

There's More to Life than Making Plans: Plan Management in Dynamic, Multi-agent Environments

Abbreviated Title: Plan Management in Dynamic Environments

Martha E. Pollack
Department of Computer Science
and Intelligent Systems Program
University of Pittsburgh
Pittsburgh, PA 15260
pollack@cs.pitt.edu

John F. Horty
Philosophy Department
and Institute for Advanced Computer Studies
University of Maryland
College Park, MD 20742
horty@umiacs.umd.edu

Abstract:

For many years, research in AI plan generation was governed by a number of strong, simplifying assumptions: that the planning agent is omniscient, that its actions are deterministic and instantaneous, that its goals are fixed and categorical, and that its environment is static. More recently, researchers have developed expanded planning algorithms that are not predicated on such assumptions. But changing the way in which plans are formed is only part of what is required when the classical assumptions are abandoned. The demands of dynamic, uncertain environments mean that in addition to being able to form plans—even probabilistic, uncertain plans—agents must be able to effectively manage their plans. In this paper, which is based on a talk given at the 1998 AAAI Fall Symposium on Distributed, Continual Planning, we first identify reasoning tasks that are involved in plan management, including commitment management, environment monitoring, alternative assessment, plan elaboration, meta-level control, and coordination with other agents. We next survey approaches we have developed to many of these tasks, and discuss a plan management system we are building to ground our theoretical work, by providing us with a platform for integrating our techniques and exploring their value in a realistic problem. Throughout, our discussion is informal, and relies on numerous examples; the reader can consult the various papers cited for technical details.

Keywords: plan management, plan generation, agents, reasoning about actions, decision theoretic reasoning, execution monitoring, agent coordination

1 Introduction

Research on planning within AI has led to the development of computational techniques for generating a plan, or course of action, to achieve a specified goal from a specified initial state, given definitions of the available actions. Until recently, however, much of this research has been governed by a number of simplifying assumptions, notably the following:

- The planning agent is omniscient: it knows all the relevant facts about its environment.
- The actions that the agent can perform have definite outcomes.

- The goals presented to the agent are categorical, i.e., they are either achieved or not, and there is no notion of partial satisfaction.
- The agent is the only source of change in the environment: there are neither exogenous events nor other agents.
- The goals presented to the agent remain unchanged throughout the process of planning and execution.
- The actions that the agent can perform can be modeled as instantaneous state transducers: they have neither temporal extent, nor fixed times of occurrence.

These assumptions have several key consequences. First, if they hold, there is no need to interleave planning and execution, because the agent has all the knowledge it needs at planning time. Second, in such circumstances, a plan can always be executed successfully: everything is guaranteed to go “according to plan,” and replanning is thus unnecessary. Third, because the goals are all known at the outset, and remain fixed throughout the planning and execution process, there is no notion of planning problems competing with one another for the planning agent’s attention. In short, these assumptions preclude the need for active management of the planning process: there is no need to decide which planning or replanning tasks to attend to when, or how much detail to include in a plan now and how much to defer until later.

Of course, most of these assumptions—which were introduced, after all, only to delimit a tractable domain for initial investigation—fail to hold in the kinds of realistic situations in which planning is called for. Consequently, a number of researchers in AI have explored techniques for extending the classical planning paradigm, relaxing some of its overly strong assumptions. Techniques now exist for generating conditional plans with branches whose performance is keyed to the outcome of sensing actions [Peot and Smith, 1992, Collins and Pryor, 1995], removing the need to assume omniscience (the first assumption). Representations of actions with probabilistic outcomes have been developed and used in systems that generate plans whose probability of success exceeds a given threshold [Kushmerick et al., 1995, Goldman and Boddy, 1994], eliminating the assumption of deterministic actions (the second assumption). These extensions have been combined in probabilistic, conditional planning systems [Draper et al., 1994, Blythe, 1998, Onder and Pollack, 1999]. The assumption of categorical goals (Assumption 3) has been removed in utility-based and decision-theoretic planning systems, which can generate plans for goals with rich, time-dependent utility functions [Williamson and Hanks, 1994, Haddawy and Hanks, 1998]. Work on modeling planning as a (fully or partially observable) Markov decision process (MDP) [Boutilier et al., 1999] can be seen as aimed at simultaneously removing the first assumption.

Other recent work has addressed the assumption of a static environment (the fourth and fifth assumptions). One approach has been to rely on techniques for planning under uncertainty, folding exogenous changes into the predicted outcomes of actions [Blythe, 1996, Hanks et al., 1996]; this is also the approach used in the MDP framework. Another body of work, often called “reactive planning” or plan-execution systems [Firby, 1994, Firby, 1996, Gat, 1992, Georgeff and Ingrand, 1989],

addresses run-time system behavior, and deals with the problem of dynamic environments by supplementing high-level plans, such as those produced by classical planners, with mechanisms for translating those plans into low-level behaviors that are responsive to changes in the world. For the most part, however, this work has concentrated on the effective execution of current activities, but has not provided sufficient consideration to issues of managing future commitments. Work on deliberation scheduling [Horvitz, 1997, Zilberstein and Russell, 1996, Greenwald and Dean, 1994, Dean and Boddy, 1988], also gives up Assumption 5, and develops techniques for deciding which goals to attend to when. But to date these techniques have had somewhat limited applicability, and they have not been applied specifically to questions of deciding how much detail to include in plan, nor to managing the process of replanning—although in principle deliberation scheduling techniques could be applied to such problems, if we could accurately model the costs and benefits of individual plan expansion and replanning tasks. Finally, while there has been recent work on developing planners that reason about rich temporal constraints [Bacchus and Kabanza, 1996], it has again focused primarily on the problem of generating plans for a fixed goal, to be subsequently executed in a static environment.

In sum, while the planning community has made significant progress on developing plan generation mechanisms that do not require the strong assumptions made in earlier work, it has continued, for the most part, to focus on the problem of plan generation. Certain exceptions exist, as noted in the previous paragraph, but the majority of research done continues to focus on algorithms for more efficient plan generation. Yet, when the assumptions of classical planning are abandoned, it becomes important not only to rethink plan generation algorithms, but also to consider a wider range of reasoning processes that are involved in managing plans in dynamic, multi-agent environments.

2 An Example

We can illustrate the types of reasoning required in plan management with a simple example of kind of reasoning that humans seem to be capable of performing on a daily basis.

