Qualitative Planning under Assumptions: A Preliminary Report

Nir Friedman
Department of Computer Science
Stanford University
Stanford, CA 94305
nir@cs.stanford.edu

Daphne Koller
Computer Science Division
University of California, Berkeley
Berkeley, CA 94720
daphne@cs.berkeley.edu

Abstract

Most planners constructed up to now are *qualitative*: they deal with uncertainty by considering all possible outcomes of each plan, without quantifying their relative likelihood. They then choose a plan that deals with the worst-case scenario. However, it is clearly infeasible to plan for every possible contingency. Even beyond the purely computational considerations, planning for highly unlikely worst-case scenarios can force the agent to choose an overly cautious plan with low utility. One common way to avoid this problem is to make assumptions about the behavior of the world, i.e., assume that certain contingencies are impossible. In this paper, we analyze the paradigm of qualitative planning under assumptions, using decision-theoretic tools. We present conditions that guarantee the existence of *optimal* assumptions (ones inducing the agent to choose the plan with maximum expected utility). Finally, we sketch how assumptions can be constructed for a certain restricted class of planning problems.

1 Introduction

Any system that acts in the real world has to deal with uncertainty, both about the exact state of the world and about the effects of performing actions. While there are many forms of planning under uncertainty, we can generalize and say that two main methods appear in the literature: qualitative planning and quantitative (decision-theoretic) planning. Qualitative planners maintain a model of the possible states in which the world might be, but do not attempt to quantify the relative likelihood of each such possibility. Such planners are typically cautious: they try to construct a plan that guarantees the agent certain minimal utility (e.g., reaching the goal) in all circumstances [GN87]. (see also [McD87] for a critical view of this approach.) Quantitative planners maintain a probabilistic model of the world, where each possible outcome is assigned a probability. Such planners construct a plan that achieves the maximum *expected utility*. The debate on the appropriateness of each method is beyond the scope of this paper, but it is clear that each has its merits in certain application areas. In this paper we take a slightly different view. We examine qualitative planners and use decision theory as an external tool to measure the quality of the generated plans.

As we said, a qualitative planner considers all possible outcomes of each plan, and chooses a plan that works best in the worst case (a *maximin* plan). A naive implementation of this approach suffers from two major disadvantages. The first is computational: it causes the agent to waste resources on planning for situations that will typically not arise during the execution of the plan. But even more importantly, planning for the worst case can force the agent to choose "defensive" plans that perform well in the (perhaps very unlikely) worst case, even if their overall performance is suboptimal. In fact, since in many situations no plan is guaranteed to reach the goal in the worst case, a qualitative planner may be unable to differentiate between alternative plans, and may simply give up. "Real-world" planning agents (whether human or artificial) typically circumvent these problems by making *assumptions* about the behavior of the world, and then planning under these assumptions. Intuitively, an assumption made by the agent eliminates, in the agent's mind, some of the behaviors available to nature. Consider, for example, an agent who has an action for turning the key in the ignition of a car. The car may start or fail to start. Without making assumptions, a qualitative planning agent must consider both outcomes possible. On the other hand, the agent can make the assumption that if he turns the key, the car will start. The behavior in which the car fails to start is "assumed away" by the agent. In general, as in this example, assumptions are defeasible: the car can, in fact, fail to start. Systems that use assumptions must monitor the execution of the plan, and replan (using a new set of assumptions) when it fails to achieve its predicated effect.

We can view the so called *traditional planners* as doing qualitative planning under assumptions. The popular *STRIPS assumption* [FN71] postulates (among other things [Lif86]) that each plan only has one conceivable outcome. This is an implicit assumption about the world. Since assumptions are defeasible, actual systems must monitor the execution of the plan, and replan if necessary [FHN72, Wil88]. Certain planners can be viewed as making assumptions (which are later dropped) in order to quickly derive an abstract or approximate plan [Sac74, Elk90, GH92]. In this paper, we ignore the computational benefits of making assumptions, and focus on the issue of constructing assumptions that induce the agent to construct a better plan.

In order to determine whether an assumption results in a better plan, we need to judge when one plan is better than another. In this paper, we use the ideas of *decision theory* to measure the value of plans. More precisely, we postulate an underlying probabilistic model for the world, and define the value of the plan in terms of its expected utility. A similar approach has also been used in the context of knowledge compilation [LSF86, Hor87, HBH89]. For example, in the previous example, we assume that there is some *probability* that the car will fail to start, and use that probability to assign expected utilities to various plans that involve turning the key in the ignition. We can then compare the expected utility of the plan chosen by the agent with no assumptions to that of the plan chosen by the agent when certain assumptions are made. This allows us to determine which assumptions are beneficial. Note that we do not assume that the probabilities in our model are known to the agent. Rather, we use them as an external measure of the performance gained by a qualitative planner from making certain assumptions. This type of analysis can be used by the designer of the agent, as a tool for constructing good assumptions. Even if the designer does not know the precise probabilities, he or she can evaluate an assumption relative to several conceivable models, and verify that it is reasonable in all of them.

