# Satisficing and Learning Cooperation in the Prisoner's Dilemma

**Jeff L. Stimpson**
Computer Science Department
Brigham Young University
Provo, UT 84602
jstim@cs.byu.edu

**Michael A. Goodrich**
Assistant Professor of Computer Science
Brigham Young University
Provo, UT 84602
mike@cs.byu.edu

**Lawrence C. Walters**
Associate Professor of Public Policy
Brigham Young University
Provo, UT 84602
larry_walters@byu.edu

## Abstract

The prisoner's dilemma is a useful model for studying the balance between self-interest and group-interest in multi-agent systems. Although many strategies have been developed that perform well, most of these strategies make strong assumptions about the information available to the agent. It is in this context that we describe a satisficing learning strategy for the prisoner's dilemma and present evidence that stable outcomes other than the Nash equilibrium are possible. In addition, we offer empirical evidence that under typical circumstances, mutual cooperation is the most likely outcome and identify conditions under which two satisficing agents will learn to cooperate.

## 1 Introduction

In situations involving several interacting agents, each agent is often forced to choose between two types of behavior: those that benefit the group as a whole, and those that lead to rewards for the individual at the expense of the group. The situation becomes interesting when, in the long run, poor outcomes for the group lead to negative consequences for each individual.

The iterated prisoner's dilemma is an elegant and well-known example of such circumstances that has been studied in a wide variety of disciplines. A typical payoff matrix for the prisoner's dilemma is given in Figure 1. The dilemma is

| (A's payoff, B's payoff) | | Agent B's Choice | |
|---|---|---|---|
| | | Cooperate | Defect |
| Agent A's Choice | Cooperate | (3, 3) | (1, 4) |
| | Defect | (4, 1) | (2, 2) |

**Figure 1:** A typical payoff matrix for the prisoner's dilemma.

that every pair of actions is either unstable or sub-optimal. Formally stated, the unique Nash equilibrium is the only outcome that is not Pareto optimal. Mutual defection is the dominant strategy in the sense that a player will be better off by defecting regardless of what his or her opponent does. Yet if both players make this "rational" decision to defect, both receive less than if they had cooperated.

In searching for an effective strategy in the prisoner's dilemma, we look for a strategy exhibiting flexible behavior. It should cooperate whenever mutual cooperation is possible, but it must be able to defect when it is apparent that its opponent is unwilling to cooperate. Many such strategies have been developed and studied, but often these strategies involve at least one of the following assumptions:

- players are aware of the structure of the game such as the other players, the other player's possible actions, and the relationship between the actions and the payoffs,
- players are immediately aware of other player's decisions,
- players are aware of the other player's payoffs,
- players are aware that they are in a game situation, meaning that they are aware that the actions of other agents are affecting their outcomes

In computer simulations these requirements are easily met, but in real-world situations they may be quite limiting. For example, the prisoner's dilemma can be extended to multiple players. If there are many players choosing from many actions, keeping track of the game structure may be unrealistic in terms of storage requirements and computational capacity. In other cases, information about the structure of the game may not even be available to the decision maker. Finally, although situations analogous to a prisoner's dilemma are common occurences, they are rarely thought of in terms of game theory. Instead, we are more interested in meeting specific goals.

Removing these assumptions from the prisoner's dilemma takes the problem out of game theory and into areas of machine learning. It is in context of these types of situations that we consider a satisficing strategy for the prisoner's dilemma. Specifically, the purpose of this paper is to present the strategy and then (1) show that stable outcomes other than the Nash equilibrium frequently occur and (2) describe the circumstances under which two agents employing a satisficing strategy will learn to cooperate.

## 2 Related Literature

The prisoner's dilemma was conceived in the 1950s to question some of the basic tenets of game theory. Standard rational decision mechanisms, such as minimax, lead to mutual defection and poor outcomes for both players. Since then there have been numerous attempts to "solve" the prisoner's dilemma by showing that mutual cooperation is rational after all. The most influential of these has been Axelrod's work in the repeated prisoner's dilemma [1984]. He shows that mutual cooperation is rational and stable when the following conditions hold: (1) the future is important, (2) there is sufficient difference between payoffs for mutual cooperation and mutual defection, and (3) one is facing an adaptive opponent. In summary, Axelrod shows that rationality in repeated-play games is not tantamount to Nash equilibrium.

