Multiagent Systems

Challenges and Opportunities for Decision-Theoretic Planning

Craig Boutilier

■ In this article, I describe several challenges facing the integration of two distinct lines of AI research: (1) decision-theoretic planning (DTP) and (2) multiagent systems. Both areas (especially the second) are attracting considerable interest, but work in multiagent systems often assumes either classical planning models or prespecified economic valuations on the part of the agents in question. By integrating models of DTP in multiagent systems research, more sophisticated multiagent planning scenarios can be accommodated, at the same time explaining precisely how agents determine their valuations for different sources or activities. I discuss several research challenges that emerge from this integration, involving the development of coordination protocols, the reasoning about lack of coordination, and the predicting of behavior in markets. I also briefly mention some opportunities afforded planning agents in multiagent settings and how these might be addressed.

The design and control of multiagent systems is an area of considerable interest in AI. Increasingly, intelligent agents, in the course of pursuing their own objectives, are required to interact with other self-interested agents. Furthermore, even when each of a group of agents has identical interests (for example, when acting on behalf of a single user or organization), physical or computational considerations can make it desirable to have these agents make decisions independently (Groves 1973). In broad terms, one might characterize research in game theory and multiagent systems as studying mechanisms by which such agents "coordinate" their activities.

At the risk of oversimplification, AI research on planning and decision making in multiagent systems can roughly be broken into two categories:[1] First, substantial research has been devoted to the extension of classical planning methodology to multiagent systems. In such work, classical planning assumptions, such as deterministic actions, complete system knowledge, and specific goals, are adopted, with an emphasis on coordinating agent activities either during planning or plan execution. The SHAREDPLANS model of Grosz and Kraus (1996) is an example of such a model. Such approaches allow for sophisticated reasoning on the part of agents but do not allow for alternative goals of different value or deal with explicit (quantified) uncertainty.

Second, economic models have also proven popular. In this research, agents are assumed to have specific valuations for different resources (which can include the ability to execute their plans) and interact indirectly through a market of some type. Wellman's (1993) work on market-oriented programming is an excellent example of this approach. These models generally allow agents to have multiple, substitutable objectives of different values and prescribe behavioral strategies that (sometimes implicitly) deal with uncertainty in the behavior of other agents. However, work using economic approaches generally assumes that agents enter the market with well-defined valuations, specified a priori, that influence their behavior (for example, the prices they are willing to bid in an auction). Generally, one ignores how these valuations arise (why a resource is worth a certain amount to the agent, how the valuation might change over time or as circumstances vary, or how the agent obtains alternative resources).

Both approaches are useful: Models that make simplifying assumptions generally allow one to study interesting aspects of a problem without getting bogged down in unnecessary detail. However, the assumptions must be relaxed when we try to represent and solve realistic decision problems. Ultimately, models are required that incorporate both types of reasoning: (1) the sequential planning and explicit reasoning about coordination that takes places in planning models and (2) the economic reasoning and decision- and game-theoretic trade-offs dealt with in economic models.

Decision-theoretic planning (DTP) might provide a middle ground to capture the best features of both approaches, reflecting the type of sophisticated reasoning required in realistic multiagent settings. DTP is currently receiving significant attention in the AI planning community, generalizing classical models to deal with uncertainty in action effects; uncertainty in knowledge of the system state; multiple (possibly competing) objectives of different priority; and ongoing, process-oriented planning problems (Boutilier, Dean, and Hanks 1999). These features make it an attractive way to view multiagent decision problems. Not only does it generalize classical multiagent planning models in suitable ways, decision-theoretic modeling also permits agents to represent their uncertainty about the behavior of other agents and can account for the specific valuations adopted by agents for different resources and courses of action. Rather than simply assume the existence of valuations, the economic behavior of agents can be related directly to their domain-level decision processes.

Rather than simply assume the existence of valuations, the economic behavior of agents can be related directly to their domain-level decision processes.

Unfortunately, there has not been much work on the application of DTP techniques to multiagent systems in the AI community. Conceptually, there is not much difficulty in formulating multiagent DTP problems. Arguably, Markov decision processes (MDPs) (Puterman 1994) are the de facto conceptual model for DTP problems. Extending MDPs to the multiagent setting leads more or less directly to certain general models of repeated games developed in game theory (Myerson 1991; Shapley 1953). However, even though the underlying model might be straightforward, a number of important research challenges must be addressed in this extension. These challenges include developing appropriate solution concepts for these models of multiagent systems, exploiting (single-agent) DTP representational and computational techniques in multiagent settings, and integrating explicit economic reasoning.

In this article, I detail some of these challenges. Specifically, I focus on two key issues that arise when extending existing models of DTP to multiagent settings: (1) how one defines the value of behaving according to certain coordination protocols and (2) how one might integrate reasoning about economic activity with more general planning activities. Both of these issues involve extending game-theoretic and economic techniques to situations involving more general, sequential DTP models. I also briefly mention a few other challenges facing DTP and opportunities for improved behavior afforded by the existence of other agents.

I use MDPs as the underlying model to illustrate these challenges. Although I don't delve into many technical details in this article, adopting a formal model at the outset helps make the issues and intuitions I want to convey precise. I initially focus on fully cooperative interactions. In general, however, planning agents will need to plan in the presence of noncooperating agents. For this reason, I also include a few remarks on the use of economic models in DTP.

The remainder of the article is structured as follows: I first introduce multiagent MDPs and discuss coordination problems. Next, I describe the difficulties posed by coordination problems for sequential decision making and propose a method for decision making given the constraints imposed by specific coordination protocols. I then point out some of the reasons for wanting to integrate DTP with economic models of resource allocation. I conclude with a brief mention of some other interesting opportunities that arise when tackling DTP in multiagent settings.