As we suggested, making assumptions can lead to an increase in the expected utility of the agent. The assumptions can provide a balance to the worst-case planning procedure used by qualitative planners: if the agent makes an optimistic, but reasonable, assumption that a certain contingency will not occur, it prevents him from choosing actions based on that (unrealistic) worst case. In the example above, the assumption that the car will start is usually a good one. With this assumption, the agent will plan as if the car will start, and simply attempt to start it when necessary. Although it is annoying if the assumption turns out to be false and the car does not start, the probability of this happening is quite low, and the actual loss of utility is rarely very high. On the other hand, without this assumption, the agent would have to plan for the worst case, which would induce him to test the car some time before it is needed (so that if necessary he would be able to fix it in time). The expected utility of this plan is typically lower than that of the plan made under the assumption. Note that the benefit of making this assumption depends not only on the probability that the car will fail to start, but also on the cost to the agent if this is the case. For example, this assumption would not be beneficial if having the car fail to start causes the agent to miss an important business appointment. In this case, it is desirable for the agent to plan for the worst case. The difference between this case and the previous one is not related to the probability of the event; it is caused only by the drastic change in the utility function.

The rest of this paper is organized as follows. In Section 2, we define a formal model for a planning problem as a game between the agent and nature, and discuss connections between the structure of the problem and the assumptions that the agent can make. We also present a motivating example. In Section 3 we investigate the existence of assumptions that induce the agent to choose "good" plans. There are clearly cases where there is no assumption that induces the agent to choose the "best" plan (the one of maximum expected utility). This is not surprising: there are clearly situations that cannot be fully represented using a qualitative model. However, we describe one important class of domains where an optimal assumption (inducing a choice of the best plan) does exist. We also describe an algorithm for constructing the best possible assumption in the general case. Finally, in Section 4, we investigate specifically the issue of constructing good assumptions. We show by example that this is, in general, a more subtle problem than it seems. We sketch how, in planning domains with strong limitations on the type of uncertainty encountered by the agent, good assumptions can be effectively constructed. We conclude with discussion and open questions.

2 The framework

It is convenient to describe any specific planning problem as a tree representing a game between the agent and nature [LR57]. The nodes of the tree denote instantaneous snapshots of the world at different points in time. The internal nodes of the tree are decision points either for the agent or for nature (for simplicity, we will refer to the agent as "he" and to nature as "she"); the outgoing edges at a node represent possible decisions at that node. The leaves of the tree represent final states, and are labelled with the utility of that state for the agent. As usual in the literature [DW91], we represent utilities as real numbers. Intuitively, each path from the root to a leaf represents one possible sequence of events leading to a final state. Traditional planning problems can be represented in this form by setting the utility at each leaf to be a certain high value if the goal is achieved (and a low value if it is not), minus the cost of the actions taken on the way [FS77].

In general, of course, the agent may not have complete information about the state of the world [Moo85]. In order to represent this, the decision nodes belonging to the agent are partitioned into *information sets* $V_1^A, \ldots, V_a^A$. Intuitively, the agent cannot differentiate between different nodes in the same information set. Hence, all nodes in an information set V^A must have the same set of choices, denoted $C(V^A)$. (Otherwise the agent could differentiate between the nodes). We also define information sets $V_1^N, \ldots, V_n^N$ for nature. While this definition has an intuitive interpretation — nature can also make decisions "in ignorance" (independently) of previous events — the information sets of nature are best interpreted differently. We return to this issue at the end of this section.