Suppose you have a class to teach next Monday morning, and sometime the week before, you decide to prepare your lecture for that class over the weekend. You don't yet decide exactly when you will prepare the lecture, because your other weekend plans aren't fixed yet, but you are confident that you will find some time over the weekend for lecture-preparation. On Friday, before you leave the office for the weekend, you decide on the general topic for the lecture, so that you can bring home the books you'll need to do the lecture preparation. Saturday morning you decide to spend the day running errands and to go to a movie that evening; you will prepare your lecture sometime Sunday. However, Saturday afternoon, you receive a phone call from a friend, informing you that she has an extra ticket to the football game on Sunday afternoon. You are not sure whether you will be able to finish your lecture Sunday morning. You could attempt to do that, and only use the football ticket if you are successful. However, you realize that this is unfair to your friend, who could find someone else with whom to go to the game if you let him know today. You therefore revise your plans, and decide not to go to the movie, but instead to begin preparing your lecture Saturday evening. If you don't finish, you'll still have Sunday morning to complete it, before going

to the football game.

Although this scenario might appear frivolous, there are two important reasons for trying to understand the reasoning processes that support it and similar cases. The first is as an exercise in cognitive science: the ability to manage one's activities in a dynamic environment is an essential component of human cognitive capacity, and thus worth exploring as part of the effort to understand the human mind. The second is as a prerequisite to designing better computational tools. It is generally agreed that intelligent autonomous agents would be extremely useful in applications ranging from military to industrial to educational. But for many of these applications, the agents involved will need to be able to perform the kinds of plan management reasoning illustrated in our example.

So let's consider the kinds of reasoning tasks that are involved. The intelligent agent—"you" in this story—must be capable of performing (at least) the following reasoning tasks:

Plan Generation: You certainly need to be able to perform classic plan generation. In the story, for instance, you need to recognize that a precondition for preparing the lecture at home is that you have the materials you need at home; therefore you need to plan to bring them home with you.

Commitment Management: Intuitively, at least, it seems that agents don't just generate new plans, but they also commit to them. By committing to a plan \mathcal{P} , an agent tends to rule out new options that would conflict with \mathcal{P} . Having committed to teach your class on Monday, you will in general rule out activities that preclude your teaching that class. But commitment is not absolute: in the current story, you do abandon your commitment to go to the movies, in order to be able to attend the football game. Agents need to be able to reason about the relative strength of their commitments, to decide which, if any, they are willing to give up.

Environment Monitoring: In a dynamic environment, many events may occur which either indicate a problem with your existing commitments or suggest a new opportunity. However, in general you will not be able to attend to all of these, but must somehow focus on those that are potentially most relevant and significant. In the current story, you attend to the invitation to attend the football game. But there are plenty of other potential opportunities that you ignore (and that we've therefore left out of the story), such as the call from a telemarketer offering you a new credit card, and the "for-sale" signs you see posted in front of neighborhood houses.

Alternative Assessment: Once you have determined that the environment presents a new opportunity, you need to be able to assess the costs and benefits of the options with which you are presented, and you need to do this taking account of the context in which those options will be performed. Going to the football game not only provides the benefit of enjoying the game, but also incurs a cost, since it requires you to skip the movie you had planned to see.

Plan Elaboration: In dynamic environments, agents often have commitments to partially specified plans, and may even begin executing those plans before they are complete. However,

there are constraints on what must be elaborated when. You need by Friday afternoon to have elaborated your lecture plan enough to determine which books and papers to carry home with you. On the other hand, you can defer your decision about exactly when to prepare your lecture. But once until you receive the phone call about the football tickets, you need to make a firmer commitment to the lecture preparation time, in order to ensure you can both complete the preparation and attend the game.

Meta-Level Control: For some activities, you will want to do a lot of careful planning, while for others, you may be willing to live with a less-than-optimal solution. For example, you may not bother to spend the time to determine the minimal-distance route that allows you to carry out all your errands.

Coordination with Other Agents: When environments have multiple agents in them, a wide range of issues arise involving the coordination of your own plans with those of the other agents. In our story, you decide not to adopt the wait-and-see plan of going to the movies and then starting your lecture preparation Sunday morning, going to the football game only if you've completed the preparations. This plan is unacceptable because it has a negative impact on your friend's plan to find someone with whom to attend the football game.

Reasoning tasks such as these have been the focus of our work on plan management. In the remainder of this paper, we will briefly describe the challenges posed by these tasks, and will survey some of the work we have done to date in addressing them. Our discussion will necessarily be informal and will rely on numerous examples; the reader can consult the papers we cite for technical details. We will omit discussion of three of plan management tasks listed: plan generation, about which a huge amount has already been written; meta-level control, for which we refer the reader to [Dean and Boddy, 1988, Russell and Wefald, 1991, Greenwald and Dean, 1994, Zilberstein and Russell, 1996, Horvitz, 1997]; and coordination issues, which are discussed in the paper by Grosz, Hunsberger, and Kraus in this same volume [Grosz et al.,]. After sketching our approach to individual reasoning processes, we will address the question of integrating these approaches, by describing a plan management system we are developing.

3 Commitment Management

Our earliest work on plan management was done a decade ago, in collaboration with Michael Bratman and David Israel. It addressed the significance of commitment for resource-bounded agents in dynamic environments [Bratman et al., 1988, Pollack, 1992]. Building on earlier work by Bratman [Bratman, 1987], who had argued that commitment to future plans helps a resource-bounded agent frame its subsequent reasoning and thereby limit the amount of computational resources it requires, we developed an architecture later called IRMA: the Intelligent Resource-Bounded Machine Architecture, depicted in Figure 1. The key idea in IRMA is that agents should, in general, bypass full-fledged deliberation about new options that conflict with their existing commitments, unless those options can be easily recognized as potentially special in some way. To

achieve this balance between commitment to existing plans and sensitivity to particularly important new options, we postulated within IRMA a *filtering mechanism* with two components: the first checks the compatibility of a new option against existing plans, while the second checks whether the new option is *prima facie* important enough to warrant deliberation even under incompatibility.

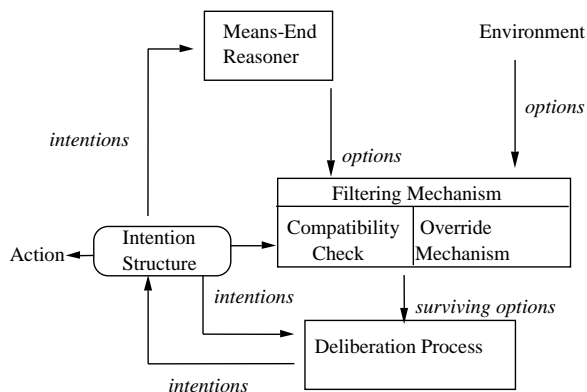


Figure 1: The Intelligent Resource-Bounded Architecture, from [Bratman et al., 1988]

Note that an IRMA agent may sometimes behave suboptimally, i.e., perform actions other than those it would have selected if it had engaged in full-fledged deliberation. It may also sometimes perform wasted reasoning, i.e., engage in deliberation about a new option that results in a decision to maintain the *status quo*. Nonetheless, an IRMA agent may attain a better overall level of performance than an agent who attempts full-fledged deliberation about every option it faces, because the time costs associated with such deliberation would typically result in significant missed deadlines and opportunities.

