Learning versus Evolution in Iterated Prisoner's Dilemma

Philip Hingston
Edith Cowan University
2 Bradford Street, Mount Lawley
Western Australia 6050
Email:p.hingston@ecu.edu.au

Graham Kendall
University of Nottingham
School of Computer Science and IT
Jubilee Campus,
Nottingham, NG8 1BB, UK
Email:gxk@cs.nott.ac.uk

***Abstract*- In this paper, we explore interactions in a co-evolving population of model-based adaptive agents and fixed non-adaptive agents playing the Iterated Prisoner's Dilemma (IPD). The IPD is much studied in the game theory, machine learning and evolutionary computation communities as a model of emergent cooperation between self-interested individuals. Each field poses the players' task in its own way, making different assumptions about the degree of rationality of the players and their knowledge of the structure of the game, and whether learning takes place at the group (evolutionary) level or at the individual level. In this paper, we report on a simulation study that attempts to bridge these gaps. In our simulations, we find that a kind of equilibrium emerges, with a smaller number of adaptive agents surviving by exploiting a larger number of non-adaptive ones.**

I. Introduction

Two topics of great interest in evolutionary computation are the interaction of learning and evolution, and evolutionary game theory. Both are important in application areas such as multi-agent systems, economics, politics, and biological modelling [1]. In each of these, the common thread is the study of systems of interacting autonomous individuals in a population, whether the individuals are artificially created intelligent agents, human beings and human institutions, or other biological organisms. How does such a collection of individuals "learn" to come to some accomodation with each other? What will be the equilibrium set of behaviours? Will there be any equilibrium? How can cooperative behaviours evolve?

Previous approaches to answering these questions have concentrated either on adaptive agents, using modified machine learning techniques, or non-adaptive agents evolving under selection pressures determined by game payoffs. In each of the application areas listed above, both the evolutionary and learning aspects are important. This study is a first step in the direction of combining the two. We report on a simulation study that concerns both topics, which explores what happens when learning and evolution interact in an evolutionary game scenario. In the spirit of Axelrod's well-known simulation study, we explore interactions in a co-evolving population of model-based adaptive agents and fixed non-adaptive agents playing Iterated Prisoner's Dilemma (IPD).

In the remainder of the paper, we first introduce the Iterated Prisoner's Dilemma and review previous learning and evolution-based approaches to its study. We then describe the model we have designed for a combined study of learning and evolution. Experimental results are then presented and discussed, and we conclude with some comments on possibilities for future work.

II. Iterated Prisoner's Dilemma

Imagine yourself in a situation where you must choose either to cooperate with another agent or try to exploit him, knowing that he may get a chance to retaliate later. The Iterated Prisoner's Dilemma is an elegant model invented to study cooperation between self-interested individuals in such a situation. It is wonderfully simple to describe and yet capable of demonstrating surprisingly complex and subtle phenomena. Here is an informal description of the "one-shot" Prisoner's Dilemma game [2]:

> (T)wo prisoners suspected of a serious crime are held in different cells, and each is offered the following deal by the district attourney: "If you confess and the other prisoner does not confess, you will be let free; if the other prisoner confesses too, you will receive a moderate sentence. If neither of you confess, you will receive a smaller sentence than if you both confess; if the other confesses but you do not, you will receive the maximum sentence."

While this description casts the problem as an entertaining puzzle, the reasons for studying this game are more serious. It is a model used to study human and natural systems in which cooperation between self-interested individuals is observed or desired. It was introduced by Flood and Dresher in the early 1950's in

studies applying game theory to global nuclear strategies [3]. It has also been applied to problems in psychology, economics, politics, and biology.

A more formal definition of the game follows: in a game of *Prisoner's Dilemma (PD)*, two players simultaneously choose a move, either cooperate (*c*) or defect (*d*). There are thus four possible outcomes for each encounter: both cooperate (*cc*), the first player cooperates, while the second defects (*cd*), vice versa (*dc*), and both players defect (*dd*). We denote this set of outcomes $E = \{cc, cd, dc, dd\}$. Each player receives a payoff after each encounter, as shown in the table:

TABLE 1 - PAYOFFS FOR PRISONER'S DILEMMA

		Second player	
		c	d
First player	c	R,R	S,T
	d	T,S	P,P

In the table, the first player's move determines the row, the second player's move determines the column, and the pair *(X, Y)* in the corresponding cell indicates that the first player's payoff is *X* and the second player's payoff is *Y*.

In defining a Prisoner's Dilemma game, certain conditions have to hold. Firstly, the order of the payoffs is important. The best a player can do is *T* (Temptation to Defect). The worst a player can do is to get the Sucker payoff, *S*. If the two players cooperate then the Reward for that Mutual Cooperation, *R*, should be better than the Punishment for Mutual Defection, *P*. Therefore, the following must hold.

$$T > R > P > S$$

Secondly, players should not be allowed to get out of the dilemma by taking it in turns to exploit each other. That is, taking turns should not be as good an outcome as mutual cooperation. Therefore, the reward for mutual cooperation should be greater than the average of the payoff for the temptation and the sucker:

$$R > (S + T) / 2$$

To be definite, we will choose the commonly used values $T = 5$, $R = 3$, $P = 1$, and $S = 0$.