The idea of applying game theory to learning in multi-agent systems is far from new. For example, Minimax-Q [Littman, 1994] is a reinforcement learning algorithm that learns the Nash equilibrium in zero-sum, or purely competetive, stochastic games. Further work such as [Hu and Wellman, 1998] has attempted to extend the same idea to general-sum stochastic games. Typically, the focus of this literature has been towards learning the Nash equilibrium. While this may be a desirable property in many circumstances, this approach has drawbacks. First, these algorithms usually require significant assumptions and knowledge about the game structure that can be quite limiting. Second, in light of Axelrod's work, in a repeated-play situation, the Nash equilibrium may not be the only stable solution with desirable properties.

Like much of the work done in the prisoner's dilemma, the concept of satisficing came about as a modification of rationality. Traditional rational choice theory holds that an agent faced with a decision will choose the alternative that maximizes a utility function. However, as noted in [Conlisk, 1996] and others, there is little empirical evidence that people make decisions in this manner; indeed evidence strongly suggests otherwise. As a replacement, Herbert Simon has proposed satisficing. He explains the difference between optimizing and satisficing: "A decision maker who chooses the best available alternative according to some criteria is said to optimize; one who chooses an alternative that meets or exceeds specified criteria, but that is not guaranteed to be either unique or in any sense the best, is said to satisfice" [Simon, 1997]. Rather than calculating optimal actions, a satisficing agent simply selects an alternative that meets a set of aspiration levels. As long as these aspiration levels are being met, the agent can continue to act without expending any search costs. When aspiration levels are not met, a search is executed until a satisfactory alternative is found.

In order to handle a variety of environments, aspiration levels can be adaptive. According to Simon, "if it turns out to be very easy to find alternatives that meet the criteria, the standards are gradually raised; if search continues for a long while without finding satisfactory alternatives, the standards are gradually lowered" [Simon, 1997].

We see several advantages in applying satisficing to multi-agent systems. First, because satisficing is simple and flexible, it can be applied when information, storage space, and execution time are limited. This means that agents do not need complex models of other agents. Satisficing is also robust—even if the environment changes (or initial information about the environment is wrong), a satisficing algorithm can typically adapt.

## 3 A Satisficing Strategy For the Prisoner's Dilemma

Applying Simon's satisficing algorithm to the prisoner's dilemma is straightforward. In this paper, we adapt the algorithm and notation presented in [Karandikar, *et al.* 1998]. The state at time $t$ for an agent using this strategy is given by the pair $(A_t,\alpha_t)$ where $A_t$ is an action in {C, D} and $\alpha_t$ is the current aspiration level. The players' actions determine the payoffs, $\pi_t^A$ and $\pi_t^B$. After receiving a payoff $\pi_t$ an agent employing a satisficing strategy updates its state in two steps. First, if $\pi_t \geq \alpha_t$ then $A_{t+1} = A_t$, otherwise $A_{t+1} \neq A_t$. Then, aspirations are updated as a weighted average between the current aspiration level and the received payoff. This update rule is given by equation (1) where $0 \leq \lambda \leq 1$.

$$\alpha_{t+1} = \lambda\alpha_t + (1-\lambda)\pi_t \qquad (1)$$

It is worth pointing out that the decision algorithm makes no use of the payoff matrix or the actions of the other players. Thus it can be applied to situations where this information is either complex or unknown. All that is needed is the ability to associate a payoff with an action. In addition, it is important to note that this algorithm requires three parameters for each agent: the update rate $\lambda$, an initial action $A_0$ and an initial aspiration $\alpha_0$.