A *plan* π for the agent is a *strategy*: a function dictating a choice in $C(V^A)$ at each information set

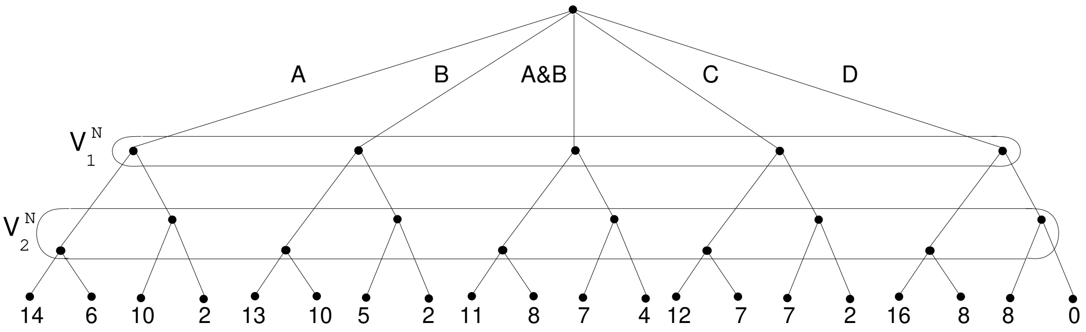

Figure 1: The tree describing Example 2.1

V^A. Let $\mathcal{P}$ denote the set of possible plans; clearly $\mathcal{P} = C(V_1^A) \times \cdots \times C(V_a^A)$. We can analogously define the possible strategies for nature. A strategy β for nature describes the *behavior* of nature at all possible situations. Let $\mathcal{B}$ denote the set of possible behaviors. Clearly, a plan π and a behavior β determine a unique leaf in the tree, with utility $U(\pi, \beta)$.

As suggested by the introduction, the same planning situation has two different interpretations: *external* and *internal*. The external interpretation is ascribed by us, when reasoning about the system. There, we assume that nature's decisions are made according to some fixed probability distribution at each of her information sets. This probability distribution determines a probability distribution over nature's behaviors in a straightforward manner. Let $\Pr(\beta)$ denote the resulting probability for the behavior β. We can now define the *expected utility* of a plan π as $EU(\pi) = \sum_{\beta \in \mathcal{B}} \Pr(\beta) U(\pi, \beta)$. In this model, the game between the agent and nature reduces to a one-player game, also known as *decision under risk*.

The agent, as we assumed, does not know the probabilities on nature's decisions. His *internal interpretation* of the tree describes the possible outcomes that might be reached by each plan. Since the agent attempts to reach a certain utility (e.g., the goal) in all possible outcomes, he must plan for the worst-case one, in effect considering nature as an adversary. With no assumptions, the agent perceives the utility of a plan π in terms of what utility is guaranteed by the plan, e.g., $PU(\pi) = \min_{\beta \in \mathcal{B}} U(\pi, \beta)$. The agent chooses the plan that with the best perceived utility. An assumption eliminates, in the agent's mind, some of the options available to nature. More precisely, an assumption $\mathcal{A}$ is some subset of the behaviors in $\mathcal{B}$ that the agent considers possible. Thus, an agent making an assumption believes that certain behaviors of nature cannot happen. This agent would choose a plan π maximizing his *perceived utility given* $\mathcal{A}$: $PU(\pi|\mathcal{A}) = \min_{\beta \in \mathcal{A}} U(\pi, \beta)$. In this case, we say that $\mathcal{A}$ *supports* π.

Example 2.1: Consider a mobile robot planning an excursion. There are two places — P_A and P_B — on the robot's planned route where a problem can occur, forcing the robot to take a detour. The probability of the first problem is 0.4, of the second is 0.5, and the two problems occur independently. The robot has a number of options: (A) take a detailed map showing the optimal detour for P_A; (B) take a detailed map showing the optimal detour for P_B; (A&B) take both detailed maps; (C) take a single large-scale map showing suboptimal detours for both; (D) take nothing. The costs associated with the various aspects of the journey are as follows: the optimal detour at P_A costs 4; the one at P_B costs 3; both suboptimal detours cost 5; both detours cost 8 when taken without any map; carrying a map costs 2. The utility of the agent is simply the negation of his cost; for simplicity we present the example directly in terms of the cost. Figure 2 shows the tree corresponding to this planning problem. At V_1^N, nature chooses whether there is a problem at P_A; left corresponds to yes and right to no. At V_2^N nature makes the analogous choice for P_B. The values at the leaves correspond to costs; for example, the cost at the leftmost leaf is 2 (for carrying one map) plus 4 (for an optimal detour at P_A) plus 8 (for a no-map detour at P_B).

