Essays on Bounded Rationality and Strategic Behavior in Experimental and Computational Economics

by

James R. Leady

A dissertation submitted in partial fulfillment
of the requirements for the degree of
Doctor of Philosophy
(Economics)
in The University of Michigan
2011

Doctoral Committee:

Professor Yan Chen, Chair
Professor Scott E. Page
Professor Carl P. Simon
Associate Professor Julianne Berry Cullen, University of California, San Diego

Dedication

I would like to dedicate this dissertation to my beautiful wife,
Michelle, for her encouragement, sacrifice, and patience,
and to my wonderful children, Joshua and Kayla.

Acknowledgments

Special thanks are due to Yan Chen and Scott Page for guiding my research and professional development throughout the whole Ph.D. program. I am grateful to Carl Simon and Julie Cullen for their guidance and many valuable comments on preliminary versions of the chapters in my dissertation.

For Chapter 1, I am thankful for the inspiration, guidance, and support provided by Yan Chen. I also thank Rick Riolo for his design and programming assistance. Comments by participants of the International Meeting of the Economic Science Association, June 2000, at New York University, and participants of the Fall 2000 Public Finance Seminar at the University of Michigan are gratefully acknowledged. I also recognize the Center for the Study of Complex Systems at the University of Michigan for use of its computing resources.

Chapters 2, 3, and 4 evolved from discussions with Scott Page and John Miller, directors of the 1999 Graduate Workshop in Computational Economics at the Santa Fe Institute, whom I thank especially. Participants at this workshop as well as discussants at the 74th Annual Meetings of the Southern Economic Association provided valuable suggestions and comments. Again, I am grateful to the Center for the Study of Complex Systems at the University of Michigan for computing support.

Table of Contents

List of Tables

Table

List of Figures

Figure

Chapter 1

If Nobody Is Going There Anymore Because It's Too Crowded, Then Who Is Going? Experimental Evidence of Learning and Imitation in the *El Farol* Coordination Game

1.1. Introduction

This paper concerns Yogi Berra's colorful observation: "Nobody goes there anymore. It's too crowded." A well-known problem in economics concerns the manner in which externalities, crowding in particular, can cause individual optimizing behavior to conflict with social optimizing behavior. In some coordination problems, however, externalities create problems that are even more serious: they make it difficult for agents to even know *how* to optimize their own individual behavior.

Crowding can occur in systems that have some fixed capacity. A person's participation in an event may make that event less profitable or desirable for others who also participate. For example, the effects of crowding are experienced by one who lacks offers while selling a house during a buyer's market or by one who experiences slow downloads while accessing CNN.com during a major crisis. In some cases, crowding effects can be reduced by an external mechanism that explicitly coordinates the participants' actions. Often, however, such mechanisms do not exist, and people must implicitly learn to coordinate actions in order to limit the negative effects of crowding.

Negative effects may build gradually, as in a large realty market where each additional home listing has only a marginal negative effect on the selling prices of existing listings. In other cases, crowding can reach a critical point. One too many web surfers seeking the latest news during a crisis may cause a popular web site to crash. The latest disappointing profit report may send a stock market into a nosedive.

This paper analyzes a theoretical problem comparable to the latter two examples. The *El Farol* Bar Game[1] first introduced by Arthur (1994) features abrupt congestion effects when people cannot explicitly coordinate either directly or through some external mechanism. Arthur provided a simple description of the game:

> N people decide independently each week whether to go to a bar that offers entertainment on a certain night. For concreteness, let us set N at 100. Space is limited, and the evening is enjoyable if things are not too crowded—specifically, if fewer than 60 percent of the possible 100 are present. There is no sure way to tell the numbers coming in advance; therefore a person or agent *goes* (deems it worth going) if he expects fewer than 60 to show up or *stays home* if he expects more than 60 to go. Choices are unaffected by previous visits; there is no collusion or prior communication among the agents; and the only information available is the numbers who came in the past weeks (408).

His simulations found that mean attendance converges always to 60. Moreover, an ever evolving "ecology" of strategies emerges such that, collectively, they predict this threshold attendance accurately *on average*. There still exists swings about the mean attendance, suggesting that the artificial agents are not fully coordinating.

The current study relaxes the limit on available information so that people are partially aware of others' strategies. The goal of the study is to identify ways people learn when given limited information about the success of strategies used by others. A hybrid design was developed consisting of laboratory experiments and computer

[1] The game is also known as the Santa Fe Bar Problem.

simulations. The basic design (see section 1.3) asks each subject in a group to develop a strategy to play the Bar Game. Artificial agents implementing these strategies then play against each other in a computer simulation. Subjects are informed of results, which include partial information about the success of others' strategies. Subjects then develop another strategy, and this process repeats several times.

This method resembles Axelrod's Prisoner's Dilemma tournaments (1984). The inclusion of a controlled laboratory experiment differentiates this project from Axelrod's tournaments. His tournaments were more informal and lacked a tangible incentive structure for the participants (although one could there were argue reputational incentives); this project uses grade points to motivate participation. Further, the participant pool varied between Axelrod's two iterations; here the same subjects are used throughout. While this experiment follows in the same spirit as Axelrod's tournaments, it proceeds with a rigorous design to discover whether people can learn to coordinate and in what way they do so.

Discovering how people behave in the Bar Game is important because, despite its colloquial name, the game models several real-world settings. This coordination game is studied in the economics and complex systems literatures as a model of stock market behavior. In a model with a bounded population, increasing agent sophistication (reasoning ability) at the individual level may lead to decreasing diversity at the population level. A more sophisticated agent can potentially do better, but a less diverse population can lead to inefficient social outcomes (Johnson, *et al*., 1998). In models of the related Minority Game[2], as agent sophistication increases, aggregate payoffs first rise but then fall. A bound on rationality improves the system's efficiency (Savit, Manuca and Riolo, 1997; Manuca, Li, Riolo and Savit, 1998; and Li, Riolo and Savit, 1999a, 1999b). In studies of the Bar Game in the computer science literature, systems with boundedly rational agents can lead to better outcomes than those with fully rational agents (Greenwald, Mishra and Parikh, 1998; Bell, Sethares and Bucklew, 1999; and Bell, *et al*., 1999).

The studies mentioned above each use computer simulation. When modeling a complex adaptive system, a simulation can be an effective method (Holland and Miller, 1991). If one is trying to investigate ways that learning occurs, however, this approach is limited because the dynamics of learning must be specified in the simulation. To overcome this liability of a purely computational model, this project implements a new experimental approach combining computer simulations involving artificial agents with a laboratory experiment involving human subjects.

The experimental approach has proved useful in uncovering the nature of learning dynamics in coordination games (see Ochs, 1995; Van Huyck, Battalio, and Beil, 1990; Van Huyck, Cook, and Battalio, 1994; Duffy and Ochs, 1999). In particular, the Two-Island Game of Meyer, Van Huyck, Battalio, and Saving (1992) shares many features with the Bar Game investigated in this paper. These researchers found that though the symmetric equilibrium correctly described variable means, naïve subjects did not behave

---

[2] The Minority Game is a restricted form of the Bar Game (Greenwald, Mishra and Parikh, 1998; Challet and Zhang, 1997; Li, Riolo and Savit, 1999a, 1999b; Manuca, Li, Riolo and Savit, 1998; and Savit, Manuca and Riolo, 1997). In this game it is always best to take the action that differs from what the majority of people are taking.

in accordance with standard theory and were not able to learn to coordinate on useful precedents.

The Bar Game presents an environment in which it is difficult to coordinate, but this study differs from previous ones both in methodology and focus. This project employs the strategy method (Selton, 1967). This method differs from the standard economics experiment, in which a subject takes a single *action* during a round of play. With the strategy method, a subject submits instead a *strategy* for play. A strategy specifies a complete plan of action that prescribes what the subject will do in every possible situation. The strategy method may grant the observer insight into subjects' motivation (Mitzkewitz and Nagel, 1993). It also allows her to acquire data on information sets not actually reached during play of the game. However, the strategy method has some limitations. It eliminates the observation of timing decisions and their effect during play of a repeated game (Roth, 1995). Furthermore, its hypothetical character may make it unrealistic as an abstraction of a natural environment (Brandts and Charness, 2000). Despite these limitations, the strategy method proves valuable because of the added insight it provides.