Before moving into an analysis of the algorithm, a simple illustration is worthwhile. Given that $\lambda = 0.5$, $A_0 = C$, and $\alpha_0 = 4.0$, consider the example in Figure 2.

| t | Tit-for-Tat | $A_t$ | $\pi_t$ | $\alpha_t$ |
|---|---|---|---|---|
| 0 | C | C | 3 | 4 |
| 1 | C | D | 4 | 3.5 |
| 2 | D | D | 2 | 3.75 |
| 3 | D | C | 1 | 2.87 |

**Figure 2:** A brief example of a satisficing strategy against a tit-for-tat strategy

In this example, a satisficing agent is playing against a tit-for-tat strategy that simply cooperates on the first move and then repeats its opponent's last move on subsequent iterations. Initially, both players cooperate, receiving a payoff of 3. However, because this payoff is less than the satisficing agent's aspiration of 4, $A_1$ = D and the aspirations are updated as an average of the old aspiration and the new payoff.

## 4 Cooperation Among Satisficing Agents

Before describing our results in detail, we make two observations. First, reinforcement learning has been applied to the prisoner's dilemma with mixed results. In [Sandholm and Crites, 1996], several types of Q-learners were shown to play optimally against a fixed tit-for-tat strategy. However, due to the interaction of their learning, these Q-learners had difficulty playing optimally against each other. Second, although the satisficing algorithm described in the last section is simple, the dynamic interaction between two agents is difficult to theoretically characterize. Thus, in this paper we restrict our analysis to two satisficing agents playing against each other. In addition, we focus on presenting empirical evidence of circumstances under which these two agents will learn to cooperate.

In order to extend the notation to a two-player game, we introduce $B_t$ and $\beta_t$ as the second player's action and aspiration level respectively. For simplicity, $\lambda$ is set to the same value for both players. We also generalize the payoff matrix by setting the off-diagonal payoffs to (0,1) and (1,0) and then use $\sigma$ as the reward for mutual cooperation and $\delta$ as the reward for mutual defection with the constraints that $0 < \delta < \sigma < 1$ and $\sigma > 0.5$. This modified payoff matrix is shown in Figure 3.

| (A's payoff, B's payoff) | | Agent B's Choice | |
|---|---|---|---|
| | | Cooperate | Defect |
| Agent A's Choice | Cooperate | $(\sigma, \sigma)$ | (0, 1) |
| | Defect | (1, 0) | $(\delta, \delta)$ |

**Figure 3:** Generalized payoff matrix for the prisoner's dilemma

### 4.1 Convergence and Stability

Before presenting our results, we discuss the possible outcomes of a repeated prisoner's dilemma played by satisficing agents. The simplest outcome is convergence to a pair of actions ($A$, $B$). This occurs when $\alpha_t \le \pi_t^A$ and $\beta_t \le \pi_t^B$ meaning that both players are satisfied with their current payoffs and thus both players will repeat their actions indefinitely. At subsequent iterations, $\alpha$ will asymptotically approach $\pi^A$ and $\beta$ will asymptotically approach $\pi^B$. This can be seen as an equilibrium in the sense that neither player has an incentive to change, given their goals and what they have learned about their environment.

A second possible outcome is convergence to some action cycle, meaning that both players repeat a sequence of action pairs indefinitely. As a formal definition we say that the players have converged to a cycle of duration $N$ at time $\tau$, if for all $t > \tau$, and all $k$ such that $0 \le k \le N-1$, $A_{t+k} = A_{t+k+N}$ and $B_{t+k} = B_{t+k+N}$.

A third and final possibility to consider is that the interaction between two agents is entirely chaotic. This is at least very unlikely, as throughout our research the process has always converged to some stable outcome regardless of the payoff matrix or initial conditions. However, this remains to be shown theoretically.

Figure 4 is a brief illustration of the complexity of the process. It depicts the outcome as a function of the initial aspirations for three possible game structures and initial actions. Clearly there is no simple mathematical characterization of the relationship between game structure and initial parameters and convergence to cooperation. However, empirical results presented in the next section do allow us to identify conditions under which these agents will learn to cooperate.