	$-EU$	$\{\beta_-\}$	$\{\beta_A\}$	$\{\beta_B\}$	$\{\beta_A, \beta_B\}$	$\{\beta_{A\&B}\}$
(A)	7.6	2	**6**	10	10	14
(B)	6.7	2	10	**5**	10	13
$(A\&B)$	7.1	4	8	7	8	**11**
(C)	**6.5**	2	7	7	**7**	12
(D)	7.2	**0**	8	8	8	16

Table 1: Expected and perceived costs of plans

The first column of Table 1 shows the expected cost of each of the agent's five actions, computed using the probabilities for nature's 4 behaviors: β_- — no problems; β_A — problem only at P_A; β_B — problem only at P_B; and $\beta_{A\&B}$ — problems at both places. The next five columns show, respectively, the agent's perceived cost for all actions under the following assumptions: no problems occur; problem at P_A; problem at P_B; problem at precisely one of P_A, P_B; problems at both places.[1] The action of maximum expected utility (minimum expected

[1] Since the agent always chooses the worst-case behavior among the ones allowed by the assumption, the remaining ten possible assumptions all reduce to one of the five described. For example, any assumption containing $\beta_{A\&B}$ induces the same perceived utility as the assumption $\{\beta_{A\&B}\}$, any assumption not containing $\beta_{A\&B}$ but containing both β_A and β_B reduces to $\{\beta_A, \beta_B\}$, and so on.

cost) in the first column is underlined, as are the actions of maximum perceived utility in the remaining columns. The action of maximum expected utility is (C); the only assumption listed that supports that action (in terms of perceived utility) is the one that asserts that precisely one of the two problems occurs. ∎

Consider the "optimal" assumption $\{\beta_A, \beta_B\}$ from the example. Nature's decisions at P_A and P_B are presumably independent, while the assumption dictates that they are correlated (a problem occurs at one place iff it did not occur at the other). Intuitively, it seems more natural to allow only *local assumptions*: assumptions that separately restrict nature's choices at each of her information sets. More precisely, $\mathcal{A}$ is a *local assumption* if it can be defined as a Cartesian product of subsets $L_i \subseteq C(V_i^N)$ for $i = 1, \ldots, n$; that is, if $\mathcal{A} = L_1 \times \cdots \times L_n$. Clearly, such a product is a subset of $\mathcal{B}$, so that a local assumption is a legitimate assumption. Note that local assumptions can be represented much more compactly than standard assumptions. The representation of a local assumption is linear in the size of the tree describing the planning problem. On the other hand, the representation of a general assumption is, at worst, the number of behaviors of nature; this is typically exponential in the size of the tree.

As the example shows, there may be plans that are supported by some general assumption, but not by a local one. On the other hand, when attempting to find good assumptions, it is clearly simpler to search the space of local assumptions. We now provide a condition on nature's information sets which guarantees that we can safely restrict attention to local assumptions. We say that nature has *perfect recall* if for any of her information set V^N, and for any two nodes v_1, v_2 in V^N the sequences of nature's decisions leading to these two nodes is exactly the same. In particular, this means that the paths leading to these two nodes must traverse precisely the same sequence of nature's information sets. Intuitively, if nature takes different decisions on the way to two nodes in the tree, and nature never "forgets", then she would be able to differentiate between the two nodes, so that they could not be in the same information set. Note that we can similarly state an assumption of perfect recall for the agent.

Theorem 2.2: *If nature has perfect recall, then any assumption $\mathcal{A}$ is local, i.e., there are $L_i \subseteq C(V_i^N)$ for $i = 1, \ldots, n$, such that $\mathcal{A} = L_1 \times \cdots \times L_n$.*

But what do information sets for nature even mean? In our probabilistic model, it only forces us to use the same probability distribution over choices at all nodes in the information set. Given a tree, we can always refine nature's information sets by partitioning each information set into smaller sets. Since the probability distributions at the nodes do not change, this does not change the expected utility of the agent's plans. However, it does change nature's strategies and local strategies. In the tree of Figure 2, for example, we placed all the decision points regarding P_B in the same information set, representing the fact that this decision is independent of the one regarding P_A (and both are independent of the agent's decision). Hence, nature does not have perfect recall: in V_2^N she does not remember her previous decision at V_1^N. We can transform this tree into one where nature has perfect recall by partitioning the information set V_2^N into two — V_{2y}^N and V_{2n}^N; the assignment of each node to a new information set is according to which decision was taken at V_1^N: y and n respectively. In the figure, the higher nodes correspond to V_{2y}^N and the lower nodes to V_{2n}^N The probability of the final outcomes (the choice of locations with a problem) is the same in both models. However, nature's strategies in each case are different, and hence so are the possible assumptions the agent can make. Note that the assumption $\{\beta_A, \beta_B\}$ is not a local assumption in the first model (the one with imperfect recall). On the other hand, in the new model, the corresponding assumption *is* local: it can be represented by $L_1 = \{y, n\}$, $L_{2y} = \{n\}$, and $L_{2n} = \{y\}$.

