# Search Control of Plan Generation in Decision-Theoretic Planners

**Richard Goodwin**
rgoodwin@watson.ibm.com
T. J. Watson Research Center, IBM
Route 134, P. O. Box 218,
Yorktown Heights, NY. USA 10598
Phone: (914) 945-2651 Fax: (914) 945-4527

**Reid Simmons**
reids@cs.cmu.edu
School of Computer Science
Carnegie Mellon University
Pittsburgh, PA, USA 15213-3890
Phone: (412) 268-2126 Fax: (412) 268-5576

## Abstract

This paper addresses the search control problem of selecting which plan to refine next for decision-theoretic planners, a choice point common to the decision theoretic planners created to date. Such planners can make use of a utility function to calculate bounds on the expected utility of an abstract plan. Three strategies for using these bounds to select the next plan to refine have been proposed in the literature. We examine the rationale for each strategy and prove that the optimistic strategy of always selecting a plan with the highest upper-bound on expected utility expands the fewest number of plans, when looking for all plans with the highest expected utility. When looking for a single plan with the highest expected utility, we prove that the optimistic strategy has the best possible worst case performance and that other strategies can fail to terminate. To demonstrate the effect of plan selection strategies on performance, we give results using the DRIPS planner that show that the optimistic strategy can produce exponential improvements in time and space.

## Introduction

In this paper, we address the search control problem of selecting which abstract plan to refine next for decision-theoretic planners. This choice point is common to the decision theoretic planners created to date including Haddawy's DRIPS planner (Haddawy & Doan 1994) and Williamson's Pyrrhus planner (Williamson & Hanks 1994) and our own Xavier route planner (Goodwin 1996a). In Kambhampati's unified framework for classical planning, this choice point is analogous to selecting which plan set component to work on when split and prune search is used (Kambhampati & Srivastava 1996). The major difference from classical planning is that decision-theoretic planners can calculate bounds on the expected utility of abstract plans using a utility function. These bounds can then be used to guide search.

We begin by describing the search control decision to be made and give examples from three planners. We then present three strategies for making the choice: selecting the plan with the highest upper bound on expected utility, selecting the plan with the highest lower bound on expected utility and selecting the plan with the lowest upper bound on expected utility. We show that selecting the plan with the highest upper bound on expected utility is optimal, in terms of the number of plans that must be expanded. We also show that this strategy has the best possible worst case performance for any strategy that uses bounds on expected utility. The DRIPS planner is then used to demonstrate how the theoretical results apply to a hierarchical refinement planner and infinite plan spaces using a large medical domain.

## Plan Selection Problem

As with most classical planners, the decision-theoretic planners created to date can be cast as refinement planning using split and prune search[1]. Conceptually, a decision-theoretic planner begins with a single abstract plan that represents all possible plans and refines it until a plan with the highest expected utility is found. The expected utility for an abstract or completely refined plan is calculated using a utility model and/or a probability model. Abstract plans have a range of expected utility, since they represent sets of plans.

A pseudo code outline of a generic decision-theoretic planning algorithm is given in figure 1. It includes two choice points: selecting a plan from the set of partial plans at the frontier of the search, the frontier set, and selecting a refinement to apply. In this paper, we address the first choice.[2] We argue for acceptance of our generic algorithm as a general model for decision-theoretic planners by casting three diverse planners in this framework.

DRIPS is a decision-theoretic planner with probabilistic actions that uses an abstraction hierarchy, similar to an abstract task network, to represent actions. A utility function is used to calculate both an upper and lower bound on expected utility for each abstract plan. The top node in the hierarchy represents the universal plan. The refinement selection step corresponds

[1] See (Kambhampati & Srivastava 1996) for a detailed analysis mapping classical AI planners to a refinement model.

