Bounded Rationality in the Iterated Prisoner's Dilemma

Jennifer Tillett
Department of Computer Science
University of British Columbia
Vancouver, B.C., Canada V6T 1Z4
jtillett@cs.ubc.ca

ABSTRACT

In diverse fields such as computer science, economics and psychology, bounded rationality has emerged as an important research topic. Models which assume the existence of perfectly rational agents seem inadequate for many real-world problems where agents often lack perfect rationality. Classes of imperfect rationality include the conditions of incomplete knowledge, memory, information or computational ability.

In this paper, bounded rationality is investigated in the context of the Iterated Prisoner's Dilemma (IPD). Six papers are surveyed which address issues of optimality and cooperation in repeated games with applications to IPD, using either a machine learning or game theory approach. Each paper imposes some bound on players' rationality; these approaches to bounding are classified and compared based on their results and applicability.

Categories and Subject Descriptors

I.2.11 [**Distributed Artificial Intelligence**]: Multiagent systems

General Terms

bounded rationality, Prisoner's Dilemma

1. INTRODUCTION

In diverse fields such as computer science, economics and psychology, bounded rationality has emerged as an important research topic, because often models of perfectly rational agents are inadequate for real-world problems. As [6] writes, the "'perfect rational man' paradigm" is dissatisfying because in real situations "decision makers are not equally capable of analyzing a situation even when the information available to all of them is the same."

Simon defines 'substantive rationality' as simply choosing the best action and 'procedural rationality' as finding the approximately best action [8]. With substantive rationality, agents are assumed to have unlimited reasoning power, and to be willing to use all of their reasoning resources to extract more payoff, while under procedural rationality, agents expend due deliberation but have access only to limited reasoning power. In bounding rationality, the actions available to agents are only procedurally rational. When players are thus bounded, predictions about the analysis players engage in to select and implement strategies is changed, and one can begin to consider outcomes which were previously considered irrational. [6] identifies several categories of bounded rationality, including limits of knowledge, memory, information and computational ability.

The well-known game of Prisoner's Dilemma is often used as a test ground for models of bounded rationality. However, it is not just a toy setting. In real-world arenas as diverse as politics and biology, Prisoner's Dilemma is often played out in ways not predicted by strictly rational models. Spawning sea bass learn to divide their sex roles and U.S. Senators regularly trade votes, challenging the rational model that predicts agents will never cooperate [1]. The Iterated Prisoner's Dilemma (IPD) is the playing of repeated stage games of Prisoner's Dilemma, where the number of iterations may or may not be known by the players in advance. In this context, each player must adopt a strategy to determine whether to cooperate or defect in each stage game. Axelrod [1] has discussed that in IPD, there is an opportunity for players to settle on a cooperative equilibrium, in which each player cooperates to ensure the most mutually satisfying outcome. In [1] he shows that cooperation is a stable, rational outcome when: (1) the future is important; either the chance of the game continuing is high or players are not incorporating knowledge of the game's endpoint into their reasoning [6], (2) the difference in payoff between mutual cooperation and mutual defection is sufficiently large, and (3) players are adapting their strategies.

In this survey, different approaches to strategizing in IPD are compared. All of the six papers surveyed use either a machine learning or game theory approach and investigate issues of optimality and cooperation in repeated games, with applications to IPD. The papers all bound the rationality of players in some way, and the authors investigate the effect of bounded rationality on strategy selection and implementation. The various methods of bounding players' rationality are classified and compared based on their results and wider applicability to other game theoretic situations.

Table 1: Prisoner's Dilemma Payoff Matrix

	C	D
C	R, R	S, T
D	T, S	P, P

2. PRISONER'S DILEMMA

Table 1 is the normal form of the game of Prisoner's Dilemma, in which each agent has the choice to Cooperate (C) or Defect (D). Mutual cooperation yields each player the Reward payoff (R), while mutual defection yields them both a Punishment payoff (P). If one player cooperates and the the other defects, the defector gains the Temptation payoff (T) while the cooperator gets the Sucker's payoff (S). Any assignation of values to the payoffs preserves the game so long as $T > R > P > S$ and $2R > (T + S)$. The latter condition assures that it is not rational for agents simply to alternate taking advantage of each other.

