Evolving Strategies for the Prisoner's Dilemma

JENNIFER GOLBECK
Computer Science Department
University of Maryland, College Park
College Park, MD
USA
golbeck@cs.umd.edu http://www.cs.umd.edu/~golbeck

Abstract: - This paper investigates the use of Genetic Algorithms (GA's) to evolve optimal strategies to the Prisoner's Dilemma, a classic game theory problem. The problem was heavily studied in the 1980's, but using more advanced computing techniques, this research extends the existing body of research. My hypothesis is that populations evolve to exhibit two traits: the ability to defend against defectors, and the ability to cooperate with other cooperators. Two successful, well studied strategies which embody these traits, Pavlov and Tit for Tat, are used as controls. Populations that do not possess these traits *a priori* are evolved and then compared to the performance of the controls. The results presented here strongly indicate that the hypothesized traits are present in all evolved populations.

Key-Words: - genetic algorithms, Prisoner's dilemma, game theory, evolutionary computation

1 Introduction

The Prisoner's Dilemma is a traditional and elegant model for studying decision making and self-interest. It has been studied extensively to model behavior from petty theft to nuclear war. Much of the current body of research has focused on which strategy in the game is "best." Different strategies have been analyzed, competed in tournaments, and even subjected to Darwinian selection. In the latter case, strategies were usually played against populations which were considered representative of the body of possible strategies.

Two strategies in particular have been found to be evolutionarily successful: Tit for Tat, which has long been touted as one of the best strategies in the game[3], and Pavlov which has been shown to be an excellent strategy in more recent literature[7,8,9]. In looking at this problem, we were not concerned with which particular strategy was superior. After all, to use the game as a model of behavior in complex systems, it is unlikely that an actor's behavior can be neatly summarized as perfectly fitting the Tit for Tat model, the Pavlov model, or any other deterministic model. Even those models that introduce randomness and probability are not capturing the essence of the question. Indeed, the behavior of decision makers may not be accurately modeled by a deterministic strategy, but neither can their decisions accurately be explained as "random" or "probabilistic." Instead of searching for a single strategy to accurately capture complex and situationally dependent behavior, we are interested in the *traits* of those strategies which have proven to be highly effective in the past. Isolating those traits facilitates the development of a general set of strategies which all should perform well.

My hypothesis is that superior strategies have two traits in common. First, they can defend against defectors, and second, they can exploit the advantages of mutual cooperation with other superior players. Both the Tit for Tat and Pavlov strategies mentioned before have these traits. In this paper, a genetic algorithm evolves strategies from five different initial populations. By testing the behavior of those evolved populations in the presence of defectors and cooperators it can be shown in this study that they perform identically to the Tit for Tat and Pavlov strategies. It follows that superior strategies, as discovered through genetic algorithms, share the two behavioral traits of the hypothesis.

2 The Prisoner's Dilemma

The Prisoner's Dilemma is a decision model. Anecdotally described, two people, indicated here as player 1 and player 2, are arrested for espionage and placed in separate interrogation rooms. Each player has the option to cooperate with his peer, or to defect against him. If both players cooperate (C), they are rewarded with a shorter sentence of length R, while if both defect (D), they receive a penalty P. If one player defects and the other cooperates, the defector receives the best payoff (or temptation payoff) T for exploiting his opponent. The opponent is punished with the sucker payoff S, the worst possible outcome [11]. Here we declare [R,S,T,P] = [3,0,5,1]. The payoff matrix for this game, indicating the points received by each player in each situation, is shown in figure 1 (scores listed as [player 1, player 2]).

Player 2	Player 1: Cooperate	Player 1: Defect
Cooperate	3,3	5,0
Defect	0,5	1,1

Fig 1: Payoff Matrix for the Prisoner's Dilemma

Nash Equilibrium describes what move each player will make to maximize her score based on correct assumptions about the other player's actions. In the prisoners' dilemma, regardless of what one player does, the other player will be better off defecting. If player 1 cooperates, player 2 will get 5 points for defecting instead of 3 for cooperating. If player 2 defects, player 1 is still better off to defect as well, giving him a score of 1 instead of 0. Each player will use this logic, resulting in a Nash Equilibrium of Defect, Defect.

