Learning and Adaption in Multiagent Systems

David C. Parkes and Lyle H. Ungar
Computer and Information Science Department
University of Pennsylvania
200 South 33rd Street, Philadelphia, PA 19104
dparkes@unagi.cis.upenn.edu ungar@cis.upenn.edu

Abstract

The goal of a self-interested agent within a multiagent system is to maximize its utility over time. In a situation of strategic interdependence, where the actions of one agent may affect the utilities of other agents, the optimal behavior of an agent must be conditioned on the expected behaviors of the other agents in the system. Standard game theory assumes that the rationality and preferences of all the agents is common knowledge: each agent is then able to compute the set of possible equilibria, and if there is a unique equilibrium, choose a best-response to the actions that the other agents will all play.

Real agents acting within a multiagent system face multiple problems: the agents may have incomplete information about the preferences and rationality of the other agents in the game, computing the equilibria can be computationally complex, and there might be many equilibria from which to choose. An alternative explanation of the emergence of a stable equilibrium is that it arises as the long-run outcome of a repeated game, in which bounded-rational agents adapt their strategies as they learn about the other agents in the system. We review some possible models of learning for games, and then show the pros and cons of using learning in a particular game, the *Compensation Mechanism*, a mechanism for the efficient coordination of actions within a multiagent system.

keywords: game theory, mechanism design, learning

Introduction

Multiagent systems can be viewed as games where the artificial agents are bounded-rational utility-maximizers with incomplete information about the other agents in the system. The problem for designers of multiagent systems is to establish rules of the game that encourage agents to choose strategies that optimize *system-wide* utility. Game theory is a useful tool because it predicts the strategies that rational agents will choose to play in a particular game. We can use this as a normative theory: given a mechanism what will rational agents do? However game theory typically makes assumptions that go far beyond the rationality-assumption that is made in classical economics.

In game theory it is not enough for agents to choose strategies that are optimal, given their beliefs about each other's intended choices. Either the preferences and rationality of all the agents must be *common knowledge*, or the agents must know each other's intended choices and the choices must form a strategic equilibrium. This assumption of common knowledge is unpalatable for multiagent systems, where it is often the existence of private information that motivates decentralization in the first place. The key to applying game-theoretic techniques to the design of multiagent systems is our ability to weaken these assumptions while maintaining desirable system properties, such as efficient coordination and robustness to manipulative behavior.

We introduce learning as a method for bounded-rational agents to adapt to an optimal strategy in a game of incomplete information. The aim is to demonstrate that agents using simple learning techniques can reach the same solution as rational agents playing in a game of complete information. The learning model that an agent uses must suit the particular dynamics of multiagent systems: the agents in the system can change, and existing agents can adapt and change their strategies as they learn. An agent must also take meta-decisions about when to choose a best-response strategy given its current knowledge, and when to deviate and gather more information about the preferences of the other agents.

This paper is organized as follows: we first discuss the application of game theory to the problem of mechanism design in a multiagent system; then we consider general models of learning that are applicable to multiagent systems; we illustrate a simple learning model from game theory on some two-player games;

and finally we consider a concrete problem and a concrete mechanism, and demonstrate the system with and without learning. The problem is one of multi-commodity flow. We extend the *Compensation Mechanism* to allow learning and adaption, and show that even bounded-rational agents with private preferences will converge to a system-wide optimal solution.

Mechanism Design

The goal of system design is to achieve good coordination of heterogeneous, self-interested, bounded-rational agents within a dynamic system. We assume that the system designer has no knowledge about the preferences of the agents in the system, and is unable to perform a global optimization and prescribe strategies. The problem of mechanism design is how to establish rules of the game that promote truthful revelation of preferences and allows efficient coordination (Kraus 1996). The system designer is only able to *indirectly* influence the actions of the agents through an appropriate reward and penalty structure. We would like to design a game where the actions that emerge as the optimal strategies for self-interested agents are also the actions that achieve the system-wide optimal coordination.