Our initial implementation of an IRMA-based agent showed the feasibility of the concept, but our experimentation was limited to a simple, highly abstract domain (the Tileworld) [Pollack and Ringuette, 1990]. Similar results were found by other researchers [Kinny and Georgeff, 1991, Kinny et al., 1992]. We later generalized the testbed to include multiple agents, and we found preliminary evidence that the IRMA notion of filtering could also be a useful mechanism for facilitating agent coordination [Ephrati et al., 1995], but again, our experiments were limited to an abstract domain. However, many of the ideas in IRMA are included in the dMARS systems developed at the Australian Artificial Intelligence Institute by Michael Georgeff and his colleagues [Rao and Georgeff, 1995]; dMARS has been used in a variety of real-world applications ranging from space shuttle malfunction handling to air traffic control to air combat simulation. We are also using a filtering mechanism in our Plan Management Agent (PMA), described later in this paper. Although we have modified our view of the exact nature of filtering—we no longer view compatibility as an all-or-nothing property, but instead reason about the compatibility of various layers of a hierarchical plan—we still subscribe to the view that it is important for resource-bounded agents in dynamic environments to maintain fairly stable sets of commitments, which guide and focus their subsequent reasoning and actions.

4 Environment Monitoring

The theory of filtering, as embodied in IRMA, suggests a way of screening out options that have already come to your attention. In fact we believe that an agent’s commitments play an even stronger role: they tend to restrict the range of options that the agent attends to in the first place. For instance, when you wake up on a typical weekday morning, you already have a plan for the day—go to work, go to the gym, pick up the kids at school, etc.—and you do not spend a lot of time seriously considering conflicting alternatives, such as taking a spur-of-the-moment trip to Hawaii. It’s not just that these options arise and then are filtered out, it’s rather that in general the options don’t even arise in the first place.

So what’s involved in monitoring “the right” features of the environment? As far back as the STRIPS planning system, researchers suggested that if an agent was engaged in plan-based behavior, the plan itself could provide guidance about what to monitor. Thus triangle tables [Fikes et al., 1972], which map the existing state of the world to preconditions of steps in the plan, probably represent the earliest approach to execution monitoring. A more recent example is the SAGE system for information gathering, which also bases its decision of what to monitor on the structure of its current plans [Knoblock, 1995].

In our own work [Veloso et al., 1998], done jointly with Manuela Veloso and Michael Cox, we have argued that an agent should carefully monitor not only environmental features that affect the plans it holds, but also features of the plans it has rejected during prior reasoning. More specifically, we view planning as a decision-making process, during which the rationale behind each decision should be recorded. Plan rationales will often include references to conditions that resulted in the rejection of an alternative plan, in favor of the one that was actually adopted.

For instance, suppose that you plan to use frequent flier miles on an airplane ticket that costs \$800. If an airline price war then erupts, and the price of the ticket drops substantially, say to \$200, you might prefer to change your plan and pay for the ticket, saving your frequent flier miles for a more expensive flight. Note that the change in price will not affect the success of your plan—you still could fly with the ticket you hold. But the rationale for your original decision to use frequent flier miles was that the flight was expensive; this rationale now no longer holds. Rationale-based monitoring can thus not only prevent the failure of previous plans, but can also lead to higher-quality plans.

Rationale-based monitoring is a three-stage process involving (i) generating monitors that represent environmental features that affect plan rationale; (ii) deliberating, whenever a monitor fires, about whether to respond to that monitor; and (iii) transforming the plan as warranted. In [Veloso et al., 1998], we described a preliminary implementation of this approach in the Prodigy planning system [Veloso et al., 1995]; more recently, with Colleen McCarthy, we have adapted the idea to a partial-order causal-link planner [Pollack and McCarthy, 1999].

However, approaches to monitoring that are based on the agents’ existing plans are necessarily incomplete. Although, as we have suggested, agents do not attend to every feature of their environment, neither do they attend only to environmental features directly related to their previous planning episodes. You don’t typically consider taking a trip to Hawaii every morning, even if your

morning newspaper always carries advertisements for airlines, but once in a long while, you just might entertain this possibility—especially if you live in Pittsburgh and it’s early March. A theory of how that happens has not yet been formulated.

5 Alternative Assessment

Suppose that a new option has arisen, that you have attended to it, and that you have determined that it is worth deliberating about. In order to perform this deliberation, it is necessary for you to assess the costs and benefits of the option. Moreover, this assessment must be done taking account of the context of your existing commitments: in our running example, you need to recognize that attending the football game will require you to give up the movie you’d planned to see. In this case, the context increases the cost of your new option, relative to its cost if done in isolation. But context can also decrease the cost of a new option—for example if some of the steps required for the new option can be merged with steps you already intend.

The theory of rational choice, as formulated in the economic and philosophical literature [Jeffrey, 1983, Keeney and Raiffa, 1993], provides a richly developed model of alternative assessment. In this theory, agents are seen as evaluating alternative actions by reference to a probability distribution over their possible outcomes together with a utility function defined on those outcomes: in the simplest case, the agent combines probability and utility into a notion of expected utility defined over actions, and then chooses some action whose expected utility is maximal.

However, classical decision theory does not completely address certain challenges that arise in designing alternative assessment procedures for agents in dynamic environments. We have been developing a framework that does [Horty and Pollack, 1998], and it differs from classical decision theory in two important ways. First, where decision theory assumes that the utility of an outcome is given as part of the background setting, we observe that the overall desirability of an option presented to an agent is often not immediately apparent; we are explicitly concerned with the mechanism through which it might be discovered. In particular, we have so far concentrated on the case in which the option presented to the agent has a known benefit, but requires some effort—the execution of a plan—for its achievement. In order to evaluate the overall desirability of the option, the agent thus has to arrive at some assessment of the cost involved in achieving it.

Second, we require that our theory accommodate the fact that agents have computational resource bounds. Towards this end, we cast it within the theoretical framework that models resource-bounded agents as always operating against the background of some current set of intentions, as discussed above in the section on Commitment Management. In contrast to standard decision theory, where actions are evaluated in isolation, we develop a model in which the options presented to an agent are evaluated against a background context provided by the agent’s current plans—commitments to future activities, which, at any given point, may themselves be only partially specified.