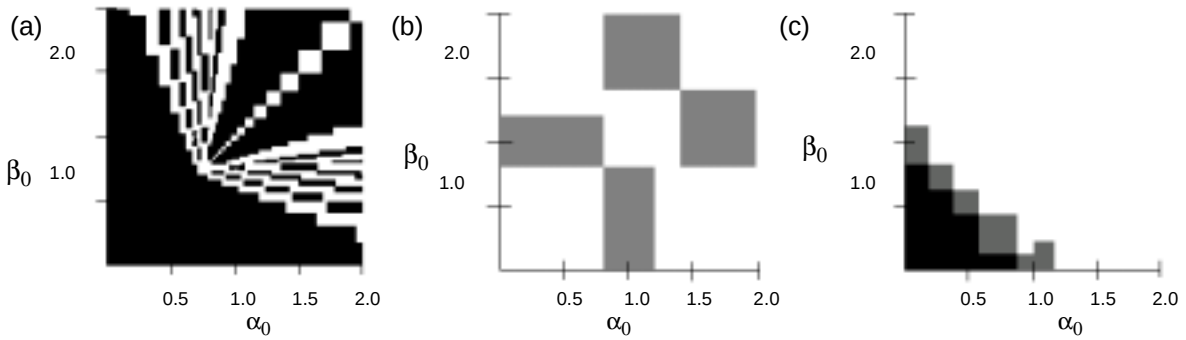


**Figure 4:** These three graphs show the relationship between initial aspirations and the final outcome for three different game structures. For each pair of initial aspirations in the graph, the outcome of the game was recorded. White indicates convergence to mutual cooperation, black indicates convergence to mutual defection, and gray indicates convergence to some cycle. In Figure 4a, $A_0$ = D, $B_0$ = D, $\sigma$ = 0.8, $\delta$ = 0.7, and $\lambda$ = 0.9. In Figure 4b, $A_0$ = C, $B_0$ = C, $\sigma$ = 0.8, $\delta$ = 0.5, and $\lambda$ = 0.5. In Figure 4c, $A_0$ = D, $B_0$ = C, $\sigma$ = 0.6, $\delta$ = 0.5, and $\lambda$ = 0.8.

## 4.2 General Results

We set up a simulation that randomly selects the parameters for a game from uniform distributions as described in Table 1.

| Parameter | Min. Value | Max. Value |
|---|---|---|
| $\alpha_0$, $\beta_0$ | 0.5 | 2.0 |
| $\lambda$ | 0.1 | 0.9 |
| $\sigma$ | 0.51 | 1.0 |
| $\delta$ | 0.1 | $\sigma$ |
| $A_0$, $B_0$ | 50% = C, 50% = D | |

**Table 1:** Distribution of parameters for simulations

The simulation then runs a repeated prisoner's dilemma until the process converges to some action pair or some action cycle. The final outcomes of 5,000 of these simulations are displayed in Figure 5.

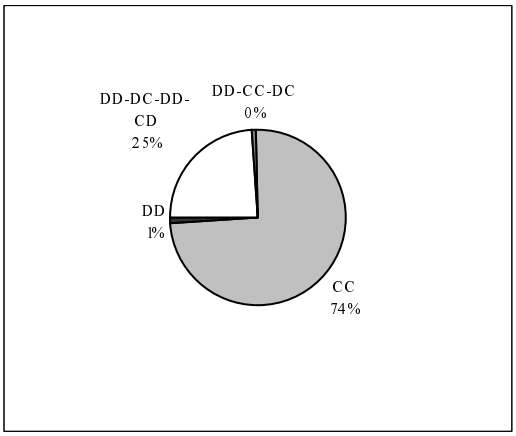


**Figure 5:** Frequencies of each of the possible outcomes from 5,000 trials. Parameters were randomly selected as described in Table 1.

It is interesting to note that every game converged to one of four possibilities: mutual cooperation, mutual defection, some variation on DD-DC-DD-CD, or some variation on DD-CC-DC.

## 4.3 Factors Leading to Cooperation

As shown previously, convergence to mutual cooperation is the most frequent outcome in a prisoner's dilemma played by two satisficing agents. Several factors influence this learning process between interacting agents. These are:

- initial aspirations,
- structure of the payoff matrix,
- learning rate, and
- initial actions

The remainder of this section focuses on analyzing how each of these parameters affect convergence to mutual cooperation.

**Initial Aspirations**

Figure 6 shows a contour plot of the frequency of mutual cooperation as a function of initial aspirations. It is clear that high aspirations are more likely to lead to cooperation. At first this may appear counter-intuitive—players with high aspirations might be unwilling to settle for cooperation. However, in most circumstances, both players are able to learn that they cannot expect more than mutual cooperation in the long run. On the other hand, players with low aspirations tend to remain satisfied with mutual defection or settle into cycles.

