Appears in *Artificial Intelligence*, 57(1):43-68, 1992

THE USES OF PLANS *

Martha E. Pollack

Department of Computer Science
and
Intelligent Systems Program
University of Pittsburgh
Pittsburgh, PA 15260

* This is a slightly revised transcription of a lecture presented upon receipt of the Computers and Thought Award at the 12th International Joint Conference on Artificial Intelligence in Sydney, Australia, on August 26, 1991. The lecture was prepared and given while the author was employed at the Artificial Intelligence Center at SRI International.

1 Introduction

The goal of Artificial Intelligence is the design of systems that behave intelligently. We want to build intelligent *actors*, not just intelligent thinkers. Indeed, it is not even clear how one could assess intelligence in a system that never acted—or, put otherwise, how a system could exhibit intelligence in the absence of action. It is therefore not surprising that planning has always been an important landmark on the map of artificial intelligence. Intelligent creatures, like us or the agents we want to design, inhabit complex environments, which they manipulate in complex ways. To do this, they need to reason about what to do. They need to plan.

Or so the commonsense story goes. Of late there have been a number of challenges to this story. Some have suggested that most of the deliberate, apparently planned quality of intelligent action is actually ephiphenomenal. They argue that intelligent action can be achieved without anything like what we would want to call reasoning. A second set of challenges is methodological. Some researchers have argued that while the process of planning may, in the end, turn out to be relevant to the design of intelligent systems, it is not yet fruitful to attempt to understand planning. There are other problems, such as enabling locomotion or accurate vision, that must be solved first. Until we solve these problems, we will not even be sure that we need to develop planning mechanisms. Brooks's Computers and Thought lecture is an important example of both kinds of challenge to AI models of planning [9].

In this paper, I will argue that, contrary to these challenges, planning deserves its central place on the AI map. I will claim that intelligent agents are planning agents, and that philosophical and commonsense psychological theorizing about the process of planning can provide useful insights into the question of agent design. The theories I have in mind are not restricted to how agents can *form* plans. Much of my research has concerned the ways in which intelligent agents *use* their plans. I will describe some of that research, and will argue that plans are used not only to guide action, but also to control reasoning and to enable inter-agent coordination. These uses of plans make possible intelligent behavior in complex, dynamic, multiagent environments.

2 Planning

We can begin by asking what exactly we mean by "planning". For many years, planning had a quite specific meaning in AI: it was the process of formulating a program of action to achieve some specified goal. You gave a planning system a description of initial conditions and a goal, and it produced a plan of action whose execution in a state satisfying the initial conditions was guaranteed to result in a state satisfying the goal. These plans were akin to recipes for achieving the goal. Your goal might be to have a chocolate cake. In the initial state, you might have eggs, milk, and chocolate, a pan and a working oven. In these conditions, a valid plan might be to go the store to buy some flour, return home, preheat the oven, mix the ingredients, pour the mixture into the pan, and put it in the oven for 45 minutes.

Traditional AI planning systems like STRIPS [22], NOAH [63], and SIPE [71], were designed to construct just this kind of plan—except usually the goal was something like a tower of three blocks instead of a cake. The point of building these planning systems was to investigate the process of constructing plans, or of concocting recipes, if you will. By and large, very little attention was paid to the uses to which the constructed plans were put.

To be fair, the computed plans were always meant to control a system's subsequent action. And in a few cases, this actually happened. Shakey the robot is a notable example [49]: after it figured out how to form a row of blocks, it went ahead and tried to do it. Sometimes Shakey ran into trouble. The plans it computed, using STRIPS, were incomplete, in that they did not cover all contingencies. In the movie that was made to document Shakey's performance [65], Shakey forms and begins to execute a plan to move a block into a doorway. At the climactic moment, a Gremlin appears and, unbeknownst to Shakey, moves a different block into Shakey's planned path. Shakey subsequently discovers the block and thus determines that its plan is unexecutable. Undaunted, Shakey computes an alternative plan for achieving its goal.

Because there is always the possibility of a system encountering unanticipated circumstances, like the block in Shakey's path, many planning systems were augmented with replanning capabilities. Usually, these consisted of techniques for reusing as much of the already computed plan as possible. The "triangle table" that Shakey exploited in replanning is an early example of a plan reuse mechanism [50]. Plan reuse has today become the object of increasingly sophisticated study (e.g.,[38]). But, although traditional systems allowed for replanning, they assumed that during replanning, the world was, once again, unchanging. Thus the fundamental picture underlying work in replanning is virtually identical to that underlying traditional AI planning. There is a reasoning module that has knowledge about the world, and about actions and their effects; it has a goal; and its job is to produce a plan of action to achieve that goal. The only difference in the case of replanning is that the reasoner may also make use of previously computed, and presumably partially correct plans. As researchers in case-based reasoning have argued, much of planning itself—not just replanning after failure—should probably operate thus [3, 27].

2.1 The Challenge of Dynamic Environments

We so far have a picture in which plans are produced by a reasoning process, a planner, and then are used to guide action. More or less hidden in this picture are some crucial assumptions. One is that an agent's goals are directly given to it, either by its designer or by some other "user" who plays the role of task-master. Moreover, goals are given to the agent one at a time. The agent is given a goal, it computes a plan for achieving it, and then, at least in principle, it executes that plan. The environment is quiescent; the agent is the only force acting on it. So nothing of significance happens while the agent is forming its plan. And nothing happens while the agent is executing that plan, except what the agent itself causes to happen.

Of course, Shakey's Gremlin was, in one sense, an exogenous force: it caused significant changes in Shakey's environment. But Shakey was never cognizant of the Gremlin. It did not know that the Gremlin had moved the block; it only knew that the world was not the way it expected it to be. From Shakey's perspective, the problem was simply that its original model of the world was wrong. So when Shakey attempted to replan, it did not worry about whether the Gremlin would reappear and once again disrupt things. Whether it was planning or replanning, Shakey assumed that the world was static. Under this assumption, plan formation (and plan reformation) can be allowed arbitrarily much time. They are constrained only by the patience of the person using, or observing, the planning system.

Shakey was lucky, because its environment was nearly static. The Gremlin was an aberration who seldom appeared. The real world, though, is not like this. Real environments are dynamic. They are populated by multiple agents that can and do effect change. Because they are dynamic, real environments may change while an agent is reasoning about how to achieve some goal, and these changes may undermine the assumptions upon which the agent's reasoning is based. While you are trying to figure out which grocery store has the best price for flour for the cake you plan to make, your children may come along and drink up the milk. If you then spend arbitrarily long trying to figure out where to buy milk, the store at which you plan to buy flour may close. And if you then spend a long time recomputing the best plan for buying flour, you just may lose your appetite for cake before you are done.

Real environments may also change while an agent is executing a plan, and, again, they may change in ways that make the plan invalid. While you are on your way to the store, the grocers may call a strike. As if this were not enough, real environments may change in ways that do not invalidate a current plan, but instead offer new possibilities for action. The problem is that the new possibilities may only be good for a limited time. It may not do for an agent to finish executing its current plan before considering whether to act on some new option. While you are mixing the ingredients for your cake, you may notice that a grease fire has started, and that you therefore have a new option: putting it out. It is probably not a good idea to wait until you have completed making the cake before you weigh the pros and cons of putting out the fire. Similarly, if your phone rings, you might not want to wait until the cake is in the oven before considering whether to answer it.

Agents in real, dynamic environments need to be receptive to many potential goals, goals that do not typically arise in a neatly sequential fashion. They need to decide what to respond to, and when. Intelligent behavior depends not just on being able to decide how to achieve one's goals, but also on being able to decide which goals to pursue in the first place, and when to abandon or suspend the pursuit of an existing goal.

Dynamic environments are ubiquitous. There is a large and growing number of AI applications, current or planned, that involve deploying agents in dynamic environments [45]. These applications include equipment-malfunction monitoring [23], process management for manufacturing [58], medical patient monitoring [33], crisis management [13], and support for combat personnel [4]. In addition, as the examples in this paper demonstrate, even humbler applications, such an office errand robot or a household assistant, require mechansims for coping with changing environments.