Game theory offers a principled approach to the problem of mechanism design, but introduces some new problems. Standard game theory makes very strong assumptions about common knowledge within a system. A necessary condition for a stable equilibrium is that each agent plays a best-response to the strategy of every other agent: this is known as a *Nash equilibrium*. The preferences and rationality of all the agents in the system must be common knowledge for agents to compute and play this equilibrium.

The system designer can weaken these assumptions by implementing a mechanism that has a *dominant* best-strategy for every agent in the system, (e.g. the *sealed-bid second price auction* (McAfee & McMillan 1987)), or allow for a repeated game where agents learn, and *adapt* to a Nash equilibrium. The learning dynamic permits bounded-rational agents with incomplete knowledge about the preferences of the other agents to converge to an optimal strategy. An adaptive system also brings wider gains over a prescriptive system. It is essential for a dynamically changing system (maybe agents are entering or leaving), it is robust to incorrect assumptions about the rationality of other agents in the system (what if another agent is "stupid"?), it permits humans to enter the game, and it allows for dynamic changes in preferences.

General Models of Learning

The general learning problem can be formulated as learning a *(state-action)* or *(state-action-value)* function that represents the best action that an agent should take given a past history of payoffs and actions of the other agents in the system. The *state* of the world models all of the information that an agent uses to adapt its strategy. An agent with perfect recall can store the complete history of his actions, the actions of the other agents, and the payoffs received. A bounded-rational agent with limited recall must *forget* some of the past history of the game. Possible approaches are to extract features, maintain a fixed window into the past, or store a summary of the past (such as a distribution over past actions). The main approach to learning within multiagent systems is *reinforcement learning*. The tendency to choose an action in a given state is strengthened if it produces favorable results, weakened if unfavorable.

The most important choice for an agent within an adaptive multiagent system is whether to model the other agents in the system and compute optimal actions based on this model and knowledge of the reward structure of the game (*model-based learning*) (Gmytrasiewicz & Durfee 1995), or to directly learn the expected utility of actions in a given state (*direct-learning*). A direct learning approach has been proposed for sequential games, where agents increase the probability of playing actions that have met with success in previous periods. This is a version of *Q-learning*, where agents modify the worth of a strategy according to recent experience of success or failure. This model has been used with considerable success in simple games such as the *Prisoner's Dilemma* (Sandholm & Crites 1995). An example of a model-based approach is *fictitious play*, which is presented in the next section.

Learning within multiagent systems is, in general, very hard. The main problem is the dynamic nature of the system: the strategies that other agents play will be continually changing as they learn and adapt within the system. An equilibrium of the system need not be a Nash equilibrium if agents have inaccurate models about the preferences of the other agents (Hu & Wellman 1996). This introduces the possibility of *strategic* adaptive play. A strategic agent, agent A, might choose to model the learning method of another agent, agent B. Agent A can choose to make non-optimal short-term actions in order to deceive agent B into learning an incorrect model of Agent A's preferences, and then Agent A can take long-term advantage of this incorrect model.

We consider two types of agents within a system:

myopic-learning agents that use a simple, short-term learning model, and *strategic-learning* agents, that consider the long-term equilibrium of the system and model the *learning* process of the other agents when taking decisions.

The mechanism should satisfy *myopic-optimality*: the equilibrium for a system of myopic-learning agents should converge to the equilibrium that is predicted in game theory for a system of rational agents with complete information. Good mechanism design should allow a simple myopic-learning agent to do reasonably well, even against strategic-learning agents. We would also like to prove an upper-bound on the gains that an agent expect from strategic action. Strategic-learning is complex, so a mechanism should be simple enough to allow myopic-learning agents to do well.

It is interesting to note that there are worlds where there are no gains from strategic-learning. The dynamic process of *price-tatônnement* to a general-equilibrium in a competitive market is a good example. In a sufficiently large economy the agents have no market power, and cannot influence the future direction of prices. Myopic-learning is optimal here: the best that an agent can do is to assume that the prices remain unchanged. This is still approximately true in a large, but finite, economy.