[2] The second choice is addressed in (Goodwin 1996b).

| Strategy | Select Plan with | Plan Selected | Argument |
|---|---|---|---|
| Conservative | $max(\underline{EU})$ | plan 1 | Improve the worst case by raising the best lower bound to prune other plans. |
| Prune | $min(\overline{EU})$ | plan 2 | Try to prune nearly pruned plans. |
| Optimistic | $max(\overline{EU})$ | plan 3 | Focus on potentially optimal plans. |
| Reckless | $min(\underline{EU})$ | plan 4 | No rationale. |

Table 1: Plans selection strategies.

```
Frontier Set fs = UniveralPlan;
LOOP
  // If the frontier set contains a fully
  // refined plan with maximal EU, then stop.
  IF ∃p∈FS fullyRefined(p) ∧ ∀p'∈FS EU(p) ≥ EU(p')
    return p;
  ELSE
    // Select a plan to refine, choice point 1.
    Plan p = SELECT(fs);
    // Select a refinement procedure for p, choice point 2.
    Refinement r = SELECT(refinement(p));
    // Refine the selected plan.
    fs = fs - p + r(p);
    // Remove dominated plans.
    fs = PRUNE(fs);
  END-IF
END-LOOP
```

Figure 1: Pseudo code for decision-theoretic planner.

to selecting an abstract action to refine. Refining an action produces one plan for each possible refinement of the selected action. The resulting plans are added to the set of plans at the frontier of the search and dominated plans are pruned. One plan dominates another if its lower bound on expected utility is greater than the upper bound on expected utility for the other plan. The plan selection step corresponds to a selecting an abstract plan from the frontier set to refine.

**Pyrrhus** is a decision-theoretic planner based on a classical partial order planner, SNLP. A utility model is added to allow the planner to calculate an upper bound on utility for a partially ordered plan by ignoring resource and precondition conflicts. The initial universal plan is the empty plan. The possible plan refinements are the refinements for partial order planners, including action addition, promotion and demotion. The plan selection step corresponds to selection of a partial order plan from the frontier set to refine.

**The Xavier Route Planner** is a domain specific planner that uses a probabilistic model of robot navigation to plan routes for a mobile robot using an abstract topological map of a building. The planner uses an iterative version of A* search that generates progressively longer routes to the goal. The length of the route can be used to calculate a lower bound on travel time, equivalent to an upper bound on utility. However, the expected travel time is usually longer since the robot can miss turns and doorways may be blocked. The universal plan is the set of all possible routes and is represented only implicitly. The possible refinements include generating another base route using the iterative A* search, refining a partial plan by expanding an topological node and planning for a contingency, such as a locked door. These refinements flesh out the plan and narrow the bounds on expected utility. As with the other two planners, the plan selection step is to select a plan from the frontier set.

## Strategies

In this section, we present four strategies for selecting plans for refinement when plans are characterized by ranges of expected utility. We discuss the rationale used to support each of these strategies.

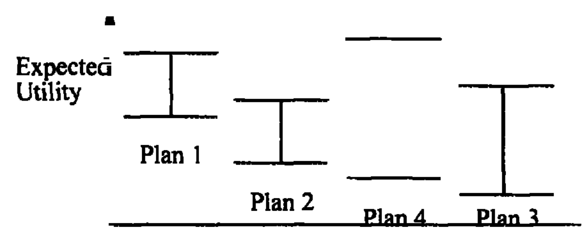


Figure 2: Plans selected by each of the four strategies.

Strategies for selecting a plan to refine focus effort either on pruning the search space or on exploring parts of the search space likely to contain plans with high expected utility. A conservative approach selects the plan with the greatest lower bound on expected utility, $max(\underline{EU})$, plan 1 in figure 2. There are two lines of argument that support this choice: plan 1 gives the best guarantee on the minimum utility and this bound has to be raised the lease to prune other plans. By raising the least lower bound on EU, it minimizes the maximum regret, or difference between the expected utility that it achieves and the highest expected utility that is possible[3]. This argument is more applicable when trying to find a nearly optimal solution with minimal effort. The problem with the pruning argument is that

[3]See (Loomes & Sugden 1982; Loomes 1987) for information on regret theory, an alternative to maximizing expected utility.

the effort expended to refine plan 1 is not necessarily needed to find the optimal solution. This strategy was originally used in the DRIPS planner and resulted in poor performance (see empirical results section).

A related pruning strategy attempts to prune "nearly dominated" plans, like plan 2 in figure 2. It selects plans with the minimal upper bound on expected utility, $min(\overline{EU})$. Such plans are almost dominated and a small amount of work may lower the upper bound on expected utility enough to allow the plan to be pruned. The problem is that this strategy expends effort on plans that are likely sub-optimal. It also suffers when there are many nearly pruned plans equivalent to plan 2, since this strategy must expend effort to prune each one.