At the center of our approach is a specification of the cost of a new option in context, which we take simply to be its marginal cost. That is, the cost of carrying out a new plan \mathcal{P} in the context of existing commitments \mathcal{C} is simply the cost of carrying out both \mathcal{P} and \mathcal{C} , less the cost of carrying

out \mathcal{C} alone: $\kappa(\mathcal{P}/\mathcal{C}) = \kappa(\mathcal{P} \cup \mathcal{C}) - \kappa(\mathcal{C})$, where κ denotes the cost function, and \mathcal{P}/\mathcal{C} denotes \mathcal{P} in the context of \mathcal{C} .

To use this definition, we need to define the cost of a plan. We do this by assuming that primitive actions have specific costs, and that the cost of a complete, fully scheduled plan is the sum of the costs of all the distinct steps in it. The requirement that we only count distinct steps is important, because if an agent can merge steps of the same type, it only incurs the cost of a single such step. Consider the example depicted in Figure 2. For instance, if while running your errands, you drive to the mall to buy a new shirt, and at the same time drive to the mall to buy sneakers, you only incur the cost of one trip to the mall. In traditional plan generation, this would be handled by a step re-use function. When a new plan is added to an existing context of commitments, explicit consideration of step merging becomes crucial. For further discussion of step merging, including algorithms for efficient merging, see [Yang, 1997]. We associate the cost of an incomplete or unscheduled plan with the cost of the least expensive way in which it might be carried out; i.e., the cost of an arbitrary plan \mathcal{P} is the cost of the least expensive fully scheduled, complete plan \mathcal{Q} such that \mathcal{Q} is a refinement of \mathcal{P} , where refinements are defined as in [Kambhampati et al., 1995].

Although we have defined the notion of the cost of a plan as the least expensive method of executing it, we do not necessarily assume that a planning agent knows the true cost either of its background plan or of any new options under consideration. Instead, the agent may only have an estimate of the cost of its plans. We view estimates as intervals that bound the true cost of a plan; they are thus related to the interval measures of plan cost used in the decision-theoretic plan generation literature [Williamson and Hanks, 1994, Haddawy et al., 1995, Goodwin and Simmons, 1998]. We have developed an algebra for reasoning about cost estimates: in particular, it specifies how to subtract estimates, so that an estimate of $\kappa(\mathcal{P}/\mathcal{C})$ can be derived from estimates of $\kappa(\mathcal{P} \cup \mathcal{C})$ and $\kappa(\mathcal{C})$, according to the definition above. The derived interval estimate of cost in context is useful because, in many cases, it allows an agent to accept or reject an option without calculating its specific cost, as illustrated in Figure 3. Suppose that an agent with background plan \mathcal{C} is considering a new option \mathcal{P} with benefit $\beta(\mathcal{P})$. Suppose further that given its estimated costs (intervals) for \mathcal{C} and \mathcal{P} it is able to derive a estimated cost interval for \mathcal{P}/\mathcal{C} . If the agent knows that the benefit of \mathcal{P} is greater than the upper bound of this interval, then he is justified in adopting it, since \mathcal{P} 's cost in context is necessarily less than its benefit; likewise, the agent is justified in rejecting \mathcal{P} if its known benefit is less than the lower bound of the estimated interval of its cost in context. It is only when the benefit is inside the interval that the agent is forced to refine its estimates further before making a rational decision. We have developed anytime algorithms for this refinement step, which, again, we are using in PMA.

6 Plan Elaboration

We have already seen that in dynamic environments, planning and execution will often be interleaved. Agents must therefore be able to decide how much detail to include in their plans now and when to add more detail. This question has largely been ignored in the AI literature, although Blythe's work [Blythe, 1998] is a notable exception. With Nilufer Onder, we have so far been inves-

tigating a restricted form of it [Onder and Pollack, 1999, Onder, 1999]: given that a large number of contingencies may arise during the execution of a plan in a dynamic, uncertain environment, which contingencies should be dealt with when the plan is initially formed, and which should be deferred and dealt with on an as-needed basis at execution time?

To address this question, we associate a contingency with the failure of an action in a plan to have its desired outcome. In our running example, you plan to write your lecture Saturday evening, but you recognize the possible failure of this action: you may not be able to complete the lecture on time. You therefore create a contingency plan, to complete it Sunday morning before the football game, if need be.

In general, when the agent determines that a generated plan includes an action that may fail, there are three approaches to improving it. *Corrective repair* involves planning a response if the failure should occur. *Preventive repair* involves enhancing the plan to include steps that lessen the likelihood of the failure. (In conformant planning, sufficiently many steps are added to guarantee that the plan will succeed, regardless of the action's outcome [Goldman and Boddy, 1996].) Finally, *replacement* involves modifying the plan, to remove the potentially problematic action and replace it with an alternative.

To decide which contingencies to focus on, we rely on two facts: first, contingencies in a plan may have unequal probability of occurrence; and second, a plan may have multiple goals, each of which has some associated value. It therefore makes sense to focus on those repairs that will have the greatest expected positive impact on the plan. To compute this, we first determine the value of all the goals that the agent will fail to achieve if some contingency C occurs, and then weight this by the probability of C 's occurrence. Subtracting this value from the optimal value of the plan gives us the expected value of the plan under the contingent failure. Next, we estimate the expected value of the plan that will result if we perform a certain type of repair. The difference between these two numbers gives us the expected value of performing the repair. We have developed algorithms for performing these computations, as well as for doing the actual process of generating contingent plans, and have also developed a number of heuristics to increase the efficiency of this process.

Of course, in reality the importance of having contingency plans will also depend on the likely difficulty of replanning “on-the-fly” for particular contingencies: even if a contingency is relatively likely to occur and to affect high-value goals, if it is extremely easy to correct during plan execution, it may not be necessary to plan in advance for it. This issue will be dealt with in later work.

7 Integration in PMA: The Plan Management Agent

Each of the individual reasoning processes described above is important, but it is also important to be able to integrate these processes with one another. To explore the challenge of integration, as well as to build a realistic platform on which to experiment with our ideas, we have been building PMA: the Plan Management Agent [Pollack et al., 1999]. PMA is an intelligent software system that is intended to aid a user in managing a potentially large and complex set of plans. It is thus related to two major classes of software systems: personal electronic calendar systems and workflow systems.

Commercially available electronic calendar systems, published by major software companies, essentially provide electronic interfaces to written calendars. They typically have advanced GUIs, and provide linkages to contact databases and email; some also provide capabilities for automated meeting scheduling. However, these systems suffer from a highly impoverished representation for activities: they can only model simple events and recurring simple events. Simple events are blocks of time with a single property—“free” or “busy”; a “free” activity is allowed overlap with other “free” activities, but a “busy” activity cannot overlap with other activities. Recurring simple events are simple events that recur at regular intervals, e.g., every Tuesday from 4-5pm. Labels and textual information can be attached to each event, but these are not used in any sophisticated way by the system; they are stored only for the human user’s information.