Multiagent Markov Decision Processes

I begin by presenting standard (single-agent) MDPs and describe their multiagent extensions (see Boutilier, Dean, and Hanks [1999] and Puterman [1994] for further details on MDPs). A fully observable MDP $M = \langle S, A, \Pr, R \rangle$ comprises the following components: S is a finite set of states of the system being controlled. The agent has a finite set of actions A with which to influence the system state. Dynamics are given by $\Pr : S \times A \times S \rightarrow [0, 1]$; here $\Pr(s_i, a, s_j)$ denotes the probability that action a, when executed at state s_i, induces a transition to s_j. $R : S \rightarrow \Re$ is a real-valued, bounded reward function. The state s is known at all times; that is, the MDP is fully observable.

An agent finding itself in state s^t at time t must choose an action a^t. The expected value of a given course of action (defined later) depends on the specific objectives. A finite horizon decision problem with horizon T measures the value of π as

$$E\left(\sum_{t=0}^{T} R\left(s^{t}\right) \mid \pi\right)$$

(where expectation is taken w.r.t. Pr). Infinite horizon problems can also be defined in this framework.

For a problem with horizon T, a nonstationary policy $\pi : S \times \{1\ldots, T\} \rightarrow A$ associates with each state and stage-to-go $t < T$ an action $\pi(s, t)$ to be executed at s with t stages remaining. An optimal nonstationary policy is one with maximum expected value at each state-stage pair.

A simple algorithm for constructing optimal policies is value iteration (Puterman 1994). Define the t stage-to-go value V^t function by setting $V^0(s_i) = R(s_i)$ and

$$V^t(s_i) =$$
$$R(s_i) + \max_{a \in \mathcal{A}} \left\{ \sum_{s_j \in S} \Pr(s_i, a, s_j) V^{t-1}(s_j) \right\}$$

Intuitively, $V^t(s_i)$ denotes the value (the expected reward accumulated) of being in state s_i and acting optimally for t stages. One sets $\pi(s_i, t)$ to be the action a maximizing the right-hand term, terminating the iteration at $t = T$.

MDPs allow one to model planning problems where action effects are uncertain, and different objectives have different priorities and might even conflict. As an example, consider a robot charged with satisfying coffee requests and delivering mail. It might be rewarded more highly for satisfying coffee requests and, all things being equal, will defer mail delivery until no more coffee is needed. However, if there is some uncertainty associated with getting coffee (for example, the robot's ability to pour coffee is limited) or if getting coffee has some nasty side effects (for example, the robot generally spills coffee on the floor when it wanders down the hallway), the expected value of coffee delivery might be reduced to give mail delivery a higher priority—it might even be that the robot decides never to attempt to get your coffee! An MDP formulation of such a planning problem allows the robot to make rational trade-offs involving the risks, rewards, and uncertainty inherent in the problem.

The Multiagent Extension

Now assume that a collection of agents is controlling the process. For the time being, assume that each agent has the same objective; for example, I might have a pair of robots that can deliver both coffee and mail. The goal is to define optimal joint behavior for the pair; for example, having one robot pick up the coffee and the other the mail will result in the quickest user satisfaction. The individual actions of agents interact in that the effect of one agent's actions might depend on the actions taken by others. I'll assume for now that the agents are acting on behalf of some individual; therefore, each has the same utility or reward function R. The system is fully observable to each agent.

I model this system formally as a multiagent Markov decision process (MMDP). MMDPs are much like MDPs with the exception that actions (and possibly decisions) are distributed among multiple agents. An MMDP $M = \langle \alpha, \{A_i\}_{i \in \boldsymbol{\alpha}}, S, Pr, R \rangle$ consists of five components. The set α is a finite collection of n agents, with each agent $i \in \boldsymbol{\alpha}$ having at its disposal a finite set A_i of individual actions. An element $\langle a_1 \ldots a_n \rangle$ of the joint action space $\mathcal{A} = \times A_i$ represents the concurrent execution of the actions a_i by each agent i. The components S, Pr, and R are as in an MDP, except that now they refer to joint actions $\langle a_1 \ldots a_n \rangle$

With the joint action space as the set of basic actions, an MMDP can be viewed as a standard (single-agent) MDP. Specifically, because there is a single reward function, the agents do not have competing interests; so, any course of action is equally good (or bad) for all. *Optimal joint policies* are optimal policies over the joint action space; these can be computed by solving the (standard) MDP $\langle A, S, Pr, R \rangle$ using an algorithm-like value iteration.

As an example, consider the MMDP illustrated in figure 1. It consists of two agents $a1$ and $a2$, each with two actions a and b that can be performed at any of the six states. All transitions are deterministic and are labeled by the joint actions that induce the transition. The joint action $\langle a, b \rangle$ refers to $a1$ performing a and $a2$ performing b and others similarly (with * referring to any action taken by the corresponding agent). At the source state s_1, $a1$ alone decides whether the system moves to s_2 (using a) or s_3 (using b). At s_3, the agents are guaranteed a move to s_6 and a reward of 5, but at s_2, both agents must choose action a, or both must choose b, to move to s_4 and a reward of 10; choosing opposite actions results in a transition to $s5$ and a reward of –10. The set of optimal joint policies are those where $a1$ chooses a at s_1 ($a2$ can choose a or b), and $a1$ and $a2$ choose either $\langle a, a \rangle$ or $\langle b, b \rangle$ at s_2. The value function determined by solving the MMDP for the optimal joint policy is the *optimal joint value function* and is denoted $\tilde{V}$. The t-stage value at s_1, $\tilde{V}^t(s_1)$, is given by $10[t+1/3]$.