This paper contributes to the existing knowledge of complex systems by integrating experimentation using human subjects into a computational model. This methodology is advantageous because no *a priori* assumptions are needed on the structure of the learning dynamics. The paper also contributes to the economics literature by providing evidence that people use imitation behavior as a method of learning and by demonstrating a variation on the strategy method.

1.2. Theoretical Properties

Consider the one-shot Bar Game with N agents indexed by i. The bar has a capacity of c. The agents have identical preferences toward the attendance and consider the bar to be not crowded if the realized attendance does not exceed capacity, *i.e.*, $A \leq c$. Each agent independently chooses an action $a^i \in \{0, 1\}$, where 0 and 1 can be interpreted as "stay home" and "go to the bar." The realized attendance is $A = \sum_i a^i$. Table 1.1 presents the payoffs for the generic case[3].

Table 1.1 indicates that an agent's payoff depends both on her choice of action and the actual attendance. For instance, an agent who "goes" receives a payoff of $\overline{G}$ if $A \leq c$ (the bar is not crowded) and receives a payoff of $\underline{G}$ otherwise. The strategy "stay" is the security strategy in the stage game and always leads to a payoff of S. This payoff structure features a discontinuity in marginal payoffs as the attendance increases beyond capacity. As long as attendance is below capacity, increasing the attendance by one person does not affect payoffs. However, once the bar reaches capacity, increasing the attendance by one extra person causes payoffs to decrease for all who decide to go. While one can relax this discontinuous payoff construction to some degree without drastically altering outcomes, the critical mass aspect is relevant to many economic problems. The discontinuous payoff structure gives the model its typically fluctuating attendance pattern (Arthur, 1994). Why a completely deterministic computer simulation leads to a random appearing attendance pattern can be explained analytically (Zambrano, 2004).

[3] This is the payoff structure for the canonical game as described in Arthur (1994). The Minority Game (see footnote 2) has c = N/2, and its payoff matrix is ((1, 0), (0, 1)).

In pure strategies any outcome in which exactly c agents attend is a Nash equilibrium of the stage game, and there are $\binom{N}{c}$ combinations of these outcomes. Keeping the capacity c a fixed proportion of N, the number of Nash equilibria in this game grows exponentially with the size of N. The presence of multiple equilibria makes it very difficult for the agents to coordinate. The aggregate payoff for these equilibria, which are Pareto optimal, is $c * \overline{G} + (N - c)S$. There is a symmetric mixed strategy Nash equilibrium in which homogeneous agents choose "go" with the same probability, which approaches c/N as N gets large. Computing the symmetric mixed strategy equilibrium, the expected payoffs π^i to "go" and "stay" for agent i are, respectively:

$$E(\pi^i \mid a^i = 1) = \underline{G} * \Pr(A > c \mid a^i = 1) + \overline{G} * \Pr(A \leq c \mid a^i = 1) \tag{1a}$$

$$E(\pi^i \mid a^i = 0) = S \tag{1b}$$

Setting these expectations equal and letting p represent the common probability to "go," we get:

$$\underline{G} + (\overline{G} - \underline{G}) \Pr(A \leq c \mid a^i = 1) = S, \text{ or}$$

$$\underline{G} + (\overline{G} - \underline{G}) \sum_{i=0}^{c-1} \binom{N-1}{i} p^i (1-p)^{N-1-i} = S \tag{2}$$

Note that when all agents play the mixed strategy p^* that solves equation (2), each agent's expected payoff is only S. The aggregate expected payoff from this symmetric mixed strategy is simply $N * S$, which is strictly less than $c * \overline{G} + (N - c) S$ since $S < \overline{G}$. The pure strategy Nash equilibria described above Pareto dominate the symmetric mixed strategy equilibrium. This equilibrium is only as efficient as the outcome in which all agents choose to "stay," which is the agents' security strategy.

While the properties of the stage game are relatively straightforward, they become more interesting and complex in the repeated game. The only information available to an agent in the standard version of the finitely repeated Bar Game is the common history of aggregate attendance at the bar during every previous round (Arthur, 1994, 408). The symmetric mixed strategy equilibrium of the stage game described above is also a Nash equilibrium for the finitely repeated game. A feature of this equilibrium that one may find desirable is its fairness—all agents receive the same expected payoff. Other fair but more efficient equilibria exist. For example, one symmetric pure strategy Nash equilibrium is for agents to cyclically change their actions such that each agent chooses "go" (*i.e.*, $a^i = 1$) exactly c out of every N rounds, and attendance is exactly c in every round. Of course, the question of how agents could learn to coordinate their actions in this cyclic manner is the crux of this problem, and the answer depends heavily on the population's heterogeneity.

Population heterogeneity plays a strong role in the aggregate performance of agents in this coordination problem. If all agents are using the same deterministic strategy (so that each responds the same way as her peers in any given round), then all will choose to either "go" or "stay" each round. The first case leads to the least efficient outcome, and the second case leads to the security outcome. For example, if all agents adapt using the Cournot best-response or fictitious-play algorithms, the attendance will oscillate between zero and N. Aggregate payoffs will average $N/2\ (S + \underline{G})$ in the Cournot case (and less with fictitious-play), which is strictly less than the aggregate security payoff $N * S$. In this game heterogeneity of actions opens the way for more efficient outcomes.

An understanding of how agents learn to implicitly coordinate actions and reach these more efficient outcomes demands some specific knowledge of the strategy space. As the strategy space for this game is vast, a methodical search through the space will likely be fruitless. Another approach is to discover a subset of pertinent strategies that may be used by real people. This objective—to investigate which classes of strategies are utilized by subjects to play this game in a laboratory setting—is incorporated into the design of the experiment.

1.3. Experimental Design

This experiment was designed to evaluate how human subjects learn to coordinate in a finitely repeated version of the *El Farol* Bar Game. To gain insight into the learning process, it is helpful to understand how subjects update strategies. Therefore strategies were collected from subjects in the laboratory and then implemented in computer simulations. Due to the time required to convert the subjects' written strategies into computer code, the collection and simulation steps did not occur concurrently. The procedural detail of each step follows.

Laboratory Procedures

The experiment consisted of eight repetitive sessions over the course of a semester for each of two independent cohorts. Figure 1.1 below summarizes the design and illustrates the interaction of the laboratory and computational components of the experiment.

As one reads from top to bottom, this figure displays a progression in perspective from the experiment (eight weeks long) to session (once weekly meeting with subjects in the lab) to trial (one simulation of the finitely repeated Bar Game on the computer). Starting at the top, the aggregate experiment consists of two treatment groups which each participate in eight sessions. Moving down the figure, each session consists of three tasks: reviewing previous sessions' results, formulating new strategies, and running the simulations for the Bar Game trials. The first two tasks took place in the lab, but the simulations occurred separately after the subjects departed. The bottom of the figure shows the detail of a trial, which consisted of a single simulation of the Bar Game played for 100 rounds using ten subjects' strategies.

At the beginning of each session (after the first), the subjects received reports of previous sessions that included statistics for the group as a whole, for each individual, and for several categories of strategies. After subjects reviewed the data, each formulated a strategy to be played in the Bar Game. Their strategies were then coded and the simulations were run.

This experiment consisted of two independent sessions with subjects recruited from two upper-level undergraduate economics classes from January 20 to March 30, 2000[4]. The only difference between the two treatments (other than the subject pools—see footnote 4) was in the payoff structure, so that one treatment group had a greater incentive to play "go" relative to "stay," as illustrated in Table 1.2. Each of these treatments lasted for eight sessions. Each sub-table shows a subject's payoff for a

---

[4] The selection of the subject pool from these two courses had two undesirable characteristics. The first class (treatment A) was an experimental economics course that introduced economic principles through experiments; the second (treatment B) was a political economy course that used game theory to analyze political institutions. The potential bias was mitigated, however, by the fact that in neither case did the curricula cover repeated games during the course of the experiment. A few of the subjects had been exposed to a variation of the game during a pilot experiment in an earlier semester.

particular action given the *ex post* attendance for a single round. Other experimental parameters are summarized in Table 1.3, with Table 1.3.a showing those parameters common across all sessions, and Tables 1.3.b and 1.3.c listing the participation by session for treatments A and B, respectively.