In the single stage game, D is the dominant strategy for each player, no matter what strategy the other player uses, and hence {D,D} is the unique Nash Equilibrium. In the finitely repeated game, {D,D} is still the only equilibrium, due to backward induction. The other 3 outcomes are Pareto optimal, so the only sub-optimal outcome is the only equilibrium.

3. DEFINITIONS

In a 2-player game, a strategy for Player 1 is *optimal* if it is the best strategy in response to a specific strategy of Player 2. A strategy is *dominant* if it is a player's best strategy no matter what the other player plays. For infinitely repeated games, optimality and domination can be defined thusly:

Let π_i^G be the payoff for player i in game G. Fix a strategy σ_2 in the set of Player 2's strategies $\sum_2^G$. Let G^∞ be the limit of the means game and G^δ be the discounted game.

For G^∞: A strategy σ_1 is *optimal* if for every strategy $\sigma_1' \in \sum_1^G$

$$\pi_1^{G^\infty}(\sigma_1, \sigma_2) - \pi_1^{G^\infty}(\sigma_1', \sigma_2) \geq 0. \quad (1)$$

For G^δ: A strategy σ_1 is *optimal* if for every strategy $\sigma_1' \in \sum_1^G$

$$\liminf_{\delta \to 1^-} (\pi_1^{G^\delta}(\sigma_1, \sigma_2) - \pi_1^{G^\delta}(\sigma_1', \sigma_2)) \geq 0. \quad (2)$$

A strategy is ϵ-*optimal* when 0 is replaced with -ϵ in the above equations. A strategy σ_1 is *dominant* if for every strategy $\sigma 2$ in $\sum_2^G$, σ_1 is optimal.

A *cooperative equilibrium* is a pair of strategies in Nash equilibrium, such that, when played against each other, each player receives an average payoff of R for each round [4]. An *optimal equilibrium* is an equilibrium for which there are no other equilibria of greater value.

4. OVERVIEW OF PAPERS' APPROACHES

The six papers examined in this survey can be classified by two different approaches to implementing or choosing a strategy in the IPD problem. In three papers focusing on machine learning, [9] and [7] both present experimental results from applying machine learning to an agent's choice of strategy, while [2] focus on a theoretical analysis of reinforcement learning. Both of the experimental papers present reinforcement learning algorithms for boundedly rational players which they hope will lead to optimal (or optimal with respect to some bound) equilibria or an otherwise stable situation in which agents learn to cooperate. The remaining set of papers surveyed, [3], [4] and [5], approach the IPD problem from a strict game theory perspective, in which players choose a strategy computationally. These papers focus on players with limits on their computational ability and examine the complexity required to achieve various levels of optimality and equilibria.

4.1 Machine Learning

In addition to its traditional assumption of rationality, game theoretic analysis often assumes perfect knowledge of the game being played. However, that may be unrealistic for agents in real-world situations, where often there are too many game elements to keep track of and even deliberately hidden information about the game structure. Machine learning can be applied in situations where players are not aware of: the structure of the game (including the other player's possible actions and the relationship between the actions and payoffs); the other player's decisions (at least immediately), the other player's payoffs, and the fact that the player is in a game situation (i.e. that other agents' actions are affecting the player's outcomes) [9]. All of the papers addressing machine learning incorporate some form of this limited knowledge.

Both [9] and [7] present reinforcement learning algorithms for IPD which they hope will lead to optimal (or optimal with respect to some bound) play or agents learning to cooperate. [2] examines reinforcement learning to try to establish theoretical guarantees about convergence to (optimal) equilibria.