MMDPs, although a natural extension of MDPs to multiagent settings, can also be viewed as a type of stochastic game, as formulated by Shapley (1953). Stochastic games were originally formulated for zero-sum games only (which we see alleviates certain difficulties), but I focus on the (equally special) case of cooperative games. By allowing each agent to have a different reward function, an MMDP

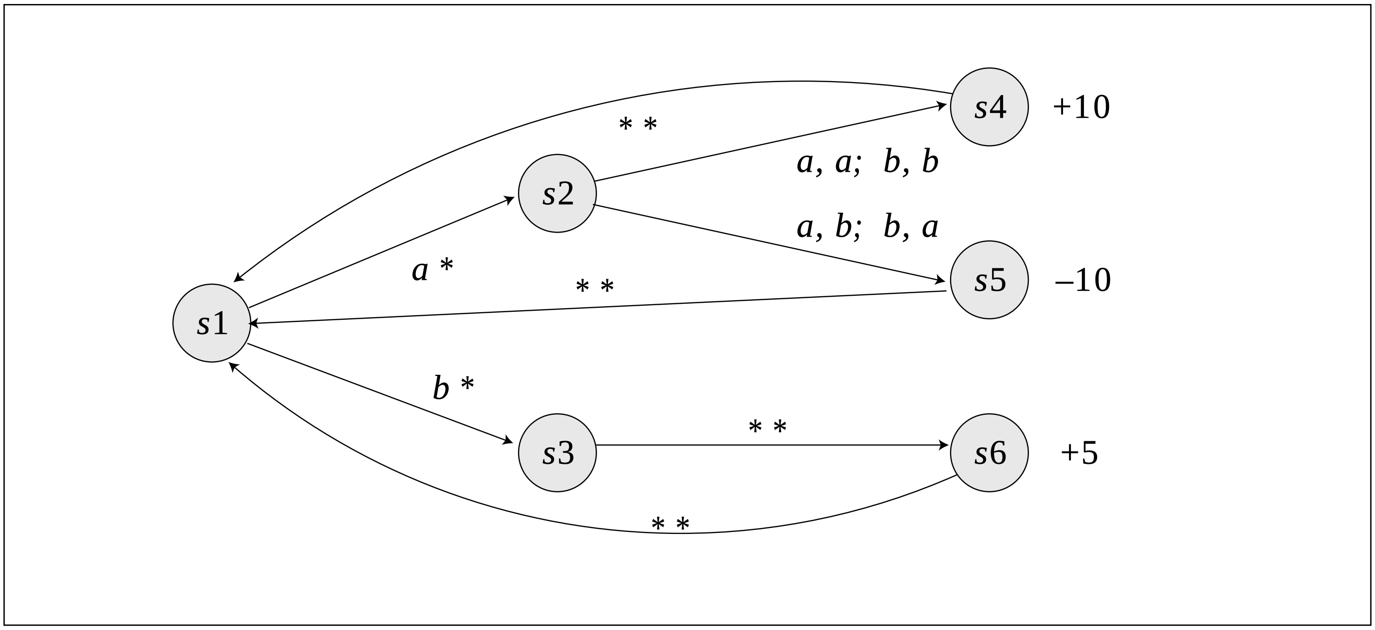

Figure 1. A Simple Multiagent Markov Decision Process with a Coordination Problem.

model can provide an even more general setting for studying multiagent interactions (of which zero-sum games are a special case).[2] Interactions that are not fully cooperative are discussed later.

Coordination Problems and Coordination Mechanisms

The previous example MMDP has an obvious optimal joint policy. Unfortunately, if agents *a*1 and *a*2 make their decisions independently, this policy might not be able to be implemented. There are two optimal action choices at s_2: ⟨*a*, *a*⟩ and ⟨*b*, *b*⟩. If, say, *a*1 decides to implement the former and *a*2 the latter, the resulting joint action ⟨*a*, *b*⟩ is far from optimal. This is a classic coordination problem: There is more than one optimal joint action from which to choose, but the optimal choices of at least two agents are mutually dependent. Notice that the uncertainty about how the agents will play s_2 makes *a*1's decision at s_1 hard to evaluate (this issue will be taken up later). In the intuitive example of mail-coffee–delivery robots, there can be situations in which it is just as good for either robot to get the coffee (and the other to get the mail), but if both start heading toward the mail room, coffee delivery will be delayed.

In the absence of a central controller that selects a single joint policy to be provided to each agent, ensuring coordinated action choice among independent decision makers requires some *coordination mechanism*. Such a mechanism restricts an agent's action choices, perhaps based on its history. I describe some of these mechanisms later, including learning and conventional and communication techniques.

A *coordination problem* can be described intuitively as any situation in which there is more than one optimal joint action such that if every agent selects an individual action that is part of some optimal action, the resulting joint action is not guaranteed to be optimal. In such a case, the agents have to agree—in some sense—which joint action to adopt because if they choose independently, they run the risk of behaving suboptimally. We call the individual actions available to agent *a*1 that are part of some optimal joint action the *potentially individually optimal* (PIO) actions for *a*1. Given the MMDP in figure 1, a coordination problem exists at s_2 if we focus attention on the immediate reward obtained at the subsequent state. Each agent has two PIO actions, *a* and *b*.[3]

A *coordination mechanism* is a protocol by which agents restrict their attention to (generally) a subset of their PIO actions in a coordination problem. A mechanism has a *state*, which summarizes relevant aspects of the agent's history, and a decision rule for selecting actions as a function of the mechanism state. Although such rules often select actions (perhaps randomly) from among PIO actions, there are circumstances where non-PIO actions can be selected (for example, if the consequences of uncoordinated action are severe).