Example: Myopic-learning agents

A simple form of model-based learning that has been suggested as a good model for learning a Nash equilibrium is *fictitious play* (Fudenberg & Levine 1997). The agents use information about the past choices and payoffs of their opponents to update their own beliefs about opponent choices in the next round of the game, and then choose optimal responses to these beliefs. Each agent models the other agents with a stationary distribution of strategies, and computes its best-response to its current model of the other agent: the strategy that maximizes its expected-utility in the final state of the game. If fictitious play converges to a single pure strategy then it is guaranteed to be a Nash equilibrium.

Fictitious play has mainly been studied in the context of learning in two-player *normal-form* games, although extensions have been considered to multi-player games. The normal-form representation of a game assumes that there is one-round and the strategies of each agent are announced simultaneously. The game is represented as a set of possible strategies for each agent, and a payoff matrix that details the payoff that each agent will receive given the particular strategies played by the other agents in the game. The strategy profile that rational agents choose to play must be a Nash equilibrium: the strategy of each agent is the best-response (in a utility-maximizing sense) to the strategy of every other agent. This is necessary for an equilibrium to be *self-enforcing*, because then no agent has an incentive to deviate.

Consider game (a) in figure 1. The strategy space for agent 1 is $\{U, D\}$ and the strategy space for agent 2 is $\{L, M, R\}$. The payoff matrix represents the payoffs that the agents will receive in every outcome of the game. For example, agent 1 and agent 2 will receive payoffs of 4 and 2 respectively at the outcome (D, R). The only pure Nash equilibrium in this game is the strategy profile (U, M). This strategy profile can easily be learned with fictitious play (Fudenberg & Kreps 1993).

The problem with fictitious play is that we cannot expect convergence to a pure strategy, even when one exists. Consider game (b) in figure 1. The only change from game (a) is that the payoff of agent 2 in the outcome (U, M) has been changed from 4.7 to 4. The result of fictitious play is a cycle $(U, L) \rightarrow (U, R) \rightarrow (B, R) \rightarrow (U, L)$. The empirical frequencies actually converge to those of a mixed strategy[1] for the game, although the agents do not play the prescribed randomized strategy - they play the cycle.

		Agent 2		
		L	M	R
Agent 1	U	5, 1	8, 4.7	2, 3
	D	2, 3	2, 1	4, 2

(a)

		Agent 2		
		L	M	R
Agent 1	U	5, 1	8, 4	2, 3
	D	2, 3	2, 1	4, 2

(b)

Figure 1: Two normal-form games

Example: Strategic-learning agents

The assumption made in fictitious play, that agents do not try to influence future play but merely play a best-response to their current model of the world, is hard to justify. Each agent must believe that the other agents are non-adaptive and have a a stationary distribution of strategies. This is a clear inconsistency - why should an agent believe that other agents are so different from itself?

The normal-form game in figure 2 demonstrates the non-optimality of myopic-learning. Under fictitious

[1] A mixed strategy for an agent is defined by a probability distribution over a set of pure strategies.

play the best short-term strategy for agent 1 is D, for any possible model of agent 2 (this is a *dominant* strategy). Eventually agent 2 will choose to play L, as the expectation that agent 1 will play D increases. Fictitious play converges on the unique pure Nash equilibrium for the game, (D, L). However, consider what happens if agent 1 chooses to play U each time. Eventually agent 2 will play R, and agent 1 will receive a payoff of 3 for the rest of the game. The key observation is that agent 1 can exploit its beliefs about the way that agent 2 will model agent 1 in order to increase its payoff.

		Agent 2	
		L	R
Agent 1	U	1, 0	3, 2
	D	2, 1	4, 0

Figure 2: Another normal-form game

The Multicommodity Flow Problem

To better understand the pros and cons of using game theory to design a mechanism and solve a coordination problem, we consider a concrete problem: solving the multcommodity flow problem using the Compensation Mechanism, and an extension of it which includes learning (both described below).