These subject pools were selected for two reasons. The combination of the strategy method and computer simulations made it infeasible to finish the experiment within the standard length of a couple of hours. Using classes and conducting the experiment during the course of a semester maintained the subject pool and allowed the use of this unique methodology. Second, I used grade points, rather than money, to provide subjects with incentives. Each subject had the option of using the experimental score to substitute for her lowest homework score[5]. The use of grades to induce value in an experiment can elicit high levels of motivation (Friedman and Sunder, 1994, 43; see also Isaac, Walker and Williams, 1994; Kormendi and Plott, 1982; Marimon, Spear, and Sunder, 1993). Each course formed an independent treatment group[6].

At the beginning of the first session, the subjects received written instructions[7] and heard them read aloud. To ensure all subjects understood the instructions, the subjects completed a brief quiz on the payoff structure, and any wrong answers were publicly explained. Following the instructions, each subject formulated a strategy for the

[5] Each homework assignment was 5% of the total course grade. For each class, the median grade was B+.

[6] One subject was enrolled in both courses, but only participated in the treatment group with the class that met earlier.

[7] The instructions used a neutral language in the description of the game. In particular, the word “event” was used in place of the word “bar.” Copies of the written instructions and strategy submission form are available upon request.

first session's simulations of the Bar Game and recorded it on a strategy submission form (see footnote 7). The only restrictions placed on the form of the strategy were completeness, consistency, and feasibility (it could not condition on unavailable information). As the subjects turned in their strategies, they were checked to ensure they did not violate the restrictions listed above. This completed the laboratory session with the subjects. Next the strategies were encoded into a computer program, and the simulations were run, producing summary results to be presented in the next session.

In the second session, the subjects listened to oral instructions and then received summarized results from the first session. These results included:

(a) The average per-round payoff and attendance for all previous sessions. The use of per-round figures maintained consistency with the payoff matrix.

(b) A listing of each subject's performance in the previous round (see Figure 1.2 for a typical listing). The performance statistics included each subject's average per-round payoff, the percentage of rounds her strategy selected "go," and the percentage of rounds it made the *ex post* optimal selection. The data were sorted from highest to lowest payoff. The listing identified subjects only by their university ID numbers to maintain anonymity

(c) A listing of various categories of the subjects' strategies. Figure 1.3 presents an example. Section 1.4 discusses the categories in detail. For consistency, a single set of categories was used for both treatments; this set remained constant throughout all sessions of the experiment. These categories were developed after the first laboratory session was finished and were determined from the actual set of strategies the subjects submitted. If a strategy could be classified into two or more separate categories, then it was included in all categories into which it fit. Each category included a generic example. The data presented for each category included the number of subjects who used a strategy from the category and the average per-round payoff received by strategies in the category.

The subjects received the results in the form of written handouts, overhead projections, and oral summaries. The instructions were then briefly reviewed, and the subjects had an opportunity to ask questions before receiving the payoff matrix either on the chalkboard or on the strategy submission forms. Then each subject formulated a new (possibly the same) strategy. Again, submitted strategies were reviewed for completeness, consistency and feasibility. Session 1 lasted approximately 30 minutes, and sessions 2-8 each lasted about 10-15 minutes. After the session the strategies were encoded and the simulations were conducted. This process continued for eight sessions.

At the end of the experiment, the instructor for the courses from which the subjects were drawn included each subject's average payoffs in her course grade. This inclusion provided the necessary incentive for active participation. The instructor divided each subject's average payoff for the whole experiment by the maximum average payoff achieved, and then scaled these relative payoffs to a point value corresponding to one homework assignment. At the end of the semester, she gave each subject the option to substitute her experiment score for a low homework score (see footnote 5). In all but a few cases, subjects included experiment scores as part of their course grades.

Simulation Procedures

After the laboratory session was complete, each subject's strategy was encoded into a computer program[8], which played out the simulations of the Bar Game. During each trial (100 rounds of an iterated Bar Game played with 10 subjects) , the simulation used the following algorithm:

1. Select ten subjects' strategies randomly from a uniform distribution without replacement and load their strategies into the simulation[9].
2. Implement simultaneously these ten strategies[10]. Every round each strategy prescribes an action ("go" or "stay"), possibly conditioning on the common attendance history, its private payoff history, and/or its private action history. (In round 1, no historical information is yet available.) Each strategy then updates its private action history.
3. Determine attendance at the event for this round, and update the collective attendance history.
4. Allocate payoffs and, for each of the ten strategies, update its private payoff history.
5. Repeat procedure starting at step two until 100 rounds of the simulation are complete.

---

[8] The simulation was coded in ObjectiveC using the Swarm v1.3.1 libraries. It was compiled and run on Hewlett-Packard machines running the HP-Unix operating system. A listing of the code is available from the author by request.

[9] The program used the Mersenne Twister MT19337 high quality pseudo-random number generator provided in the Swarm libraries. It has a period of 2^{19337} or 10^{6001}.

[10] Strategies are technically activated sequentially in a single computer processor; however, the program's schedule makes this detail transparent, effectively engaging the strategies as if they acted simultaneously.

Each trial game was an independent run on the computer. The number of trials conducted in a particular session was 100 times the number of subjects participating in that session, resulting in each subject's strategy engaging in approximately 1000 trials. This format minimized small group-size bias in the play. Earlier pilot experiments had revealed that 100 rounds per trial were sufficient to observe full interaction of the strategies played.

The results of the hybrid laboratory and computer experiments follow.

1.4. Results

Attendance varied widely about its mean level, which was near the capacity, as in Arthur (1994). The subjects submitted strategies that ranged from the very simple (single action, deterministic) to the complex (composites of deterministic, stochastic, and conditional). Some strategies resembled classic ones such as Cournot best-response, fictitious-play, reinforcement learning, and pure strategies; however, many were hybrids of these strategies.

Table 1.4 presents the mean attendance and payoff for each treatment group by session. A particular strategy's reported payoff for a session is the average over all rounds of all trials in which it played during that session. Table 1.4 then presents the mean of all strategies' payoffs. Figures 1.4 (attendance) and 1.5 (payoffs) present the same data in graphical form. For Treatment A, though the mean attendance was 6.03, close to the capacity and socially optimal level of 6.00, the attendance generally increased throughout the experiment. The attendance started at a low of 3.98 in session 1, increased from the previous session 5 out of 7 times, and ended on a high of 6.99 in session 8. A simple linear regression of the mean attendance on session numbers results in a positive coefficient on the session variable that is statistically significant, confirming an increasing pattern. In contrast, the attendance pattern of Treatment B did not display an increasing trend.

The payoff graphs for each group nearly mirror the attendance graphs after rotating them 180 degrees about the horizontal axis. For Treatment A, the average payoff decreased as mean attendance continued to increase beyond the capacity. Likewise for Treatment B, losses and gains in the mean payoffs were matched by shifts in attendance above and below the bar's capacity.

<u>Result 1</u>: Subjects used strategies that could generally be classified into ten categories. These categories[11], listed in the order of most used to least used, were:

(1) "Always" go.

(2) Streaking and/or alternating strategy, for example: *"Go" for rounds 1-30, "stay" for rounds 31-80, and alternate between "go" and "stay" rounds 81-100 ("go" in round 81).*

(3) Condition choice on previous round's attendance, such that "go" is prescribed if attendance was below some threshold. For example, *If attendance last round* < *4, then "go," otherwise "stay."*

(4) Condition action on subject's own payoff history.

(5) Condition action on some attendance average.

(6) Condition action on previous round's attendance, such that "go" is prescribed if

[11] Figure 1.3 displays the categorical data in the form presented to the subjects for treatment B in session 6. These results were presented in cumulative form so that in any given session, subjects could see the aggregate and categorical results for all previous sessions.

attendance was above some threshold. For example, *If attendance last round > 4, then "go," otherwise "stay."*

(7) "Always" stay.