If both players were to change their moves to Cooperate, they each would triple their score. Another approach to analyzing game theoretic problems is the concept of Pareto Efficiency. A solution is Pareto Efficient if there is no other solution that makes one player better off without making the other player worse off. In the case of the Prisoner's Dilemma, mutual cooperation is the Pareto Efficient solution.

If players met and played several games in a row (the iterated Prisoner's Dilemma), the Nash Equilibrium of mutual defection becomes increasingly inefficient. If individuals are able to remember some set of prior games with their opponent, then they can each develop a more complicated strategy to maximize their score.

Some strategies have no advantages over the single game. A player who cooperates regardless of previous behavior (AllC) or who always defects (AllD) will score no better than their memory-less counterpart. Much research suggests, however, that the Tit For Tat (TFT) strategy is very successful. This strategy simply states that a player should repeat the opponent's move of the previous round. In earlier research, TFT has been shown to outperform most other strategies [2]. Another strategy shown to perform well against a wide range of opponents is the Pavlov strategy. This strategy, also known as Win Stay Lose Switch or Simpleton, cooperates when rewarded for cooperating or punished for defecting, and defects otherwise. In a memory-one system, where players can remember their own move and their opponent's move from the previous game only, Pavlov players would cooperate on a history of mutual cooperation and mutual defection. Since they are rewarded with a score of 3 for mutual cooperation, Pavlov players continue to cooperate. With a history of DD, players will also choose too cooperate in the next round since they had been punished with a low score of 1. On the other hand, Pavlov players would defect on a history of DC since they had just been rewarded with the best score of 5 points for defection, and would also defect with a history of CD since they had just been severely punished with a score of 0 for cooperating. This strategy was shown to perform as well or better than any other strategies in the memory-one iterated Prisoner's Dilemma [9].

The TFT and Pavlov strategies have been widely studied. What features do they have in common that makes them consistently competitive? Both are able to use mutual cooperation, the Pareto Efficient solution, to maximize their score, and both are able to defend themselves from receiving Sucker payoffs from a defector. This

research searches for these two traits in evolved populations.

3 The Genetic Algorithm

Genetic algorithms lend themselves well studying strategies in the prisoner's dilemma. Each player is represented by its strategy. In the memory-three game used in this study, each player's strategy must address sixty-four possible histories. We use the set of moves to create a 64 bit string which represents each player in the algorithm. Table 1 shows string position, the history it represents, and a sample strategy.

After calculating fitness, which is described in the next section, this study implements roulette wheel selection, also called stochastic sampling with replacement [4]. In this stochastic algorithm, the fitness of each individual is normalized. Based on their fitness, individuals are mapped to contiguous segments of a line, such that each individual's segment is equal in size to its fitness. A random number is generated and the individual whose segment spans the random number is selected. The process is repeated until the correct number of individuals is obtained.

Table 2 shows a sample population with calculated and normalized fitness. Figure 2 shows the line with selection probability for ten individuals using a roulette wheel implementation. Individual 1 has a normalized fitness of approximately 0.20 which gives it a 1 in 5 chance of being selected. Individual 10 has the lowest fitness, with a normalized value of 0.02. If an individual had a fitness of zero, it would have no chance of being selected to propagate into the new population. Random points are selected on this line to select individuals to reproduce. Children's chromosomes (strategies) are produced by single point crossover at a random point in the parent's chromosome.

The mutation rate was 0.001 which produced approximately one mutation in the population per generation, and the recombination rate was set at 0.8.

4 The Simulation

Simulations in this study utilized a genetic algorithm to evolve strategies for the Prisoner's Dilemma. Each simulation began with an initial population of twenty players represented by their strategies.