Stimpson, Goodrich and Walters [9] present a satisficing solution, which imposes this bound on an agent's rationality: rather than choosing an optimal action which maximizes the agent's utility, the agent simply chooses an action whose payoff meets some aspiration level. As long as an agent's aspiration level is being met, he can play without having to search for a maximal action. However, after each play the user's aspiration level is recalculated, as the weighted average between the last payoff and the current aspiration level. In their experiments, play between 2 satisficing agents always led to stable outcomes (mutual cooperation, mutual defection, or a cycle of outcomes). They also determined that mutual cooperation was most likely to occur when agents' initial aspirations were high, when the cooperation payoff was significantly higher than the defect payoff, when both players chose the same initial action, and when the learning rate (the rate at which aspirations were updated) was high.

Sandholm and Crites [7] also present a reinforcement learning algorithm. The authors used Q-learning, where the Q-value of a state-action pair $Q(s, a)$ is updated when that action is taken from that state, based on the immediate payoff of the action, the maximum discounted payoff possible from the next state and the learning rate. To ensure that every action is eventually tried from each state while actions with high value estimates are preferred, an exploration mechanism is introduced into Q-learning. Here the authors chose the Boltzmann distribution, which includes a temperature parameter allowing for annealing. Q-learning is guaranteed to converge to correct Q-values given several constraints: the environment must be stationary and Markovian, Q-values are stored in a lookup table, every state-action pair continues to be visited, and the learning rate is decreased appropriately over time [7]. However, in the multiagent setting these guarantees do not hold because the other agent's state can't be observed and because the environment is not stationary when the other agent is learning or otherwise adapting. To test approaches to the multiagent setting, the authors ran experiments playing Q-learners against unknown opponents playing Tit-for-Tat and other Q-learners, varying 3 factors which bounded the rationality of the agents: the length of the history the agent could maintain, lookup tables vs. recurrent neural networks, and the length of the exploration schedule. Every Q-learner learned to play optimally against Tit-for-Tat, but against other Q-learners the agents with the best results had longer history windows, lookup table memories and longer exploration schedules.

Though Claus and Boutilier [2] focus on common interest repeated games, their interest is in investigating reinforcement learning as a means for teaching coordination to agents in any sequential decision problem. They first distinguish independent learners, who ignore the existence of other agents, and joint action learners, who learn the value of both their own actions and those of other agents. Reinforcement learning is introduced when agents are also unaware of the rewards which result from joint actions. The authors chose as their algorithm Q-learning with a Boltzmann distribution, as in [7]. They identify 4 conditions required to ensure convergence to equilibria:

- The learning rate λ decreases over time such that $\sum_{\lambda=0}^{t} \lambda = \infty$ and $\sum_{\lambda=0}^{t} \lambda^2 < \infty$.
- Each agent samples each of its actions infinitely often.
- The probability $P_t^i(a)$ of agent i choosing action a is nonzero.
- Each agent's exploration is exploitive. In other words, $\lim_{t\to\infty} P_t^i(X_t) = 0$, where X_t is a random variable denoting the event that some non-optimal action was taken based on i's estimated values at time t.

However, practical considerations mean that convergence may take a very long time, due to factors such as decaying exploration and the number of observations required to shift away from a strategy with much evidence. There also is no guarantee that this method will converge on optimal equilibria. 3 strategies for biasing a joint action learner's search toward optimal equilibria are explored, using *myopic heuristics*, which focus only on the current state.

4.2 Game Theory

In the three game theory papers, bounded rationality is explored for its guarantees on both optimality and cooperation. Each paper models agents' bounds on rationality by restricting their computational ability. [5] attempt to model a human level of rationality with finite state automata, using the number of states as a measure of complexity. [3] model situations in which computers are used in decision making, and thus compare the sets of strategies returned by a range of computational devices, from fully rational agents to Turing machines to finite automata. Finally [4] investigate a minimal bound on agents' computational ability which still enables cooperative behavior, as well as a computationally tractable model in which sub-optimal play may be a player's optimal solution.