Mechanisms can guarantee immediate coordination or eventual coordination or provide no such assurances. To illustrate, I list some simple (and commonly used) coordination methods. Several of these are learning mechanisms, whereby coordination is achieved through repeated interaction.

Randomization

Randomization is a learning mechanism that requires agents to select a PIO action randomly until coordination is achieved (that is, an optimal joint action is selected by the group). At this point, the agents play the optimal joint action forever. Assume that actions are selected according to a uniform distribution. The mechanism has $k + 1$ states, one denoting coordination on each of the k optimal joint actions and one denoting a lack of coordination; it changes to a coordinated state once the corresponding optimal action is played. Often only two mechanism states need to be distinguished—(1) coordinated and (2) uncoordinated—if all coordinated values are identical.[4] The randomization protocol clearly guarantees eventual coordination at a rate dictated by the number of agents and the number of choices available to them. We can view this protocol as a finite-state machine (FSM). The FSM for the coordination problem at s_2 in figure 1 is (partially) illustrated in figure 2. A related technique is *fictitious play*, a learning technique commonly studied in game theory (Fudenberg and Levine 1998; Brown 1951) that can be applied to fully cooperative games (Boutilier 1996; Monderer and Shapley 1996). Unlike randomization, it leads to faster coordination as the number of agents and actions increase (Boutilier 1996).

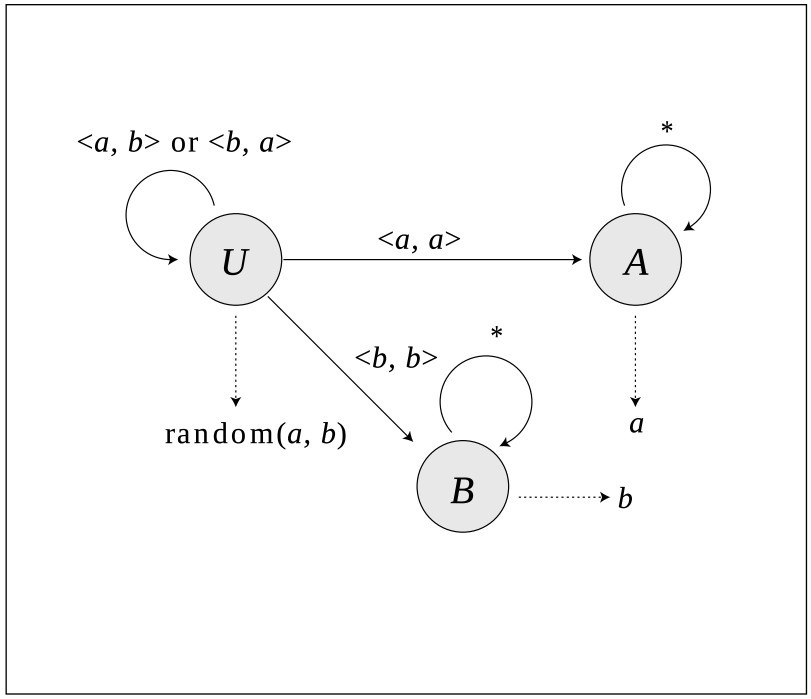

Figure 2. A Simple Finite-State Machine for the Randomization Mechanism.

The solid arrows denote state transitions, labeled by input (observed joint actions); the dashed arrows indicate output (action choices).

Lexicographic Conventions

Conventions or social laws (for example, driving on the right-hand side of the road) are often used to ensure coordination (Shoham and Tennenholtz 1992; Lewis 1969). Lexicographic conventions can be applied to virtually any coordination problem. Given some commonly known total ordering of both agents and individual actions, the set of optimal actions can totally be ordered in several different ways. Lexicographic conventions ensure immediate coordination but can have substantial overhead because they require that each agent have knowledge of these orderings. This might be reasonable in a fixed setting but might be harder to ensure over a variety of decision problems (for example, involving different collections of agents). In contrast, the learning models described earlier can be viewed as metaprotocols that can be embodied in an agent once and applied across multiple decision problems.

Communication

A natural means of ensuring coordination is through some form of communication. For example, one agent might convey its intention to perform a specific PIO action to another agent, allowing the other agent to select a matching PIO action. There are a number of well-known difficulties with devising communication and negotiation protocols, involving issues as varied as synchronization and noisy channels. I do not delve into such issues here, but even when some agreed-on negotiation protocol is in place, realistically, communication has some cost, some risk of failure or misinterpretation, and delays the achievement of goals in a way that must be modeled if optimal behavior is to be attained. As such, I would claim that one must model communication using actions in an MMDP that have effects not on the underlying system state but on the "mental state" of the agents involved.

Sequential Optimality with Coordination Problems

Coordination problems arise at specific states of an MMDP but must be considered in the context of the sequential decision problem as a whole. It is not hard to see that coordination problems such as the one at s_2 in figure 1 make the joint value function misleading. For example, $\tilde{V}^1(s_2) = 10$ and $\tilde{V}^1(s_3) = 5$, suggesting that $a1$ should take action a at $s1$ with 2 stages to go. However, $\tilde{V}^1(s_2)$ assumes that the agents will select an optimal, coordinated joint action at s_2. As discussed earlier, this policy might not be able to be implemented. Generally, $\tilde{V}$ will overestimate the value of states at which coordination is required and, thus, overestimate the value of actions and states that lead to them.