What a game theorist asks is: as a perfectly rational player, playing another perfectly rational player, what should you do in such a game?

- Suppose you think the other player will cooperate. If you cooperate then you will receive a Reward of 3 for mutual cooperation. If you defect then you will receive a payoff of 5 for the Temptation to defect payoff. Therefore, if you think the other player will cooperate, you should defect, to give you a payoff of 5.
- But what if you think the other player will defect? If you cooperate, then you get the Sucker payoff of zero. If you defect then you would both receive the Punishment for mutual defection of 1 point. Therefore, if you think the other player will defect, you should defect as well.

So, you should defect, no matter what option your opponent chooses (the strategy *d dominates* the strategy *c*). Of course, the same logic holds for your opponent. And, if you both defect you receive a payoff of 1 each, whereas, the better outcome would have been mutual cooperation with a payoff of 3 each. This is the dilemma and the reason for interest in models that promote mutual cooperation.

In other games, there may not be a dominant strategy, and other notions of "solving" the game are used. One such notion is that of a *Nash equilibrium*, in which the two players adopt a pair of strategies such that neither player can get a better payoff by deviating from their strategy. In other words, each strategy is a *best response* to the other. Depending on the game, there may be no Nash equilibrium, a unique one, or many equilibria.

Aside from the strategies *c* and *d*, there is another kind of strategy that can be considered, in which players are allowed to use randomness (e.g. a roll of a die) to decide their moves. In game theory these are called *mixed* strategies or *stochastic* strategies, whereas those without randomness are called *pure* strategies.

In contrast to the rational conclusion of mutual defection, in real-life instances of Prisoner's Dilemma, cooperation is often observed. Why is it so? One suggested explanation is that in real life, the players would have an expectation that they may meet the same opponent in the future, and he might remember a previous defection and take revenge by defecting on us next time we play.

This leads us to consider the *Iterated Prisoner's Dilemma (IPD)*: In a game of repeated or Iterated Prisoner's Dilemma, the players play a sequence of games of PD against each other.

In the iterated game, player strategies are rules that determine (perhaps stochastically) a player's next move in any given game situation (which can include the history of the game to that point). Each player's aim is to maximize his total payoff over the series.

If you know how many times you are to play, then one can argue that the game can be reduced to a one-shot Prisoner's Dilemma. The argument is based on the observation that you, as a rational player will defect on the last iteration - that is the sensible thing to do because you are in effect playing a single iteration. The same logic applies to your opponent. Knowing that your opponent will therefore defect on the last iteration, it is sensible for you to defect on the second to last one, as your action will not affect his next play. Your opponent will make the same

deduction. This logic can be applied all the way back to the first iteration. Thus, both players inevitably lock into a sequence of mutual defections.

One way to avoid this situation is to use a regime in which the players do not know when the game will end. In game theory terms, "Nature" can be introduced as a third player, which decides whether to continue the game. For example, Nature could toss a (possibly biased) coin to decide. If the players know the probability, d , that the game continues, then from their point of view, it is equivalent to an infinite game where the values of payoffs in each successive round are discounted by a factor d .

Depending on the value of d and on various other parameters, different Nash equilibria are possible, where both players play the same strategy. Some well-known examples are:

- **Tit-for-tat**: cooperate on the first move, and play the opponent's previous move after that;
- **Grim**: cooperate on the first move, and keep cooperating unless the opponent defects, in which case, defect forever;
- **Pavlov:** cooperate on the first move, and on subsequent moves, switch strategies if you were punished on the previous move.

Many variations of IPD have been studied, using a variety of approaches. Variations include different classes of strategies, noisy moves, noisy payoffs, alternating non-simultaneous moves, signalling and so on. Approaches used include game theory, evolutionary methods, and machine learning. We briefly review the latter two in the next sections.

A. Axelrod's tournaments

Around 1980, Robert Axelrod staged two round-robin "tournaments" between computer programs designed by participants to play IPD. Many sophisticated programs were submitted. In each case, the winner was Anil Rapaport's submission, a program that simply played tit-for-tat. In 1987, Axelrod carried out computer simulations using a genetic algorithm (nowadays it would be called a co-evolutionary simulation) to evolve populations of strategies playing the IPD against each other [4]. In these simulations, tit-for-tat-like strategies often arose, but other, more complicated strategies sometimes evolved that outperformed tit-for-tat in particular populations. Axelrod used this to illustrate that there is no "best" strategy for playing the IPD in such an evolving population, because success depends on the mix of other strategies present in the population.