A third strategy takes an optimistic view and works only on plans with the greatest upper bound on expected utility, $max(\overline{EU})$. It concentrates effort on the potentially optimal plans rather than trying to prune the search space. Selecting plan 4 in figure 2 for refinement can lead to plans that are less optimal than either plan 1 or plan 2. However, using the optimistic strategy when looking for a plan with the highest expected utility requires the fewest plan expansions. The next section presents the proof of this result and the bounds on worst case performance.

The three strategies and their arguments are summarized in table 1. A fourth strategy, without rationale, selects the potentially worst plan, the one with the minimum lower bound on expected utility, $min(\underline{EU})$. It is included for completeness.

## Proofs of Optimality

This section presents and proves two theorems on the optimality of the optimistic plan selection strategy. We show that, when looking for all plans with the highest expected utility, the optimistic strategy is refines the fewest plans. When looking for a single plan with the highest expected utility, we prove a bounds on the worst case performance of the optimistic strategy and prove that this is the best possible bounds for any strategy. Finally, we show that strategies that focus on pruning the search tree can be arbitrarily worse, and in some cases fail to terminate.

### Optimal Strategy for Finding All Optimal Plans.

**Theorem 1** *Selecting, for expansion, a plan with the greatest upper bound on expected utility expands the fewest plans when finding all plans with the highest expected utility.*

**Proof:** The proof relies on showing that every abstract plan that, at some point in the search, has a maximal upper bound on expected utility must be refined. Since the method expands only the plans that are required to be expanded, the method expands the fewest number of plans. The proof that such plans need to be expanded can be shown by contradiction.

1. The optimistic strategy, $\mathcal{O}$, refines plans in a monotonically non-increasing order of the upper bound on expected utility, $\overline{EU}$.
   - The abstract plans at the frontier of the search are sorted by $\overline{EU}$, and the plans with the highest $\overline{EU}$ are refined first. [By definition of the optimistic strategy.]
   - No node beyond the frontier of the search has an $\overline{EU}$ higher than the next node to be refined.
     Any node beyond the frontier of the search is a descendent of one of the nodes at the frontier.
     All descendents of a node have $\overline{EU}$ less than or equal to the $\overline{EU}$ for all ancestors. (Otherwise, the bounds are not valid.)
   - Therefore, no unrefined plan can have a higher $\overline{EU}$ than the next plan to be refined and plans are refined in monotonically non-increasing order of $\overline{EU}$.
2. All plans with $\overline{EU} \geq \overline{EU}_{optimal}$ must be expanded in order to show that all optimal plans have been found.
   - Suppose it were possible to determine that there were no more optimal plans without expanding an abstract plan with an $\overline{EU} \geq \overline{EU}_{optimal}$. This is a contradiction since the unexpanded plan could include an optimal refinement.
3. Any plans with $\overline{EU} < \overline{EU}_{optimal}$ are pruned by $\mathcal{O}$ when a plan with $EU_{optimal}$ is found. This happens before any plans with $\overline{EU} < \overline{EU}_{optimal}$ are refined since plans are refined in monotonically non-increasing order of $\overline{EU}$.
4. Since the $\mathcal{O}$ strategy refines only the set of plans that are required to be refined, it refines the fewest number of plans.

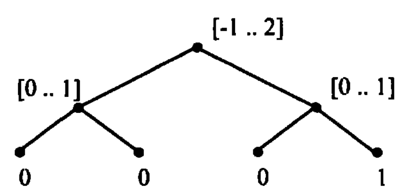


Figure 3: If the right sub-tree is refined, an optimal solution is found without refining the left sub-tree. If the left sub-tree is refined first, then the right sub-tree must also be refined.

Generally, it is sufficient to find a single plan with the highest expected utility, since all such plans are equally valued. When looking for a single optimal plan, the optimistic strategy has problems with the boundary cases where abstract plans have an upper bound on expected utility equal to $\overline{EU}_{optimal}$. Figure 3 shows a search tree where the two children of the root node are abstract plans with the same upper bound on expected utility. One of the abstract plans has a refinement with the same upper bound on expected utility, but the other one does not. By guessing correctly, only the right sub-tree needs to be expanded to find a plan with the highest expected utility and prove that it is optimal. Guessing incorrectly requires both sub-trees to be expanded.