Papadimitriou and Yannakakis [5] investigate the achievement of equilibria given varying levels of complexity in strategy implementation and strategy selection. For a finite automaton implementing a strategy, the complexity of the strategy can be represented as the amount of memory needed, or the number of states in the automaton. A previous result by Myhill-Nerode held that for a pair of strategies of size $s_I(n)$, $s_{II}(n) < n$ and ≥ 2 in an n-round Prisoner's Dilemma neither automaton is capable of backward induction because neither can count to n and thus there exists a cooperative equilibrium. The finite automaton which executes the Tit-for-Tat strategy has only 2 states and so satisfies this condition. The authors of [5] wished to establish other, less-restrictive bounds on memory which would still foster cooperative behavior. They prove that while a pair of strategies of size $s_I(n)$, $s_{II}(n) > 2^n$ still results in mutual defection by backward induction, for all subexponential complexities players can achieve equilibria arbitrarily close to the cooperative payoff.

Theorem 1 (Papadimitriou and Yannakakis). *For $\epsilon > 0$, let $c_\epsilon = \epsilon/6(1+\epsilon)$. For every $\epsilon > 0$, n the n-round Prisoner's Dilemma played by automata with sizes bounded by $s_I(n)$, $s_{II}(n)$, if at least one of the bounds is smaller than $2^{c_\epsilon n}$, then there is a (mixed) equilibrium with average payoff for each player at least R-ϵ.*

If *both* players' strategies are bounded subexponentially, [5] prove that there is a (mixed) equilibrium with an average payoff for each player arbitrarily close to any Pareto optimal outcome with value greater than P. They also prove a generalization of these results to other repeated games and begin an investigation of the complexity of strategy selection in general games. They discover that they can classify the complexity for the problems of: (a) choosing a best response; (b) whether a game has a pure equilibrium; (c) whether a given payoff can be achieved in mixed equilibrium, given a zero-sum game; and (d) whether a given payoff can be achieved in mixed equilibria in a general game. These problems are members of the complexity classes NP, $\sum_2^p$, EXP, and NEXP, respectively.

Instead of attempting to model more 'human' levels of rationality with finite automata, Fortnow and Whang [3] model situations in which computers are used in decision making. They compare the sets of strategies returned by a range of computational devices and prove that certain relationships of optimality and domination will hold whenever certain computational devices are used. The sets of strategies under consideration are: (a) *rational* strategies, or the set of all possible strategies which a rational player can achieve; (b) *recursive* strategies, which are the strategies which can be computed by a Turing machine that halts on all inputs; (c) *polynomial-time* strategies, which are the set of all strategies computed by a polynomial-time Turing machine; and (d) *regular* strategies, which are those strategies which can by realizable by finite automata. By definition, $(a) \supset (b) \supset (c) \supset (d)$.

The authors introduce 2 new variations on the definitions of optimality and domination. A strategy for player i is *eventually optimal* if, after some grace period in which the other player's strategy can be learned, i's strategy is the optimal strategy among all strategies i could play given the history of the interaction so far. A strategy for player i is *eventually dominant* if it is eventually optimal against every strategy of the other player. They prove that, for any game, there is a recursive strategy which is eventually dominant for the class of rational strategies against regular strategies. However, for Prisoner's Dilemma, they prove that the lower bound for the number of rounds required for *any* strategy to become eventually dominant over regular strategies is exponential in the number of states in the minimal automaton that implements the regular strategy. For comparison, the result for the game of Matching Pennies is much better: there is a polynomial-time strategy that dominates all finite automata and converges in a polynomial number of rounds. The authors also found that polynomial-time strategies are quite robust against even rational strategies. They prove that in Prisoner's Dilemma there is a polynomial-time strategy against which there is no eventually optimal rational strategy. There is also a polynomial-time strategy with an optimal rational response but which no Turing machine guaranteed to halt can find (no recursive strategy can find even an ϵ-optimal response).