Axelrod's simulations illustrate a different approach to studying the IPD – one in which the players are not perfectly rational, and solutions evolve rather than being deduced.

B. Evolutionary Game Theory

Evolutionary game theory is an adaptation of game theory that concerns games played by populations of players, in which expected payoffs are frequency dependent, as in Axelrod's simulations. Maynard Smith provides a nice discussion of the differences between the game theory perspective and the evolutionary game theory one ([5], p195). The players are not assumed to be rational – instead they play whatever strategy their genes tell them to play. It is the genes that now have an "interest" in maximising payoffs (of their phenotypes, relative to the other phenotypes in the population). The questions to be answered concern the dynamics and equilibria of populations of alternative strategies, undergoing evolution (i.e., replication, variation and fitness-based selection). One of the major application areas is the case where players are organisms in a natural population, and strategies are alternative behaviours.

The notion of a Nash equilibrium has a strong parallel in evolutionary game theory – that of an *evolutionarily stable strategy* (ESS) [5]. Roughly speaking, an ESS is a set of proportions of alternative strategies that is stable against small perturbations in those proportions. ESS's have been used to explain why natural populations of organisms are observed to contain specific proportions of different types of individuals (e.g., why gender ratios tend to remain constant in natural populations).

C. Machine Learning

The machine learning community also has an interest in IPD and other games. Their interest is in how adaptive agents can learn to play these games. In contrast to game theory, agents are not assumed to be *perfectly* rational. In contrast to evolutionary game theory, the focus is not on populations, but on individual agents, whose behaviours change as they learn.

Carmel et al ([6],[7],[8]) describe a model-based approach for learning a finite automaton model of a fixed opponent in a two player game, and deriving a best response to the inferred strategy. Reinforcement learning has been adapted for learning to play multi-player games by Littman [9] and Hu et al. [10]. Recently, Tekol et al. used an ant colony algorithm to discover strategies for IPD [11].

III. AGENT DESIGN

In this study, we combine elements of the evolutionary simulation and machine learning approaches to IPD. We want to see what happens when rational thought meets evolutionary forces. We have chosen an IPD tournament as the battlefield, and an evolutionary simulation a la Axelrod as the method of study.

Imagine, then, a population of IPD-playing agents undergoing evolution. During their lives, individual agents

meet each other in encounters where they may choose to cooperate or defect, and they receive payoffs as listed in Table 1. Those that get higher payoffs attain higher reproductive fitness – that is, they contribute more viable offspring in the next generation. The choices that they make are prescribed by genetically determined strategies. Mutations can transform one strategy into another, causing the child to play differently from its parent. This is the scenario that Axelrod and many others have simulated.

Into this mix, imagine injecting a new kind of mutation, that transforms the child into a different kind of player – one who tries to understand the strategies used by his fellows, and to use this understanding to get higher payoffs for himself – an intelligent, adaptive, exploitative player. This is the scenario that we simulated.

In order to make our description complete, we need to list the implementation choices that we made – What do the fixed strategies look like? How are they mutated? How is fitness defined? What selection scheme do we use? How do the adaptive agents learn, and how do they figure out how to exploit their fellows? We describe our choices in the following sections, but we hasten to point out that the particular choices were made for convenience, and we believe that the outcome would be similar if we chose differently. That is, we are not claiming that these are the best or the only choices that could be made.

D. Fixed Strategy Representation

For this study, we have chosen to restrict the fixed strategies under consideration to a class of finite memory stochastic strategies (also sometimes called *behavioural* strategies) that can be described in terms of fixed set of probabilities. This is general enough to represent quite complicated strategies, but it does not include, for example, some strategies defined by finite state automata (see, e.g., [13]). However, most of the well-known strategies for IPD fit into the framework. Formally, an nth-order strategy requires a function

$$C_n : E^n \to [0,1]$$

where $C_n(e_1, e_2, \cdots, e_n)$ gives the probability of cooperation on the next move, given that the previous *n* encounters in the current game were $e_1, e_2, \cdots, e_n$. Similar functions are also required for the first move of the game (as there are no previous encounters), the second move, and so on up to the *(n-1)th* move. We denote these functions $C_1, C_2, \cdots, C_{n-1}$. A pure strategy is one where the value of each function is always either 0 or 1, otherwise the strategy is stochastic.

For example, the strategy of always cooperating is a pure 0-order strategy having $C_0 = 1.0$. A completely random strategy is a mixed 0-order strategy in which $C_0 = 0.5$. 0-order strategies are those that ignore the other player's moves. The optimal counter-strategy for any 0-order strategy is to always defect (another 0-order strategy).

Tit-for-tat is a pure first-order strategy having $C_0 = 1.0$ and

$$C_1(cc) = 1.0$$
$$C_1(cd) = 0.0$$
$$C_1(dc) = 1.0$$
$$C_1(dd) = 0.0$$

We adopt the convention that each player considers himself to be the "first" player, so that his own move is listed first in each encounter (although actually the players move simultaneously). In this study, in order to keep things simple, we only use first order strategies.