(8) Trigger-like strategy, for example: *For rounds 1-10, "go." If average attendance through round 10 is less than 6, then "go" for all remaining rounds, otherwise "stay" for all remaining rounds*.

(9) Mixed strategy, for example: *"Go" with probability 0.6, "stay" with probability 0.4.*

(10) Other.

Discussion: The first nine categories comprised 99.5 percent of strategies submitted by the subjects, with only 3 of 623 strategies falling into the tenth category "Other." Some subjects submitted strategies that could potentially be classified into multiple categories. For instance, such a strategy might dictate to play a mixed strategy the first half of the rounds, and then alternate between go and stay that last 50 rounds. In these cases, the strategy was included in all applicable categories, except for "Always" stay and "Always" go.

Strategies were classified as "Always" go and "Always" stay if they dictated the action at least 95 percent of the time, as it was not uncommon for a subject to "stay" in round 1 and "go" in rounds 2-100 (or vice versa). This behavior may have been a sophisticated signaling effort on the subject's part, or it may have been influenced by the design of the strategy submission form. To help subjects avoid submitting incomplete strategies that conditioned on past information, the form explicitly asked subjects what their choice was in round 1. The category "Streaking and/or alternating strategy" was mostly composed of strategies that either cycled repeatedly through some series of actions or played the same action for extended rounds before switching to the other. Strategies in the third category listed above were very much like Cournot best-response, except they did not always condition on attendance relative to capacity, as best-response would dictate in this game. In general, these strategies dictated to go if the previous round's attendance was below some threshold. Category (6) was just the opposite: these strategies directed to go if the previous attendance was above some threshold (again these strategies did not always condition on attendance relative to capacity). Category (4), conditioning on payoff history, is indicative of reinforcement learning, while category (5), conditioning on attendance averages, is indicative of fictitious play-like behavior. Trigger strategies in category (8) prescribed the same action for all remaining rounds after some condition is met in a particular round. Often the condition was the subject's own payoff or the average attendance up to that point in the game.

Tables 1.5.a and 1.5.b present session data for these categories for Treatments A and B, respectively. These tables include the number of strategies, and payoff means and standard deviations for each category by session. Table 1.5.c summarizes the treatments for all sessions.

Even though the researcher defined the categories upon completion of the first session, relatively small standard deviations of payoffs characterized the categories. Referring to Tables 1.5.a and 1.5.b, the third row of data for each category shows its sample standard deviation of payoffs. In most cases, the categories' figures are less than those for the session as a whole (shown in the last row of each table). Considering both treatments, the only categories that did not display tighter standard deviations of payoffs than the treatment session as a whole were categories (4) and (5)—those strategies that condition on payoff history and an attendance average. Each of the other seven defined categories was distinguished by the relative homogeneity of its payoffs. This results holds strongest for categories (1), (3), and (7)—those that prescribe to "Always" go, go if the previous round's attendance was below some threshold, and "Always" stay[12].

The category data aggregated across all sessions that Table 1.5.c presents provides additional support that payoffs within categories were bunched closely. Here, eight of nine categories have lower payoff standard deviations than all strategies combined in at least one treatment, and four categories show the same for both treatments. However, these data are not as illuminating as those presented by session. Category (1), for example, had very similar payoffs within each session, but the variance between sessions was high.

One contribution of this project is the comprehensive list of strategies that subjects developed. The use of the strategy method with only minimal guidelines placed on strategy structure enabled the subjects to submit imaginative responses. This method allows one to gauge behavior that people may use in strategic settings similar this game, with a word of caution. One should not expect this specific listing of elicited strategy classes to generalize to other games, even those similar to the Bar Game. What one can take from this result is the expectation of heterogeneity across, but not necessarily within, strategy categories.

[12] There are slight deviations in the payoffs for "Always" Go and "Always" Stay from subject to subject within a round because these categories allow up to 5 percent of the rounds to use some other rule.

Result 2: Subjects imitated the most successful strategies.

Discussion: For Treatment A, the strategy category with the highest payoff in a particular session never had fewer subjects use it in the following session, and in all but one case had more subjects use it in the following session. Table 1.6 presents the results. Column 2 lists the category receiving the highest average payoff in session t. Column 3 gives the absolute increase in the number of subjects whose strategies fell within the category from session t to t+1, and column 4 lists the corresponding percent increase. For session 7, two categories are listed as best because of the small difference (less than 1 percent) in their payoffs.

The imitation effects are more pronounced for Treatment B. With this group, the top-performing category always increased in usage, except in session 2. This session had two top-performers, one whose usage increased by 8 subjects, and another whose usage dropped by one subject.

In this treatment, the strategy "Always" go appears to be the focal strategy—subjects used it a total of 104 times, making it much more popular than the two runner-ups that were used 62 times each. Subjects using this strategy received the highest payoff in five sessions. After each of these five sessions, the number of subjects using this strategy increased prominently, with an average gain of 6.8 subjects (out of about 40) per session. Because this strategy was a focal point both in popularity and payoffs, subjects may have chosen to imitate it. Players who perform poorly tend to imitate the strategies of those they see doing better (Axelrod, 1996). This evidence from both treatments strongly suggests that some subjects imitated the best performing strategy of previous session or sessions. The imitation appears limited to the best strategies; more general replicator effects are not observed.

Result 3: In both treatments, the following categories of strategies grew in usage, while all others decreased in usage:

- Category (1): "Always" go (from 7 subjects in session 1 to 34 subjects in session 8).
- Category (5): Condition choice on some attendance average (fictitious play-like) (from 3 to 8 subjects).
- Category (9): Mixed strategy (from 1 to 5 subjects).

Discussion: These strategies increased in usage in both treatments. High payoffs cannot explain the growing popularity of these three categories; the payoff rank (averaged over the whole experiment) of these three were fifth, second, and eighth out of ten, respectively. The growing use of the "Always" go strategy may be due to it being the top-ranked category in several sessions, as discussed in Result 2. The fictitious play-like category was the top performer in two sessions, and overall received the second highest payoff averaged over all sessions and treatments. The mixed strategy usage is more difficult to explain, and may be due to the small sample size.

Result 4: Play did not converge to the pure strategy equilibrium of the stage game, in which attendance equals capacity, in either treatment.

Discussion: An inspection of Table 1.4 and Figure 1.4 reveals that the mean attendance did not converge to the capacity attendance (six). Figure 1.4 especially highlights the difficult nature of this coordination problem. This result is partially limited by the number of sessions during which subjects had the opportunity to learn.

This variation of the Bar Game presented additional information to the subjects that may have assisted them to coordinate their actions to a higher degree. Specifically, in this experiment subjects were able to observe the strategies and payoffs of the others, albeit in a reduced form. This additional information did not appear to help the subjects to improve their payoffs; Treatment B's payoffs had a high variance without a trend and Treatment A's payoffs appeared to decrease throughout the experiment (see Figure 1.5). Also, subjects' payoffs were inefficient relative to random strategies. Table 1.4 lists the payoff efficiencies[13] for each session and shows the mean efficiency in Treatment A was 49.5 percent and in Treatment B was 32.5 percent. Both of these values are much less than the efficiencies obtained by strategies which predict an attendance randomly drawn from a uniform distribution U(0, 10); a simple simulation shows these efficiencies are 63.6 percent and 58.7 percent, respectively. While the motivation for including this extra information was not to assist the subjects to improve payoffs nor to help play converge to the Nash equilibrium, the results do illustrate the difficulty the subjects faced in this coordination problem.

This experiment has shown that subjects did appear to imitate successful strategies when playing a finitely repeated Bar Game in a laboratory setting. While the evidence of this result is somewhat limited, it does suggest that subjects did not behave in accordance with strict rationality. Widespread imitation behavior in this game can be dangerous as it can cause peoples' actions to become overly homogeneous, which would

[13] Payoff efficiency measures the percentage of potential payoff above the minimum payoff that the players receive and is equal to (actual joint payoff – minimum joint payoff)/(maximum joint payoff – minimum joint payoff) (Bednar et al., 2010).

result in poor outcomes. Despite this concern, there was evidence of imitation behavior in both treatments.