Over the past four or five years, there has been a growing appreciation of the problems faced by agents in dynamic environments. To some, these problems have suggested that our underlying picture is all wrong—that instead of conceiving of agents as deciding on goals and on plans for achieving them, we should envision agents whose behavior is generated by much simpler, more direct processes that involve little or no explicit symbol manipulation [1, 9, 10, 37, 59]. We are counseled to build such agents by "hard-wiring" them to do the right thing in response to perceptually detectable features of their environment. The ultimate feasibility of this approach is an empirical question, but there are many researchers, including me, who are skeptical and who believe that knowledge-compilation techniques will provide only one piece of the intelligent-behavior puzzle [14, 20, 25, 43, 57, 60].

The reasons for skepticism are wide-ranging. There are practical, engineering concerns: obviating planning puts an enormously heavy burden on a system designer. There are philosophical concerns: accounts of rationality and responsibility depend upon conceiving of behavior as resulting from explicit deliberation. There are empirical concerns: although psychologists debate the exact relation between language and cognition, many problem-solving skills, including some that resemble planning, seem to require a level of linguistic competence. So planning may only be possible for agents capable of the kind of explicit symbol manipulation inherent in human language. Perhaps most significantly, there is the tremendous explanatory power of planning theories. By beginning with the premise that agents explicitly reason about what actions to perform, we put within our grasp a large set of well-explored theories of agent behavior.

2.2 Controlling Reasoning

Rather than give up on the idea that agents reason about what actions to perform, many researchers have been investigating the principles by which agents control that reasoning. Indeed, such investigations constitute something of a renaissance in planning research. Broadly speaking, there have been two main approaches taken to the control of reasoning: explicit control and implicit control.

In explicit control, an agent reasons about the potential value of its candidate reasoning tasks. Explicit reasoning about reasoning, or meta-reasoning, has been a topic of interest in the expert-systems community for some time, dating back at least to the TEIRESIAS system [15]. Expert systems include meta-level control in order to increase the efficiency of their base-level reasoning. A medical expert system, for example, may engage in meta-reasoning about which of its candidate hypotheses are most likely to explain some observed symptoms, and can then entertain those hypotheses first.

Similarly, explicit meta-level reasoning can be used to increase the efficiency of plan formation. As far back as 1977, Feldman and Sproull suggested the use of decision-theoretic techniques to estimate the likelihood that alternative partial plans could be successfully expanded to form complete, executable plans for a given goal. Attention could then be focused on the highest scoring candidates [21]. Just a few years later, the MOLGEN system was developed; it used traditional AI planning techniques to form meta-plans that then guided domain-plan formation [69]. But this early work on meta-planning, like the concurrent work on planning and replanning, assumed an agent with one goal at a time, situated in a more or less static environment.

As I have already noted, dynamic environments stress an agent further, and so amplify the need for control of reasoning. Explicit control of reasoning for agents in dynamic environments has been explored, again primarily using the tools of decision theory [5, 16, 18, 34, 35, 39, 61, 62]. In addition to these theoretical investigations, programming systems have been built to support the implementation of explicit meta-level reasoning processes. The Procedural Reasoning System (PRS) typifies the approach [23, 24]. Using PRS, a system designer can encode both object-level and meta-level reasoning procedures in a uniform way. Blackboard systems have also been used for this purpose [17].

The alternative to explicit control of reasoning is the development of agent architectures that directly incorporate principled control strategies. An agent with implicit meta-level control will not always need to reason about what (object-level) reasoning options to pursue at any time. Instead, it will have built-in strategies for focusing its reasoning. Of course, the researcher advocating this approach needs to articulate carefully what those control strategies are—otherwise, she will be back in the position of simply counseling the design of agents that "do the right thing," but now at the meta-level. It turns out that one can articulate and provide detailed arguments for at least some rational control strategies, as I will show in Section 3 when I describe the research I have been conducting over the past few years.

2.3 Resource Bounds and Satisficing

Before turning to my research, though, I want to point out that just below the surface of all this work on the control of reasoning lies a set of ideas that pre-dates the establishment of AI as a discipline. It has now been about 35 years since Simon introduced the idea of "satisficing" [66, 67, 68]. He argued that, contrary to the prevailing slogan of decision theory, a rational agent is *not* one who always chooses the action that does the most to satisfy its goals given its beliefs. A rational agent simply does not have the resources always to determine what that optimal action is. Instead, rational agents must and should attempt only to "satisfice," that is, to make good enough, though perhaps non-optimal, decisions about their actions. Sometimes an agent just has to got ahead and act on the basis of the reasoning it has already done, even if additional reasoning might lead it to do something else.

To a large extent, the AI planning renaissance is about the tradeoff between getting around to acting, and spending enough time thinking. Control strategies, whether explicit or implicit, effectively determine when sufficient resources have been allocated to a decision problem—that is, whether a solution so far computed is "good enough," or whether the agent's current circumstances allow it the luxury of further consideration. It is worth noting that we were more or less able to ignore the challenge of resource bounds until we began to worry about systems *using* the plans they computed. As I have already said, first-generation planning systems could afford to reason indefinitely. Of course, their users might become restless, but the correctness of the plan they eventually produced was in no way threatened by the amount of time it took them to produce it. After all, the computed plan was tagged with a set of initial conditions, and it was simply assumed that those were the conditions that would hold at the time the plan was executed.

But useful planning does not happen in a vacuum. Current planning research aims at designing systems capable of inhabiting real, dynamic environments. Thus it is concerned with how systems can respond to opportunities in their environment, adopt appropriate goals, form plans for achieving those goals, and execute (at least some of) those plans successfully. Under these conditions, thinking too long can have dire effects. When TEIRESIAS provided meta-level control for the medical expert-system MYCIN, it enabled MYCIN to make its diagnosis more quickly. The correctness of the diagnosis—that is, the likelihood that it identifies the cause of a fixed set of symptoms—is independent of the amount of time it took for TEIRESIAS plus MYCIN to arrive at it.

This contrasts dramatically with the Guardian system, which is intended to monitor and assist in the management of an intensive care patient in real time [31, 33]. If Guardian spends too long diagnosing and developing a treatment for some perceived problem, its solution may be very wrong. For although the proposed treatment may have been perfect for a patient with the symptoms as described at the beginning of its reasoning process, by the end of the reasoning process, the patient's condition may be significantly different. In a dynamic environment, meta-level control is needed not only to improve the efficiency of reasoning, but also the accuracy and utility of the results of reasoning.

3 Using Plans to Constrain Reasoning

I want next to consider a specific set of proposals about the rational control of reasoning that my colleagues and I have developed over the past few years [7, 44, 55, 56]. This work begins with a question: what is the point of forming plans? As I have already stated, most agents inhabit dynamic environments. Any plan they make may be rendered invalid by some unexpected change. The problem is bad enough for plans that will be carried out immediately, like your plan to make a chocolate cake as soon as you figure out where to buy flour. The problem of potential change is exacerbated for plans for future activities, like my plan to fly to California in a week, or my plan to attend my grandmother's 80th birthday party in October. The more distant the intended execution time of some plan, the less that can be assumed about the conditions of its execution. So why should resource-bounded agents bother with such plans for the future?

This is a question that has considered in detail by Bratman [6], who argues that agents form plans in large part *because* of their resource bounds. An agent's plans, he claims, serve to frame its subsequent reasoning problems so as to constrain the amount of resources needed to solve them. Here is an example of how this is supposed to work. During the spring of 1991, I engaged in some serious deliberation about the best plan for my trip to IJCAI. This was quite a complicated deliberation problem, because I not only had to arrange an international trip, to Sydney, where the IJCAI meeting was being held, but I also knew that my family and I had to move cross-country at roughly the same time. I had to weigh the costs and benefits of adding a vacation with my husband onto the IJCAI trip against those of leaving our small daughter with my parents during that time and of not arriving at our new home until a few days before beginning a new job; I had to consider the problem of ensuring that the moving truck containing our furniture did not arrive before we did; I had to coordinate the transport of our cars cross-country; I had to think about the ramifications of starting a new job in a state of jet lag; and so on.

There were a huge number of relevant factors to consider, and I spent a fair amount of time trying to sort them out and come up with the best plan I could. And then I adopted that plan; I committed to it. Thereafter, I reasoned on the presumption that I would leave for Syndey on Aug. 19, and fly back on the 30th; that my husband and daughter would fly to the East coast on the 19th and visit with my in-laws while our furniture was in transit; and so on. I no longer weighed all the factors to continually recompute the best plan; I had settled on a plan and I was generally committed to it. My beliefs about all kinds of things continued to change, but very few of these belief changes caused me to reassess my plan. Instead, I used my plan to frame other decision problems. For example, I now had a range of dates that were suitable for the moving company to come and collect my furniture, dates that were compatible with the travel plans to which I had already committed.