A more realistic estimate $V^1(s_2)$ of this value would account for the means available for coordination. For example, if a lexicographic convention were in place, the agents are assured of optimal action choice, whereas if they randomly choose PIO actions, they have a 50-percent chance of acting optimally (with value 10) and a 50-percent chance of miscoordinating (with value –10). Under the randomization protocol, $V^1(s_2) = 0$ and $V^1(s_3) = 5$, making the optimal decision at s_1, with two stages to go, opting out of the coordination problem—$a1$ should choose action b and move to s_3.

Unfortunately, pursuing this line of reasoning (assuming a randomization mechanism for coordination) will lead $a1$ to always choose b at s_1, no matter how many stages remain. If we categorically assert that $V^1(s_2) = 0$, we must have that $V^t(s_3) > V^t(s_2)$ for any stage $t \geq 1$. This reasoning ignores the fact that the coordination mechanism in question doesn't require the agents to always randomize: Once they have coordinated at s_2, they can choose the same (optimal) joint action at all future encounters at s_2. Clearly, $V^1(s_2)$ depends on the state of the coordination mechanism. If the agents have coordinated in the past, then $V^1(s_2) = 10$ because they are assured coordination at this final stage; otherwise, $V^1(s_2) = 0$. By the same token, $V^t(s_2)$ depends on the state of the mechanism for arbitrary $t \geq 1$, as does the value of other states.

The optimal value function V is not a function of the system state alone; it also depends on the state of the mechanism. By expanding the state space of the original MMDP to account for this, we recover the usual value function definition. In this example, I define the *expanded MMDP* to have states of the form $\langle \mathcal{S}, \mathcal{C} \rangle$, where $\mathcal{S}$ is some system state, and $\mathcal{C}$ is the state of the randomization mechanism. More precisely, given the FSM defining the randomization protocol in figure 2, the possible states of the mechanism are U (the agents are uncoordinated), A (the agents have coordinated on joint action $\langle a, a \rangle$ and B (they have coordinated on $\langle b, b \rangle$).Transitions induced by actions are clear: Each action causes a system state transition as in the MMDP, but the mechanism state changes from U to A or B only if the agents choose action $\langle a, a \rangle$ or $\langle b, b \rangle$ at s_2 (and never reverts to U). The coordination protocol also restricts the policies the agents are allowed to use at s_2. If they find themselves at (expanded) state $\langle s_2, U \rangle$, they must randomize over actions a and b. As such, the transition probabilities can be computed easily: $\langle s_2, U \rangle$ moves to both $\langle s_4, A \rangle$ and $\langle s_4, B \rangle$ with probability 0.25 and moves to $\langle s_5, U \rangle$ with probability 0.5. Note that the protocol has nothing to say about choices at states other than s_2.

The expanded MMDP is essentially the cross-product of the original MMDP and the FSM defining the protocol. The protocol restricts choices at the state where a coordination problem exists, but otherwise, actions can be chosen in any way that maximizes expected value. Specifically, optimal policy-construction algorithms such as value iteration can be applied to the expanded MDP. In this example (using C to refer to mechanism states A or B), we have $V^t \langle s_2, U \rangle < V^t \langle s_3, U \rangle$ or all stages $t < 8$, but $V^t \langle s_2, U \rangle \geq V^t \langle s_3, U \rangle$ for $t \geq 8$. Thus, if the agents have not coordinated with eight or more stages to go, $a1$ will still opt in at s_1 and move to s_2 but will avoid s_2 with fewer than 8 stages to go. This example shows how knowledge of the state of the coordination mechanism allows the agents to make informed judgments about the (long-term) benefits of coordination, the costs of miscoordination, and the odds of (immediate or eventual) coordination. With a sufficient horizon, the short-term costs are worthwhile given the long-term expected payoffs.

In Boutilier (1999), I describe a general value-iteration algorithm that takes as input an MDP and a coordination mechanism and constructs an optimal policy subject to the constraints that coordination problems are resolved using the specific coordination mechanism. This model is able to associate a well-defined sequential value to MMDP states by incorporating the mechanism state where necessary. Shapley's (1953) stochastic games provide a related sequential multiagent decision model with a well-defined value for game states. This value, however, is a consequence of the zero-sum assumption, which removes the reliance of state value on the selection of a (stage game) equilibrium. In particular, it does not apply to fully cooperative settings where coordination problems arise.[5]

Design of Coordination Protocols

By casting the coordination problem in a sequential framework, one can pose interesting questions about the design of specific coordination mechanisms and how well they can be expected to perform for the class of decision problems an agent is expected to face. The value of imposing a given coordination protocol can be defined using the value function induced in a given MMDP when the protocol is adopted. There might be costs associated with imposing the protocol, but these costs can be weighed against any potential advantages (gain in performance).

As an example, a lexicographic protocol might induce immediate coordination. The increase in expected value over, say, a randomization protocol can be measured precisely in the expanded state model and used to decide whether the overhead required to incorporate the convention is worthwhile. For any specific decision problem and collection of agents, it can be hard to justify the use of a randomization protocol. However, when the collection of agents (and their corresponding abilities) change over time, ensuring the feasibility of a lexicographic protocol can be difficult. In this sense, a randomization protocol can be considered a metaprotocol: It applies to a wide range of situations and does not need to be predefined. Similarly, endowing agents with the ability to communicate their intended actions can be evaluated by trading off the object value of an MMDP (or class of MMDPs) with the costs associated with providing agents with communication apparatus, appropriate negotiation protocols, and so on.