Workflow systems [Georgakopoulos et al., 1995, Nutt, 1996, Mahling et al., 1995] constitute another class of systems aimed at helping users manage their routine activities. In contrast to personal calendar systems, workflow systems employ richly structured representations of activities (or processes), and they use these representations to ensure that information and tasks flow to the appropriate people in an organization in a timely fashion. Modern workflow systems support “document management, imaging, application launching, and/or human coordination, collaboration, and co-decision” [Georgakopoulos et al., 1995, p. 121]. On the other hand, they tend to have limited capabilities for handling uncertainty, for replanning when a problem is detected, and for reasoning about the relative value of alternative ways to perform a given task. PMA is being designed to include just these sorts of reasoning capabilities, using the techniques discussed above.

PMA is a joint project of our research groups at the University of Pittsburgh and the University of Maryland, and includes the efforts of Ioannis Tsamardinos and Carl Anderson. Other efforts to apply AI plan generation and execution technology to develop intelligent workflow-style systems include the IWCM project at SRI International [Berry and Myers, 1998] and the Enterprise Project at the Artificial Intelligence Applications Institute [Drabble et al., 1998, Stader and Jarvis, 1998]. Like PMA, the IWCM project is relatively new, and comparisons between the projects are not yet possible. Although some of the functionality of PMA overlaps with that of the Enterprise project, the approaches taken in the two efforts are quite different.

To illustrate the behavior of PMA, and show how it includes the types of reasoning processes already discussed, we describe a sample interaction with it. PMA has knowledge of the structured activities—the “plans” or “procedures”—that its user typically performs. For instance, a PMA for use in a physician’s office would know the steps involved in carrying out diagnostic procedures, preparing a patient for surgery, and handling insurance forms. The activity of preparing a patient for surgery might include, say, organizing a preliminary battery of tests, assembling and scheduling the surgical team, booking the operating room, etc. Many of these tasks would themselves decompose into structured activities: carrying out a single test might involve scheduling that test, tracking the lab work, entering the results into the patient’s record, and calling the patient for follow-up work if necessary.

Imagine that a physician (or nurse) specifies the goal of performing a particular diagnostic test on a patient. PMA immediately posts commitments to various tasks pertaining to that goal in an internal knowledge base—the “schedule.” It also updates the graphical display that includes

a calendar and to-do list. In this example, the posted commitments might include scheduling the test, obtaining the necessary background information before the test date, reminding the patient 48 hours before the test date, and so on. Once the user indicates that the test has been scheduled for a certain date—December 15, say—the temporal information associated with the related procedures will be updated accordingly; for example, a calendar entry will then appear reminding the user to notify the patient on December 13. Furthermore, if this test is just one of a battery of tests, and the scheduled December 15 date places it too near another test with which it might interfere, PMA will notice this conflict and notify the user, suggesting an alternative schedule that avoids the conflict. It may also suggest to the user that an operating room should be scheduled now, even though the actual deadline for the reservation has not yet occurred, because there is limited flexibility in the schedule to handle the situation should the operating rooms become unavailable at the desired time.

This scenario illustrates the main capabilities that we are building into PMA:

- The PMA user can commit to activities that have rich temporal and causal structure. She does not need to specify separate commitments to each component of the activity.
- The PMA user can make partial commitments: for instance, she can commit to performing a particular activity without yet specifying the exact time at which it will occur, or she can specify that she wants to commit to a particular goal, without yet specifying exactly which plan she will use to achieve that goal.
- When the user extends her commitments (e.g., by specifying a particular time or a particular plan for a goal), PMA propagates the new commitment to all affected parts of the activity. In the example above, when the user specifies that the test should be scheduled for Dec. 15, a patient reminder is automatically scheduled for Dec. 13.
- Whenever the user attempts to form a new commitment, PMA performs temporal and causal reasoning to determine whether it is consistent with the user's previous commitments. If PMA determines that certain additional constraints are required to ensure consistency, it notifies the user of those additional constraints, which we call "forced constraints." If PMA determines that there is a conflict between the new commitment and prior commitments, it suggests ways to resolve the conflict.
- PMA can assess the cost of executing a plan in the context of existing commitments, and notify the user if the cost fails to exceed some specified threshold.
- As time passes, PMA monitors the execution of the user's activities, and reminds the user when deadlines are approaching. It also reasons about the tightness of the schedule: for instance, if there is little slack at some future periods, PMA may suggest taking early action.

The initial version of PMA has been implemented on a Pentium platform, using Allegro Common Lisp for Windows. To date, we have implemented the first five capabilities just listed, using the methods discussed earlier in the paper, but so far they are only implemented for non-hierarchical

plans. The extension of these capabilities to hierarchical activities, and the implementation of execution monitoring, are part of our ongoing effort. Our knowledge base so far contains plans for an academic user, rather than a medical one. Figure 4 illustrates the user's view of PMA.

8 Conclusions

As the strong assumptions of classical planning are being abandoned, the possibility of constructing powerful, autonomous agents is increasing. We have argued however, that better planners, which can generate plans for dynamic, uncertain, multi-agent environments, are not enough. There's more to life than plan generation! Autonomous agents in dynamic, multi-agent environments also need to be able to manage the plans they generate. They need to determine which planning problems and opportunities to consider in the first place. They need to be able to weigh alternative incomplete plans, and to decide among competing alternatives. They need to be able to form incomplete plans now, adding detail later, and they thus need to be able to decide how much detail to include now and when to add more detail. They need to be able to integrate plans with one another, and to decide when to treat an existing plan as an inflexible commitment and when, instead, to consider modifications of it. And they need to be able to do all this in a way that comports with the inherent bounds on their computational resources.

We have described some of the challenges inherent in these reasoning tasks, and sketched some of the work we have done to address them. We also described PMA, the Plan Management Agent, a system we are developing to ground our theoretical work, by providing us with a platform for integrating our techniques and exploring their value in a realistic problem.

But we have only scratched the surface and there remains a lot more work to be done on modeling and developing computational techniques for plan management in dynamic, multi-agent environments.

Acknowledgements

We would like to thank our students who have participated in various aspects of the work described here: Colleen McCarthy, Nilufer Onder, and Ioannis Tsamardinos (University of Pittsburgh), and Carl Anderson (University of Maryland). We also thank our other collaborators on the papers cited here, and our colleagues who provided feedback on several versions of this talk that were given in the last year. Our work has been supported by the Air Force Office of Scientific Research, contract F49620-98-1-0436 (to Pollack), and by the National Science Foundation, grants IRI-9619579 (Pollack) and IRI-9619562 (Horty).