These strategies make up part of the genome of our populations of IPD-playing agents. In the first generation, stochastic strategy probabilities are assigned randomly by sampling from a uniform distribution with range [0,1), while the pure strategy probabilities are each equally likely to be 0 or 1.

Mutation of a strategy is carried out as follows: For each probability value, *p*, in the genome, first determine whether mutation is to occur (using the specified mutation rate). If so, either

1. "flip" the value of *p* to *1.0 – p*, or
2. add a small value, sampled uniformly from the range [-0.05,0.05), with the new value of *p* adjusted if necessary to stay in the range [0,1].

The choice between the two mutation types is done with equal probability for a stochastic strategy, while the first type is always chosen for pure strategies.

E. Adaptive Agents

We want our adaptive players to compete with non-adaptive ones, so they must be quick learners. Although reinforcement learning is a very general learning method, we judged it to be too slow for our purposes. Therefore, we devised our own adaptive players specifically for playing IPD against first-order strategies. We use a method analogous to that of Carmel et al. ([6],[7],[8]). Our adaptive players maintain models of each opponent's play, and use these models to determine a counter-strategy for each opponent.

An *opponent modeling agent* (*OMA*) of order *n* maintains a summary of the moves made by each opponent depending on the (up to *n*) previous encounters with this opponent. We call this a *model* of the opponent. A model is a set of functions $M_0, M_1, \cdots, M_n$ analogous to the functions in a strategy, except that the value of each *M* is a pair *(X, Y)*, where *X* reflects how often this opponent has cooperated given the particular sequence of previous encounters in the past, and *Y* reflects how often they have defected. These models can then be used to compute an

estimate of the opponent's playing strategy, so that a counter-strategy can be devised. One way for an OMA to do this is described below.

1) Updating Rule

Any player that learns to play better over time must do so by adjusting internal parameters of some kind. The method a player uses to do this is called his *updating rule*. Here we describe the updating rule for OMA's. Initially, that is, before our OMA has played any games, it has no opponent models at all, but has a randomly generated "default" model. For example, the default model could be one having $M_0 = (2,0)$ and

$$M_1(cc) = (2,0)$$
$$M_1(cd) = (0,2)$$
$$M_1(dc) = (2,0)$$
$$M_1(dd) = (0,2)$$

This first-order model suggests an opponent who plays tit-for-tat. Each time the OMA meets a new opponent, he creates a clone of the default model as an initial model of the opponent. Thus, the OMA is starting from scratch each time he meets a new opponent.

Each time a move is played, the OMA updates the relevant function value in the opponent model. He does this by multiplying each element of the pair by a forgetting factor, $g < 1$, and incrementing either the cooperation value, *X*, or the defection value, *Y*. An increment value of $2.0 \times (1.0 - g)$ is chosen to keep the total of the pair values constant (in this case equal to 2.0). For example, if the previous outcome against this opponent was *cc*, $M_1(cc)$ is (1.5, 0.5), and both players cooperate this time, then $M_1(cc)$ becomes $(1.5g + 2.0 \times (1.0 - g), 0.5g)$.

One problem with this scheme is that some game positions may seldom or, worse, never be reached, so the part of the model dealing with these positions would remain inaccurate. We describe below, in 3), how this problem can be handled.

This explains how the OMA updates his opponent models. Now we come to the question of how these models are used by the OMA to select its moves.

2) Choice Rule

In the next few sections, we describe the method an OMA uses to select moves, sometimes called his *choice rule*. Opponent models provide an OMA with a way to anticipate how likely it is for an opponent to play a particular move in a given situation. For example, in the situation described above, we could guess that, as $M_1(cc)$ is (1.5, 0.5), the opponent will cooperate with probability 1.5/2.0, or 0.75. In short, the model can be converted to a strategy for the opponent, by using the ratio *X/(X+Y)* to estimate the probability of cooperation in each situation.

If the opponent's strategy is indeed a first-order strategy, and if its parameters are known, then we can derive a best response analytically, as shown below. Let us put ourselves in the position of the OMA. Suppose that we know that our opponent is playing strategy $\{C_0, C_1\}$. The initial part of the game tree is shown in Figure 1.