Several terms are used in this section. A *game* refers to one "turn" in the Prisoner's Dilemma. Both players make simultaneous moves, and each are awarded points based on the outcome. A *round* is a set of games between two players. Rounds in this study are 64 games long. A *cycle* is completed when every player has played one round against every other player.

To determine fitness, each player was paired with every other for one round of 64 games. Players did not compete against themselves. Since there are sixty-four possible histories, this number of games ensures that each reachable history is visited at least once. After each *game*, the players' scores are tallied and their histories are updated. Players maintain a performance score which is the sum of the points that they receive in every game against every player. The maximum possible performance score is 6080: if a player defected in every game and his opponents all cooperated in every game he would receive 5 points X 64 games X 19 opponents. For a player who is able to mutually cooperate in every game, the performance score would be 3,648 (3 points X 64 games X 19 opponents).

After a full cycle of play, players are ranked according to their performance score, and selected to reproduce. Recombination occurs, children replace parents, and the cycle repeats.

At the end of each generation, the total score for the population is tallied. This value is the sum of the scores of all members in the population.

String Position	Represented History	Move
0	CCCCCC	C
1	CCCCCD	D
2	CCCCDC	D
3	CCCCDD	D
4	CCCDCC	C
5	CCCDCD	C
6	CCCDDC	C
7	CCCDDD	D
8	CCDCCC	C
9	CCDCCD	D
10	CCDCDC	D
11	CCDCDD	D
12	CCDDCC	D
13	CCDDCD	D
14	CCDDDC	C
15	CCDDDD	C
16	CDCCCC	C
17	CDCCCD	C
18	CDCCDC	D
19	CDCCDD	C
20	CDCDCC	C
21	CDCDCD	D
22	CDCDDC	D
23	CDCDDD	C
24	CDDCCC	C
25	CDDCCD	C
26	CDDCDC	D
27	CDDCDD	C
28	CDDDCC	D
29	CDDDCD	C
30	CDDDDC	C
31	CDDDDD	D

String Position	Represented History	Move
32	DCCCCC	D
33	DCCCCD	C
34	DCCCDC	D
35	DCCCDD	D
36	DCCDCC	C
37	DCCDCD	C
38	DCCDDC	D
39	DCCDDD	D
40	DCDCCC	D
41	DCDCCD	C
42	DCDCDC	C
43	DCDCDD	C
44	DCDDCC	C
45	DCDDCD	D
46	DCDDDC	D
47	DCDDDD	C
48	DDCCCC	C
49	DDCCCD	D
50	DDCCDC	C
51	DDCCDD	C
52	DDCDCC	C
53	DDCDCD	C
54	DDCDDC	D
55	DDCDDD	D
56	DDDCCC	C
57	DDDCCD	C
58	DDDCDC	D
59	DDDCDD	D
60	DDDDCC	C
61	DDDDCD	C
62	DDDDDC	D
63	DDDDDD	C

Table 1: The table above shows the strategy for a randomly generated player. The corresponding 64 bit string used to represent this strategy in the algorithm is
CDDDCCCD CDDDDDCC CCDCCDDC CCDCDCCD DCDDCCDD DCCCCDDC CDCCCCDD CCDDCCDC

Individual	1	2	3	4	5	6	7	8	9	10
Fitness	27	22	18	15	13	12	9	8	4	3
Normalized Fitness	0.20	0.17	0.14	0.11	0.10	0.10	0.07	0.06	0.03	0.02

Table 2: Sample Population showing fitness and normalized fitness for each individual

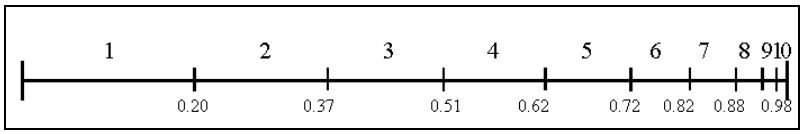

Figure 2: Continuous line divided for the 10 individuals of the sample population. Random points generated along this line determine which individuals will be selected for reproduction.