This then, illustrates Bratman's fundamental insight: that agents commit to their plans, which then frame, and thereby constrain, their subsequent reasoning. Their plans tell them what to reason about: in general they try to figure out how to execute the plans to which they are already committed. And their plans tell them what not to reason about: they will not, in general, give full consideration to options that are incompatible with their existing commitments.

3.1 An Agent Architecture

These ideas served as the starting point for an architecture for resource-bounded reasoning called IRMA, the the Intelligent, Resource-Bounded Machine Architecture [7], which is depicted in Figure 1.

One assumption made in IRMA is that agents need to reason about how to achieve their goals. This is means-end reasoning, or what I earlier described as "recipe concoction." I will discuss the process of means-end reasoning further in Section 3.6. For now, imagine a program that computes plans from first principles, like STRIPS or NOAH, or one that retrieves them from a cache, like the case-based planners, or something else along these lines. What is important is that an IRMA agent will concoct recipes for its existing intentions, not for arbitrary ends. This much is not a departure from traditional AI planning systems, which compute recipes for given goals.

A second assumption in IRMA is that agents not only need to figure out how to achieve their goals, they also need to decide which goals to adopt in the first place. This is deliberation. A deliberation mechanism will be some routine that is given a small set of options, weighs their likely outcomes, and selects the option or options whose potential value exceeds some threshold. The options that are selected by the deliberation process become the agent's new intentions.

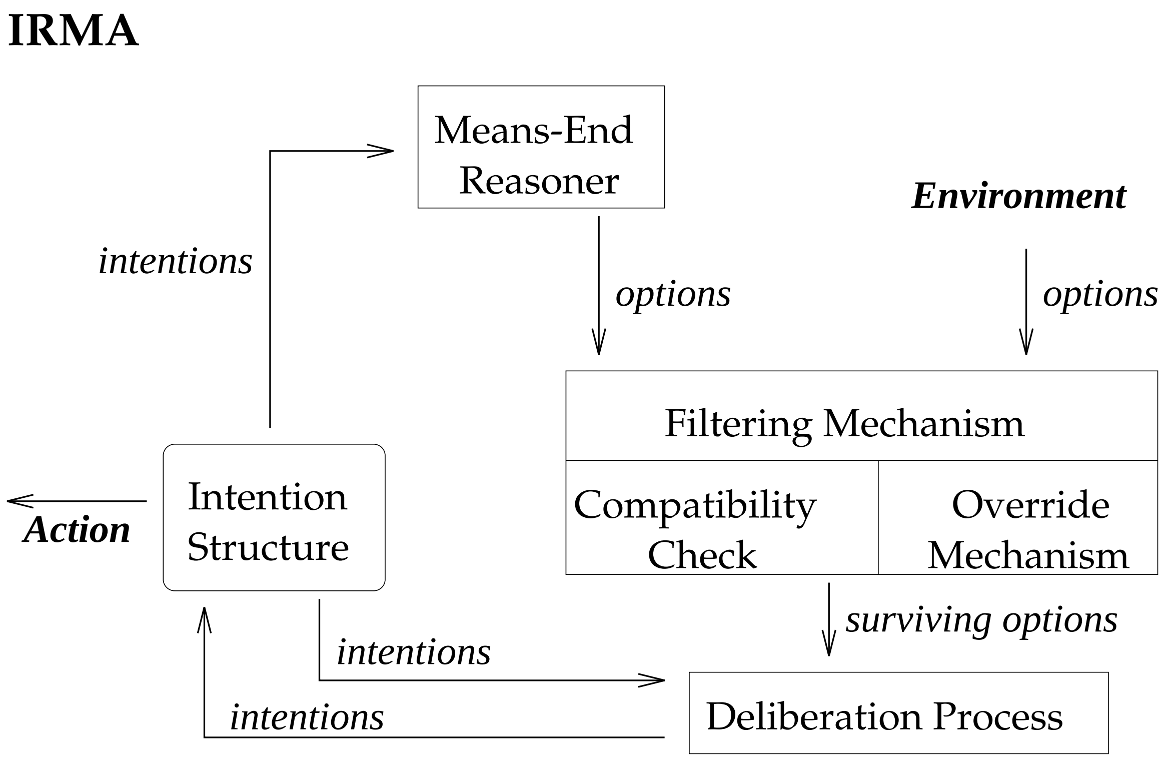

Figure 1: IRMA

But where do the options come from? On the one hand, they may be suggested by changes in the environment. When I first learned that IJCAI was going to be held in Sydney, I was implicitly presented with a new option: that of attending the conference there. Alternatively, options may be suggested by the means-end reasoner, as candidate ways of achieving adopted goals. When I began to think about how to get to Sydney, I constructed a variety of recipes: flying nonstop on Qantas, arranging to stop in Honolulu on the way, etc.

In IRMA, options from both sources are passed through a filtering mechanism, shown in the center of the diagram. The filtering mechanism performs a quick-and-dirty check to see whether an option is compatible with the agent's existing plans. If it is not, the option can be immediately dismissed, without being subject to full-fledged deliberation. After all, if the agent is committed to carrying out the plans it has adopted, there is, in general, no point weighing options that will render those plans impossible.

I have qualified this last comment, because, of course, it *is* sometimes valuable to consider options that are incompatible with one's plans. An agent that never reconsidered its adopted plans, regardless of what unanticipated changes it encountered, would be unlikely, in the long run, to behave intelligently. I illustrated this earlier with the example of the grease fire. To allow for prudent reconsideration, IRMA's filter contains a second module: the override mechanism. This mechanism encodes classes of environmental characteristics to which the agent is sensitive. An option that triggers an override sensitivity will be subject to full-fledged deliberation, even if it is incompatible with the agent's already adopted plans.

Above I described the process by which I settled on a plan to travel to IJCAI. That plan included the step of flying Qantas to Sydney on Aug. 19. A few weeks after I made my reservations for this flight, I learned that some air carriers had significantly reduced their prices. As a result, I constructed alternative options that involved flying these other carriers, and I deliberated about whether to adopt one of these alternatives. It turned out that none of the alternatives was good enough. I either had to leave weeks too early, or fly a route that involved no less than four stop-overs. In the end, I went ahead with my original plan. But the possibility of saving a large amount of money *prima facie* was sufficient for me to engage in reconsideration, despite the effort that reconsideration entailed.

My reasoning was quite different for other options that I encountered. For instance, after I had settled on my plan, a colleague asked whether I would give a lecture in her class on Aug. 23. I did not spend a lot of time deliberating about whether to revise my plans in order to give the lecture. I quickly determined that giving the lecture was incompatible with my existing plans, and that it was not *prima facie* the kind of thing worthy of serious consideration nonetheless. So I quickly dismissed the option of doing it.

3.2 The Effect of Filtering

We can now turn to the question of evaluating an IRMA agent. To do this, we track its behavior over a period of time, and record what happens each time a new option arises. Each option will be seen to be either compatible or incompatible with the agent's existing plans. We will consider only the cases in which the option is deemed incompatible. The agent may or may not end up deliberating about such an option, depending upon whether or not it triggers an override. If the agent does deliberate, that deliberation may or may not lead to a change of plans.

I have enumerated the possibilities in Table 1. Cases 1 and 2 are where the agent engages in deliberation about whether to change its plan, as indicated by a "Y" ("yes") in the first column. This is the class of situation I was in when I learned about the availability of inexpensive fares to Australia.

	Triggers override: Deliberation Occurs	Deliberation leads to change of plan	Deliberation *would have* led to change of plan	Example
1	Y	Y		Changes to Continental
2	Y	N		Sticks with Qantas
3	N		N	Does Not Give Lecture
4	N		Y	Would have Given Lecture

Table 1: Partial Taxonomy of Practical-Reasoning Situations: Options Deemed Incompatible

Case 1 is where the deliberation leads to a change of plans, as indicated by the second column of the table. This was not what happened in my case, but it might have been. We can call this the "Changes to Continental Airlines" case. Case 2 is where the deliberation does not result in a change, as was true in my example. We can call this the "Sticks with Qantas" case. It is obvious that in Case 1, deliberation turns out to be worthwhile, while in Case 2, there is wasted effort.