1.5. Conclusion

This experiment has collected a sampling of actual strategies that human subjects used when faced with a difficult coordination problem. Rather than specifying a functional form to which subjects' strategies had to conform, this project sought to record their behavior in a less restrictive environment. This sample provides insight into subject behavior and displays its heterogeneous nature. The experiment has also provided suggestive evidence that subjects do use imitation behavior as a form of learning in this type of game. Despite employing this learning behavior, subjects did not improve collective outcomes throughout the experiment.

Besides providing support for imitation as an observed form of learning in coordination settings, perhaps the strongest contribution of this present work is its novel hybridization of traditional laboratory experiments involving human subjects with computational simulations. Inspired by Axelrod's (1984) tournaments, this project employed accepted experimental controls that his study lacked. The combination of methods allows the testing of theories in procedurally difficult situations. Further, the unrestrictive manner in which the strategy method was employed provided a descriptive catalog of real people's strategic decisions. Necessarily, each of these implementation choices involved some drawbacks.

This methodology contributes to both the complex systems and the experimental economics fields by providing a fresh approach one can take in the design of a research project. Intertwining the dual approaches of studying human subjects in laboratory settings and computational agents in simulation settings allows each technique to complement the other. The incorporation of agent-based computational methods assists in the design of more complex laboratory experiments. Concurrently, the inclusion of living, breathing "agents" allows the validation of computational algorithms.

Table 1.1: Generic payoff structure.

	Not Crowded	Crowded
Go	$\overline{G}$	$\underline{G}$
Stay	S	S

with $\overline{G}$ > S > $\underline{G}$

Table 1.2: Treatment payoffs.

Table 1.2.a: Treatment A.

	Not Crowded	Crowded
Go	2	0
Stay	1	1

Table 1.2.b: Treatment B.

	Not Crowded	Crowded
Go	4	0
Stay	1	1

Table 1.3: Experimental parameters.

Table 1.3.a: Parameters common across sessions.

Number of Sessions	8	
Trial Group Size for Bar Game (N)	10	
Capacity (c)	6	
Rounds of Bar Game per Trial	100	
Payoffs	Treatment Group A	Treatment Group B
"Go" & not crowded	2	4
"Go" & crowded	0	0
"Stay"	1	1

Table 1.3.b: Treatment A sessions.

Session	1	2	3	4	5	6	7	8
Participants	38	37	38	35	35	38	37	37
Trials	3800	3700	3800	3500	3500	3800	3700	3700

Table 1.3.c: Treatment B sessions.

Session	1	2	3	4	5	6	7	8
Participants	49	43	43	44	29	40	40	40
Trials	4900	4300	4300	4400	2900	4000	4000	4000

Table 1.4: Mean attendance and payoffs.

Session	Treatment A			Treatment B		
	Attendance	Payoffs	Payoff Efficiency	Attendance	Payoffs	Payoff Efficiency
1	3.98 (0.937)	1.35 (0.328)	67.5%	6.11 (1.08)	1.67 (0.445)	41.8%
2	6.25 (1.67)	0.87 (0.136)	43.5%	6.85 (0.813)	1.13 (0.103)	28.3%
3	5.52 (0.953)	1.19 (0.178)	59.5%	7.33 (0.760)	0.73 (0.146)	18.3%
4	6.16 (0.969)	1.05 (0.0809)	52.5%	5.59 (1.35)	1.80 (0.571)	45.0%
5	6.39 (0.750)	1.02 (0.0598)	51.0%	6.74 (1.04)	1.22 (0.240)	30.5%
6	6.89 (0.744)	0.70 (0.189)	35.0%	7.38 (0.844)	0.77 (0.111)	19.3%
7	6.02 (1.09)	1.07 (0.148)	53.5%	5.66 (0.713)	1.99 (0.734)	49.8%
8	6.99 (0.744)	0.66 (0.179)	33.0%	6.94 (0.798)	1.07 (0.100)	26.8%
Mean	6.03 (0.950)	0.99 (0.235)	49.5%	6.58 (0.704)	1.30 (0.472)	32.5%

Standard deviations are listed in parentheses.

Table 1.5.a: Number of strategies, payoff means and standard deviations for each category, treatment A.

			Session							
Strategy Categories			**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**
1	"Always" Go	n	1	7	9	12	12	15	11	15
		μ	1.89	0.97	1.31	1.12	1.04	0.63	1.22	0.53
		σ	***0.000***	***0.012***	**0.151**	***0.011***	***0.016***	**0.153**	***0.011***	***0.017***
2	Alternating or Streaking	n	10	15	11	8	8	10	10	9
		μ	1.41	0.93	1.12	0.95	0.97	0.69	0.98	0.64
		σ	***0.147***	**0.075**	***0.091***	**0.051**	0.074	**0.128**	**0.106**	**0.116**
3	Condition on Previous Attendance: If $A_{t-1} < x$, then Go	n	11	11	4	3	3	3	7	3
		μ	1.67	0.67	1.31	1.01	0.99	0.93	0.86	0.92
		σ	**0.271**	***0.051***	**0.107**	**0.051**	***0.018***	**0.113**	***0.061***	***0.077***
4	Condition on Payoff History	n	2	2	9	4	4	5	3	3
		μ	1.21	1.00	1.11	1.07	1.07	0.76	1.11	0.80
		σ	**0.287**	***0.015***	0.233	**0.073**	0.092	0.287	**0.103**	0.307
5	Condition on Some Attendance Average	n	0	0	1	5	6	4	5	3
		μ	-	-	1.48	1.08	1.04	0.89	1.07	0.89
		σ	-	-	-	0.086	0.071	0.275	**0.121**	**0.135**
6	Condition on Previous Attendance: If $A_{t-1} > x$, then Go	n								
		μ	1.01	-	0.99	0.95	0.93	0.47	1.23	0.62
		σ	***0.008***	-	-	**0.076**	-	-	-	0.202
7	"Always" Stay	n	8	0	3	2	0	0	1	2
		μ	1.00	-	1.00	1.00	-	-	1.00	1.00
		σ	***0.003***	-	***0.003***	***0.005***	-	-	-	***0.004***
8	Trigger-like	n	2	3	4	3	2	1	0	0
		μ	1.65	0.98	1.22	1.03	1.00	0.82	-	-
		σ	**0.329**	***0.018***	**0.164**	***0.027***	**0.047**	-	-	-
9	Mixed	n	0	0	1	2	1	1	0	1
		μ	-	-	1.35	1.03	1.01	0.61	-	0.54
		σ	-	-	-	0.094	-	-	-	-
10	Other	n	1	0	0	0	0	0	2	0
		μ	1	-	-	-	-	-	1.11	-
		σ	-	-	-	-	-	-	**0.134**	-
	All	n	38	37	38	35	35	38	37	37
		μ	1.35	0.87	1.19	1.05	1.02	0.70	1.07	0.66
		σ	0.337	0.143	0.181	0.079	0.063	0.198	0.150	0.188

Notes:

1. The first, second, and third rows for each category indicate: n = number of strategies, μ = their mean payoff, and σ = the sample standard deviation of their payoffs.
2. Those category payoff standard deviations noted in **bold** were less than the standard deviation of all payoffs for the session. Those category payoff standard deviations noted in ***bold italic*** were at least fifty percent less than the standard deviation of all payoffs for the session.

Table 1.5.b: Number of strategies, payoff means and standard deviations for each category, treatment B.