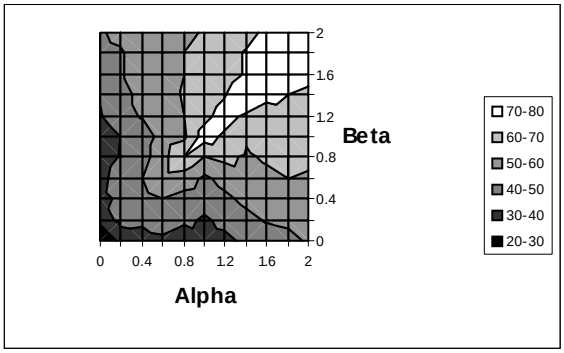


**Figure 6:** A contour plot of the percentage of trials out of 1,000 that converged to mutual cooperation as a function of initial aspirations. Light colors indicate that in most of the trials with the given initial aspirations, the agents learned to cooperate. Dark colors indicate that few of the trials led to mutual cooperation. Parameters other than $\alpha_0$ and $\beta_0$ were selected randomly as described in Table 1.

**Structure of the Payoff Matrix**

The structure of the payoff matrix can also have considerable influence over the ability of the agents to converge to learn to cooperate. Figure 7 shows the frequency of mutual cooperation as a function of $\sigma$ and $\delta$.

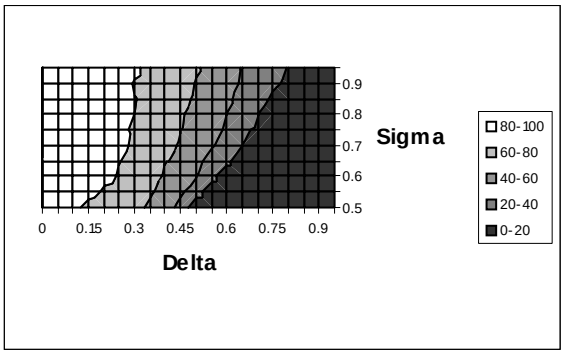


**Figure 7:** A contour plot of the percentage of trials out of 1,000 that converged to mutual cooperation as a function of each ($\delta$, $\sigma$) pair. Light colors indicate that most of the trials converged to mutual cooperation, while dark colors indicate that few of the trials converged to cooperation. Parameters other than $\delta$ and $\sigma$ were chosen randomly according to Table 1.

Note that cooperation is most likely when $\delta$ is small and $\sigma$ is large. This is expected because the distinction between cooperation and defection blurs when $\sigma$ and $\delta$ are close together. This type of behavior seems typical of non-optimizing algorithms. In describing his work in modeling human behavior, Arthur writes that human behavior (and his algorithm), "appear to 'discover' and exploit the optimal action with high probability, *as long as it is not difficult to discriminate*. But beyond a perceptual threshold, where differences in alternatives become less pronounced, non-optimal outcomes become more likely" [Arthur, 1991].

**Initial Actions**

To study the effects of initial actions on cooperation, we ran four sets of simulations, holding different initial actions constant each time. The percentages of samples that converge to cooperation for each group are shown in Table 2.

| Initial Actions | % of Cooperation |
|---|---|
| Random | 73.7 % |
| CC | 81.6 % |
| DD | 81.6 % |
| DC or CD | 66.7 % |

**Table 2:** Percentage of cooperation out of 1,000 trials as a function of initial actions. Parameters other than $A_0$ and $B_0$ where chosen according to Table 1.

While initial actions do not appear to be as significant as other factors, note that cooperation occurs with the same percentage regardless of whether the initial actions are cooperation or defection as long as both players choose the same action.

**Learning Rate**

The rate at which the aspirations are updated also has a considerable effect on whether mutual cooperation is learned. Figure 8 shows the relationship between $\lambda$ and the percentage of trials that converged on mutual cooperation. As $\lambda$ increases, the frequency of cooperation increases as well. The only exception is when $\lambda = 1$ (and thus aspirations are not updated at all), leading to virtually no cooperation.