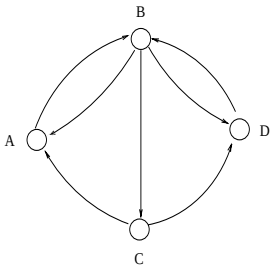

Figure 3: The Simple Network

The multicommodity flow problem is the task of allocating flows over a network to minimize the total system-wide cost, subject to satisfying certain shipping requirements. Congestible arcs[2] within the network represent externalities: an increase in flow down a shared arc due to one agent increases the per-unit cost on that arc, and has a direct effect on the costs of the other agents that are using the arc. We assume that the pricing structure of the network is known to all of the agents in the system, but that the goals, preferences and rationality of the agents are private knowledge.

We consider the case of two agents shipping over the network in figure 3 (Wellman 1993). Agent 1 has to ship 10 units of cargo from A to D, and agent 2 has to ship 10 units from D to A. Each agent chooses a shipping strategy that will meet her goal at a minimum cost. The shared arc $B \rightarrow C$ represents an externality, and we wish to use a mechanism with an incentive structure that promotes efficient system-wide usage of this arc.

The Compensation Mechanism - without learning

The *Compensation Mechanism* (Varian 1994) is a *one-shot* two-stage extensive-form game that implements efficient outcomes in a multiagent system. The mechanism is designed to solve market failures due to externalities in classic economics, a canonical example of which is the *Tragedy of the Commons* (Hardin 1968). The idea behind the mechanism is simple: the agents all report the compensation that they require for the actions of the other agents, and the other agents consider this level of compensation when choosing their best strategy. There is now an incentive for the agents to consider the wider effects of their actions. The mechanism is designed in such a way that an agent can not gain from misrepresenting the level of compensation that it requires.

Consider the choice of shipping level, x, down the arc $B \rightarrow C$ by agent 1. The preference of agent i over the shipping level x is represented as a utility function, $u_i(x)$. The game has two stages: an *announcement stage* and a *choice stage*. In the announcement stage, agent 2 announces p_2, the penalty that agent 1 will pay to the center in compensation for the effect that her actions will have on agent 2. Agent 1 simultaneously announces p_1, the reward that agent 2 will receive from the center. In the choice stage of the game, agent 1 chooses the level of shipping, x, that maximizes her payoff given the penalty structure announced. The payoff to agent 1 is

$$\pi_1(p_1, p_2, x) = u_1(x) - p_2 x - (p_1 - p_2)^2 \qquad (1)$$

Notice that the penalty that agent 1 pays for shipping x units is independent of the level of compensation that agent 1 announces that agent 2 should receive. The penalty term $(p_1 - p_2)^2$ provides an incentive for agent 1 to announce the same level of compensation as requested by agent 2. The payoff to agent 2 is

$$\pi_2(p_1, p_2, x) = u_2(x) + p_1 x \qquad (2)$$

[2] A congestible arc has a per-unit cost that increases as the total volume shipped increases.

The unique subgame perfect[3] Nash equilibrium (SPNE) of the game has agent 1 being taxed at a rate equal to the marginal effect that her shipping level has on agent 2. See (Varian 1994) for a proof. This is precisely the amount of taxation required for agent 1 to choose a socially-optimal (total utility- maximizing) shipping level. The transfer prices announced at the SPNE are:

$$p_1 = p_2 = -\frac{\partial u_2(x)}{\partial x} \tag{3}$$

The problem is that all of the agents must be able to solve the SPNE of the game, which requires common knowledge of preferences, goals, and rationality. Consider that both agents must announce, in the first stage of the game, agent 2's marginal utility for the shipping level of agent 1 at the level of shipping that agent 1 will choose in the second stage of the game. This requires that each agent know the preferences and rationality of the other agent, which is unrealistic for a multiagent system.