			Session							
	Strategy Categories		**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**
1	"Always" Go	n	6	10	18	4	12	21	14	19
		μ	2.23	1.22	0.64	2.66	1.45	0.71	2.77	1.13
		σ	***0.050***	***0.036***	***0.023***	***0.021***	***0.022***	***0.027***	***0.025***	***0.038***
2	Alternating or Streaking	n	11	11	8	11	5	6	4	6
		μ	1.46	1.05	0.76	1.67	1.15	0.73	1.82	0.97
		σ	**0.377**	**0.093**	**0.096**	**0.471**	0.300	**0.094**	**0.628**	***0.046***
3	Condition on Previous Attendance: If A_{t-1} < x, then Go	n	19	9	6	10	8	5	2	3
		μ	1.64	1.15	0.9	1.65	0.97	0.82	1.09	1.02
		σ	**0.432**	**0.064**	**0.119**	***0.262***	**0.148**	0.120	***0.096***	***0.017***
4	Condition on Payoff History	n	4	5	2	4	3	1	0	1
		μ	1.56	1.1	0.92	1.94	1.24	0.87	0	1.04
		σ	0.579	0.117	**0.118**	0.681	**0.130**	-	-	-
5	Condition on Some Attendance Average	n	3	4	3	4	1	2	6	5
		μ	1.6	1.13	0.97	2.17	1.22	0.94	2.52	1.13
		σ	**0.395**	0.178	0.197	0.592	-	0.238	***0.198***	0.122
6	Condition on Previous Attendance: If A_{t-1} > x, then Go	n	6	6	4	3	1	0	2	1
		μ	1.68	1.12	0.57	2.41	1.02	0	1.28	0.99
		σ	0.545	**0.104**	***0.060***	**0.402**	-	-	**0.429**	-
7	"Always" Stay	n	0	1	1	5	1	3	7	0
		μ	0	1	1	1.01	1	1	1	0
		σ	-	-	-	***0.011***	-	***0.000***	***0.006***	-
8	Trigger-like	n	5	2	1	1	1	1	2	4
		μ	2	1.22	0.83	2.21	1.37	0.87	2.35	1.17
		σ	***0.171***	**0.079**	-	-	-	-	***0.003***	0.134
9	Mixed	n	1	0	3	2	2	4	6	4
		μ	1.22	0	0.73	1.72	1.17	0.79	1.31	0.9
		σ	-	-	***0.071***	0.665	***0.077***	**0.090**	***0.071***	***0.027***
10	Other	n	1	0	0	0	1	0	0	0
		μ	2.15	0	0	0	1.12	0	0	0
		σ	-	-	-	-	-	-	-	-
	All	n	49	43	43	44	29	40	40	40
		μ	1.67	1.13	0.73	1.80	1.22	0.77	1.99	1.07
		σ	0.450	0.106	0.149	0.581	0.233	0.112	0.763	0.106

Notes:

1. The first, second, and third rows for each category indicate: n = number of strategies, μ = their mean payoff, and σ = the sample standard deviation of their payoffs.
2. Those category payoff standard deviations noted in **bold** were less than the standard deviation of all payoffs for the session. Those category payoff standard deviations noted in ***bold italic*** were at least fifty percent less than the standard deviation of all payoffs for the session.

Table 1.5.c: Number of strategies, payoff means and standard deviations for each category, treatment summaries.

Strategy Categories			Treatment A	Treatment B
1	"Always" Go	n	82	104
		μ	0.942	1.349
		σ	0.315	0.755
2	Alternating or Streaking	n	81	62
		μ	0.965	1.214
		σ	**0.242**	**0.470**
3	Condition on Previous Attendance: If $A_{t-1} < x$, then Go	n	45	62
		μ	1.079	1.299
		σ	0.405	**0.428**
4	Condition on Payoff History	n	32	20
		μ	1.016	1.349
		σ	**0.239**	**0.518**
5	Condition on Some Attendance Average	n	24	28
		μ	1.029	1.599
		σ	**0.179**	0.678
6	Condition on Previous Attendance: If $A_{t-1} > x$, then Go	n	13	23
		μ	0.908	1.343
		σ	**0.218**	0.642
7	"Always" Stay	n	16	18
		μ	1.000	1.003
		σ	***0.004***	***0.007***
8	Trigger-like	n	15	17
		μ	1.135	1.594
		σ	**0.266**	**0.540**
9	Mixed	n	6	22
		μ	0.928	1.082
		σ	0.306	**0.348**
10	Other	n	3	2
		μ	1.073	1.635
		σ	**0.115**	0.734
	All	n	297	331
		μ	0.99	1.31
		σ	0.290	0.597

Notes:

1. The first, second, and third rows for each category indicate: n = number of strategies, μ = their mean payoff, and σ = the sample standard deviation of their payoffs.
2. Those category payoff standard deviations noted in **bold** were less than the standard deviation of all payoffs for the treatment. Those category payoff standard deviations noted in ***bold italic*** were at least fifty percent less than the standard deviation of all payoffs for the treatment.

Table 1.6: Evidence of imitation behavior.

Table 1.6.a: Treatment A.

Session (t)	Best Strategy Category in Session t	Increase in Usage in Session t+1	Percent Increase
1	"Always" Go	+ 6	600%
2	Condition On Payoff History	+ 7	350%
3	Condition On Some Attendance Average	+ 4	400%
4	"Always" Go	+ 0	0%
5	Condition On Payoff History	+ 1	25%
6	Cond. On Previous Attendance: If $A_{t-1} < x$, then Go	+ 4	133%
7*	Cond. On Previous Attendance: If $A_{t-1} > x$, then Go	+ 1	100%
	"Always" Go	+4	36%

Table 1.6.b: Treatment B.

Session (t)	Best Strategy Category in Session t	Increase in Usage in Session t+1	Percent Increase
1	"Always" Go	+ 4	67%
2*	"Always" Go	+ 8	80%
	Trigger-like Strategy	- 1	-100%
3	"Always" Stay	+ 4	400%
4	"Always" Go	+ 8	200%
5	"Always" Go	+ 9	75%
6	"Always" Stay	+ 4	133%
7	"Always" Go	+ 5	38%

* In both Session 7 of Treatment A and Session 2 of Treatment B, there were virtual ties for the best performing strategy categories.

Figure 1.1: Experiment design.