While the maximum score for an individual is 6080, the maximum score for a population of 20 cannot be 20 times that. For one individual to score the maximum, all others must score very low. The highest cumulative score achievable in an individual game is 6, when both players receive a score of 3 for mutual cooperation. Mutual defection would have a total game score of 2 (1 point each), and mixed plays, with one cooperator and one defector, have a game total of 5 (5 for the defector plus 0 for the cooperator). Thus, the highest score that a population can achieve is 72,960 (3 points X 64 games X 19 opponents gives 3,648 per player X 20 players total). In the end, the fitness of a population is measured by what percentage of the highest possible score is achieved. A population with a total score of 63,480, for example, would have a population fitness of 50% (63,480 / 72,960).

In these experiments, each evolutionary simulation ran for 200,000 generations. This gave plenty of time for genomic stabilization without requiring too much computing time. Results for much longer selected simulations (up to 4,000,000 generations) were not significantly different.

5 Methodology

To test whether or not a population had each of the two traits described in the hypothesis, players' behavior in these experiments is compared to the behavior of both the Tit for Tat and Pavlov populations. A straight comparison, however, would not yield productive results; through the evolutionary process, noise is introduced into any population, even those well evolved.

Consider a population that has evolved Tit for Tat like behavior. That population is likely using only one quarter to one third of its encoded genes because many of the possible histories are not achievable by a Tit for Tat player (i.e. DCDCDC where a cooperating player never defends against the defecting opponent). This means that an individual might look very little like an unevolved Tit for Tat player. This noise in the gene can show up in simulations because each player starts with a randomly generated initial history. Thus, it is possible for a Tit for Tat player to start with and visit several strategies which would not otherwise be reachable in play alone. If a noisy Tit for Tat player winds up with an initial history that visits a "corrupted" gene (which is not unlikely), play can take a very non-Tit for Tat like path by playing though these genes which are less affected by selection. Thus noise is added into the Tit for Tat and Pavlov populations when using them for comparison to discount the effect of noise in the evolved populations and focus in on the actual evolved behavior.

Five distinct populations were used to compare behavior before and after evolution. Tit For Tat and Pavlov, as discussed previously, were the two control populations for this experiment. Both have the inherent ability to exploit mutual

cooperation and defend against defectors. Three other populations were respectively comprised of AllC players, of AllD players, and of independently randomly initialized players.

To measure the performance of populations, the average fitness over the last 10,000 generations of each simulation was studied. Starting with the five initial populations, each was evolved for 200,000 generations. This evolution was simulated several hundred times for each initial population.

Significance was calculated by the standard 2 tailed t-test for data sets with unequal variance. Each population was compared to the Tit for Tat and Pavlov controls.

6 Results

Recall the hypothesis: Evolved populations of players develop 1) the ability to defend against defectors, and 2) the ability to take advantage of mutual cooperation. Below, the results are outlined which support this hypothesis. Statistical data generated for these results is contained in Table 3.

After a period of evolution and play as described earlier, the average performances of the five populations were statistically equal. The natural question to ask is whether or not this equality came about as a result of "random drift" of the populations, or because they were evolutionarily driven in that direction. Random drift occurs when strategies are recombined and mutated without selection. Since there is no pressure to perform, there is no meaning associated with the genomic makeup of a population after evolution. Each specific gene has occurred simply by chance mutation or recombination, and the performance of such a population is generally low. By turning off the selection mechanism in the genetic algorithm, results for a random drift population were generated. The evolved populations all performed well above the level of the random drift population, indicating that they exhibit evolutionarily preferred traits (shown in Table 3). The next step is to test whether or not the improved performance is due to the presence of the traits described in the hypothesis, namely, if they are defensive and cooperative abilities.