We also want to consider situations in which the agent does *not* deliberate about some option that is perceived to be incompatible with its existing plans: Cases 3 and 4. This is the class of situation I was in when I quickly dismissed the option of giving a lecture in my colleague's class. For our analysis, we need to determine what the agent would have done, if it had deliberated. This is relatively easy for the designer of an artificial agent to do; it is sometimes harder for a human who is introspecting. In the current example, we can imagine that even if I had reasoned more about whether to give the class lecture, I would not have changed my plans. So this is an example of Case 3. In this case, not deliberating was just right, since deliberation would not have led to a change in plans anyway.

But there are times when the agent would have done something different if it had engaged in further deliberation. If I had thought more about my colleague's class, I might have remembered that it was attended by some important visiting researchers whom I had hoped to meet—and then might have realized that giving the lecture in class would provide me with an opportunity to meet them. This added advantage might have been sufficient for me to change my plans. Then this would have been an instance of a Case 4 situation, which I will accordingly call "Would have given lecture." In a Case 4 situation, the agent performs *locally suboptimal* behavior. Because it does not engage in full-fledged deliberation, it misses an opportunity to perform an action it might otherwise have chosen, given more reasoning resources.

This analysis has been simplified in a number of ways; further details can be found in [7]. The point to note here is that IRMA, and in particular, its filtering mechanism, embody the tradeoff between spending enough time thinking and getting around to acting. Putting it somewhat coarsely, in Cases 1 and 2, the inexpensive fare cases, thinking takes precedence over acting; and the reverse is true in Cases 3 and 4, the class lecture cases. The challenge for the designer of an IRMA-agent is to tune the filtering mechanism so that, to the extent possible, the agent does not end up in Case 2 ("Sticks with Qantas") or in Case 4 ("Would have given lecture"). In Case 2, there is wasted reasoning, such as my reasoning about whether to switch to Continental, when, in the end, I stuck with Qantas. In Case 4, there is suboptimal behavior: not giving the class lecture when it would have been better to do so.

The challenge is to balance appropriate sensitivity to environmental change against reasonable stability of the system's adopted plans. But we should not expect perfection. Resource bounded agents have to trade occasional wasted reasoning and locally suboptimal behavior for global success.

3.3 Relation to Decision Theory

IRMA agents are, clearly, planning agents. They form plans, and they adopt a distinctive attitude towards those plans. We can say that they intend them, or that they are committed to them. This commitment pervades the practical reasoning of an IRMA agent. Their commitment to their plans is what distinguishes IRMA agents from strict Bayesian decision-theoretic agents.

I began this paper by claiming that resource-bounded agents in dynamic environments need to reason about their actions. I said they need to plan. In saying this, I used the term "plan" somewhat loosely, equating it with any sort of reasoning about action. One can certainly imagine agents who plan, in this broad sense, without forming the kinds of commitment to their plans that is the hallmark of an IRMA agent. For example, we can envision an agent who, at each point in time, reasons about what action to perform, given the totality of its current beliefs and desires. You may recognize this as a somewhat uncharitable caricature of a Bayesian deliberator [36]. The caricature is uncharitable because the decision theorists never intended a model of such unconstrained reasoning. Their presumption instead is that each decision involves a limited set of options. The problem of selecting the relevant set of options has been called the "small world problem" [64]. Decision theorists have had little to say about general solutions to the small world problem, but progress has been made on it from an AI perspective [8, 70].

In IRMA, an agent's commitment to its plans plays a large role in framing its decision problems. Its commitment often defines decision problems, in which the options are alternative ways of carrying out the plans. And its commitment often prunes decision problems, by ruling out options incompatible with its plans. With this in mind, we can distinguish between two related claims that I am making in this paper. First, I am arguing that agents need to reason about their actions—to plan, in the broader sense of planning. Second, I am arguing that agents need to commit to their plans. This commitment, in my view, is the essence of useful planning. It is certainly the essence of IRMA. For the reminder of this paper, when I speak of planning, it will be in this latter, more restrictive sense.

3.4 Evaluating Filtering Strategies

We can now return to the central challenge that confronts the designer of an IRMA agent. The control principles embodied in IRMA provide only partial guidance in designing an intelligent agent, since there is still the problem of tuning the filtering mechanism. Can we say anything more about the conditions under which it pays to be especially sensitive to environmental change, and those under which it pays to be more stubbornly committed to one's plans?

To address the question, the Tileworld testbed system was built [56].[1] Tileworld comprises two parts: a simulated dynamic environment with an associated set of tasks, and an embedded agent incorporating an IRMA-design. Both the agent and the environment are highly parameterized, and a menu-based interface makes it easy for an experimenter to vary the parameter settings. For example, an experimenter using the Tileworld can control the average rate of change in the environment. The experimenter can also control the properties of the tasks the agent faces, creating an environment in which all tasks are of roughly equal difficulty, or one in which some tasks are much easier than others. It is possible to create environments in which tasks have hard deadlines, or ones in which the tasks have high value only if they are

[1] Marc Ringuette and Michael Frank both contributed greatly to the design and implemenation of Tileworld.

completed by the deadline, after which their value decays continuously. Various other environmental conditions are also under the experimenter's control. In addition, the Tileworld includes an embedded IRMA agent, and the experimenter can tune its filtering mechanism, to vary its degree of commitment to its existing plans.

Thus to conduct experiments using Tileworld, one sets the parameters to establish an environment and an agent with characteristics of particular interest, and then generates simulations. The initial state of each simulation is randomly generated, and the simulation evolves over time, in accordance with the parameter settings, but subject to the influence of pseudo-random variation. During each simulation, the agent performs Tileworld tasks, and the system keeps track of how successful it is. In other words, one conducts experiments in which the independent variables are the characteristics of the environment and of the agent, and the dependent measure is the agent's performance. The goal is to develop an understanding of the relation between agent design choices and environmental factors.

Researchers at several sites have found Tileworld to be a good starting point for exploring their ideas on agent design. A notable example is Kinny's work [41, 42], which investigated variants of the commitment strategy proposed in IRMA. Figure 2, taken from [42], shows the results of one of his experiments. The graph plots the success of different IRMA-agents on the y-axis, against the rate at which their environments were changing on the x-axis. There are three agents shown: an agent that commits strongly to its plans, labeled the "bold" agent; an agent somewhat more open to reconsideration, labeled "normal"; and an agent prone to reconsideration, labeled "cautious". In all cases, the bold agent performs best.

In a subsequent experiment, means-end reasoning was made more "expensive", that is, each decision about how to achieve a given goal was made to take more time, relative both to the rate at which the agent could move and

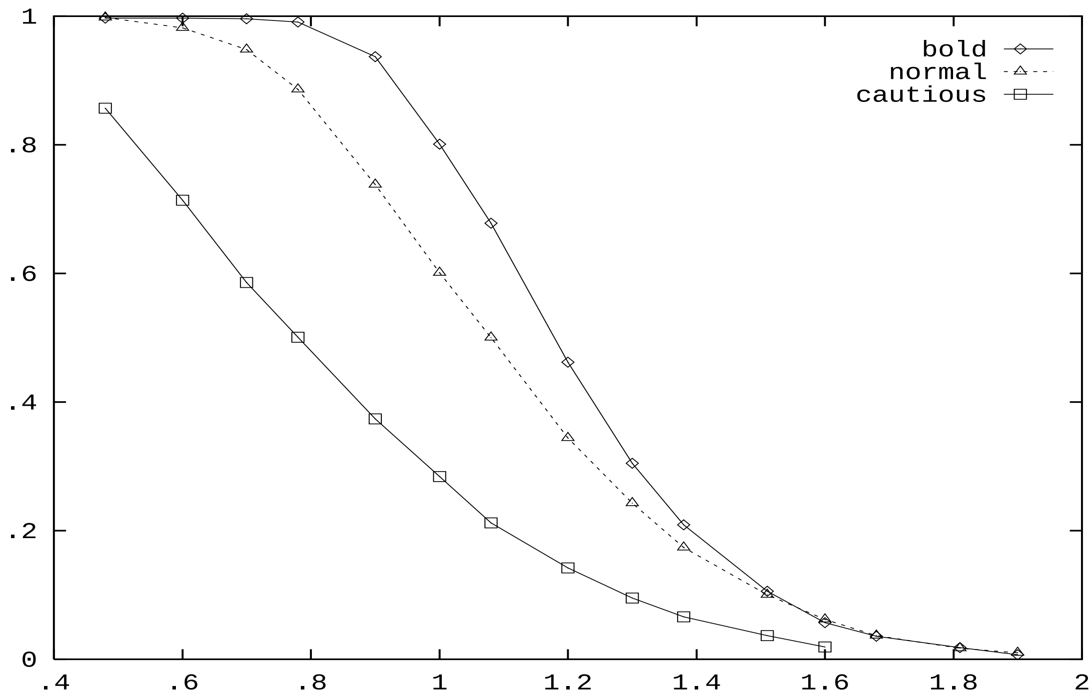

Figure 2: Performance of Baseline Agent (Source: D. Kinny, IJCAI-91)