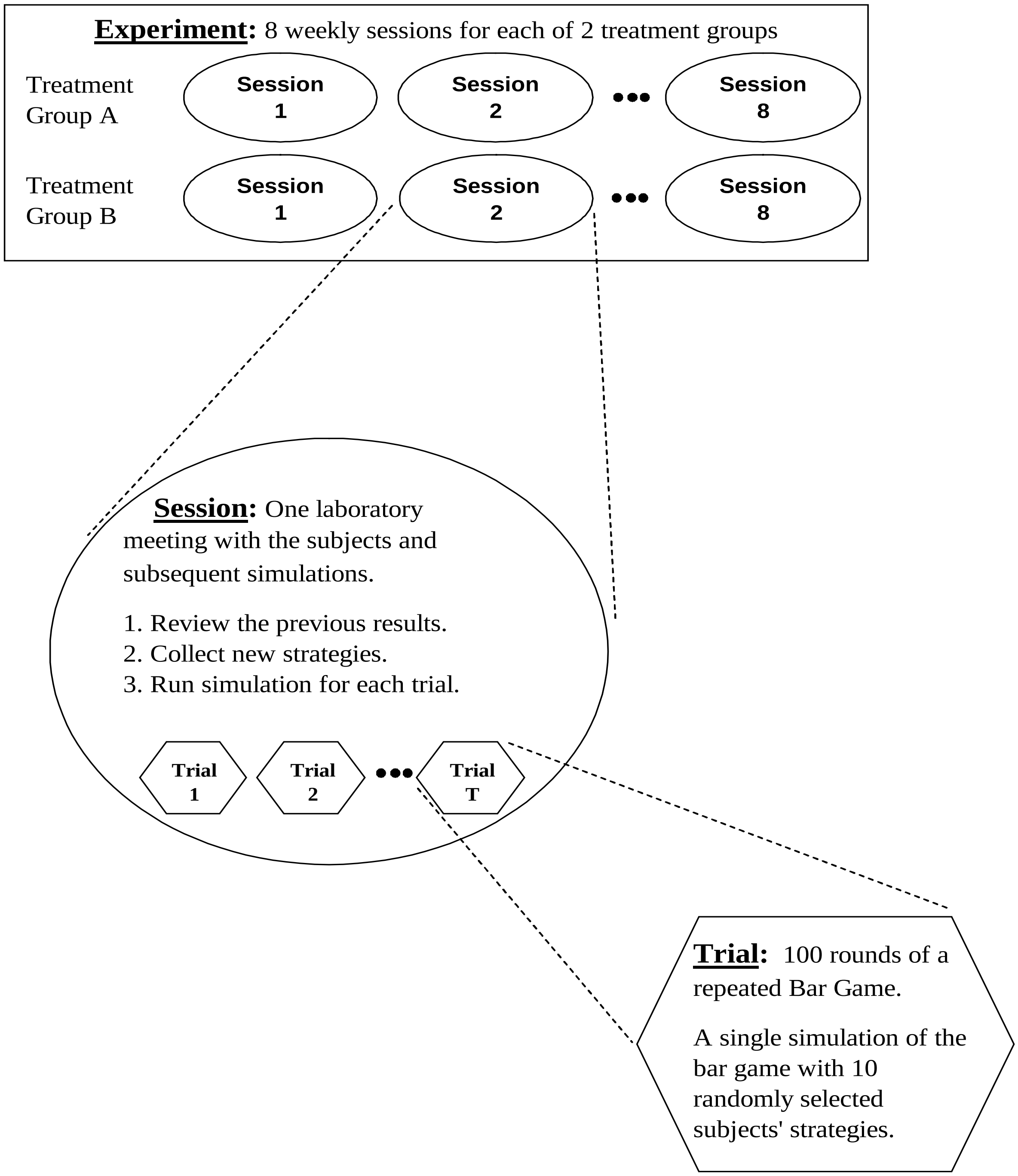


Figure 1.2: Typical individual results as presented to the subjects.

Group A -- Session 5 Individual Results

Payoff: per round is from a possible range of 0 - 2.
%Go: The % of rounds your strategy chose "Go".
%Right: The (% of rounds your strategy chose "Go" and attendance turned out to be 0 - 6) + (% of rounds your strategy chose "Not Go" and attendance turned out to be 7 - 10).

U of M ID	Payoff	%Go	%Right
xxx-xx-xxxx-x	1.185	80.7%	64.1%
xxx-xx-xxxx-x	1.165	49.6%	53.8%
xxx-xx-xxxx-x	1.132	24.9%	46.1%
xxx-xx-xxxx-x	1.063	100.0%	53.2%
xxx-xx-xxxx-x	1.063	99.0%	52.8%
xxx-xx-xxxx-x	1.053	100.0%	52.6%
xxx-xx-xxxx-x	1.050	100.0%	52.5%
xxx-xx-xxxx-x	1.048	99.0%	52.1%
xxx-xx-xxxx-x	1.047	95.0%	50.9%
xxx-xx-xxxx-x	1.044	89.0%	49.6%
xxx-xx-xxxx-x	1.044	100.0%	52.2%
xxx-xx-xxxx-x	1.042	99.0%	51.8%
xxx-xx-xxxx-x	1.038	100.0%	51.9%
xxx-xx-xxxx-x	1.037	100.0%	51.8%
xxx-xx-xxxx-x	1.032	100.0%	51.6%
xxx-xx-xxxx-x	1.032	67.7%	44.7%
xxx-xx-xxxx-x	1.021	21.8%	37.8%
xxx-xx-xxxx-x	1.021	22.0%	37.8%
xxx-xx-xxxx-x	1.017	23.2%	35.3%
xxx-xx-xxxx-x	1.015	90.0%	47.3%
xxx-xx-xxxx-x	1.012	56.8%	44.7%
xxx-xx-xxxx-x	1.009	79.0%	45.7%
xxx-xx-xxxx-x	1.006	10.0%	32.8%
xxx-xx-xxxx-x	1.004	21.2%	35.5%
xxx-xx-xxxx-x	1.001	100.0%	50.0%
xxx-xx-xxxx-x	0.998	0.3%	31.8%
xxx-xx-xxxx-x	0.983	3.3%	30.6%
xxx-xx-xxxx-x	0.980	3.7%	31.7%
xxx-xx-xxxx-x	0.979	76.0%	41.5%
xxx-xx-xxxx-x	0.965	43.4%	34.5%
xxx-xx-xxxx-x	0.946	51.0%	34.8%
xxx-xx-xxxx-x	0.945	75.0%	39.5%
xxx-xx-xxxx-x	0.935	10.8%	25.9%
xxx-xx-xxxx-x	0.819	49.0%	24.9%

(ID's are masked in this paper for privacy)

Figure 1.3: Treatment group B results as presented during the experiment in session 6.

Group B -- Session 5 Results

	Session 1	Session 2	Session 3	Session 4	Session 5
Average **Payoff** (per round):	1.67	1.13	0.73	1.80	1.22
Average **Attendance** (per round):	6.11	6.85	7.33	5.59	6.74

Strategy categories *Examples in italics*	Session 1		Session 2		Session 3		Session 4		Session 5	
	Times Used	Avg Payoff	Times Used	Avg Payoff	Times Used	Avg Payoff	Times Used	Avg Payoff	Times Used	Avg Payoff
Condition on Previous Attendance: If FEW went, then Go *Ex: If previous attendance < x then "Go"; otherwise "Not Go".*	19	1.64	9	1.15	6	0.90	10	1.65	8	0.97
Condition on Previous Attendance: If MANY went, then Go Ex: *If previous attendance > x then "Go"; otherwise "Not Go"*	6	1.68	6	1.12	4	0.57	3	2.41	1	1.02
Trigger Strategy Ex: *After x rounds, if my payoff > y, then always "Go" thereafter; otherwise always "Not Go".*	5	2.00	2	1.22	1	0.83	1	2.21	1	1.37
Alternating or Streaking Strategies Ex: *"Go" every nTH round, "Not Go" all other rounds.* Ex: *Rounds 1 to x: "Go"; rounds x+1 to y: "Not Go"; rounds y+1 to 100: "Go".*	11	1.46	11	1.05	8	0.76	11	1.67	5	1.15
Always "Go" (at least 95% of the time).	6	2.23	10	1.22	18	0.64	4	2.66	12	1.45
Always "Not Go" (at least 95% of the time).	0	N/A	1	1.00	1	1.00	5	1.01	1	1.00
Condition Choice on Previous Payoff(s) Ex: *If I'm averaging less than x points per round, then "Go"; otherwise "Not Go".*	4	1.56	5	1.10	2	0.92	4	1.94	3	1.24
Condition Choice on Some Attendance Average. Ex: *If the average attendance of all previous rounds < x, then "Go"; otherwise "Not Go".*	3	1.60	4	1.13	3	0.97	4	2.17	1	1.22
Randomized Strategy Ex: *Choose "Go" with probability p; otherwise "Not Go".*	1	1.22	0	N/A	3	0.73	2	1.72	2	1.17

Figure 1.4: Average attendance.