Mor and Rosenschein [4] implemented the most minimal bound on agents' rationality, by simply limiting the computation time allotted to each agent. To begin, agents are restricted to the Finite Time Repeated Prisoner's Dilemma (FTPD) shown in Table 2, where for N clock ticks, players each play an action at each clock tick, and both score 0 if one player waits (doesn't choose an action). This design ensures that two unboundedly rational players will want to play each time (because W is dominated by D for both players). If a complexity bound is placed on both players' rationality such that any "compare" action takes the player at least one clock tick, the authors present a proof that there is a cooperative equilibrium in every FTPD, as long as $R > 0$. The authors prove that complexity bound is a weaker bound than being restricted to Turing machines. The authors go on to propose a further tweak to the game: allowing a player to opt out (stop playing against the current opponent) if the opponent is waiting or not cooperating, and be rematched with another opponent. They prove that opting out is a rational strategy, so long as there is a positive probability of being rematched with a cooperative opponent, and rematching is instantaneous. The authors also consider a model for cases when maximizing expected payoff is too expensive computationally or a player lacks complete knowledge of the other players in the population. In these cases of bounded rationality, players can pursue a satisfying strategy in the FTPD with opt-out, where instead of maximizing expected payoff, players maximize their security level (the lowest payoff possible with a given strategy). They prove that the satisfying payoff approaches the maximizing payoff as the number of alternative cooperative opponents and the length of the game increase.

Table 2: Finite Time Prisoner's Dilemma Payoff Matrix

	C	D	W
C	R, R	S, T	0
D	T, S	P, P	0
W	0	0	$H \leq 0$

5. COMPARISON

The approaches of these papers represent only a small portion of the work on bounded rationality. Therefore we cannot draw any conclusions about their comprehensiveness in terms of covering the current state of the art. However, we can clarify what the methods surveyed contribute to our understanding of bounded rationality.

5.1 Methods

The three machine learning approaches differ in that [9] and [7] offer only empirical analysis, while [2] investigates theoretical guarantees in addition to presenting empirical results. All three of the machine learning algorithms are robust and achieve stable outcomes, but are not guaranteed to converge to optimal equilibria. Further, it is not clear whether the learning algorithms presented are meant to represent any particular class of 'real-world' agent.

The game theoretic papers in which computational ability is bounded offer theoretical certainties, though it is often unclear what types of agents are modeled by a particular computational device. One important thread of comparison among the papers is the focus on optimality and cooperation. In Prisoner's Dilemma, these can coincide, as cooperative equilibrium is the optimal equilibrium, but for general games this is not always the case. [3] look only at optimality and domination without considering cooperation, while the machine learning papers [9] and [7] look only for cooperation/equilibria. The remaining papers sought to balance these two concerns.

5.2 Bounds on Rationality

In each paper, some bound on players' rationality is explored. These can be generally categorized under Rubinstein's classes of bounded rationality: incomplete knowledge, memory, information and computational ability.

Rubinstein [6] identifies three costs on information: acquisition, memory and communication. Agents may be unwilling or unable to perform acquisition operations, store information in memory or participate in the process of communication necessary to gain information about their situation. The conditions in [2] bound players' information, by exploring the effects of ignoring or being unaware of other agents, being unaware of the actions of the other agents, and being unaware of the rewards associated with various actions. Their results show that reinforcement learning allows the costs of information gain to be minimized, while still converging to equilibria, though often at the expense of time.

[9]'s satisficing solution also imposes a bound on information, or rather allows a bound to be present; agents do not need to be aware of any elements of the game structure, other than being able to observe a payoff and and associate it with his last action.

Rubinstein [6] describes memory as the ability to record and retain information about changes to the game structure. It can also generally be thought of as the difference between perfect and imperfect recall. [7] bound the players' memory in Q-learning, experimenting with changes to: the length of the history the agent could maintain, lookup tables vs. recurrent neural networks, and the length of the exploration schedule. [5]'s bound on the number of states in finite automata is also a bound on the agents' memory, in that with fewer states agents are unable to represent all the possible states of the game, and so distinct histories may be viewed as the same.