References

- [Bacchus and Kabanza, 1996] Bacchus, F. and Kabanza, F. (1996). Planning for temporally extended goals. In *Proceedings of the 13th National Conference on Artificial Intelligence*.
- [Berry and Myers, 1998] Berry, P. M. and Myers, K. L. (1998). Adaptive process management: An AI perspective. In *Proceedings of the Workshop Towards Adaptive Workflow System*. Available at <http://www.ai.sri.com/berry/publications/cscw98-sri.html>.
- [Blythe, 1996] Blythe, J. (1996). Event-based decompositions for reasoning about external change in planners. In *Proceedings of the 3rd International Conference on AI Planning Systems*.
- [Blythe, 1998] Blythe, J. (1998). *Planning under uncertainty in dynamic domains*. PhD thesis, Carnegie Mellon University. Available from the author's Web page: <http://www.isi.edu/blythe/papers.html>.
- [Boutilier et al., 1999] Boutilier, C., Dean, T., and Hanks, S. (1999). Decision theoretic planning: Structural assumptions and computational leverage. *Journal of Artificial Intelligence Research*. To appear.
- [Bratman, 1987] Bratman, M. E. (1987). *Intention, Plans and Practical Reason*. Harvard University Press, Cambridge, MA.
- [Bratman et al., 1988] Bratman, M. E., Israel, D. J., and Pollack, M. E. (1988). Plans and resource-bounded practical reasoning. *Computational Intelligence*, 4:349–355.
- [Collins and Pryor, 1995] Collins, G. and Pryor, L. (1995). Planning under uncertainty: some key issues. In *Proceedings of the 14th International Joint Conference on Artificial Intelligence*.
- [Dean and Boddy, 1988] Dean, T. and Boddy, M. (1988). An analysis of time-dependent planning. In *Proceedings of the Seventh National Conference on Artificial Intelligence*, pages 49–54, St. Paul, MI.
- [Drabble et al., 1998] Drabble, B., Lydiard, T., and Tate, A. (1998). Workflow support in the air campaign planning process. In *Proceedings of the Workshop on Interactive and Collaborative Planning, AIPS98*, Pittsburgh, PA.
- [Draper et al., 1994] Draper, D., Hanks, S., and Weld, D. (1994). A probabilistic model of action for least-commitment planning with information gathering. In *Proceedings of the 10th Conference on Uncertainty in Artificial Intelligence*.
- [Ephrati et al., 1995] Ephrati, E., Pollack, M. E., and Ur, S. (1995). Deriving multi-agent coordination through filtering strategies. In *Proceedings of the 14th Joint International Conference on Artificial Intelligence (IJCAI)*.

- [Fikes et al., 1972] Fikes, R. E., Hart, P. E., and Nilsson, N. J. (1972). Learning and executing generalized robot plans. *Artificial Intelligence*, 3(4):251–288.
- [Firby, 1994] Firby, R. J. (1994). Task networks for controlling continuous processes: Issues in reactive planning. In *Proceedings of the Second International Conference on AI Planning Systems*, pages 49–54.
- [Firby, 1996] Firby, R. J. (1996). Modularity issues in reactive planning. In *Proceedings of the Third International Conference on AI Planning Systems*, pages 78–85.
- [Gat, 1992] Gat, E. (1992). Integrating planning and reacting in a heterogenous asynchronous architecture for controlling real-world mobile robots. In *Proceedings of the Tenth National Conference on Artificial Intelligence*, pages 802–815.
- [Georgakopoulos et al., 1995] Georgakopoulos, D., Hornick, M., and Sheth, A. (1995). An overview of workflow management: From process modeling to workflow automation infrastructure. *Distributed and Parallel Databases*, 3:119–153.
- [Georgeff and Ingrand, 1989] Georgeff, M. P. and Ingrand, F. F. (1989). Decision-making in an embedded reasoning system. In *Proceedings of the Eleventh International Joint Conference on Artificial Intelligence*, pages 972–978, Detroit, MI.
- [Goldman and Boddy, 1994] Goldman, R. and Boddy, M. (1994). Epsilon-safe planning. In *Proceedings of the Tenth International Conference on Uncertainty in Artificial Intelligence*.
- [Goldman and Boddy, 1996] Goldman, R. P. and Boddy, M. S. (1996). Expressive planning and explicit knowledge. In *Proceedings of the Third International Conference on AI Planning Systems*, pages 110–117.
- [Goodwin and Simmons, 1998] Goodwin, R. and Simmons, R. (1998). Search control of plan generation in decision-theoretic planners. In *Proceedings of the Fourth International Conference on Artificial Intelligence Planning Systems (AIPS-98)*, pages 94–101.
- [Greenwald and Dean, 1994] Greenwald, L. and Dean, T. (1994). Solving time-critical decision-making problems with predictable computational demands. In *Proceedings of the Second International Conference on AI Planning Systems (AIPS)*, pages 25–30, Chicago, IL.
- [Grosz et al.,] Grosz, B. J., Hunsberger, L., and Kraus, S. Planning and acting together. In this volume.
- [Haddawy et al., 1995] Haddawy, P., Doan, A., and Goodwin, R. (1995). Efficient decision-theoretic planning: techniques and empirical analysis. In *Proceedings of the 11th Conference on Uncertainty in Artificial Intelligence*.
- [Haddawy and Hanks, 1998] Haddawy, P. and Hanks, S. (1998). Utility models for goal-directed decision-theoretic planners. *Computational Intelligence*, 14(3):392–429.