Further analysis shows that this problem arises only when a plan with utility equal to $EU_{optimal}$ has a parent with the same upper bound on expected utility. If the right sub-tree in figure 3 had an upper bound that was greater than 1, then the optimistic strategy would be optimal, independent of the value of the upper bound on the left sub-tree. If the left sub-tree had an upper bound greater than 1, both sub-trees would have to be refined before a planner could conclude that it had an optimal plan. If the left sub-tree had an upper bound equal to or less than 1, the right sub-tree would be expanded first. The best fully-refined plan would be found and the left sub-tree would not have to be expanded.

There are two questions we need to address: Are the upper bounds on expected utility ever tight for ancestors of a plan with the highest expected utility and how big a difference can lucky guesses make? We look first at the question of tight bounds and argue that, for some domains, the expected utility bounds on partially refined plans are never tight. For such domains, the optimistic strategy is optimal even when looking for a single plan with the highest expected utility. Without this restriction, we can still show that the difference between the optimistic strategy and lucky guessing is bounded. Such is not the case for the other plan selection strategies, and we show search trees where these strategies can fail to terminate, when the optimistic strategy would terminate with an optimal plan.

**Optimal Strategy in Restricted Domains** In some domains, the upper bound on expected utility for abstract plans is never tight, and a fully-refined plan will always have an expected utility that is less than the upper bound on its abstract parent. This situation arises in domains where refining an abstract plan resolves a tradeoff and the upper bound on the abstract plan was calculated by assuming an optimal, but unachievable resolution of the tradeoff. The expected utility bound on the parent can, however, be arbitrarily close to the real value. Note that if you could show that the bound was always lowered by a minimum amount, the original bound could just be lowered by this amount. In such domains, the optimistic strategy is optimal.

**Theorem 2** *If the upper bound on expected utility for abstract plans is never the same as the bound on the expected utility of a completely expanded plan, but can be arbitrarily close, then selecting the plan with the greatest upper bound on expected utility expands the fewest number of plans when finding a single plan with the highest expected utility.*

**Proof:** The idea behind the restriction is to remove the possibility that a lucky guess can be confirmed to end the search. If there is a tie in the upper bounds on two abstract plans, there is no way to confirm that you have a plan with the highest expected utility by refining only one of them. Refining one of the plans will always give a lower expected utility for the fully-refined plan and the other abstract plan may contain a refinement that has slightly higher expected utility. The proof proceeds by showing that the optimistic strategy refines only the plans that are required to be refined in order to find the optimal solution.

1. The optimistic strategy refines plans in a monotonically non-increasing order of the upper bound on expected utility, $\overline{EU}$.
   - From the proof of theorem 1
2. The parent of an fully-refined plan with the highest expected utility will be refined before any abstract plan with $\overline{EU} = \overline{EU}_{optimal}$
   - The parent plan of an optimal plan must have $\overline{EU} > \overline{EU}_{optimal}$, since the bounds are not tight.
   - Since plans are refined in monotonically non-increasing order of $\overline{EU}$ the parent plan must be refined before plans with $\overline{EU} = \overline{EU}_{optimal}$.
     This ensures that the optimal, fully-refined plan is in the frontier set before any plans with $\overline{EU} = \overline{EU}_{optimal}$ are refined.
3. All plans with $\overline{EU} > \overline{EU}_{optimal}$ must be expanded in order to show that the optimal plan has been found.
   - Suppose it were possible to determine the optimal plan without expanding an abstract plan with an $\overline{EU} > \overline{EU}_{optimal}$. Let the difference between the unrefined abstract plan and the upper bound on expected utility of the optimal plan be $\delta EU$. To conclude that the optimal plan has been found is to conclude that refining the abstract plan would reduce the upper bound on expected utility by at least $\delta EU$. But this contradicts the assumption that the bounds can be arbitrarily tight.
4. When the optimal fully-refined plan is in the frontier set, planning stops and only plans with $\overline{EU} > \overline{EU}_{optimal}$ have been refined.
   - The stopping criteria is part of the definition of the optimistic strategy. The fact that no plans with $\overline{EU} \leq \overline{EU}_{optimal}$ are refined follows from the fact that plans are refined in order and the optimal fully-refined plan will appear in the frontier set before any plans with $\overline{EU} \leq \overline{EU}_{optimal}$ are refined.
5. Since the strategy refines only the set of plans that are required to be refined, it does the minimum amount of computation.