Four papers impose some bound on computational ability. Rubinstein [6] characterizes this bound on complexity as an agent's trade-off between choosing a good strategy and keeping his computation as simple as possible. [9] present a satisficing solution; rather than choosing an optimal action which maximizes the agent's utility, the agent simply chooses an action whose payoff meets some aspiration level, thereby bounding the amount of computation required to choose a strategy. [5] present results when players' computation is limited to a finite state automaton, where the complexity can be directly attributed to the number of states in the machine. [3] compare approaches in which one or both players use a range of computation to compute sets of strategies, from fully rational agents to Turing machines which halt on all inputs, to polynomial-time Turing machines, to finite automata. [4] measured their bound as simply the computation time allotted to each agent, which they proved was equivalent to or even less restrictive than bounding computation to a Turing machine.

A final bound on rationality which Rubinstein [6] discusses is limited foresight. Foresight is the process of predicting the possible future states of a game in order to choose a strategy, thus it is really a combination of memory and computational ability. Backward induction is an example of using foresight. However, as we have seen and as is apparent in complex games like chess, backward induction is not always possible due to limits on either memory or computational ability. The extent of an agent's foresight can vary along three dimensions: the depth of their inquiry; the quality of the sample scenarios they select, if they must only model certain possible futures; and evaluation of the future moves they have generated [6]. In experiments, humans have demonstrated significant differences in their performance along these dimensions, as well as differences in their performance in different domains. [2]'s experiments with "myopic heuristics," in which agents consider only their current state and ignore future rewards, represent a very limited foresight. Similarly [5] discusses finite automata bounded to sizes smaller than the number of rounds, which cannot perform backward induction. Both of these papers present a simple example of limited foresight, in which agents are limited only along the dimension of depth.

6. RESULTS

6.1 Extent of Bounding

Because there are various means of bounding rationality, it is difficult to rate approaches for their 'level' of bounding. However, it seems clear that [4] impose the least constraint on rationality, as they prove that their approach is at least as rational as any Turing machine. [5] and [9] seem to impose the greatest constraints among approaches which bound agents' computational ability. The satisficing solution in [9] clearly requires the least amount of computation, as agents do not engage in any computation of sets of strategies and have only two calculations to perform at each round. Among the finite automata presented in [5] there are also very simplistic agents discussed, including the 2-state finite automaton which plays Tit-for-Tat, though the authors focus on more complex finite automata and even explore strategy selection in finite automata. However, as is demonstrated in [3], the set of strategies realizable by finite automata is very limited compared to other computational devices. It seems fair to say that the lower bound on real-world computational ability is very close to the lower bounds presented in these papers, as an agent must be able to perform at least one compare operation to be considered rational.

It seems also that the game consequences of different bounds vary with the extent of bounding. [3] prove that the extent of bounding on an agent's computational ability can drastically change the sets of strategies which can respond optimally or dominantly. Similarly, [5] show that changes to agents' computational complexity can significantly alter the space of possible outcomes.

6.2 Applications

We have seen many different methods to achieve cooperation or optimality in the Iterated Prisoner's Dilemma and other repeated games, and in each paper some rationale is implicit or provided to explain how this method of bounding rationality applies to real problem-solving. In the more complex approaches such as [3] and [4] it is clear that the results apply to situations in which computers of varying capabilities will be making decisions. The remaining papers, and [4], make claims about how little computation is required to guarantee their results. However, the question remains 'which bounds on rationality really correspond to true human or other real-world forms of rationality'?

Simon [8] discusses one real problem in limiting bounded memory. In human cognition, the limits on memory are similar to the models presented in [7] and [5] because of the fact that we have limited short-term memory space and also have limits on how quickly and in what manner we retrieve stored data. However, those constraints don't capture our ability to create "templates" that allow us faster lookup if the the data is in a familiar form, nor do any of these approaches capture the algorithms humans learn to solve problems and even remember or recognize items better. As discussed in both [8] and [6], the foresight performed by expert chess players indicates that they have learned to recognize certain arrangements of pieces immediately, and can even recognize board arrangements which have evolved from these templates or 'chunks' of memory. Grandmaster chess players also have very evolved heuristics for choosing which future moves to project, which most likely are specific to the domain. There are similar corollaries for experts in other domains. None of the approaches presented for learning or computation take the formation of these templates or heuristics into account, though most of the models do not preclude the possibility of these memory features.