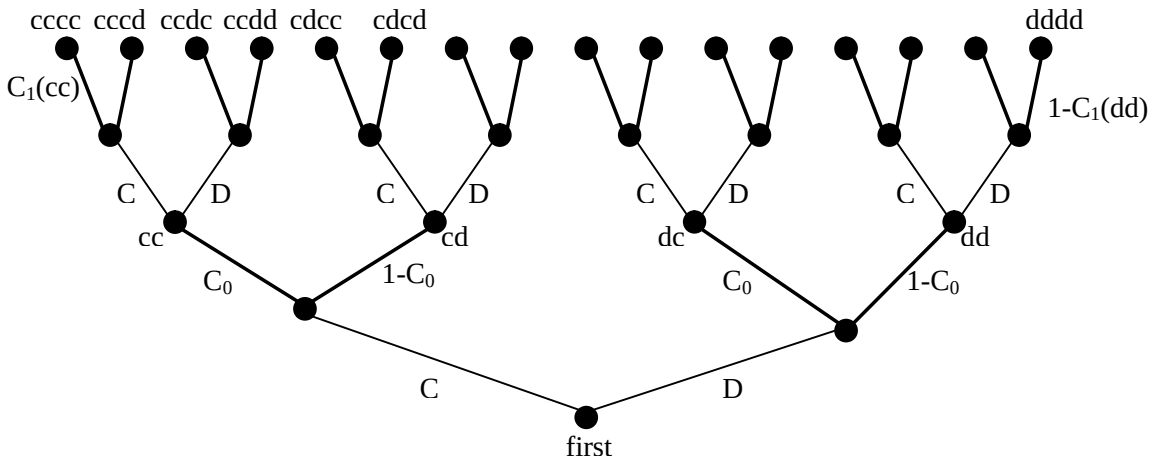

Figure 1 -Game tree for the first few moves in a game of IPD against a stochastic strategy

Our moves are labeled with *C* or *D*, while our opponent's have thicker lines and are marked with the probability of our opponent cooperating in that game position. Because our strategies are all of order 1, as far as the opponent's choice of moves is concerned, the game positions labeled *cccc*, *dccc*, etc., are the same as those labeled *cc* – i.e. the opponent will cooperate with probability $C_1(cc)$ Also, note that if, say, *C* is our next move in a best response at position *cc*, it must still be a best response at positions *cccc*, etc., (if there was a better

choice, we could use it at *cc* for the same effect). Hence, from both our points of view, the game positions ending in *cc* are equivalent, as are those ending in *cd*, those ending in *dc* and those ending in *dd*. Therefore, we need only decide on best responses for the first move, and for positions ending in *cc*, *cd*, *dc*, and *dd* – 5 positions in all. Given a choice of move for each of these 5 positions, the value of the game can be calculated: Let $V(cc)$ be the value of the game at position *cc*, by which we mean the expected discounted future payoff starting from this position. Define the value of the game for the other positions similarly. If we choose to cooperate at *cc*, then:

$$V(cc) = C_1(cc) \times (R + d \times V(cc)) + (1 - C_1(cc)) \times (S + d \times V(cd))$$

while if we choose to defect, then:

$$V(cc) = C_1(cc) \times (T + d \times V(dc)) + (1 - C_1(cc)) \times (P + d \times V(dd)).$$

Similar equations hold for the other positions, giving a system of equations that can be solved for the values $V(\_)$. Finally, the value of the game at the start of the game is either

$$V = C_0 \times (R + d \times V(cc)) + (1 - C_0) \times (S + d \times V(cd)), \text{ or}$$

$$V = C_0 \times (T + d \times V(dc)) + (1 - C_0) \times (P + d \times V(dd))$$

, depending on whether we choose to cooperate or not. The best response is that set of 5 choices which maximizes this value. There are only $2^5 = 32$ strategies to check.

This is the method for determining a best response that OMA's used in the experiments reported below. Note that using this "optimal" counter-strategy does not make the OMA's unbeatable. One reason is that the strategy is only optimal relative to the opponent's probabilities, which the OMA can only approximate. It takes a few moves to "learn" how an opponent is playing. Depending on the opponent, a mistake in the first few moves may lock in a low reward for both players (if the opponent is grim, for example). A second reason is that deliberate errors are introduced into all agent's play, as described in 3) below, so an OMA can never achieve perfect play.

The method described above can be extended to calculate a best response against any finite-order stochastic strategy. As an aside, notice that this best response is also a strategy of the same order (a pure one). Since we can find best responses for any stochastic strategy, including the pure ones, it is straightforward to check which pure strategies form Nash equilibria. Using the values computed for the subgames, it is also easy to find the subgame perfect equilibria. Which strategies turn out to be equilibria in particular cases depends on the values of *d*, *T*, *R*, *S* and *P*.

3) Adding exploration – the trembling hand

As mentioned earlier, in 1), a problem with the opponent model updating rule is that the part of the model dealing with some game positions may not be updated, because these game positions are not reached. For example, if both players play tit-for-tat, neither player ever finds out what his opponent would do if he defected. To ensure that all parts of the opponent model are updated at least periodically, we introduce occasional (in these experiments, on 1% of moves) "noisy" moves, where we deliberately play the opposite of the calculated best response. This is similar to the choice rule used in stochastic fictitious play, where randomness is introduced as a response to noisy payoffs [12], and to exploration strategies sometimes used in reinforcement learning [14]. It has also been proposed as a natural way to model "real" players, who could be expected to make errors from time to time. This device has been called the "trembling hand."