## Bounding the Advantage of Lucky Guesses

With no restrictions on the bounds of partially refined plans, the optimistic strategy can lead to expanding every abstract plan with an upper bound on expected utility that is greater than or equal to $\overline{EU}_{optimal}$. However, as discussed above, for finding a single optimal plan, it may not be necessary to expand all the plans with upper bounds equal to $\overline{EU}_{optimal}$ need be expanded. The planner needs to expand only direct ancestors of a plan with the highest expected utility.

If we modify the optimistic strategy to break ties such that nodes closest to the root of the search tree are refined first, then we can bound the difference between a lucky guess and our systematic strategy. It is also possible to show that any complete strategy has, at best, the same worst case performance. A complete strategy is one that is guaranteed to eventually find a fully-refined plan with the highest expected utility and show that it has the highest expected utility, whenever it is possible to do so.

**Theorem 3** *In the worst case, the optimistic strategy of selecting the plan with the greatest upper bound on expected utility for expansion and breaking ties in a FIFO order expands $O(B^n)$ more nodes than an optimal, omniscient strategy. Here B is the maximum branching factor and n is the minimum depth of a plan with the highest expected utility.*

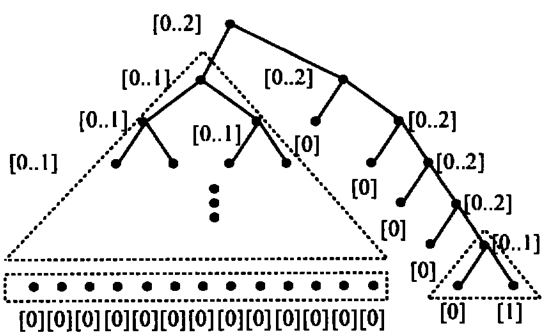


Figure 4: Worst case search tree.

**Proof:** To show that the worst case is at least $O(B^n)$, we need give only an example. To show that the bound is tight, we can show that the algorithm will always find the optimal plan in $O(B^n)$.

Figure 4 shows a worst case example for the optimistic strategy. The dashed triangles show the part of the tree that will be unrefined when all plans with $\overline{EU} > \overline{EU}_{optimal}$ have been refined. An optimal strategy would just refine the node at the lower right of the tree and find the optimal solution. The optimistic strategy will refine every other node before this node. The difference is:

$$\begin{aligned} & O(optimistic) - O(omniscient) \\ &= ((B-1)(B^{n-1}-1)+1)-1 \\ &= B^n - B^{n-1} - B + 1 + 1 - 1 \\ &= O(B^n) \end{aligned}$$

We now show that the maximum possible difference is $O(B^n)$.

1. Any strategy must expand all nodes with $\overline{EU} > \overline{EU}_{optimal}$, from the proof of theorem 2
2. When all the plans with $\overline{EU} > \overline{EU}_{optimal}$ have been expanded, either the optimal plan has been found, or there are one or more abstract plans with $\overline{EU} = \overline{EU}_{optimal}$.
   - Since all strategies must refine all plans with $\overline{EU} > \overline{EU}_{optimal}$, this point will be reached eventually. The optimistic strategy ensures that this point is reached before any plans with $\overline{EU} = \overline{EU}_{optimal}$ have been evaluated.
3. If the optimal plan has been found, then the optimistic strategy terminates, otherwise it does a breadth-first search through the plans with $\overline{EU} = \overline{EU}_{optimal}$.
   - Since the nodes with equal $\overline{EU}$ are sorted by inverse depth, they will be explored breadth-first.
4. The number of node expansions to completely expand a tree using breadth-first search, to a depth of n, is bounded by $O(B^n)$, and will necessarily find any leaf node at depth n with $EU = EU_{optimal}$
5. In the best case, an optimal strategy could refine a single plan and find an optimal solution. $O(1)$
6. The worst case difference is $O(B^n) - O(1) = O(B^n)$

Given theorem 3, we know that the optimistic strategy can be $O(B^n)$ worse than optimal, but what about other strategies? It turns out that any strategy is $O(B^n)$ worse than an optimal strategy for some search trees.