The first experiment looks for the ability to defend against defectors. In the experiment, the five unevolved, initial populations were mixed with a small set of AllD players. Fitness of those populations was calculated over the first 10,000 generations immediately following inoculation. Both Pavlov and TFT performed well, with scores around 80%. Neither Random, AllC, nor AllD came near this level. By the standard t-test, all were significantly lower than both Pavlov and TFT with $p = .01$.

The same experiment was performed with the five populations after evolution. After 200,000 generations, the populations were mixed with a small group of AllD players. Fitness was calculated over the next 10,000 generations. Looking at the average fitness of all five populations, it was found that there was no statistical difference in performance among them with $p=.01$. Additionally, comparing these results to the performance of unevolved Tit for Tat and Pavlov players, there was no statistical difference. Additionally, there was no statistical difference between performance of the inoculated populations, and the uninoculated evolved populations, indicating that defectors had no effect on performance of evolved populations.

The second part of the hypothesis predicts that populations evolve the ability to cooperate with other cooperators. Repeating the same experimental structure above, the five unevolved populations were mixed with a small set of AllC players. Tit for Tat and Pavlov again performed at nearly 80% if the maximum fitness, as did the initial population of AllC players. AllC players always cooperate by their nature. In an initial population made up entirely of AllC players, mutual cooperation is the norm. Introducing more AllC players to that initial population obviously does not change it. The prevalence of mutual cooperation explains the excellent performance of the unevolved AllC population.

Unevolved Populations inoculated with AllC

	Tit For Tat	Pavlov	Cooperate	Defect	Random
Mean	0.7954	0.7999	0.7873	0.8784	0.4384
t-test with unevolved Tit for Tat	1	0.8423	0.7179	1.9440E-05	1.9420E-45
t-test with unevolved Pavlov	0.8423	1	0.5659	3.2140E-05	4.6290E-48

Unevolved populations inoculated with AllD

	Tit For Tat	Pavlov	Cooperate	Defect	Random
Mean	0.8138	0.8240	0.7348	0.8825	0.4444
t-test with unevolved Tit for Tat	0.3974	0.2022	0.0077	7.3307E-06	1.3276E-44
t-test with unevolved Pavlov	0.5153	0.2740	0.0035	1.1989E-05	3.3702E-47

Evolved populations with no inoculation

	Tit For Tat	Pavlov	Cooperate	Defect	Random	Random Drift Population
Mean	0.7990	0.7970	0.8189	0.8045	0.8206	0.7517
t-test with unevolved Tit for Tat	0.8768	0.9427	0.2899	0.2733	0.2733	0.0091
t-test with unevolved Pavlov	0.9667	0.8961	0.3837	0.3603	0.3603	0.0029
t-test with uninoculated Tit for Tat	1	0.9310	0.3681	0.3459	0.3459	0.0046
t-test with uninoculated Pavlov	0.9310	1	0.3122	0.2940	0.2939	0.0047

Evolved populations inoculated with AllC

	Tit For Tat	Pavlov	Cooperate	Defect	Random
Mean	0.8145	0.7980	0.7977	0.8029	0.8297
t-test with unevolved Tit for Tat	0.3975	0.9117	0.9190	0.7364	0.1147
t-test with unevolved Pavlov	0.5115	0.9285	0.9163	0.8943	0.1622
t-test with uninoculated Tit for Tat	0.4911	0.9628	0.9518	0.8618	0.1562
t-test with uninoculated Pavlov	0.4277	0.9678	0.9770	0.7876	0.1228

Evolved populations inoculated with AllD

	Tit For Tat	Pavlov	Cooperate	Defect	Random
Mean	0.7981	0.8019	0.8015	0.8084	0.8248
t-test with unevolved Tit for Tat	0.9063	0.7837	0.7904	0.5696	0.1878
t-test with unevolved Pavlov	0.9345	0.9329	0.9455	0.7087	0.2561
t-test with uninoculated Tit for Tat	0.9690	0.9019	0.9132	0.6801	0.2459
t-test with uninoculated Pavlov	0.9619	0.8335	0.8421	0.6106	0.2019

Figure 3: Statistical Results

Compared to the controls and the AllC population, both Random and AllD populations performed significantly worse, at about 74% and 44% fitness respectively.