Thus, the granularity of nature's information sets determines a tradeoff between a simpler description of the game (one that also represents the independences inherent in the domain) and the expressiveness of the local assumptions that the agent can make.[2] Theorem 2.2 gives a condition guaranteeing that local assumptions are as expressive as possible.

3 Choosing good assumptions

How can we choose an assumption that supports a good plan? If we wish the agent to execute a particular plan π^*, is there an assumption that supports it? In the full paper, we present Algorithm FindAssump, which provides a partial answer to this question. In a domain where both the agent and nature have perfect recall, the algorithm determines for a plan π^* whether there exists an assumption supporting it. If such an assumption exists, the algorithm returns the most general (least restrictive) one. The algorithm works in the following manner. It computes the worst strategy $\beta^* \in \mathcal{A}$ that nature can apply against π^* (initially $\mathcal{A} = \mathcal{B}$). It also computes the worst outcome nature can force on the agent: $\max_\pi \min_{\beta \in \mathcal{A}} U(\pi, \beta)$. Since the agent plans for the worst case, he cannot guarantee himself a better utility. If he can guarantee this utility using π — i.e., if $U(\pi^*, \beta^*)$ achieves this utility — then π^* is an optimal plan relative to $\mathcal{A}$. If not, then any assumption set that includes β^* will not support π^* (another plan will always be better). Hence, it is possible to prune β^* from any further consideration. This is done by examining the path that leads to $U(\pi^*, \beta^*)$ and finding the deepest information set V_i^N where $\mathcal{A}$ allows nature a choice between two or more actions. The action that is chosen by β^* at V_i^N leads to the bad outcome $U(\pi^*, \beta^*)$; we therefore assume that nature does not take it. The algorithm then iterates with the modified assumption. If the algorithm reaches an assumption containing exactly one strategy, and it does not support π^* then it returns a negative answer. In the full paper we present the al-

[2] The size of the tree does not grow when we refine information sets. Hence, we do not lose the advantage of compact representation for local assumptions in the refinement process.

gorithm, prove its correctness, and show that it runs in quadratic time in the size of the tree.

Using a well-known theorem from game theory, in games with perfect recall the plan π^* of maximum expected utility can be computed from the game tree in linear time. We therefore obtain the following result.

Theorem 3.1: *For a planning problem represented by a game tree with perfect recall, it is possible to decide in quadratic time whether there exists an assumption supporting the plan of maximum expected utility.*

In Example 2.1, we saw that the plan of maximum expected utility is supported by some assumption. Hence, in that example, a qualitative planner, guided by an appropriately chosen assumption, can do as well as a decision-theoretic planner. Unfortunately, it is easy to construct examples showing that this is not always the case (see the full paper for details). If the optimal plan is not supported by an assumption, we may wish to find the best plan that is. This can be done by enumerating the possible plans, and ordering them by expected utility. We can then check, starting from the best plan and going down, which plan is supported by some assumption. Unfortunately, this procedure is impractical since the number of possible plans is exponential in the size of the tree. As of now, we do not have a good technique for finding the best possible assumption.

There is, however, one important class of domains in which this problem does not come up. In situations where both players have *perfect information* — all information sets contain exactly one node — the plan of maximum expected utility is always supported by an assumption. Although perfect information implies that the agent has no uncertainty about the current state of the world, there is still a significant source of uncertainty: nature's future choices. The perfect-information restriction for the agent is one that is common to most planning systems.[3] As we explained in the previous section, the perfect-information constraint on nature's information sets ensures sufficient expressive power for the set of assumptions considered by the agent. In particular, it enables the agent to make "Murphy's Law"-like assumptions, such as "If I take no maps, then everything will go wrong, and if I take maps, then no detours will even be necessary." While this type of assumption might not be consistent with the causal structure of the world, it can certainly be a useful assumption for the agent to make for the purposes of planning.

Theorem 3.2: *In any planning problem represented by a game tree with perfect information, there exists an assumption $\mathcal{A}$ supporting the plan of maximum expected utility.*

Using Algorithm FindAssump, this assumption can be found in quadratic time.

[3]Furthermore, we can often convert domains where the agent has imperfect information to ones where he does, by modelling nature's decisions as happening just before the agent discovers their outcome (even if this is not an accurate temporal description).

4 Design issues in assumption making

Algorithm FindAssump is an initial step in the construction of tools for the designer. It allows the designer to determine whether a qualitative planning agent can achieve a high level of performance in the specific domain at hand, and what assumptions can enable him to do so. For example, the designer can test whether certain "good" plans are chosen by the agent, and under which assumptions. This procedure can also allow us to analyze existing planners in order to determine when the assumptions they make are justified.