**Theorem 4** *For every strategy there exists a tree such that the strategy expands $O(B^n)$ more nodes that an optimal omniscient strategy.*

**Proof:** An adversarial argument can be used to prove this theorem.

1. Given a search strategy, an adversary can determine the order that the leaves in the tree in figure 4 will be visited.
2. The adversary can then rearrange the labels on the leaves so that the plan with the highest expected utility is visited last.
3. For any arrangement of labels on the leaf nodes, an optimal omniscient strategy can find the plan with the highest expected utility in at most n expansions, where n is the depth of the plan with the highest expected utility.

4. The minimum difference is:

$$\begin{aligned} Min(\text{difference}) &= (B-1)(B^{n-1}-1)+1-n \\ &= B^n - B^{n-1} - B + 1 + 1 - n \\ &= O(B^n) \end{aligned}$$

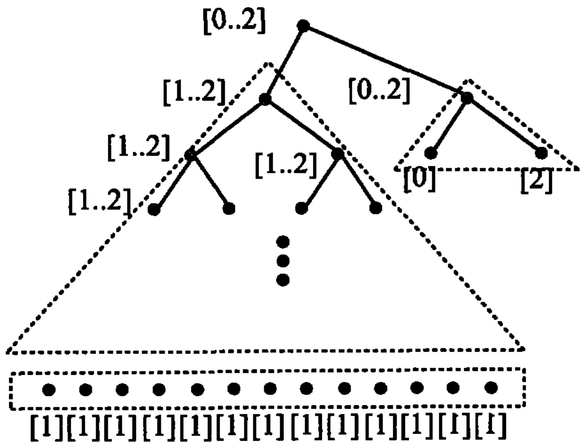


Figure 5: Worst case tree for the conservative strategy.

**Worst Case Performance** We have shown that any complete strategy can be $O(B^n)$ worse than an optimal strategy for some trees. Obviously, an incomplete strategy can fail to ever find the optimal plan. It is also interesting to note that a search tree may not be balanced or even finite. A strategy that performed depth-first search could be forced to search an arbitrarily large, or even an infinite space before finding the optimal solution, where a breadth first search would terminate after $O(B^n)$ refinements.

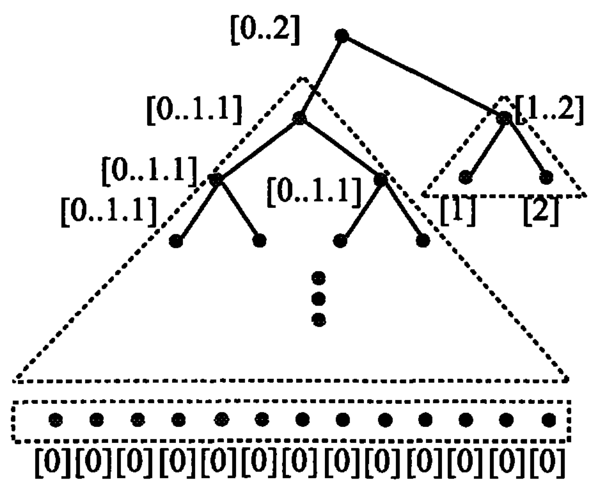


Figure 6: Worst case tree for pruning strategy.

Consider the conservative and pruning strategies. Neither one systematically refines the plans with $\overline{EU} > \overline{EU}_{optimal}$ before expanding plans with $\overline{EU} = \overline{EU}_{optimal}$. In fact, selecting nearly pruned plans can expand plans with $\overline{EU} < \overline{EU}_{optimal}$ before plans with $\overline{EU} > \overline{EU}_{optimal}$. It is possible to construct examples where these strategies will explore an arbitrarily large space before finding the optimal solution. For infinite search trees, we can construct examples where these strategies never terminate.

Consider the search tree in figure 5 where the optimal solution can be found by refining the top three nodes. The strategy of selecting the plan with the greatest lower bound on expected utility will completely expand the sub-tree on the left before refining the sub-tree on the right. The amount of work required is $O(B^{max(Depth)})$, which can be arbitrarily worse than $O(B^n)$. In fact, for infinite trees the maximum depth is infinite and the planner will never terminate.

A similar example, figure 6, shows an example where the pruning strategy expends an arbitrarily large or infinite amount of work to find the optimal plan. The worst case bounds for this strategy is the same as for the conservative strategy, $O(B^{max(depth)})$.