One can also address the problem of designing robust, computationally effective, and value-increasing coordination protocols in the framework. In a certain sense, such an undertaking can be viewed as one of designing social laws (Shoham and Tennenholtz 1992). It is also related to the issues faced in the design of protocols for distributed systems and the distributed control of discrete-event systems (Lin and Wonham 1988). However, rather than designing protocols for specific situations, metaprotocols that increase value over a wide variety of coordination problems would be the target.

Economic Methods of Coordination

A great deal of attention has been paid to the development of economic models and protocols for the interaction of agents in distributed and multiagent systems. Often, agents need access to specific resources to pursue their objectives, but the needs of one agent might conflict with those of another. A number of market-based approaches have been proposed as a means to deal with the resource allocation and related problems in multiagent systems (Wellman et al. 1998; Sandholm 1996). Of particular interest are auction mechanisms, where each agent bids for a resource according to some protocol, and the allocation and price for the resource are determined by specific rules (McAfee and McMillan 1987). Auctions have a number of desirable properties as a means for coordinating activities, including minimizing the communication between agents and, in some cases, guaranteeing Pareto efficient outcomes (Wellman et al. 1998; McAfee and McMillan 1987).

Of considerable recent interest is devising auction mechanisms for dealing with resources exhibiting "complementarities"—that is, one resource has no value without another—and substitutability—that is, one resource can be used in place of another. Research in combinatorial auctions and simultaneous actions (Rothkopf, Pekeč, and Harstad 1998; Rassenti, Smith, and Bulfin 1982; Rothkopf 1977) addresses such issues.

In sequential decision making under uncertainty, say, that involves the solution of an MDP, an agent generally considers a number of potential courses of action and settles on the one with the highest expected utility. However, when different courses of action require different collections of resources to be implemented, an agent must also plan to obtain these resources. It is clear that planning agents often require complementary resources and those that can be substituted. For example, being allocated trucks without fuel or drivers renders the trucks worthless because they cannot be used to transport goods (that is, pursue the course of action for which they were required). By the same token, once trucks and drivers are obtained for transporting material in an optimal fashion, helicopters and pilots lose any value they might have had.

If these resources are allocated using a market-based mechanism, such as an auction, then an MDP formulation of a DTP problem can be viewed as providing a framework from which the economic valuations associated with resources are drawn. Economic models generally assume that agents enter a market with prespecified valuations for specific goods (or bundles of goods) and then act according to those valuations; little attention is paid to how those valuations arise. In this sense, DTP plays a complementary role to economic models for resource allocation. For example, the value of one set of goods relative to another can be defined as the difference in expected value obtained by acting optimally with the first set of goods compared to the second.

If that's all that DTP had to offer to the use of economic methods in multiagent systems, the research challenges facing the integration would not be difficult. Often overlooked, however, is the fact that an agent must plan not only for its domain-level objectives but also for its plan-suitable economic behavior. Specifically, an agent must make decisions about which resources to bid for (or purchase), when to bid for them, how much to pay, and so on. In economic models, the planning is generally restricted to behavior within a given market or auction: what to bid for a good given a fixed valuation, how to apportion one's endowment over multiple goods in simultaneous auctions, things of this nature. However, little effort has been directed toward planning at the higher level or to ways of integrating reasoning and planning about economic activities with domain-level planning. It seems to me that such an integration is crucial to both research in DTP and research in economic mechanisms applied to multiagent systems.

Reasoning and planning with economic models is clearly important to the enterprise of DTP. If we want to extend the reach of intelligent planning agents to more (and more sophisticated) domains, we must enable them to interact with other agents. There is little doubt that much of this interaction will be mediated by economic transactions of various sorts—that's simply the way much of the world works. If we don't provide our agents with the ability to plan in such a world, we severely restrict the types of activity in which planning agents can engage. Planning an activity as simple as traveling to a conference requires economic planning and not just domain-level planning.[6] Work on planning to bid in a sequence of auctions (Boutilier, Goldszmidt, and Sabata 1999; Hausch 1986) makes a start in this direction; issues such as the order in which to pursue resources, at what point to pursue them, under what conditions, and so on, are clearly critical.

Conversely, research on the design of economic protocols for multiagent interaction must eventually account for the fact that agents do not always enter markets with clear (or unchanging) valuations. It often makes sense for an agent to execute part of a domain-level plan to see if some uncertainty is resolved before committing to the purchase of specific resources. If a course of action has a particular outcome, a specific resource might have very different value to the agent, in which case it might pay to put off bidding for a resource. Of course, if anticipated demand for the resource is expected to increase over time, it might be wise to obtain it beforehand. Economic protocols that are designed to account for the sequential nature of an agent's policy, and the inherent uncertainty in many courses of actions, will offer increased value to all parties involved.

Concluding Remarks

Apart from the challenges mentioned previously, a number of other opportunities present themselves to agents that are required to plan in settings where other agents are present. Many of these issues have been explored in the multiagent systems and distributed AI communities (see, for example, Rosenschein and Zlotkin [1994] for a treatment of several of the issues described here). It is important that they also be explored with the framework of DTP and that the decision processes involved be integrated with domain-level planning.

An agent can often discover joint courses of action that have higher expected value than any individual plan through team or coalition formation. By acting together, a team of agents can often achieve more desirable objectives than they could by acting individually. A related way in which the presence of other agents can help a planning agent is *contracting:* Paying an agent to perform a task for which the planning agent is ill suited, a planning agent can satisfy objectives it might otherwise have to forego. Of course, the design of appropriate communication and negotiation protocols is crucial here (and has received a fair amount of attention in the distributed AI literature).