Partly to allow OMA's to explore, partly for uniformity of treatment, and partly to make the agents more realistic, all the agents, adaptive and non-adaptive, used in these experiments were "equipped" with a trembling hand. This requires a change to the equations given in 2). The value of the game at position *cc*, for example, now becomes either

$$V(cc) = (1-e) \times (C_1(cc) \times (R + d \times V(cc)) + (1 - C_1(cc)) \times (S + d \times V(cd))) + e \times (C_1(cc) \times (T + d \times V(dc)) + (1 - C_1(cc)) \times (P + d \times V(dd)))$$

or

$$V(cc) = e \times (C_1(cc) \times (R + d \times V(cc)) + (1 - C_1(cc)) \times (S + d \times V(cd))) + (1-e) \times (C_1(cc) \times (T + d \times V(dc)) + (1 - C_1(cc)) \times (P + d \times V(dd)))$$

, where ε is the error rate, depending on whether the player chooses to try to cooperate or not. The other equations need similar changes, but the whole system is still a linear system that can be solved as before.

IV. Evolving Players

In the experiments described below, populations of agents were evolved using a genetic algorithm-like simulation of evolution, as follows:

1. An initial population is created.
2. A round-robin IPD tournament is held between the members of the population. Every player plays every other player in a game of IPD in which the game continues to another round with probability d. The fitness of each individual is assigned to be that player's average payoff per move in the tournament.
3. Fitness-proportionate selection is used to select parents for the next generation (using stochastic universal selection).
4. Each parent, when selected, produces one child, by a process of copying the genome of the parent (with a low mutation rate – the probability of mutation as each gene is copied), and the development of a new individual from this genome. The children become the next generation.
5. Repeat steps 2-4 until finished.

The genome of each IPD-playing agent consists of a set of probabilities for a first-order strategy, plus an additional

"smart" bit. If the smart bit is on, the agent plays as an OMA. Otherwise, he plays the fixed strategy prescribed by his genes. As well as mutation of strategies, as described in D, the smart bit may independently mutate between on and off.

In all these experiments, we used the payoffs given in Section II, and a discount rate of 0.96, giving an average game length of 25 moves. A mutation rate of 0.01 was used throughout. All the strategies used were of order 1. The first experiments establishes a baseline for non-adaptive players evolving without adaptive players present, while the second investigates interactions between coevolving adaptive and non-adaptive players. Note that a simulation with adaptive players alone would not make sense, as there is nothing to be passed on from parents to children.

Because of space limitations, we report only on experiments using pure strategies for non-adaptive players. We also carried out experiments using stochastic strategies, with similar results.

F. Experiment 1

In this experiment, we evolved populations of 100 non-adaptive pure strategy players, with the smart bit permanently turned off. We ran each simulation for 1000 generations. In each generation, we recorded the mean fitness value and percentage of cooperation in each generation as well as the percentage of grim and tit-for-tat strategies present in the population.

TABLE 2 – SUMMARY STATISTICS FOR EVOLUTION OF PURE STRATEGIES, N = 20, MEAN ± STD.DEV.

mean fitness	mean coop%	grim%	tft%
2.783±0.013	86.7±0.8	26.4±1.5	19.7±1.8

The populations evolved in a few generations to a mixture of generally cooperative players, cooperating around 87% of the time. As can be seen in Table 2, the mean reward was close to the mean of 2.783 in all the runs. The average percentages of grim and tit-for-tat strategies were around 26% and 20% respectively. Figure 2 shows a typical run, with defection initially popular, and cooperation taking over after about 20 generations. Although the mean reward and degree of cooperation of the population have stabilised, the composition of the population is constantly fluctuating, with grim and tit-for-tat always present in large numbers, appearing to be loosely tied together in a cycle of period about 100 generations. Figure 3 shows the percentages for the same typical run.