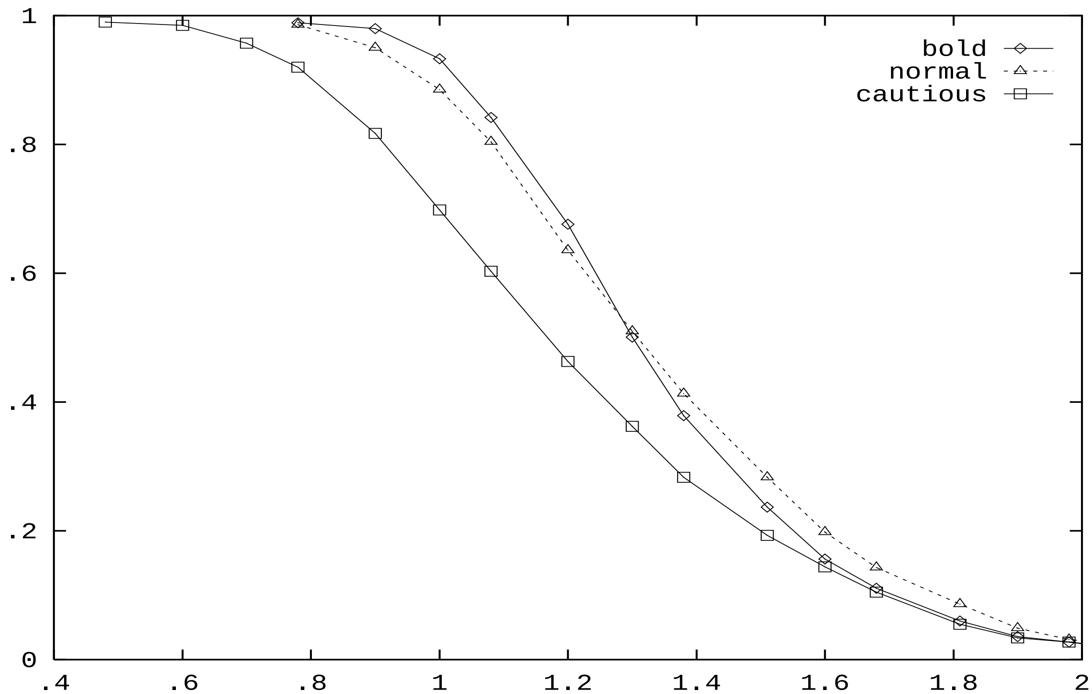

Figure 3: Performance of Costly Reasoner (Source: D. Kinny, IJCAI-91)

to the average rate of change in the environment. The results of this experiments are shown in Figure 3. A comparison of the two experiments shows that under conditions of costlier means-end reasoning, the margin between the committed agent and the uncommitted one narrows, and in really rapidly changing environments, strong commitment ceases to dominate.

The agents in these first two experiments are absolutely committed to their existing plans: they never reconsider them, regardless of any changes they perceive in their environment. Weakening this commitment in certain ways improves the agent's performance. In particular, Kinny found superior behavior in agents that immediately consider options to abandon goals they believe have become impossible. Agents who are committed to their plans except under these conditions perform better than any of the other agents studied in the first two experiments, even in the most rapidly changing environments. Details of the experiments that provide this result, along with experiments that explore other variations in the commitment strategy, can be found in [42]. The point to notice here is that Kinny's experiments provide support for the key claims in IRMA. For the agents he studied, optimal behavior was achieved by the agents that committed to their plans, except under specific, restricted kinds of environmental change. This is consistent with what IRMA predicts: agents do well to remain generally committed to their plans, except in response to identifiable classes of events, which the system designer encodes in the override mechanism.

Of course, Kinny's results are only a first step towards understanding what must be encoded in an agent's override mechanism—and, even more, his results are only a first step towards understanding the general nature of the relationship between agent design and environmental factors. Many more experiments are needed, and it will take significant effort to understand fully the implications of such experiments for complex, "real-world" applications [28]. Nonetheless, Kinny's works provides a detailed illustration of how one can use controlled experimentation, supported by a testbed like the Tileworld, to carefully investigate claims about agent design.

3.5 AI as an Experimental Science

One goal of the Tileworld system has been to has been to provide a testbed for studying a wide range of domains and tasks. Towards this end, we aimed for an abstract, almost generic environment. One important open question is whether this strategy is viable. Will it be possible to identify pertinent features of environments and tasks that can guide the design of agents intended for them? In other words, will we be able to find environmental "natural kinds", such that there are lawlike regularities between the environmental kinds and particular reasoning strategies?

This is just one of many methodological questions confronting researchers interested in using simulation techniques for performing AI experimentation [13, 19, 47]. As Cohen demonstrated in his analysis of the papers presented at AAAI90, we are, as a discipline, just learning how to perform real, systematic experimentation [12]. One hears a lot of talk about AI as an experimental science, but typically the "experiments" amount merely to writing a computer program that is supposed to validate some hypothesis by its very existence. As Newell and Simon explained it in their Turing lecture,

> Each new program that is built is an experiment. It poses a question to nature, and its behavior offers clues to an answer ... [Computer systems] are artifacts that have been designed ... and we can open them up and look inside. We can relate their structure to their behavior and draw many lessons from *a single experiment*. We do not have to build 100 copies of, say, a theorem prover, to demonstrate statistically that it has not overcome the combinatorial explosion of search in the way hoped for. Inspection of the program in light of a few runs reveals the flaw [48, p.36, emphasis mine].

It is doubtless true that we do not have to build 100 copies of a theorem prover to derive results about its performance. It is less clear that we can always get by with just a few runs. This may be possible when our mathematics is sophisticated enough to support analysis, as in the case of complexity analyses of theorem proving. But it has proven to be exceedingly difficult to provide mathematical analyses of complex behavior by systems situated in complex environments. This is why researchers in performance analysis so frequently make use of simulation techniques rather than relying exclusively on the mathematics of queueing theory.

It is time, I believe, that we in AI admit the limitations of our single-experiment methodology, and supplement it with more traditional, rigorous experimentation techniques. I am not suggesting that all AI research needs to involve controlled experimentation of the kind typified by Tileworld. Nor am I saying that all controlled experimentation that is done should rely on simulators. Where it is possible, one should attempt to gather sufficient data about the performance of actual systems operating "in the real world." But what simulation provides is a way for the experimenter to carefully monitor, control, and replicate environmental conditions, something that is often infeasible for actually deployed systems.

3.6 Overloading

Having taken a detour to discuss methodology, we can now return to the principal focus of this paper: a consideration of the ways in which intelligent agents use plans. I have already suggested that plans are useful not just for guiding action, but also for constraining reasoning. To illustrate how an agent's plans constrain its reasoning, I described the process by which I formed my plan to attend IJCAI, and then used that plan as a fixed assumption in further decision-making. But much of our quotidian reasoning involves less-extensive deliberation than did my trip to IJCAI. It is more like the decision-making you are likely to engage in when you have to buy flour for cake. You do not generate a large number of alternative recipes for getting the flour, and then carefully weigh the pros and cons of each. You are much more likely to settle fairly directly on a plan for achieving your goal of getting flour. For example, you may know that you have to go to the bank anyway, and that Franklin's supermarket is right next door, so you decide to stop at Franklin's on the way to the bank.

As the philosopher Gilbert Harman puts it:

> ...many of one's decisions are of necessity what we might call *simple decisions*. These arise when one finds oneself with a salient end E [like getting flour] and one recognizes a salient means M that will get one E [like stopping at Franklin's, because it is right next door to the bank]. In a simple case, one does not consider whether there might be some other means to E or some other end distinct from E that one might now obtain, and one disregards any other consequences of one's act. One simply forms the intention of getting E by doing M [29, p. 106].