The Compensation Mechanism - with learning

We can relax the assumptions about common knowledge by allowing a repeated game that permits learning and adaption. The agents communicate compensation levels and choose best-response flows in each round of the game, and learn the best strategy. The most straightforward model to suggest is the simple myopic-learning model (Varian 1994), where at time $t+1$:

$$p_1(t+1) = p_2(t) \tag{4}$$

$$p_2(t+1) = p_2(t) - \gamma \left[p_1(t) + \frac{\partial u_2(x)}{\partial x} \right] \tag{5}$$

where γ is a suitable constant. This is model-free learning because the agents make no attempt to consider the learning strategy of the other agents. The state of the world is defined by the actions (compensation levels) from the previous round, and the agents learn a *(state-action)* function.

The history of the flows and compensation levels for the shared arc $B \rightarrow C$ is shown in figure 4. The mechanism is activated at 30 iterations. The first graph represents the flow down the shared arc $B \rightarrow C$. The flow converges within around 20 iterations to the optimal system-wide flow (illustrated as the horizontal line). The long-run equilibrium of this dynamic system is the same as for the two-stage game, so we have myopic-optimality. See (Varian 1994) for a proof. Notice that without the mechanism the flow down the shared arc $B \rightarrow C$ is higher than the optimal level because the agents increase the flow until the marginal cost to them equals their marginal benefit, but ignore the cost to the other agent. The long-term cost to each agent is lower than the cost in the system without the mechanism.

[3] A subgame perfect Nash equilibrium is a refinement for sequential games that rules out non-credible threats.

A Comparison: with and without learning

The basic compensation mechanism is incentive compatible: agents will choose to truthfully reveal their preferences over the actions of the other agents, and select an efficient coordination solution. This property comes at an unreasonable price: the agents must have common knowledge about the preferences and rationality of the other agents in the system. We introduce learning and adaption to avoid this requirement, and demonstrate that simple myopic-agents will converge to the same solution. Learning and adaption also make the system more robust to imperfect knowledge.[4]

We can compare the computational requirements of learning to that of model-building and computation. In the traditional game-theoretic setting of common knowledge, each agent must solve an $n-$player simultaneous optimization problem for the game: each agent is modeled as optimizing payoff subject to the actions of all the other agents. The computation time for this problem is clearly no worse than the total computation time for the solution using learning because each agent could simulate the dynamic version of the game. The adaptive solution gives a $n-$times speedup through parallelization, since each agent solves its own optimization in parallel with the other agents in each round of the game. The main advantage of learning however, is not the computational savings, but that it allows us to apply game-theory to the design of mechanisms for realistic multiagent systems.

Conclusions

Game theory provides a valuable insight into mechanism design for multiagent systems. One can design mechanisms that have attractive properties: E.g., incentive compatibility so that agents will choose to reveal their true preferences (such as the *sealed-bid*

[4] We should, however, note that strategic-learning agents could play the repeated game to take advantage of the presence of simple-minded myopic-learning agents. Clearly an agent might be able to behave strategically and take local actions that influence the final outcome of the game. As an example, if agent 2 knows that agent 1 is just setting $p_1(t+1) = p_2(t)$ at every round of the game, then agent 2 might choose to increase the compensation that agent 1 must pay to the center, knowing that agent 1 will also announce an increase in the compensation that agent 2 should receive from the center.