- [Hanks et al., 1996] Hanks, S., Madigan, P., and Gavrin, J. (1996). Probabilistic temporal reasoning with endogenous change. In *Proceedings of the 11th Conference on Uncertainty in Artificial Intelligence*.
- [Horty and Pollack, 1998] Horty, J. F. and Pollack, M. E. (1998). Option evaluation in context. In *Proceedings of the 7th Conference on Theoretical Aspects of Rationality and Knowledge (TARK-98)*, pages 249–262, San Francisco. Morgan Kaufmann.
- [Horvitz, 1997] Horvitz, E. J. (1997). Models of continual computation. In *Proceedings of the 14th National Conference on Artificial Intelligence (AAAI-97)*, pages 286–293.
- [Jeffrey, 1983] Jeffrey, R. (1983). *The Logic of Decision, 2nd edition*. University of Chicago Press, Chicago.
- [Kambhampati et al., 1995] Kambhampati, S., Knoblock, C. A., and Yang, Q. (1995). Planning as refinement search: A unified framework for evaluating design tradeoffs in partial-order planning. *Artificial Intelligence*, 76(1-2):167–238.
- [Keeney and Raiffa, 1993] Keeney, R. L. and Raiffa, H. (1993). *Decisions with Multiple Objectives: Preferences and Value Tradeoffs*. Cambridge University Press, New York. First published in 1976 by John Wiley and Sons.
- [Kinny and Georgeff, 1991] Kinny, D. N. and Georgeff, M. P. (1991). Commitment and effectiveness of situated agents. In *Proceedings of the Twelfth International Joint Conference on Artificial Intelligence*, pages 82–88, Sydney, Australia.
- [Kinny et al., 1992] Kinny, D. N., Georgeff, M. P., and Hendler, J. (1992). Experiments in optimal sensing for situated agents. In *Proceedings of the Second Pacific Rim International Conference on Artificial Intelligence*, pages 1176–1182, Seoul, Korea. Korea Information Science Society.
- [Knoblock, 1995] Knoblock, C. A. (1995). Planning, executing, sensing and replanning for information gathering. In *Proceedings of the 14th International Joint Conference on Artificial Intelligence*.
- [Kushmerick et al., 1995] Kushmerick, N., Hanks, S., and Weld, D. (1995). An algorithm for probabilistic planning. *Artificial Intelligence*, 76:239–286.
- [Mahling et al., 1995] Mahling, D., Craven, N., and Croft, W. B. (1995). From office automation to intelligent workflow systems. *IEEE Expert*, 10(3).
- [Nutt, 1996] Nutt, G. J. (1996). The evolution towards flexible workflow systems. *Distributed Systems Engineering*, pages 276–294.
- [Onder, 1999] Onder, N. (1999). *Contingency Selection in Plan Generation*. PhD thesis, University of Pittsburgh Department of Computer Science. 1999.

- [Onder and Pollack, 1999] Onder, N. and Pollack, M. E. (1999). Conditional, probabilistic planning: A unifying algorithm and effective search control mechanisms. In *Proceedings of the 15th National Conference on Artificial Intelligence*.
- [Peot and Smith, 1992] Peot, M. and Smith, D. E. (1992). Conditional nonlinear planning. In *Proceedings of the First International Conference on AI Planning Systems (AIPS-92)*, pages 189–197, College Park, MD.
- [Pollack, 1992] Pollack, M. E. (1992). The uses of plans. *Artificial Intelligence*, 57:43–68.
- [Pollack and McCarthy, 1999] Pollack, M. E. and McCarthy, C. (1999). Applying rationale-based monitoring to causal-link planning. Technical Report 99-09, Univ. of Pittsburgh Dept. of Computer Science, Pittsburgh, PA.
- [Pollack and Ringuette, 1990] Pollack, M. E. and Ringuette, M. (1990). Introducing the Tileworld: Experimentally evaluating agent architectures. In *Proceedings of the Eighth National Conference on Artificial Intelligence*, pages 183–189, Boston, MA.
- [Pollack et al., 1999] Pollack, M. E., Tsamardinou, I., and Horty, J. F. (1999). Adjustable autonomy for a plan management agent. In *1999 AAAI Spring Symposium on Agents with Adjustable Autonomy*, Stanford, CA.
- [Rao and Georgeff, 1995] Rao, A. S. and Georgeff, M. P. (1995). Bdi-agents: From theory to practice. In *Proceedings of the First International Conference on Multi-Agent Systems*, San Francisco, CA.
- [Russell and Wefald, 1991] Russell, S. J. and Wefald, E. H. (1991). *Do the Right Thing*. MIT Press, Cambridge, MA.
- [Stader and Jarvis, 1998] Stader, J. and Jarvis, P. (1998). Intelligent support for enterprise modelling. Technical Report AIAI-TR-220, AIAI, Edinburgh, Scotland.
- [Veloso et al., 1995] Veloso, M. M., Carbonell, J., Pérez, M. A., Borrajo, D., Fink, E., and Blythe, J. (1995). Integrating planning and learning: The Prodigy architecture. *Journal of Experimental and Theoretical Artificial Intelligence*, 7(1):81–120.
- [Veloso et al., 1998] Veloso, M. M., Pollack, M. E., and Cox, M. T. (1998). Rationale-based monitoring for planning in dynamic environments. In *Proceedings of the Fourth International Conference on AI Planning Systems (AIPS-98)*. To appear.
- [Williamson and Hanks, 1994] Williamson, M. and Hanks, S. (1994). Optimal planning with a goal-directed utility model. In *Proceedings of the Second International Conference on Artificial Intelligence Planning Systems*, pages 176–181.
- [Yang, 1997] Yang, Q. (1997). *Intelligent Planning: A Decomposition and Abstraction Based Approach*. Springer, New York.

[Zilberstein and Russell, 1996] Zilberstein, S. and Russell, S. (1996). Optimal composition of real-time systems. *Artificial Intelligence*, 83(2).

Biographical Sketches

Martha E. Pollack is Professor of Computer Science and Professor and Director of the Intelligent Systems Program at the University of Pittsburgh. She was previously a Senior Computer Scientist at the AI Center at SRI International, and received her Ph.D. from the University of Pennsylvania. Her primary research interests are in planning, intelligent workflow and plan management, agent architectures and computational models of rationality. She is a Fellow of the AAAI, and a past recipient of the Computers and Thought Award. Her email address is pollack@cs.pitt.edu.

John F. Horty is a Professor in the Philosophy Department and the Institute for Advanced Computer Studies at the University of Maryland. He received his Ph.D. in Philosophy from the University of Pittsburgh, and has worked in a number of areas of philosophical logic and artificial intelligence, including nonmonotonic reasoning, deontic logic, the logic of action, decision theory, and the foundations of planning and practical reasoning. His email address is horty@umiacs.umd.edu.