The question, of course, for those of us who want to design intelligent agents, is: how does the agent recognize that salient means M to its end E? There are doubtless lots of strategies that an agent might use for this purpose. Case-based reasoning attempts to get at one such strategy. I will describe a different strategy—one that, perhaps not surprisingly, involves the agent once again making use of its already adopted plans.

The strategy I have in mind is one that I have called *overloading* [55], on analogy with the way that term is used in programming languages. Overloading is illustrated by the cake-making example. You have a means-end reasoning problem: getting flour. You solve it by considering the relatively small set of active plans you currently have, which includes your plan to go to the bank. *That* plan includes the step of driving to a particular location, say, the Bondi Junction Shopping Centre. You recognize that you can also use that same action, of driving to Bondi Junction, in a plan to get flour. So you decide directly, without further deliberation, to overload your intention to drive to Bondi Junction. You will use that action both for its initially intended purpose of getting you to the bank, and for the new purpose, of getting you to Franklin's, where you will buy your flour.

In terms of IRMA, you engage in a special kind of means-end reasoning. You attempt to form a plan for your goal that involves some action you already intend for another purpose. If you are successful, you immediately adopt that plan, without subjecting it to further deliberation.

3.7 The Rationality of Overloading

This example is suggestive, and seems intuitively to describe a kind of reasoning that we engage in. In fact, two observational studies of human planning behavior show that humans frequently do make use of the strategy of overloading [30, 32]. But to justify incorporating this strategy in the agents we design, we need to ask whether it is cost-effective. Is it sensible to control reasoning by overloading existing intentions?

Indeed it is. There is a certain efficiency to be gained in overloading an action—in killing two birds with one stone, so to speak. Although a plan that involves an overloaded action may not be the very best plan that the agent could use to achieve its goal, it is likely to be a reasonably good one, given the efficiency of action inherent in it. Moreover, if the agent directly adopts the plan involving the overloaded action, without generating and weighing alternatives to it, it thereby saves reasoning costs. It avoids the extra means-end reasoning that would be needed to generate the alternatives. It avoids the deliberation that would be needed to choose among them. And it cuts down on the amount of subsequent means-end reasoning needed, since it only has to determine how to perform one action—the overloaded one that subserves multiple goals—rather than figuring out how to perform a separate action for each goal.

So a plan containing an overloaded action will exhibit a certain efficiency of action, and directly adopting it will result in an added efficiency of reasoning. At least in everyday decision-making tasks, efficiency of action tends to be a significant factor in a plan's utility. So it is likely that the savings in reasoning cost will outweigh the advantages of the "best" possible plan relative to the one suggested by overloading.

We can relate this back to the plan to buy flour at Franklin's. If you had carefully considered all the ways of getting flour, scouring the newspapers for ads, computing mileages to each grocery, and so on, you might have determined that going to Safeway was marginally better than going to Franklin's. But going to Franklin's is a pretty good plan, since, after all, you can go there while you are at the bank, and save yourself an extra trip. And the marginal advantage of going to Safeway is likely to be outweighed by the effort involved in deciding on it.

Overloading thus provides a computational account of how, in some cases, an agent can perform what Harman called simple decision making. In addition, it contributes to an explanation of how simple decision making can be good decision making. Of course, not all simple decisions will be instances of overloading. Further research is needed to identify additional strategies.

Once again, we have started with a commonsense idea about how agents use their plans, and we have argued that this use makes good sense in light of resource bounds. Overloading, like filtering, is a way of satisficing. An agent who relies on overloading will sometimes engage in suboptimal behavior: if it had taken the time to generate and consider alternatives to the overloaded plan, it might have adopted one of those alternatives. But, as I have said before, local suboptimality appears to be the price that resource-bounded agents pay for global success.

4 Using Plans for Coordination and Communication

So far, I have been arguing that it is a good idea to be a planning agent, because you can use your plans to focus and constrain your reasoning. Planning, I have claimed, is advantageous even if you never interact with other agents. But the agents we build need to interact with other agents. They typically need at least to interact with us. Eventually, we are also going to want to build agents that can interact effectively with one another, because there are many tasks that cannot be satisfactorily performed by a single agent.

It turns out that, if you want to coordinate and communicate with other agents, it is extremely useful—and possibly even essential—for you and those other agents to be planners. There are two reasons for this. First, coordination between agents seems possible only because they can count on one another behaving in more or less stable ways, such as would result from an agent's commitment to its plans. I will not defend this claim here, but see Bratman [6, Chap. 5]. Second, as I will illustrate below, communication between agents is greatly facilitated by the agents reasoning about one another's plans.

4.1 Plan Recognition in Discourse

Consider once again my trip to the IJCAI meeting in Sydney. When I was forming my plans for the trip, I called my travel agent, and asked her,

> "How much does a round-trip flight to Sydney on Qantas cost?"

She could have just answered by telling me the price, but instead she said something like:

> "$1500. They fly nonstop out of Los Angeles every evening. For the same price, you can also fly out of San Francisco on Continental or Northwestern, making a stop in Honolulu."

Why did she give me this answer? I did not ask her anything about the departure times or locations of the flights, or about alternative carriers. At an intuitive level, it is not hard to see why my travel agent answered the way she did. She recognized that if I was asking about how much it cost to fly to Syndey, I must be considering a plan that involved going there. And she recognized that to form, evaluate and possibly execute this plan, I would need additional information. So she took it upon herself to be helpful, and provide that extra information. Being helpful, after all, is her job.

This style of analysis was originally formalized by Allen [2], who recognized that we could take the recipe-formation techniques of the classical AI planning systems, and effectively turn them on their heads to model plan recognition behavior. In Allen's approach, a plan-recognition system reasons both about what kinds of actions might be supported by an observed action, and about what kinds of actions might be performed in order to achieve contextually likely goals. In the current example, the observed action is my request for information about the cost of flights to Sydney; the likely goal, given that the context of the conversation is a phone call to a travel agent, is to take a trip somewhere. Plan recognition consists in chaining forward from the observed action and backwards from each candidate goal, using a set of heuristics to guide the inference, and terminating when the two action chains intersect to form a complete plan. Allen's original work on plan recognition lay the foundation for a number of later research efforts [11, 40, 46].

Suppose my travel agent had not reasoned about my plans; could I still have gotten the information I needed? Perhaps, but it would have been a tedious process. Thus, in this example, plan recognition increases the efficiency of the conversation. There are also circumstances in which plan recognition actually prevents a failure in the communication process. For instance, after I reserved my ticket for a Monday evening flight to Sydney, I said to my travel agent,

> "Please reserve a car for me in Sydney for Tuesday morning."

My travel agent did not simply assent to my request. She understood my wanting a car as part of my larger plan of taking a trip, and recognized that I must have forgotten about, or misunderstood, the implications of crossing the international date line. So she explained to me that although I would be leaving the States on a Monday evening, I would not arrive in Sydney until *Wednesday* morning. The reasoning needed for her to recognize my plan in this instance may have been almost trivial. The point is that she performed that reasoning. An agent that did not attempt to understand my utterance as part of my plans would have gone ahead and made the car reservation for Tuesday.

In my thesis research, I focused on with the problem of recognizing invalid plans, like the one I have just described [51, 52, 54]. I showed that the traditional models of plan recognition, which equate plans with recipes for action, are insufficient for this task. Traditional models are more or less restricted to searching through a fixed space of plans. With some limited exceptions, you can only recognize plans that you might yourself form. But what you really need is to be able to generate a space of a plans. To do that, you need to take seriously the idea that a plan is a structured collection of beliefs and intentions. Then plan recognition becomes the problem of ascribing plausible beliefs and intentions that are related to one another in a coherent fashion [44]. In other words, recognizing the way in which plans decompose into beliefs and intentions is a prerequisite for adequately describing the plan recognition process.

4.2 Plan Use and Plan Recognition

Enabling cooperative responses is just one of many communicative phenomena that can be accounted for in terms of plan recognition. From figuring out what object a referring expression signifies, to recognizing the structure of an extended discourse, most aspects of communication seem to be intimately linked to reasoning about the communicator's plans [26]. Yet, most research on plan recognition has taken place in isolation from the AI planning renaissance. Could a marriage of these two research projects bear any fruit?