## Empirical Comparison of Strategies

To get an idea of how the plan selection strategies compare in practice, we modified the DRIPS planner so that it could use any of the first three strategies in table 1. We then tried each strategy on a sequence of problems from the deep venous thrombosis (DVT) medical domain (Haddawy, Kahn, & Doan 1995). The domain consists of a model of how the disease can progress over time and the actions available to the doctor that include a number of tests and a treatment. The planning objective is to create a testing and treatment policy with the highest expected utility. The original encoding of the domain allowed up to three tests before a decision to treat was made. We extended the domain to use loops so that the planner can determine how many tests are useful (Goodwin 1996b).

The set of DVT problems is generated by varying the cost of fatality. For low costs of fatality, the optimal plan is not to test or treat anyone. Few plans need to be refined in order to determine the optimal plan. As the cost of fatality rises, the optimal plan includes more tests to improve the likelihood of detecting the disease and treats the patients with positive test results. Adding more tests increases the probability that all patients, including healthy patients, will suffer side effects from the tests, which can lead to death in some cases. As the cost of fatality rises, the number of tests that must be considered and the tradeoffs between not treating the disease and side-effects from multiple tests become more difficult. The planner must refine an ever increasing number of plans to determine the optimal solution, even in the best case.

The results of the experiment are shown in table 2. The number of plans generated and the CPU time are given for each strategy. The percentage differences

| Cost of Fatality in $000 | Optimistic Strategy $max(\overline{EU})$ # Plans | Time | Conservative Strategy $max(\underline{EU})$ # Plans | | Time | | Pruning Strategy $min(\overline{EU})$ # Plans | | Time | |
|---|---|---|---|---|---|---|---|---|---|---|
| 50 | 9 | 5.0 sec | 9 | 0% | 5.0 sec | 0% | 9 | 0% | 5.0 sec | 0% |
| 100 | 33 | 43.7 sec | 33 | 0% | 44.7 sec | 2% | 39 | 18% | 47.5 sec | 9% |
| 150 | 45 | 62.0 sec | 45 | 0% | 63.0 sec | 2% | 51 | 13% | 66.0 sec | 7% |
| 200 | 67 | 115.9 sec | 67 | 0% | 116.6 sec | 1% | 73 | 12% | 119.8 sec | 3% |
| 300 | 157 | 429.8 sec | 157 | 0% | 433.6 sec | 1% | 161 | 3% | 433.5 sec | 1% |
| 500 | 441 | 32.6 min | 471 | 7% | 34.7 min | 6% | 553 | 25% | 41.8 min | 28% |
| 650 | 933 | 90.1 min | 1025 | 10% | 99.0 min | 10% | 1293 | 39% | 128.9 min | 43% |
| 850 | 3105 | 8.31 hrs | $\geq$ 10007 | $\geq$ 222 % | $\geq$ 5.45 days | $\geq$ 1574% | 8481 | 173% | 22.85 hrs | 175% |

Table 2: Results from the DVT domain comparing the number of plans generated and the run time for three strategies. Percentages are relative to the optimistic strategy. All tests were run under Allegro 4.2 on a Sun 4 Sparc.

from the optimistic strategy are given for the other two strategies. For low costs of fatality, all the strategies have the same performance and refine relatively few plans and take about the same amount of CPU time. As the cost of fatality rises, the number of plans that must be expanded increases, giving the planner more opportunity to make bad plan selection choices. When the cost of fatality reaches $650,000, the conservative strategy, which was originally used in DRIPS refines about 10% more plans and uses 10% more CPU time. This extra effort is spent refining plans in an attempt to prune the search space rather than focusing on potentially optimal plans. Above a $650,000 cost of fatality, this strategy crosses a threshold where it becomes exceedingly sub-optimal. For a cost of fatality of $850,000, the planner runs for over 5 CPU days and still fails to find a solution when the optimal strategy finds a solution in 8.31 CPU hours. The planner stopped after 5.45 days because the computer ran out of virtual memory.

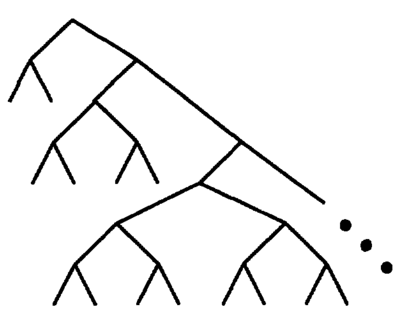

Figure 7: Shape of the search tree for the DVT domain.