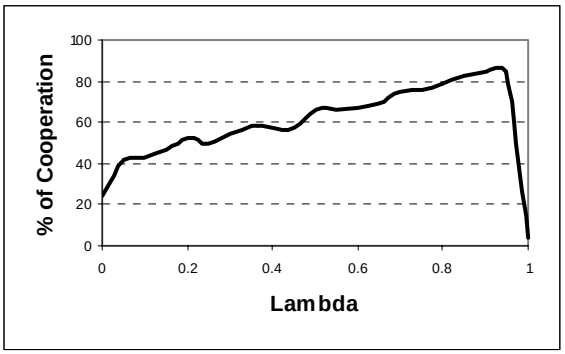


**Figure 8:** Percentage of trials out of 1,000 that converged to mutual cooperation as a function of the update rate, $\lambda$. Parameters other than $\lambda$ were selected randomly as described in Table 1.

# 5 Conclusions and Further Work

To summarize the results of the previous section, we restate five important factors that increase the likelihood that two satisficing agents will learn to cooperate:

- Agents should learn, but slowly.
- The difference between payoffs for mutual defection and mutual cooperation should be maximized.
- Agents should have high initial aspirations.
- Agents should start out with similar behavior.

As a test of these principles, we ran a final set of simulations enforcing the following conditions: $A_0 = B_0$, $\sigma - \delta > 0.4$,$1 > \lambda > 0.8$, $\alpha_0 > \sigma$, and $\beta_0 > \sigma$. Under these conditions, the agents learn to cooperate in 100% of 5,000 trials.

These results make a promising case for the use of satisficing in multi-agent systems as a way of balancing self-interest and common good when little information about the environment is available. Because agents do not directly model each other, the approach is fast, simple, and scalable to many players.

As a final note, we recognize that there are several directions for further work that should prove useful and interesting to researchers in multi-agent systems. We have limited our discussion of this satisficing algorithm to the prisoner's dilemma. However, because no assumptions about the relationships between the payoffs have been built into the algorithm, it should extend easily to other domains. In addition, the algorithm we have presented is limited to two-action decision problems with immediate feedback. Thus, the addition of a satisficing search algorithm for multiple actions is necessary and an extension to sequential decision problems would prove useful for many applications.

# Acknowledgements

The authors gratefully acknowledge the of support of the National Science Foundation under grant #CMS-9526018.

## References


[Arthur 1991] W. Brian Arthur. Designing economic agents to act like human agents: A behavioral approach to bounded rationality. *The American Economic Review* 81 (May): 353-359, 1991.

[Axelrod 1984] R.M. Axelrod. *The Evolution of Cooperation*. Basic Books, 1984.

[Conlisk 1996] John Conlisk. Why bounded rationality? *Journal of Economic Literature* 34(2): 669-694, 1996.

[Hu and Wellman, 1998] J. Hu and M. P. Wellman. Multiagent reinforcement learning: Theoretical framework and an algorithm. *Proceedings of the Fifteenth International Conference on Machine Learning*, 242-250. San Francisco: Morgan Kaufman.

[Karandikar, *et al.* 1998] R. Karandikar, D. Mookherjee, D. Ray, and F. Vega-Redondo. Evolving aspirations and cooperation. *Journal of Economic Theory*, 80:292-331, 1998.

[Littman 1994] Michael L. Littman. Markov games as a framework for multi-agent reinforcement learning. *Proceedings of the Eleventh International Conference on Machine Learning*, 157-163. San Francisco: Morgan Kaufman.

[Sandholm and Crites, 1996] Tuomas W. Sandholm and Robert H. Crites. Multiagent reinforcement learning in the Iterated Prisoner's Dilemma. *BioSystems*, 37: 147-166, 1996.

[Sen, *et al*. 1994] S. Sen, M. Sekaran, and J. Hale. Learning to coordinate without sharing information. *Proceedings of the Twelfth National Conference on Artificial Intelligence* (AAAI-94). Seattle, WA. 426-431.

[Simon 1997] Herbert A. Simon. *Models of bounded rationality.* Vol. 3, *Empirically grounded economic reason*. Cambridge, Mass. MIT Press, 1997.