The presence of other agents can help a planning agent discover decent individual courses of action for itself. Planning is a computationally intensive process. By observing the behavior of other agents with similar (or analogous) goals and abilities, a planning agent can direct its search for a good plan in an especially fruitful area of policy space. Imitation offers yet another opportunity, especially in learning situations, where the decision-making agent might not be completely aware of system dynamics or action costs. Multiagent reinforcement learning is a related topic that deserves considerable exploration.

It should be clear that the extension of DTP techniques to the solution of multiagent planning problems will play an important role in broadening the applicability, and increasing the value of, multiagent systems. A number of general, interesting research challenges emerge from this integration, and it offers the opportunity to explore fertile territory at the crossroads of AI planning, economics, game theory, distributed control, and a host of other areas.

Acknowledgments

This work was supported by Natural Sciences and Engineering Research Council grant OGP0121843, Institute for Robotics and Intelligent Systems, Phase III Project Dealing with Actions, and the Defense Advanced Research Projects Agency Co-ABS.

Notes

1. I am completing setting aside the considerable research directed toward what I view as mechanisms that support multiagent interaction: social laws, communications and negotiation strategies, agent modeling, and so on. Work in these areas, however, is often directed toward one or the other of the categories that follow.

2. Allowing partial observability allows even more general problems to be studied (Myerson 1991).

3. I discuss the sequential aspects of this problem a bit later.

4. Technically, an agent requires enough memory to know which of its potentially individually optimal actions it should perform but not those of other agents.

5. Explicit reasoning about protocol states can also allow generalization, whereby the coordinated choices learned at one state can be applied directly to similar states, allowing agents to naturally fall into roles (Boutilier 1999).

6. The recent experience of a colleague, who wished to attend a workshop at a resort, illustrates just this. His plan involved finding accommodations before booking a flight because the room was the most scarce resource at this location. His plan was "appropriate": He was unable to find accommodations for the entire workshop. By putting off the purchase of a plane ticket, he was able to make the correct travel plans.

References

Boutilier, C. 1999. Sequential Optimality and Coordination in Multiagent Systems. In Proceedings of the Sixteenth International Joint Conference on Artificial Intelligence, 478–485. Menlo Park, Calif.: International Joint Conferences on Artificial Intelligence.

Boutilier, C. 1996. Learning Conventions in Multiagent Stochastic Domains Using Likelihood Estimates. In *Proceedings of the Twelfth Conference on Uncertainty in Artificial Intelligence*, 106–114. San Francisco: Morgan Kaufmann.

Boutilier, C.; Dean, T.; and Hanks, S. 1999. Decision-Theoretic Planning: Structural Assumptions and Computational Leverage. *Journal of Artificial Intelligence Research* 11:1–94.

Boutilier, C.; Goldszmidt, M.; and Sabata, B. 1999. Sequential Auctions for Allocation of Resources with Complementarities. In Proceedings of the Sixteenth International Joint Conference on Artificial Intelligence, 527–534. Menlo Park, Calif.: International Joint Conferences on Artificial Intelligence.

Brown, G. W. 1951. Iterative Solution of Games by Fictitious Play. In *Activity Analysis of Production and Allocation*, ed. T. C. Koopmans. New York: Wiley.

Fudenberg, D., and Levine, D. K. 1998. *The Theory of Learning in Games.* Cambridge, Mass.: MIT Press.

Grosz, B. J., and Kraus, S. 1996. Collaborative Plans for Complex Group Action. *Artificial Intelligence* 86(2): 269–357.

Groves, T. 1973. Incentives in Teams. *Econometrica* 41:617–631.

Hausch, D. B. 1986. Multiobject Auctions: Sequential versus Simultaneous Sales. *Management Science* 32(12): 1599–1610.

Lewis, D. K. 1969. *Conventions, A Philosophical Study.* Cambridge, Mass.: Harvard University Press.

Lin, F., and Wonham, W. 1988. Decentralized Supervisory Control of Discrete-Event Systems. *Information Sciences* 44:199–224.

McAfee, R. P., and McMillan, J. 1987. Auctions and Bidding. *Journal of Economic Literature* 25:699–738.

Monderer, D., and Shapley, L. S. 1996. Fictitious Play Property for Games with Identical Interests. *Journal of Economic Theory* 68:258–265.

Myerson, R. B. 1991. *Game Theory: Analysis of Conflict.* Cambridge, Mass.: Harvard University Press.

Puterman, M. L. 1994. *Markov Decision Processes: Discrete Stochastic Dynamic Programming.* New York: Wiley.

Rassenti, S. J.; Smith, V. L.; and Bulfin, R. L. 1982. A Combinatorial Auction Mechanism for Airport Time-Slot Allocation. *Bell Journal of Economics* 13:402–417.

Rosenschein, J. S., and Zlotkin, G. 1994. *Rules of Encounter: Designing Conventions for Automated Negotiation among Computers.* Cambridge, Mass.: MIT Press.

Rothkopf, M. H. 1977. Bidding in Simultaneous Auctions with a Constraint on Exposure. *Operations Research* 25:620–629.

Rothkopf, M. H.; Pekeč, A.; and Harstad, R. M. 1998. Computationally Manageable Combinatorial Auctions. *Management Science* 44(8): 1131–1147.

Sandholm, T. 1996. Limitations of the Vickrey Auction in Computational Multiagent Systems. In Proceedings of the Second International Conference on Multiagent Systems, 10–13 December, Kyoto, Japan.

Shapley, L. S. 1953. Stochastic Games. *Proceedings of the National Academy of Sciences* 39:327–332.