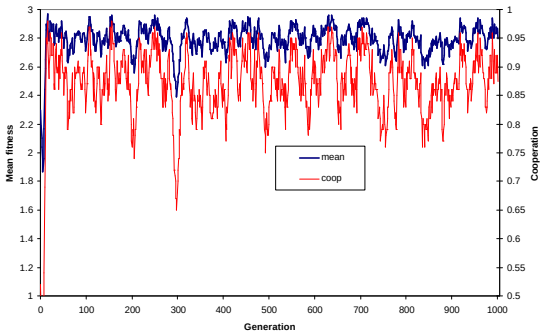

Figure 2 - A typical run with pure strategies

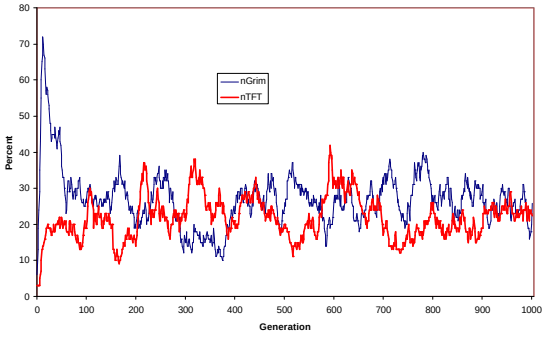

Figure 3 - Percentage of grim and tit-for-tat strategies for the run in Figure 2

G. Experiment 2

In this experiment, we used the same setup as in Experiment 1, except that we allowed the smart bit to mutate. In the initial population, all the smart bits were off. Figure 4 shows the mean fitness and level of cooperation in a typical run. The picture is similar to that of Experiment 1, with a slightly lower degree of cooperation at around 81%, and slightly lower mean rewards around 2.67. Figure 5 shows the percentage of tit-for-tat and grim strategies and also the percentage of OMA's for the same run. As Table 4 shows, a significant number of OMA's, a mean of around 13.5% of the population, is able to survive. Compared to Experiment 1, some of the grim strategies have been displaced, but the percentage of tit-for-tat strategies has actually increased. We conjecture that this increase is at the expense of more exploitable strategies, which are under pressure from the OMA's.

We must ask: how do the OMA's continue to survive, despite having a mean fitness of only 2.51 compared to the average of 2.67? The answer is that their fitness is high enough that the constant mutation prevents them from being driven to extinction. We can illustrate this using additional data collected in Experiment 2. At one point during one of the runs, we recorded the mean payoff achieved by OMA's when playing against non-adaptive

players, as well as those for OMA's against other OMA's, for non-adaptive players against OMA's, and for non-adaptive players against other non-adaptive players. Table 3 gives these payoffs. The payoffs constantly fluctuate during a run as the percentages of different kinds of non-adaptive players fluctuates. At this point, the percentage of OMA's was 15%, the mean OMA fitness was 2.52, and the overall mean fitness was 2.71. In game theory terms, the OMA "strategy" is dominated by the non-adaptive "strategy", so there could be no equilibrium - no ESS with a non-zero proportion of OMA's.

TABLE 3 - PAYOFF MATRIX FOR DIFFERENT STRATEGY TYPES

	OMA	Non-adaptive
OMA	1.69	2.64
Non-adaptive	1.79	2.88

Without mutation, the expected percentage of OMA's in the next generation would be $15 \times 2.52 / 2.71 = 13.95\%$ and the expected percentage of non-adaptive players would therefore be 86.05%. With a mutation rate of 1%, we expect on average 0.1395% of the next generation will be OMA's mutated into non-adaptive players, and 0.8605% will be non-adaptive players mutated into OMA's. Therefore, the expected percentage of OMA's in the next generation is actually $13.95 - 0.1395 + 0.8605 \approx 14.67\%$ almost back up to 15%. This unequal effect of mutation would be more pronounced if the percentage of OMA's was to fall lower. The result is that an equilibrium - a kind of modified ESS - occurs.

Thus we see that OMA's, which come into the IPD-world knowing nothing and must quickly learn everything they need to know about their fellows, can survive amongst a population of players genetically bred to play IPD instinctively. We conjecture that OMA's could do better at some other games, because in IPD, players like grim severely punish mistakes like those that a learner makes in the process of learning.

TABLE 4 – SUMMARY STATISTICS FOR COEVOLUTION OF PURE STRATEGIES WITH OMA'S, N = 20, MEAN ± STD.DEV.

mean fitness	mean OMA fitness	mean coop%	OMA %	grim%	tft%
2.67± 0.01	2.51± 0.01	80.6± 0.6	13.5± 0.7	21.7± 2.0	24.1± 2.0

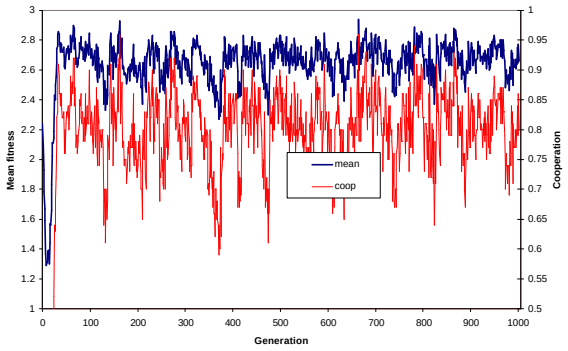

Figure 4 - A typical run with pure strategies and OMA's

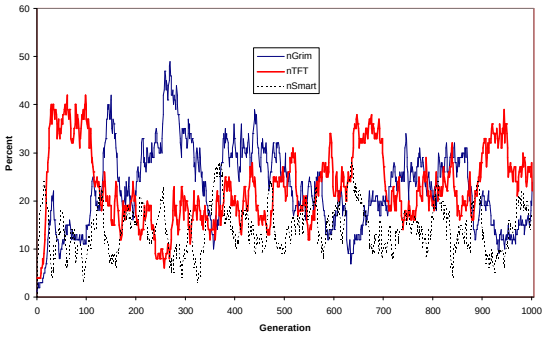

Figure 5 - Percentage of OMA's, grim and tit-for-tat strategies for the run in Figure 4

V. Conclusion

We have presented a simulation study combining evolution and learning in a competitive game setting. Previous work in this area has focussed on either evolution or learning as the mechanism by which agents determine their strategies. This paper is a first step towards studying interaction between these two mechanisms.