After 200,000 generations of evolution, the populations were again mixed with a small set of AllC cooperators. As was the case when inoculated with a defector, the fitnesses of the five evolved populations were statistically indistinguishable from one another. Comparing the performance of the evolved populations with the performance of the unevolved Tit for Tat and Pavlov populations, there was again no difference among populations. Finally, the performance of the inoculated population was

compared to that of the un-inoculated population and they were not statistically different.

7 Discussion

These results lead to several conclusions. Our first experiment shows that defectors effect all five of the evolved populations in the same way. They react identically, but does this necessarily indicate that they all have a defensive ability? Since the populations which were unable to defend against defectors *a priori* exhibit that behavior after evolution they must evolve that ability over time.

The second set of conclusions that can be drawn are those regarding mutual cooperation. Results here show that evolved populations are able to cooperate among themselves since they perform the same as the control populations in the presence of cooperators. Further, once can conclude that populations exhibit this behavior even without the experimental conditions, since there is no difference between performance in the natural, evolved environment and performance in the presence of pure cooperators.

With the results outlined above, it follows that in this experiment, evolved populations performed equivalently to Tit for Tat and Pavlov. Specifically, these experiments show that evolved populations are able to fend off defectors and mutually cooperate with other evolved individuals. Since these populations did not have such abilities *a priori*, it follows that evolution introduced this behavior over time.

Some preliminary simulations have been run to study this phenomenon in probabilistic strategies. Initial results show no difference in results between deterministic and probabilistic populations. A more thorough study would be useful in generalizing or limiting the results described here.

8 Acknowledgements

My deepest gratitude to Prof. Stuart Kurtz of the University of Chicago and Prof. Jim Reggia of the University of Maryland at College Park. Without their support and advice this paper would not have been possible.

References:

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

[2] Axelrod, Robert. "Laws of Life: How Standards of Behavior Evolve." *The Sciences* 27 (Mar/Apr. 1987): 44-51.

[3] Axelrod, Robert. *The Complexity of Cooperation*, Princeton: Princeton University Press, 1997.

[4] Baker, J. E. "Reducing Bias and Inefficiency in the Selection Algorithm." *Proceedings of the Second International Conference on Genetic Algorithms and their Application*, Hillsdale, New Jersey, USA: Lawrence Erlbaum Associates, 1987: 14-21

[5] Goldberg, David. *Genetic Algorithms in Search, Optimization and Machine Learning*, New York: Addison-Wesley Publishing Co, 1989.

[6] Holland, John. *Hidden Order: How Adaptation Builds Complexity*, New York: Persus Co, 1996.

[7] Kraines, David and Vivian Kraines. "Pavlov and the Prisoner's Dilemma", Theory and Decision, 26: 47-49.

[8] Kraines, David and Vivian Kraines. "Evolution of Learning among Pavlov Strategies in a Competitive Environment with Noise," *Journal of Conflict Resolution*, 39: 439-466.

[9] Kraines, David and Vivian Kraines. "Natural Selection of Memory-one Strategies for the Iterated Prisoner's Dilemma," *Journal of Theoretical Biology*, 203: 335-355.

[10] Mitchell, Melanie. *An Introduction to Genetic Algorithms*, Cambridge: MIT Press, 1998.

[11] Osborne, Martin J. and Ariel Rubinstein. *A Course In Game Theory*, Cambridge, Massachusetts: The MIT Press, 1994.

[12] Smith, John Maynard. *Evolution and the Theory of Games,* Cambridge: Cambridge Univ Press, 1982.