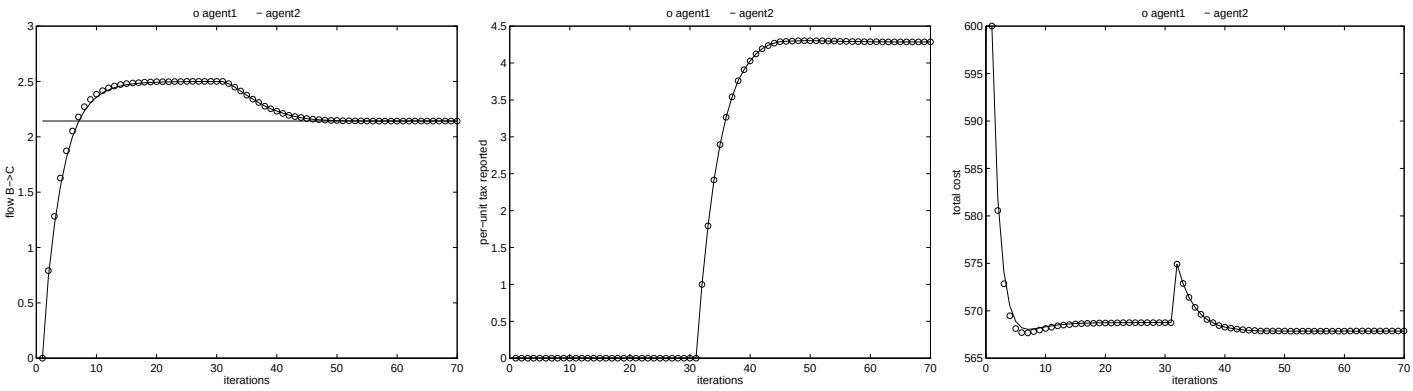

Figure 4: The compensation mechanism (a) Flow $B \rightarrow C$ (b) Compensation level (c) Cost to agent

second price auction), and Pareto efficiency so that agents will achieve good coordination even where markets traditionally fail (such as the *Tragedy of the Commons*). Without learning, game theory makes assumptions that are unreasonable for systems of bounded-rational artificial agents: E.g., that the preferences and rationality of the agents in the system are common knowledge, or the agents have an accurate model of the other agents and are able to compute the fixed point of an infinite regression of the form *she knows that he knows that she knows that...*

Learning avoids the need for common knowledge, and allows system designers to consider the performance of a mechanism within a system of bounded-rational agents with private information. We want to design systems that will allow learning while maintaining certain desirable properties of the mechanism, such as incentive compatibility and efficient outcomes. Another desirable property of a mechanism for an adaptive-system is that it is *myopically-optimal*: a system of myopic-learning agents will adapt to the same equilibrium as a system of rational agents with common knowledge. The extended compensation mechanism presented above has this property. The designer of an adaptive-mechanism should also seek to prove upper bounds on the gains that an agent can hope to achieve through strategic behavior. The mechanism should allow simple myopic-learning agents to do well, even in the presence of strategic-learning agents.

Our future work will look more closely at the tradeoffs that agents can make between deliberation and action, and build a taxonomy of learning strategies. We would like to understand how to maintain incentive-compatibility results for standard mechanisms while allowing repeated play and learning - and to design mechanisms that have the desirable property of myopic-optimality, and allow upper-bounds to be proved for the potential gains from strategic-learning.

Acknowledgments

This research was funded in part by National Science Foundation Grant SBR 96-02053.

References

Fudenberg, D., and Kreps, D. M. 1993. Learning mixed equilibria. *Games and Economic Behavior* 5:320–367.

Fudenberg, D., and Levine, D. 1997. *Theory of Learning in Games*. MIT Press. forthcoming.

Gmytrasiewicz, P. J., and Durfee, E. H. 1995. A rigorous, operational formalization of recursive modeling. In *(ICMAS-95)*.

Hardin, G. 1968. The tragedy of the commons. *Science* 162:1243–1248.

Hu, J., and Wellman, M. P. 1996. Self-fulfilling bias in multiagent learning. In *(ICMAS-96)*.

Kraus, S. 1996. An overview of incentive contracting. *Artificial Intelligence* 83:297–346.

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

Sandholm, T. W., and Crites, R. H. 1995. Multiagent reinforcement learning in the iterated prisoner's dilemma. *Biosystems* 37:147–166.

Varian, H. R. 1994. A solution to the problem of externalities when the agents are well-informed. *American Economic Review* 84(5):1278–1293.

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.