In addition, it is not clear exactly how complex human computation is, even given batteries of experiments [8]. Simon [8] has presented evidence that any agent (machine or biological) which performs computation must perform it in approximately the same way, but little evidence has been presented to indicate where on the spectrum of computational devices biological agents might fall. Further, Mor [4] points out that other approaches which involve bounds on computational ability, including [5] and [3], "exhaust" the machines, meaning that the game patterns produced are so complex that the machines must use all their computational power in order to participate. These computationally-intensive versions of game play and intricate rules do not seem to correspond to the natural occurrences of Prisoner's Dilemma cited by [1] and [8].

To strict psychological and biological discussions of human cognitive models, game theory probably has little to contribute, but it seems that some of these game theoretic models, both for machine learning and computational devices, could be pitted against humans to try to tease out similarities and differences in their reasoning processes. In addition to the discoveries presented here, a greater understanding and finer granularity of the differences among computational models would aid this process.

7. CONCLUSION

Bounded rationality has become an important research topic in many fields, because models of perfectly rational agents are often dissatisfying when applied to real-world problems. 'Procedural rationality', in which agents have only limited reasoning power, seems more appropriate for models in a large range of fields, from computer science to economics to biology and psychology.

Prisoner's Dilemma is a rich testing ground for models of bounded rationality. The results presented here utilize this deceptively simple example as an expressive mechanism for exploring limits on agents in a multitude of game situations. The two main approaches discussed here, machine learning and limits on computational abilities, have provided insights into the theoretical guarantees of various forms and levels of bounded rationality, including computational bounds under which optimal outcome and cooperative equilibrium are achieveable.

However, computational models of bounded rationality still fall short of accurate models of real-world bounded rationality. There are many possibilities for bounding rationality within the classes identified by [6], which include limits on knowledge, memory, information and computational ability. It seems likely also that other forms of bounding or refinements to these will be discovered with more human experimentation. The papers surveyed here are excellent examples of the contribution computation and game theory can make to the classification of bounded rationality in machines and biological agents. Hopefully, these theoretical insights can be combined with psychological and economics experiments to determine which models most closely approximate our world and the games we play.

8. REFERENCES

[1] R. Axelrod. *The Evolution of Cooperation.* Basic Books, New York, 1984.

[2] C. Claus and C. Boutilier. The dynamics of reinforcement learning in cooperative multiagent systems. In *AAAI/IAAI*, pages 746–752, 1998.

[3] L. Fortnow and D. Whang. Optimality and domination in repeated games with bounded players. In *Proceedings of the twenty-sixth annual ACM symposium on Theory of computing*, pages 741–749, May 1994.

[4] Y. Mor and J. S. Rosenschein. Time and the prisoner's dilemma. In *ICMAS*, pages 276–282, 1995.

[5] C. H. Papadimitriou and M. Yannakakis. On complexity as bounded rationality (extended abstract). In *Proceedings of the twenty-sixth annual ACM symposium on Theory of computing*, pages 726–733, May 1994.

[6] A. Rubinstein. *Modeling Bounded Rationality.* MIT Press, Cambridge, Massachusetts, 1998.

[7] T. Sandholm and R. Crites. Multiagent Reinforcement Learning in the Iterated Prisoner's Dilemma. *Biosystems, Special Issue on the Prisoner's Dilemma*, January 1995.

[8] H. A. Simon. *The Sciences of the Artificial.* MIT Press, Cambridge, Massachusetts, 1996.

[9] J. L. Stimpson, M. A. Goodrich, and L. C. Walters. Satisficing and learning cooperation in the prisoner's dilemma. In *Proceedings of the Seventeenth International Joint Conference on Artificial Intelligence*, pages 535–540, 2001.