We chose Iterated Prisoners' Dilemma as our example game. We chose a class of non-adaptive players that use strategies that are easy to understand and analyse, but able to represent many of the well known IPD strategies. These non-adaptive players "learn" to play IPD with each other by a purely evolutionary process – the players learn nothing during their lifetimes. For the adaptive players, we chose a custom built learner that inherits nothing from its parents, and must learn everything in its own lifetime. We found that these adaptive players are able to survive in the evolving population in significant proportions, though IPD is a tough environment for a learner, and mutation is needed to prevent their extinction. We also found that the presence of these adaptive players alters the composition of the non-adaptive portion of the population.

In this initial study, we chose simple, but non-trivial, strategies, and kept a clear distinction between learning and evolution. In future work, richer, more complex

interactions may be studied by relaxing these restrictions. For example, the non-adaptive players could use strategies defined by finite state automata. Our adaptive players start from scratch each time they meet a new opponent. It would perhaps be more realistic if knowledge gained by playing previous opponents could be kept and used. Taking this a step further, an adaptive player's default opponent model could be inherited from his parent. It would be interesting to see whether this would produce a kind of Baldwinian learning. Lastly, it would be interesting to know whether human or animal society provides examples that mirror what we saw in these simulations – can we find situations in which a minority of devious, calculating individuals is able to sustain a parasitic existence by exploiting the naïve good nature of the majority? We leave this question for those with expertise in the social sciences to ponder.

REFERENCES

[1] Axelrod, R. and D'Ambrosio, L. "Annotated Bibliography on the Evolution of Cooperation", http://pscs.physics.lsa.umich.edu/RESEARCH/Evol_of_Coop_Bibliography.html, [Accessed 12 Feb 2004], 1994.

[2] Tsebelis, G. "Nested Games: rational choice in comparative politics", University of California Press, 1990.

[3] Flood, M. "Some experimental games", Research Memorandum RM-789, RAND Corporation, Santa Monica, CA, 1952.

[4] Axelrod, R. "The evolution of strategies in the iterated prisoner's dilemma", in *Genetic Algorithms and Simulated Annealing* (L. Davis, Ed.), Pitman, 1987.

[5] Maynard Smith, J. "Evolution and the Theory of Games", Cambridge U.P., 1982.

[6] Carmel, D. and Markovitch, S. "Learning Models of Intelligent Agents", Proceedings of the Thirteenth National Conference on Artificial Intelligence and the Eighth Innovative Applications of Artificial Intelligence Conference, Vol. 2, pp 62-67, AAAI Press, Menlo Park, California, 1996.

[7] Carmel, D. and Markovitch, S. "Opponent Modeling in Multi-Agent Systems", Adaptation and Learning in Multi-Agent Systems, pp 40-52, Springer-Verlag: Heidelberg, Germany, 1996.

[8] Carmel, D. and Markovitch, S. "Exploration Strategies for Model-based Learning in Multi-agent Systems", Autonomous Agents and Multi-Agent Systems, 2, 2, pp 141-172, Kluwer, 1999.

[9] Littman, M. "Markov games as a framework for multi-agent reinforcement learning", Proceedings of the 11th International Conference on Machine Learning (ML-94), pp 157-163, Morgan Kaufmann, New Brunswick, NJ, 1994.

[10] Hu, J. and Wellman, M. "Multiagent reinforcement learning: theoretical framework and an algorithm", Proc. 15th International Conf. on Machine Learning, pp 242-250, Morgan Kaufmann, San Francisco, CA, 1998.

[11] Tekol, Y. and Acan, A. "Ants Can Play Prisoner's Dilemma", Proceedings of the 2003 Congress on Evolutionary Computation, pp 1151-1157, Canberra, Australia, 2003.

[12] Fudenberg, D. and Levine, D. "The Theory of Learning in Games". Cambridge, MA: MIT Press, 1998.

[13] Fogel, D.B. "Evolving Behaviours in the Iterated Prisoner's Dilemma", Evolutionary Computation, Vol. 1:1, pp 77-97, 1993.

[14] Jehiel, P. and Samet, D. "Learning to Play Games in Extensive Form by Valuation", NAJ Economics, Peer Reviews of Economics Publications, 3, 2001.