Shoham, Y., and Tennenholtz, M. 1992. On the Synthesis of Useful Social Laws for Artificial Agent Societies. In Proceedings of the Tenth National Conference on Artificial Intelligence, 276–281. Menlo Park, Calif.: American Association for Artificial Intelligence.

Wellman, M. P. 1993. A Market-Oriented Programming Environment and Its Application to Distributed Multicommodity Flow Problems. *Journal of Artificial Intelligence Research* 1:1–23.

Wellman, M. P.; Walsh, W. E.; Wurman, P. R.; and MacKie-Mason, J. K. 1999. Auction Protocols for Decentralized Scheduling. *Games and Economic Behavior.* Forthcoming.

Craig Boutilier is an associate professor of computer science at the University of Toronto. He received his Ph.D. in 1992 from the University of Toronto and was an assistant and associate professor at the University of British Columbia from 1991 to 1999 before returning to Toronto. His research interests include planning, reasoning, and sequential decision making; Markov decision processes and reinforcement learning; probabilistic reasoning; and knowledge representation schemes for actions and preferences. He is an associate editor with the *Journal of Artificial Intelligence* and was awarded a Killam Faculty Research Fellowship in 1997–1998. His e-mail address is cebly@cs.ubc.ca.

The 2001 AAAI Spring Symposium Series

Call for Proposals

AAAI invites proposals for the 2001 Spring Symposium Series, to be held March 26-28, 2001 at Stanford University, California.

The Spring Symposium Series is an annual set of meetings run in parallel at a common site. It is designed to bring colleagues together in an intimate forum while at the same time providing a significant gathering point for the AI community. The two and a half day format of the series allows participants to devote considerably more time to feedback and discussion than typical one-day workshops. It is an ideal venue for bringing together new communities in emerging fields.

The symposia are intended to encourage presentation of speculative work and work in progress, as well as completed work. Ample time should be scheduled for discussion. Novel programming, including the use of target problems, open-format panels, working groups, or breakout sessions, is encouraged. Working notes will be prepared, and distributed to the participants. At the discretion of the individual symposium chairs, these working notes may also be made available as AAAI Technical Reports following the meeting. Most participants of the symposia will be selected on the basis of statements of interest or abstracts submitted to the symposia chairs; some open registration will be allowed. All symposia are limited in size, and participants will be expected to attend a single symposium.

Proposals for symposia should be between two and five pages in length, and should contain:

- A title for the symposium.
- A description of the symposium, identifying specific areas of interest, and, optionally, general symposium format.
- The names and (physical and electronic) addresses of the organizing committee preferably three or more people at different sites, all of whom have agreed to serve on the committee.
- A list of potential participants that have been contacted and that have expressed interest in participating. A common way of gathering potential participants is to send email messages to email lists related to the topic(s) of the symposium. Note that potential participants need not commit to participating, only state that they are interested.

Ideally, the entire organizing committee should collaborate in producing the proposal. If possible, a draft proposal should be sent out to a few of the potential participants and their comments solicited.

Approximately eight symposia on a broad range of topics within and around AI will be selected for the 2001 Spring Symposium Series. All proposals will be reviewed by the AAAI Symposium Committee Chair (Chair: Ian Horswill, Northwestern University; Cochair: Dan Clancy, NASA Ames Research Center); and an Associate Chair. The criteria for acceptance of proposals include:

Perceived interest to the AAAI community. Although AAAI encourages symposia that cross disciplinary boundaries, a symposium must be of interest to some subcommunity of the AAAI membership. Symposia that are of interest to a broad range of AAAI members are also preferred.

Appropriate number of potential participants. Although the series supports a range of symposium sizes, the target size is around 40-60 participants.

Lack of a long-term ongoing series of activities on the topic. The Spring Symposium Series is intended to nurture emerging communities and topics, so topics that already have yearly conferences or workshops are inappropriate.

An appropriate organizing committee. The organizing committee should have (1) good technical knowledge of the topic, (2) good organizational skills, and (3) connections to the various communities from which they intend to draw participants. Committees for cross-disciplinary symposia must adequately represent all the disciplines to be covered by the symposium.

Accepted proposals will be distributed as widely as possible over the subfields of AI, and balanced between theoretical and applied topics. Symposia bridging theory and practice and those combining AI and related fields are particularly solicited.

Symposium proposals should be submitted as soon as possible, but no later than April13, 2000. Proposals that are submitted significantly before this deadline can be in draft form. Comments on how to improve and complete the proposal will be returned to the submitter in time for revisions to be made before the deadline. Notifications of acceptance or rejection will be sent to submitters around April 28, 2000. The submitters of accepted proposals will become the chair of the symposium, unless alternative arrangements are made. The symposium organizing committees will be responsible for:

- Producing, in conjunction with the general chair, a Call for Participation and Registration Brochure for the symposium, which will be distributed to the AAAI membership
- Additional publicity of the symposium, especially to potential audiences from outside the AAAI community
- Reviewing requests to participate in the symposium and determining symposium participants
- Preparing working notes for the symposium
- Scheduling the activities of the symposium
- Preparing a short review of the symposium, to be printed in *AI Magazine*.

AAAI will provide logistical support, will take care of all local arrangements, and will arrange for reproducing and distributing the working notes.

Please submit (preferably by electronic mail) your symposium proposals, and inquiries concerning symposia, to:

- Ian Horswill
  AAAI Symposium Chair
  Computer Science Department
  Northwestern University
  ian@cs.nwu.edu
  Tel: 847-467-1256 / Fax: 847-491-5258