However, the applicability of Algorithm FindAssump as a general-purpose design tool is very limited. It constructs good assumptions for a specific planning situation, i.e., relative not only to a domain, but also to a specific goal and initial state. It is clearly unrealistic to expect the designer to construct an appropriate assumption for each possible planning situation encountered by the agent. In this section, we investigate the issue of designing assumptions in a more general way, that does not depend on all the attributes of the specific planning situation.

Unfortunately, assumptions that are good "in general" are not easy to construct. Intuitively, assumptions are designed to make the best plan "look good" to the agent, i.e., raise its perceived utility. However, the choice of best plan depends on the utilities and probabilities in the domain. Consider again the example from the introduction, where the agent can assume that the car will start if he turns the key. As we pointed out, the benefit of making the assumption depends on the utilities: if the agent loses significantly from the car failing to start, the assumption that the car will start is no longer a good one. But the utilities are determined by the specific goal. We might hope to be able to construct assumptions in a hierarchical manner: to construct assumptions for actions or sub-plans, and then combine them into assumptions for a large plan. Here also, matters are not so straightforward. While the assumption that the car will start is useful on any given day, it is probably unwise to make a long term plan (such as driving cross-country) based on the assumption that the car *never* fails (which is an event whose probability is much higher). But there is an even more subtle problem: the best plan depends on the other possible plans. In the example above, we made the assumption that the car does not fail, since it supports the plan of simply starting the car when needed, which is a better plan than checking it thoroughly in advance. Now, assume that the agent also has an alternative option — taking the train — which is slightly slower but always reliable. This new plan might have a higher expected utility, but the assumption that the car always works would induce the agent to choose the car nonetheless. Thus, the usefulness of the assumption cannot be decided in isolation: it also depends on the other options available to the agent.

These examples show that the problem of making assumptions modularly is a very difficult one. But it is not hopeless. In a given domain, we can often identify classes of problems that share similar starting conditions and goals, and thereby construct assumptions that are beneficial for all the problems in the class. We can also characterize general structural properties of domains, which enable us to determine easily whether certain types of local assumptions are useful or not. In the remainder of this section, we describe one type of uncertain event for which we can determine easily whether the associated assumption is beneficial. Consider a domain where certain *contingencies* may arise. Intuitively, a contingency $\mathbf{c}$ is an uncertain event that occurs uniformly over plans, and whose costs are also uniform (the cost-uniformity is described below). That is, for each plan that may encounter $\mathbf{c}$, the probability that it occurs is the same. Furthermore, in any planning situation (a tree) either all plans encounter the contingency or none do. (Technically, it is an information set of nature that appears on any path to a leaf node in the tree.) For example, consider a domain similar to Example 2.1, where the agent must navigate around a building where there are doors that may be locked. For any given starting point and goal, the doors that the agent must pass through are the same. In such a domain, it is reasonable to suppose that the probability of a locked door, and the cost of dealing with it, is independent of the plan chosen. However, if the cost of a contingency is truly fixed, then it does not affect the planning process; in this case, assumptions become irrelevant. We therefore allow the agent to take certain *preemptive actions*, which reduce the cost of some contingency. For example, taking the key to a door is a preemptive action that reduces the cost to the agent if he finds the corresponding door to be locked; similarly, the action of taking a map in Example 2.1 is also a preemptive action. We can now define the cost-uniformity constraint, and assume that the cost of the contingency is the same over all plans where a preemptive action is taken, and the same over all plans where it is not; these two costs can, however, be different. More precisely, let β^+ and β^- be a pair of behaviors that are identical but for the fact that $\mathbf{c}$ occurs in the former but not in the latter; similarly, let $\mathbf{a}$ be a preemptive action and π^+ and π^- be two plans that are the same except that $\mathbf{a}$ is taken in the former and not in the latter. Then we assume that:

$$\begin{array}{rcl} U(\pi^+,\beta^-) & = & U(\pi^-,\beta^-) + \mathit{cost}(\mathbf{a}) \\ U(\pi^-,\beta^+) & = & U(\pi^-,\beta^-) + \mathit{cost}(\mathbf{c}|\neg\mathbf{a}) \\ U(\pi^+,\beta^+) & = & U(\pi^-,\beta^-) + \mathit{cost}(\mathbf{a}) + \mathit{cost}(\mathbf{c}|\mathbf{a}). \end{array}$$

That is, the cost of taking a preemptive action, and the costs of the contingency with and without the action, are the same for any plan and behavior.