To understand the performance of the conservative strategy ($max(\underline{EU})$), consider the search tree representation of the problem in figure 7. As the cost of fatality increases, the optimal solution includes more tests and is found lower in the tree. Because the lower bound on expected utility for the infinite right subtree is $-\infty$, the conservative strategy will completely expand each finite sub-tree to the left before expanding the infinite sub-tree to the right. This effort may be unnecessary since only the parts of the sub-tree with $\overline{EU} \geq \overline{EU}_{optimal}$ need to be expanded. Near the root of the search tree, most of the nodes in the finite subtree need to be expanded anyway. However, as the number of tests increase and the marginal utility decreases, the range of expected utility for the interior nodes decreases, increasing the likelihood that parts of the finite sub-trees can be pruned without expansion. At the same time, the size of the finite sub-trees grows exponentially with depth. At a $850,000 cost of fatality, the search tree reaches a point where the conservative strategy becomes exponentially worse than the optimistic strategy.

The pruning strategy, $min(\overline{EU})$, does worse than the conservative and optimistic strategy for small costs of fatality. It expands plans that would otherwise be pruned when an optimal plan was found. The relative performance continues to worsen as the cost of fatality and the difficulty of the planning problem increases. It does not have the same dramatic threshold effect that the $max(\underline{EU})$ strategy has, but the performance does degrade significantly. At $850,000, this strategy refines 173% more plans and uses 175% more CPU time.

Although strategies $min(\overline{EU})$ and $max(\underline{EU})$ can be arbitrarily worse than $max(\overline{EU})$, in practice the differences are mitigated by other factors. One thing to note is that the choice is limited to plans that were not pruned and are thus potentially optimal. There is also a correlation between plans selected by the optimistic strategy and plans selected by the conservative strategy since plans with high lower bounds also tend to have high upper bounds.

The results in table 2 are consistent with our theoretical results and show that using the optimistic plan selection strategy can have a significant effect on performance. We have also observed similar results for the Xavier route planner and Williamson reports similar observations for the Pyrrhus planner[4].

[4]Personal communications with Mike Williamson

## Conclusions and Future Work

In this paper, we have shown that the optimistic strategy of selecting the plan with the highest upper bound on expected utility and breaking ties in a breadth first fashion, expands the smallest number of plans under a variety of conditions. When these conditions do not hold, the difference in performance between the optimistic strategy and a lucky guess is bounded.

The question of how to select plans for expansion while taking the cost of computation into account remains open. In large search spaces, it may not be possible to find a plan with $EU = EU_{optimal}$ in a reasonable amount of time. An anytime decision-theoretic planner may do better by using the conservative plan selection strategy by minimizing regret.

## References


Goodwin, R. 1996a. *Meta-Level Control for Decision-Theoretic Planners.* Ph.D. Dissertation, Carnegie Mellon University.

Goodwin, R. 1996b. Using loops in decision–theoretic refinement planners. In Drabble, B., ed., *Proceedings of the Third International Conference on Artificial Intelligence Planning Systems (AIPS-96)*, 118–124. Edinburgh, Scotland: AAAI Press.

Haddawy, P., and Doan, A. 1994. Abstracting probabilistic actions. In de Mantras, R. L., and Poole, D., eds., *Proceedings of the Tenth Conference on Uncertainty in Artificial Intelligence (UAI-94)*, 270–277.

Haddawy, P.; Kahn, Jr, C.; and Doan, A. 1995. Decision-theoretic refinement planning in medical decision making: Management of acute deep venous thrombosis. *Medical Decision Making.*

Kambhampati, S., and Srivastava, B. 1996. Unifying classical planning approaches. Technical Report ASU CSE Technical Report 96-006, Arizona State University.

Loomes, G., and Sugden, R. 1982. Regret theory: an alternative theory of rational choice under uncertainty. *The Economic Journal* 92:805–824.

Loomes, G. 1987. Some implications of a more general form of regret theory. *Journal of Economic Theory* 41:270–287.

Williamson, M., and Hanks, S. 1994. Optimal planning with a goal-directed utility model. In Hammond, K., ed., *Proceedings of the Second International Conference on Artificial Intelligence Planning Systems (AIPS-94).* Chicago, Illinois: AAAI Press.