I believe that it could, although so far this is on the basis of reasoned speculation, rather than on any careful working-out of the technical details [53]. I will describe, in support of this claim, one more example of the intelligent use of plans. The example involves multi-agent coordination, although similar examples that involve communication could be constructed. It illustrates how plan recognition can profit from taking into account the role of an agent's plans in constraining reasoning.

You talk with a colleague one morning and she tells you that she plans to visit a nearby research laboratory later that day, and also plans to go to the bank sometime. In addition, you know that she plans to attend the monthly departmental lunch at noon. At 11:45, you see her heading out to the parking lot. Because you have a paper that you want delivered to the research laboratory, you attempt to recognize her plan: is she on her way to the laboratory? If she is, you will run after her to give her the paper. Suppose you know that it only takes about 10 minutes round-trip to get to the bank, but that when she goes to the laboratory, she will probably be gone for more than an hour. In that case, the likely conclusion is that the plan she is currently executing is to go to the bank, since going to the laboratory now, at 11:45, would be incompatible with her plan to attend the departmental lunch at noon.

Virtually all existing AI plan recognition systems would only take into account the current observation, that your colleague is heading out to the parking lot. And so they would be unable to decide what plan she is pursuing. In contrast, a plan recognition system that took into account knowledge of her other plans, along with the fact that plans are generally consistent with one another, would draw the right conclusion. But notice—and this is the key—the fact that plans are mutually consistent results from the roles that they play in an agent's mental life, from the uses to which an agent puts them. To recognize plans, we have to understand how plans are used.

5 Conclusion

In this paper, I have described some of the ways in which intelligent agents use plans. I have focused primarily on how plans can be used to constrain reasoning, and thereby facilitate survival in dynamic environments. I presented two methods for controlling reasoning—two design principles—that both involve the use of plans. The first, filtering, amounts to having a tendency to give less than full consideration to options that conflict with one's plans. The second, overloading, amounts to preferring to use one's actions for multiple purposes. Both filtering and overloading are satisficing strategies: they inevitably result in episodes of locally suboptimal behavior. But this is the price that resource-bounded agents pay for survival in complex, changing environments.

I also suggested that agents "use" one another's plans. That is, they reason about other agents' plans in order to communicate and cooperate with them. Finally, I conjectured that our models of the plan recognition process could be greatly improved by having them reflect our recent insights about the uses of plans by resource-bounded agents.

I want to conclude with a few remarks about methodology. I have presented several theories here, about how to design agents that use their plans to behave intelligently. I have not presented any theorems, and this may have surprised some of you, who may have been expecting modal, temporal, autoepistemic logic, and possible-worlds semantics, and a even few circumscriptive axioms. Where, you may be wondering, is the formalization?

Formalizing our theories is, without question, an important activity, and one in which I, like many AI researchers, have engaged. But I am concerned that, especially of late and especially in certain subfields of AI, there has been an overvaluing of formalization, and an unfortunate, concomitant devaluing of the hard work of theory formation. In some quarters, formalism hacking has replaced system hacking. Both activities are essential, but both must be supported by rich theories of the phenomena that concern us. I think we ought to adopt a new slogan in AI: theories before theorems.

In this paper, I have tried to illustrate the kind of theory formation that I think is essential to the long-term progress of our field. Such theory formation should not be confused with mere introspection. I did not argue that we should build planning agents because it somehow seems obvious to us that we plan. Rather, I began with the hypothesis that we plan—that much was based on introspection—and then asked what possible reasons we might have for planning. What benefits accrue to a planning agent? The arguments thus bear structural similarities to those given by biologists, who ask what value some expressed characteristic may have for an organism, in order to explain how that characteristic might have survived the test of evolution. Planning, I argued, has tremendous value for organisms like us and like the systems we are aiming to build.

ACKNOWLEDGEMENTS

I would like to thank the many people who, over the years, have influenced my thinking about the nature of intelligence, especially Michael Bratman, Barbara Grosz, David Israel, and Kurt Konolige, each of whom also provided help and support while I was writing the Computers and Thought lecture. The lecture was written, and most of the work described in it was done, while I was employed at the Artificial Intelligence Center at SRI International, and I am indebted to all my colleagues there, and especially to Ray Perrault, the lab director, for providing a wonderful environment in which to do AI. Preparation of the lecture was financially supported in part by a contract with the Nippon Telegraph and Telephone Corporation and by the Office of Naval Research under Contract No. N00014-89-C-0095, and by the Air Force Office of Scientific Research under Contract No. F49620-91-C-0005.

References

[1] P. E. Agre and D. Chapman. Pengi: An implementation of a theory of activity. In *Proceedings of the Sixth National Conference on Artificial Intelligence*, pages 268–272, Seattle, WA, 1987.

[2] J. Allen. Recognizing intentions from natural language utterances. In M. Brady and R. C. Berwick, editors, *Computational Models of Discourse*, pages 107–166. MIT Press, Cambridge, MA, 1983.

[3] R. Alterman. Adaptive planning. *Cognitive Science*, 12:393–421, 1988.

[4] S. B. Banks and C. S. Lizza. Pilot's associate: A cutting edge knowledge-based system application. Technical report, Wright Research and Development Center, Wright-Patterson Air Force Base, Ohio, 1990.

[5] M. Boddy and T. Dean. Solving time-dependent planning problems. In *Proceedings of the Eleventh International Joint Conference on Artificial Intelligence*, pages 979–984, Detroit, MI, 1989.

[6] M. E. Bratman. *Intention, Plans and Practical Reason.* Harvard University Press, Cambridge, MA, 1987.

[7] M. E. Bratman, D. J. Israel, and M. E. Pollack. Plans and resource-bounded practical reasoning. *Computational Intelligence*, 4:349–355, 1988.

[8] J. S. Breese. *Knowledge Representation and Inference in Intelligent Decision Systems.* PhD thesis, Stanford University, Stanford, CA, 1987.

[9] R. A. Brooks. Intelligence without reasoning. In *Proceedings of the Twelth International Joint Conference on Artificial Intelligence*, pages 569–595, Sydney, Australia, 1991.

[10] R. A. Brooks. Intelligence without representation. *Artificial Intelligence*, 47:139–159, 1991.

[11] M. S. Carberry. *Plan Recognition in Natural Language Dialogue*. MIT Press, Cambridge, MA, 1990.

[12] P. R. Cohen. A survey of the Eighth National Conference on Artificial Intelligence: Pulling together or pulling apart? *AI Magazine*, 12:16–41, 1991.

[13] P. R. Cohen, M. L. Greenberg, D. M. Hart, and A. E. Howe. Trial by fire: Understanding the design requirements for agents in complex environments. *AI Magazine*, 10:32–48, 1989.

[14] B. D'Ambrosio and M. Fehling. Resource-bounded agents in an uncertain world. In *AAAI Symposium on Limited Rationality: Working Notes*, pages 13–17, Stanford, CA, 1989.

[15] R. Davis. Meta-rules: Reasoning about control. *Artificial Intelligence*, 15:179–222, 1980.

[16] T. Dean and M. Boddy. An analysis of time-dependent planning. In *Proceedings of the Seventh National Conference on Artificial Intelligence*, pages 49–54, St. Paul, MI, 1988.

[17] R. Dodhiawala, N. S. Sridharan, P. Raulefs, and C. Pickering. Real-time AI systems: A definition and an architecture. In *Proceedings of the Eleventh International Joint Conference on Artificial Intelligence*, pages 256–261, Detroit, MI, 1989.

[18] J. Doyle. Artificial intelligence and rational self-government. Technical Report CS-88-124, Carnegie Mellon University, Pittsburgh, PA, 1988.

[19] E. H. Durfee and T. A. Montgomery. MICE: A flexible testbed for intelligent coordination experiments. In L. Erman, editor, *Intelligent Real-Time Problem Solving: Workshop Report*, pages X1–X16, Palo Alto, CA, 1990. Cimflex Teknowledge Corp.

[20] C. Elkan. Incremental, anytime planning. In *Proceedings of the Eighth National Conference on Artificial Intelligence*, pages 145–150, Boston, MA, 1990.

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

[22] R. E. Fikes and N. J. Nilsson. STRIPS: A new approach to the application of theorem proving to problem solving. *Artificial Intelligence*, 5:189–208, 1971.

[23] M. P. Georgeff and F. F. Ingrand. Decision-making in an embedded reasoning system. In *Proceedings of the Eleventh International Joint Conference on Artificial Intelligence*, pages 972–978, Detroit, MI, 1989.