In such a domain, when can we assume that a contingency does not occur? Such an assumption reduces the incentive of the agent to take appropriate preemptive actions. Hence, certain assumptions are beneficial while others are not. In a domain with complex interactions between contingencies and preemptive actions, the problem of finding an optimal assumption can be shown to be NP-hard, even if all contingencies can occur in all plans. However, if this latter condition holds, we can determine whether a given assumption is beneficial in polynomial time. Furthermore, in the special case where each preemptive action is associated with precisely one contingency, we can give a simple rule that allows us to decide whether an assumption should be made:

Theorem 4.1: *Consider a domain with contingencies* $\mathbf{c}_1,\ldots,\mathbf{c}_k$ *and the preemptive actions* $\mathbf{a}_1,\ldots,\mathbf{a}_k$*, where* $\mathit{cost}(\mathbf{c}_i|\mathbf{a}_j) < \mathit{cost}(\mathbf{c}_i|\neg\mathbf{a}_j)$ *iff* $i = j$*. Then the assumption that* $\mathbf{c}_i$ *does not occur is beneficial iff*

$$\mathit{cost}(\mathbf{a}_i) > \Pr(\mathbf{c}_i)[\mathit{cost}(\mathbf{c}_i|\neg\mathbf{a}_i) - \mathit{cost}(\mathbf{c}_i|\mathbf{a}_i)].$$

That is, the decision for each contingency can be made in isolation. The resulting assumption is optimal for any planning problem in the domain.

While this theorem deals with a rather restricted case, it is possible to extend it in two ways. We can allow a single preemptive action to affect the cost of several contingencies, or we can allow several preemptive actions to address a single contingency. It is only the combination of the two types of interaction that cause the NP-hardness result mentioned above. As an example of the first type of interaction, consider the following scenario. The agent is planning a cross-country car trip. On each day of the trip, the agent might encounter the contingency that the car fails to start (one contingency per day). A possible preemptive action is the join the American Automobile Association (AAA). This action has a fixed cost, but it reduces the cost associated with each of the above contingencies. (For example, a AAA membership allows the agent to have the car towed to a garage with no charge.) If the trip takes n days, then the AAA membership is worthwhile if $\mathit{cost}(\mathbf{AAA}) \leq n\Pr(\mathbf{break})[\mathit{cost}(\mathbf{break}|\neg\mathbf{AAA}) - \mathit{cost}(\mathbf{break}|\mathbf{AAA})]$. However, the perceived utility of each plan is the worst-case one, i.e., when all possible failures occur. Thus, without making any assumptions, a qualitative planner will choose to buy a AAA membership if $\mathit{cost}(\mathbf{AAA}) \leq n[\mathit{cost}(\mathbf{break}|\neg\mathbf{AAA}) - \mathit{cost}(\mathbf{break}|\mathbf{AAA})]$. If $\Pr(\mathbf{break})$ is small enough so that the preemptive action is not worthwhile, we need assumptions that would make its perceived utility lower. In this case, the best assumption (the one generated by our algorithm) is that the contingency **break** will not occur too often. That is, nature will not choose it more than k times where k is the largest number such that $\mathit{cost}(\mathbf{AAA}) \leq k[\mathit{cost}(\mathbf{break}|\neg\mathbf{AAA}) - \mathit{cost}(\mathbf{break}|\mathbf{AAA})]$. This seems like a very natural assumption to make. In fact, experimental results, derived from an implementation of our algorithm, demonstrate that the best possible assumptions are often very natural ones. This observation is very encouraging, since it suggests that designers of planning systems may be expected to make good assumptions.

5 Conclusions

This paper presents some preliminary results on the subject of making assumptions in qualitative planning. Our results provide a framework for rationally justifying (using decision-theoretic techniques) the assumptions that agents make. We showed that assumptions can induce a qualitative planning agent to choose a better plan, i.e., one whose expected utility is higher. In the important class of domains where the agent has perfect information, such an agent can achieve optimal utility: the utility he could have obtained had he used a decision-theoretic planning process. We also described some of the issues facing a designer of a qualitative planning system, and presented a restricted type of domain, where we can give guidance on constructing good assumptions.

The framework developed in this paper could be extended in a number of very interesting directions. The most obvious is a deeper investigation of a design methodology for making assumptions in practical planning domains. While this is a very ambitious goal, we believe that progress can be made in domains of certain types. In this paper we did not examine the computational benefits of assumption making. Such an analysis would be applicable to both qualitative and decision-theoretic planning [DKKN93].

