas Meuleau, Martha Pollack, Sailesh Ramakrishnan, and Rao Kamhampati for discussions on this subject and on early versions of these ideas. This work has been supported by the NASA IS-NRA program.

References

Blum, A., and Langford, J. 1999. Probabilistic planning in the graphplan framework. In *Proceedings of ECP'99*.

Bonet, B., and Geffner, H. 2005. mGPT: A probabilistic planner based on heuristic search. *Journal of Artificial Intelligence Research* 24:933–944.

Boutilier, C.; Brafman, R.; and Geib, C. 1998. Structured reachability analysis for Markov Decision Processes. In *Proceedings of UAI'98*.

Bryce, D.; Kambhampati, S.; and Smith, D. 2006a. Planning graph heuristics for belief space search. *Journal of Artificial Intelligence Research*. (To appear).

Bryce, D.; Kambhampati, S.; and Smith, D. 2006b. Sequential monte carlo in probabilistic planning reachability heuristics. In *Proceedings of ICAPS'06*.

Hyafil, N., and Bacchus, F. 2004. Utilizing structured representations and CSPs in conformant probabilistic planning. In *Proceedings of ECAI'04*.

Koehler, J.; Nebel, B.; Hoffmann, J.; and Dimopoulos, Y. 1997. Extending planning graphs to an adl subset. In *Proceedings of ECP'97*.

Little, I., and Thiébaux, S. 2006. Concurrent probabilistic planning in the Graphplan framework. In *Proceedings of ICAPS'06*.

Little, I.; Aberdeen, D.; and Thiébaux, S. 2005. Prottle: A probabilistic temporal planner. In *Proceedings of AAAI'05*.

McDermott, D. 1998. PDDL-the planning domain definition language. Technical report, Available at: www.cs.yale.edu/homes/dvm.

Onder, N.; Whelan, G.; and Li, L. 2006. Engineering a conformant probabilistic planner. *Journal of Artificial Intelligence Research* 25:1–15.

Rintanen, J. 2003. Expressive equivalence of formalisms for planning with sensing. In *Proceedings of ICAPS'03*.

Younes, H., and Littman, M. 2004. PPDDL1.0: An extension to PDDL for expressing planning domains with probabilistic effects. Technical report, CMU-CS-04-167, Carnegie Mellon University.

Younes, H.; Littman, M.; Weissman, D.; and Asmuth, J. 2005. The first probabilistic track of the International Planning Competition. *Journal of Artificial Intelligence Research* 24:851–887.