[24] M. P. Georgeff and A. L. Lansky. Reactive reasoning and planning. In *Proceedings of the National Conference on Artificial Intelligence*, pages 677–682, Seattle, WA, 1987.

[25] M. L. Ginsberg. Universal planning: An (almost) universally bad idea. *Artificial Intelligence Magazine*, 10:40–44, 1989.

[26] B. J. Grosz and C. L. Sidner. Attention, intentions, and the structure of discourse. *Computational Linguistics*, 12:175–204, 1986.

[27] K. J. Hammond. CHEF: A model of case-based planning. In *Proceedings of the Fifth National Conference on Artificial Intelligence*, pages 267–271, Philadelphia, PA, 1986.

[28] S. Hanks, M. E. Pollack, and P. R. Cohen. Benchmarks, testbeds, controlled experimentation, and the design of agent architectures. Forthcoming.

[29] G. Harman. *Change in View.* MIT Press, Cambridge, MA, 1986.

[30] C. C. Hayes. A model of planning for plan efficiency: Taking advantage of operator overlap. In *Proceedings of the Eleventh International Joint Conference on Artificial Intelligence*, pages 949–953, Detroit, MI, 1989.

[31] B. Hayes-Roth. Architectural foundations for real-time performance in intelligent agents. *The Journal of Real-Time Systems*, 2:99–125, 1990.

[32] B. Hayes-Roth and F. Hayes-Roth. A cognitive model of planning. *Cognitive Science*, 3:275–310, 1979.

[33] B. Hayes-Roth, R. Washington, R. Hewett, M. Hewett, and A. Seiver. Intelligent monitoring and control. In *Proceedings of the 11th International Joint Conference on Artificial Intelligence*, pages 243–249, Detroit, MI, 1989.

[34] E. J. Horvitz. Reasoning about beliefs and actions under computational resource constraints. In *Uncertainty in Artificial Intelligence*, volume 3. North Holland, Amsterdam, 1988.

[35] E. J. Horvitz, G. F. Cooper, and D. E. Heckerman. Reflection and action under scarce resources: Theoretical principles and empirical study. In *Proceedings of the Eleventh International Joint Conference on Artificial Intelligence*, pages 1121–1127, Detroit, MI, 1989.

[36] R. Jeffrey. *The Logic of Decision, 2nd edition.* University of Chicago Press, Chicago, 1983.

[37] L. P. Kaelbling. Goals as parallel program specifications. In *Proceedings of the Seventh National Conference on Artificial Intelligence*, pages 60–65, St. Paul, MI, 1988.

[38] S. Kambhampati and J. Hendler. A validation structure based theory of plan modification and reuse. *Artificial Intelligence*, To appear.

[39] K. Kanazawa and T. Dean. A model for projection and action. In *Proceedings of the Eleventh International Joint Conference on Artificial Intelligence*, pages 985–990, Detroit, MI, 1989.

[40] H. A. Kautz. A circumscriptive theory of plan recognition. In P. R. Cohen, J. Morgan, and M. E. Pollack, editors, *Intentions in Communication.* MIT Press, Cambridge, MA, 1990.

[41] D. N. Kinny. Measuring the effectiveness of situated agents. Technical Report 11, Australian Artificial Intelligence Institute, One Grattan Street, Carlton 3053, Australia, 1990.

[42] D. N. Kinny and M. P. Georgeff. Commitment and effectiveness of situated agents. In *Proceedings of the Twelfth International Joint Conference on Artificial Intelligence*, pages 82–88, Sydney, Australia, 1991.

[43] D. Kirsh. Today the earwig, tomorrow man? *Artificial Intelligence*, 47:161–184, 1991.

[44] K. Konolige and M. E. Pollack. Ascribing plans to agents: Preliminary report. In *Proceedings of the Eleventh International Joint Conference on Artificial Intelligence*, pages 924–930, Detroit, MI, 1989.

[45] T. J. Laffey, P. A. Cox, J. L. Schmidt, S. M. Kao, and J. Y. Read. Real-time knowledge-based systems. *AI Magazine*, 9(1):27–45, 1988.

[46] D. J. Litman and J. F. Allen. Discourse processing and commonsense plans. In P. R. Cohen, J. L. Morgan, and M. E. Pollack, editors, *Intentions in Communication*. MIT Press, Cambridge, MA, 1990.

[47] P. Maes. The dynamics of action seclection. In *Proceedings of the Eleventh International Joint Conference on Artificial Intelligence*, pages 991–997, Detroit, MI, 1989.

[48] A. Newell and H. A. Simon. Computer science as empirical inquiry. In J. Haugeland, editor, *Mind Design*, pages 35–66. MIT Press, Cambridge, MA, 1982. Tenth Turing Award Lecture.

[49] N. J. Nilsson. Shakey the robot. Technical Report 323, Artificial Intelligence Center, SRI International, Menlo Park, CA, 1984.

[50] N. J. Nilsson. Triangle tables: A proposal for a robot programming language. Technical Report 347, Artificial Intelligence Center, SRI International, Menlo Park, CA, 1985.

[51] M. E. Pollack. Inferring domain plans in question-answering. Technical Report 403, Artificial Intelligence Center, SRI International, Menlo Park, CA, 1986.

[52] M. E. Pollack. A model of plan inference that distinguishes between the beliefs of actors and observers. In *Proceedings of the 24th Annual Meeting of the Association for Computational Linguistics*, pages 207–214, New York, 1986.

[53] M. E. Pollack. Plan recognition beyond STRIPS. In *Second Workshop on Plan Recognition*, Detroit, MI, 1989.

[54] M. E. Pollack. Plans as complex mental attitudes. In P. R. Cohen, J. Morgan, and M. E. Pollack, editors, *Intentions in Communication*, pages 77–104. MIT Press, Cambridge, MA, 1990.

[55] M. E. Pollack. Overloading intentions for efficient practical reasoning. *Noûs*, 25(4):513–536, 1991.

[56] M. E. Pollack and M. Ringuette. Introducing the Tileworld: Experimentally evaluating agent architectures. In *Proceedings of the Ninth National Conference on Artificial Intelligence*, pages 183–189, Boston, MA, 1990.

[57] J. Pollock. *How to Build a Person: A Prolegomenon.* MIT Press, Cambridge, MA, 1989.

[58] P. Raulefs, B. D'Ambrosio, M. R. Fehling, and S. Forrest. Real-time process management for materials composition in chemical manufacturing. In *Proceedings of the IEEE Conference on Applications of Artificial Intelligence*, pages 120–125, 1987.

[59] S. J. Rosenschein and L. P. Kaelbling. The synthesis of digital machines with provable epistemic properties. In *Proceedings of the Conference on Theoretical Aspects of Reasoning about Knowledge*, pages 83–98, 1986.

[60] S. J. Russell. Execution architectures and compilation. In *Proceedings of the Eleventh International Joint Conference on Artificial Intelligence*, pages 15–20, Detroit, MI, 1989.

[61] S. J. Russell and E. H. Wefald. Principles of metareasoning. In *Proceedings of the First International Conference on Principles of Knowledge Representation and Reasoning*, pages 400–411, Toronto, 1989.

[62] S. J. Russell and E. H. Wefald. *Do the Right Thing.* MIT Press, Cambridge, MA, 1991.

[63] E. D. Sacerdoti. *A Structure for Plans and Behavior.* American Elsevier, New York, 1977.

[64] L. J. Savage. *The Foundations of Statistics, 2nd edition.* Dover Press, New York, 1972.

[65] Shakey: Experiments in robot planning and learning. Videotape. Artificial Intelligence Center, SRI International, Menlo Park, CA, 1972.

[66] H. A. Simon. A behavioral model of rational choice. *Quarterly Journal of Economics*, 69:99–118, 1955.

[67] H. A. Simon. *Models of Man.* Macmillan Press, New York, 1957.

[68] H. A. Simon. *The Sciences of the Artificial, 2nd Edition.* MIT Press, Cambridge, MA, 1982.

[69] M. Stefik. Planning and meta-planning. *Artificial Intelligence*, 16:141–169, 1981.

[70] M. P. Wellman. *Formulation of Tradeoffs in Planning under Uncertainty.* Pitman/Morgan Kaufmann, San Mateo, CA, 1990.

[71] D. E. Wilkins. *Practical Planning: Extending the Classical AI Paradigm.* Morgan Kaufmann, San Mateo, CA, 1988.