Finally, there is a close connection between assumptions and default rules. As we pointed out, assumptions are defeasible: an agent might discover that an assumption he made was wrong. In this case, the agent should stop and replan, using a new set of assumptions [FHN72, Wil88]. The process of replacing an assumption with a new one is closely related to belief revision. The connection between assumptions and defaults is not limited to this aspect. For example, in [Elk90, GH92], defaults are used in the planning process in order to focus the search on the difficult aspects of the plan. That is, there is a default assumption that certain subgoals, that are relatively easy to achieve, are in fact achievable. Our framework does not address this use of assumptions. However, we hope that it will allow us to formally justify their use. That is, in certain domains, we may be able to specify which assumptions can be "safely" made, i.e., without sacrificing the quality of the constructed plan. One special type of "safe" assumptions are these that concern contingencies that have very small probability. As in Section 4, we can examine domains that are almost deterministic: where each action results in a single outcome with a very high probability, but certain contingencies might occur. Using the techniques of Section 4, we can determine (based on the probabilities and utilities in the domain) when we can safely assume that a contingency does not happen. This extends the the idea of probabilistic semantics for defaults [Pea89] (e.g., ϵ-semantics) to incorporate utilities. We are currently investigating these and other connections between assumptions and defaults.

References

[DKKN93] T. Dean, L. P. Kaelbling, J. Kirman, and A. Nicholson. Planning with deadlines in stochastic domains. In *Proc. National Conference on Artificial Intelligence (AAAI '93)*, pages 574–579, 1993.

[DW91] T. Dean and M. P. Wellman. *Planning and Control.* Morgan Kaufmann, 1991.

[Elk90] C. Elkan. Incremental approximate planning. In *Proc. National Conference on Artificial Intelligence (AAAI '90)*, pages 145–150, 1990.

[FHN72] R. E. Fikes, P. E. Hart, and N. J. Nilsson. Learning and executing generalized robot plans. *Artificial Intelligence*, 3:251–288, 1972.

[FN71] R. E. Fikes and N. J. Nilsson. STRIPS: A new approach to the application of theorem proving to problem solving. *Artificial Intelligence*, 2:189–208, 1971.

[FS77] J. A. Feldman and R. F. Sproull. Decision theory and artificial intelligence II: The hungry monkey. *Cognitive Science*, 1:158–192, 1977.

[GH92] M. L. Ginsberg and H. W. Holbrook. What defaults can do that hierarchies can't. In *Proceedings 1992 Nonmonotonic Reasoning Workshop*, Plymouth, VT, 1992.

[GN87] M. R. Genesereth and N. J. Nilsson. *Logical Foundations of Artificial Intelligence.* Morgan Kaufmann, Los Altos, CA, 1987.

[HBH89] D. E. Heckerman, J. S. Breese, and E. J. Horvitz. The compilation of decision models. In *Proc. Fifth Workshop on Uncertainty in Artificial Intelligence*, 1989.

[Hor87] E. J. Horvitz. Problem-solving design: Reasoning about computational value, tradeoffs, and resources. In *Proc. NASA Artificial Intelligence Forum*, pages 26–43, 1987.

[Lif86] V. Lifschitz. On the semantics of STRIPS. In M. P. Georgeff and A. L. Lansky, editors, *Reasoning about Actions and Plans: Proceedings of the 1986 Workshop*, pages 1–9. Morgan Kaufmann, 1986.

[LR57] R. D. Luce and H. Raiffa. *Games and Decisions.* Wiley, New York, 1957.

[LSF86] C. P. Langlotz, E. H. Shortliffe, and L. M. Fagan. Using decision theory to justify heuristics. In *Proc. National Conference on Artificial Intelligence (AAAI '86)*, pages 215–219, 1986.

[McD87] D. McDermott. A critique of pure reasoning. *Computational Intelligence*, 3(3):151–160, 1987.

[Moo85] R. C. Moore. A formal theory of knowledge and action. In J. Hobbs and R. C. Moore, editors, *Formal Theories of the Commonsense World*, pages 319–358. Ablex Publishing Corp., Norwood, NJ, 1985.

[Pea89] J. Pearl. Probabilistic semantics for nonmonotonic reasoning: A survey. In R. J. Brachman, H. J. Levesque, and R. Reiter, editors, *Proc. First International Conference on Principles of Knowledge Representation and Reasoning (KR '89)*, pages 505–516, 1989.

[Sac74] E. Sacerdoti. Planning in a hierarchy of abstraction spaces. *Artificial Intelligence*, 5:115–135, 1974.

[Wil88] D. E. Wilkins. *Practical Planning: Extending the Classical AI Planning Paradigm.* Morgan Kaufmann, Los Altos, CA, 1988.