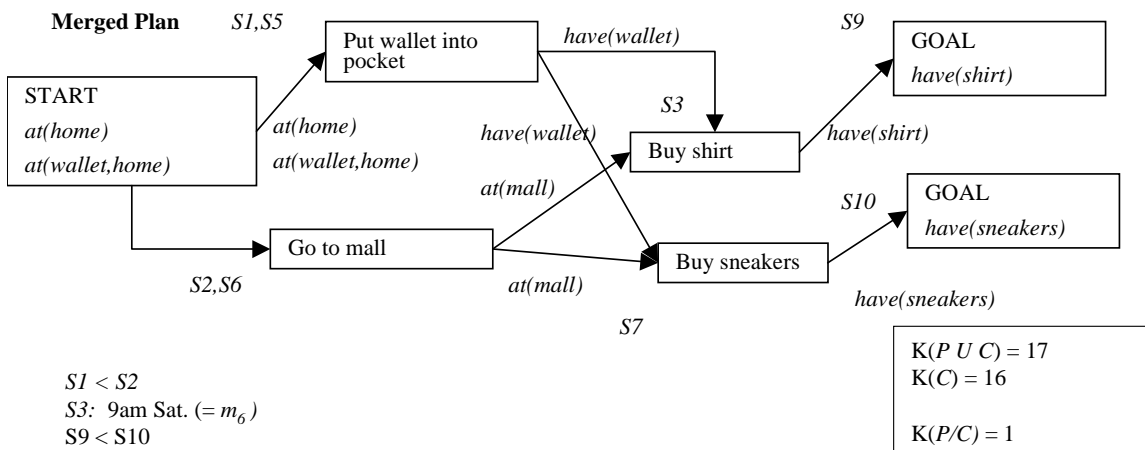
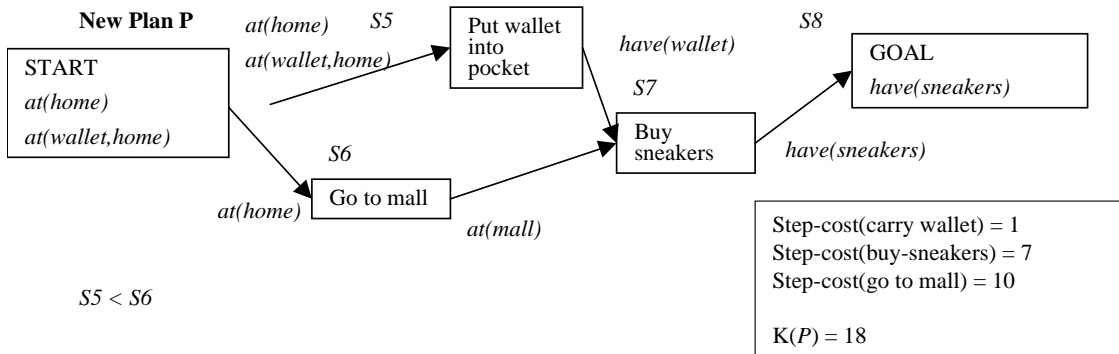


Figure 2: The Costs of Plan in Isolation and in Context

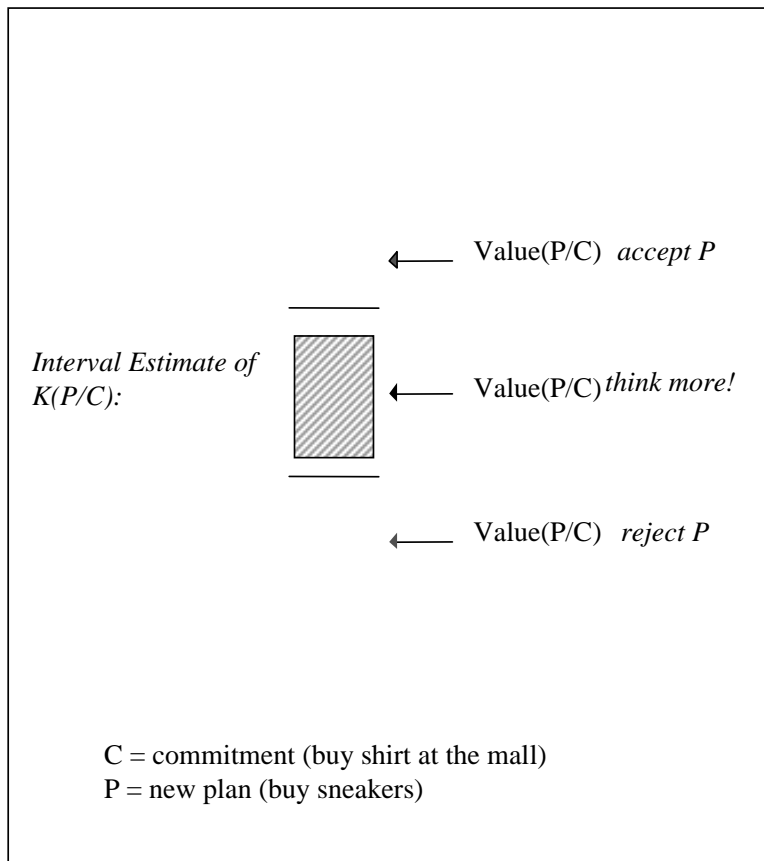


Figure 3: Using Cost Estimate Intervals to Make Decisions

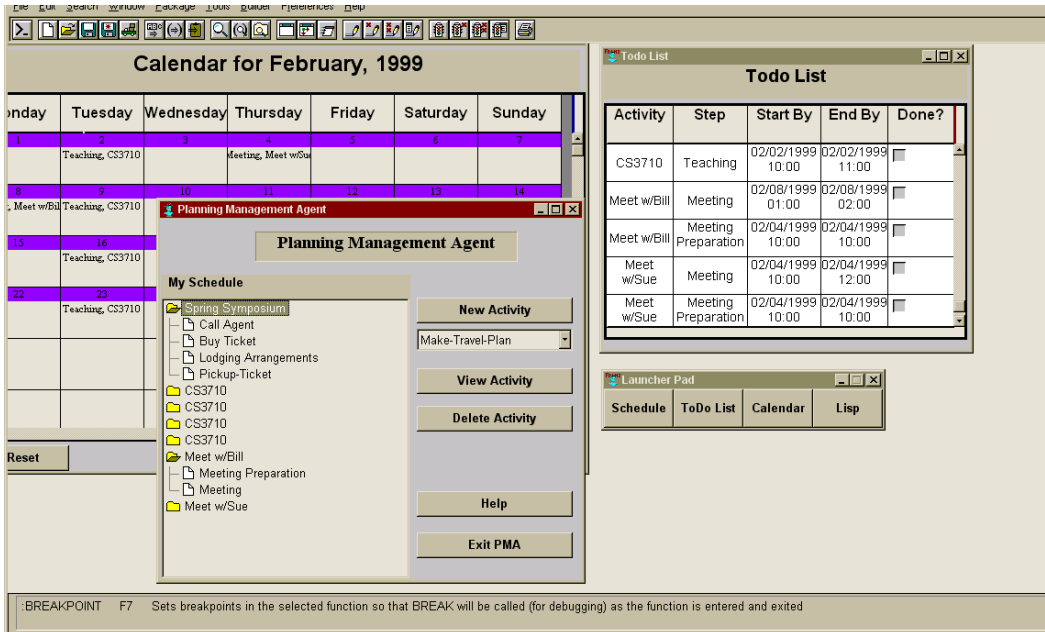


Figure 4: PMA: The User's View