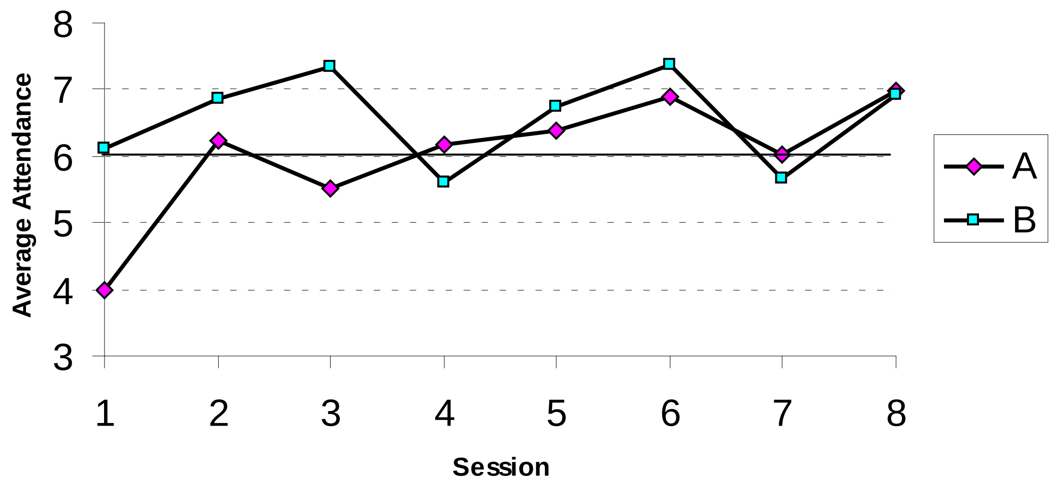
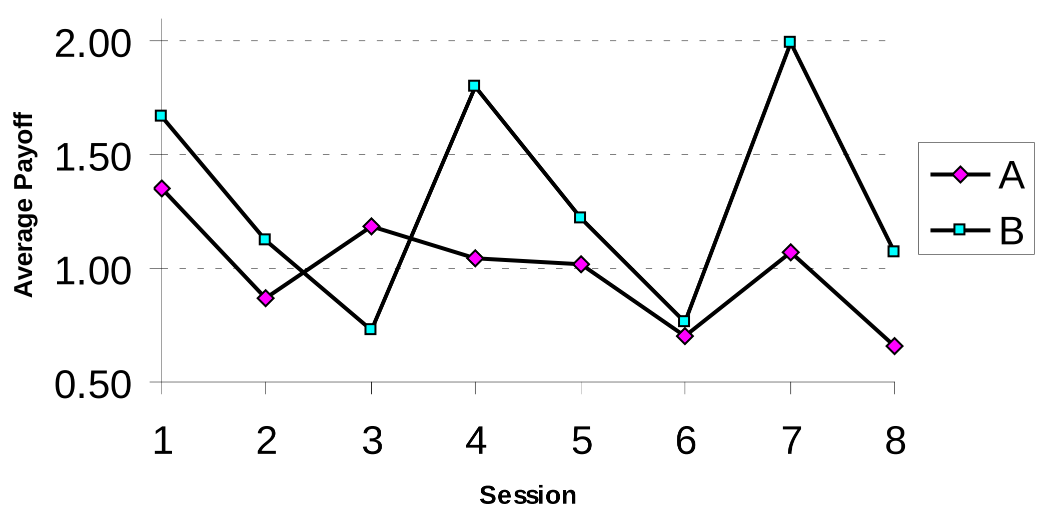


Figure 1.5: Average payoffs.



Bibliography

Arthur, W. B. 1994. "Inductive Reasoning and Bounded Rationality." *American Economic Review*, 84(2): 406-411.

Axelrod, R. 1984. *The Evolution of Cooperation*. New York: Basic Books.

Axelrod, R. 1996. "An Evolutionary Approach to Norms." *American Political Science Review*, 80(4): 222-238.

Bednar, J., Chen, Y., Liu, T. X., Page, S. 2010. "Behavioral Spillovers and Cognitive Load in Multiple Games." http://yanchen.people.si.umich.edu/papers/Multiple_Games_20101101.pdf.

Bell, A. M., Sethares, W. A., Bucklew, J. A. 1999. "Coordination Failure as a Source of Congestion in Information Networks." Mimeo available at http://ic.arc.nasa.gov/ic/people/bell/bell.html.

Bell, A. M., Sethares, W. A., Licking-Reiley, D., Wolpert, D., Tumer, K., Frank, J. 1999. "Strategic Behavior, Learning and Efficient Allocation of Network Resources." Presented at the International Congress on Networks, Groups and Coalitions, Menresa, Spain, May 1999. Mimeo available at http://ic.arc.nasa.gov/ic/people/bell/bell.html.

Berra, Y., Horton, T. 1989. *Yogi—It Ain't Over…* New York: McGraw Hill Publishing Company.

Brandts, J., Charness, G. 2000. "Hot vs. Cold: Sequential Responses and Preference Stability in Experimental Games." *Experimental Economics*, 2(3): 227-38.

Challet, D., Zhang, Y.-C. 1997. "Emergence of Cooperation and Organization in an Evolutionary Game." *Physica-A*, 246(3-4): 407-18.

Duffy, J., Ochs, J. 1999. "Emergence of Money as a Medium of Exchange: An Experimental Study." *American Economic Review*, 89(4): 847-877.

Friedman, D., Sunder S. 1994. *Experimental Methods: A Primer for Economists*. Cambridge, UK: Cambridge University Press.

Greenwald, A., Mishra, B., Parikh, R. 1998. "The Santa Fe bar problem revisited: theoretical and practical implications." *Proceedings of the Festival on Game Theory: Interactive Dynamics and Learning*, SUNY Stony Brook, July 1998.

Holland, J. H., Miller, J. H. 1991. "Artificial Adaptive Agents in Economic Theory." *American Economic Association Papers and Proceedings*, 81(2): 365-370.

Isaac, R. M., Walker, J. M., Williams, A. W. 1994. "Group Size and the Voluntary Provision of Public Goods: Experimental Evidence Utilizing Large Group Sizes." *Journal of Public Economics*, 54: 1-36.

Johnson, N. F., Jarvis, S., Jonson, R., Cheung, P., Kwong, Y. R., Hui, P. M. 1998. "Volatility and agent adaptability in a self-organizing market." *Physica-A*, 258(1-2): 230-6.

Kormendi, R., Plott, C. 1982. "Committee Decisions Under Alternative Procedural Rules: An Experimental Study Applying a New Nonmonetary Method of Preference Inducement." *Journal of Economic Behavior and Organization*, 3: 175–95.

Li, Y., Riolo, R., Savit, R. 1999a. "Evolution in Minority Games I. Games with a Fixed Strategy Space." Center for the Study of Complex Systems (University of Michigan) Technical Report PSCS-99-001.

Li, Y., Riolo, R., Savit, R. 1999b. "Evolution in Minority Games II. Games with a Variable Strategy Space." Center for the Study of Complex Systems (University of Michigan) Technical Report PSCS-99-003.

Manuca, R., Li, Y., Riolo, R., Savit, R. 1998. "The Structure of Adaptive Competition in Minority Games." Center for the Study of Complex Systems (University of Michigan) Technical Report PSCS-98-11-001.

Marimon, R., Spear, S., Sunder, S. 1993. "Expectationally-Driven Market Volatility: Experimental Evidence." *Journal of Economic Theory*, 61: 74-103.

Meyer, D. J., Van Huyck, J. B., Battalio, R. C., Saving, T. R. 1992. "History's Role in Coordinating Decentralized Allocation Decisions." *Journal of Political Economy*, 100(2): 292-316.

Mitzkewitz, M., Nagel, R. 1993. "Envy, Greed and Anticipation in Ultimatum Games with Incomplete Information: An Experimental Study." Discussion paper B–181. March. Universitat Bonn. *International Journal of Game Theory*, 22: 171–98.

Ochs, J. 1995. "Coordination Problems." In *The Handbook of Experimental Economics*, eds. J. H. Hagel & A. E. Roth,195-251. Princeton, NJ: Princeton University Press.

Roth, A. 1995. "Bargaining Experiments." In *The Handbook of Experimental Economics*, eds. J. H. Hagel & A. E. Roth, 253-348. Princeton, NJ: Princeton University Press.

Rubinstein, A. 1998. *Modeling Bounded Rationality*. Cambridge, MA: MIT Press.

Savit, R., Manuca, R., Riolo, R. 1997. "Adaptive Competition, Market Efficiency, Phase Transitions, and Spin Glasses." Center for the Study of Complex Systems (University of Michigan) Technical Report PSCS-97-12-001.

Selton, R. 1967. *Die Strategiemethode Zur Erforschung Des Eigeschrankt Rationalen Verhaltens Im Rahmen Eines Oligopolexperiments*." In *Beitrage Zur Experimentellen Wirt Schaftsforschung*, ed. H. Sauermann, H., 136-68. Tubingen: J.C.B. Mohr.

Van Huyck, J. B., Battalio, R. C., Beil, R. O. 1990. Tacit Coordination Games, Strategic Uncertainty, and Coordination Failure. *American Economic Review*, 80(1): 234-252.

Van Huyck, J. B., Cook, J. P., Battalio, R. C. 1994. Selection Dynamics, Asymptotic Stability, and Adaptive Behavior. *Journal of Political Economy*, 102 (5): 975-1005.

Zambrano, Eduardo. 2004. The Interplay Between Analytics and Computation in the Study of Congestion Externalities: The Case of the El Farol Problem. *Journal of Public Economic Theory*, 6 (2): 375–395.

Chapter 2

A Framework to Investigate Multiple Game Environments Using Computer Experiments

2.1. Introduction

This study develops a framework that employs artificial adaptive agents in computer experiments to investigate the effects of multiple game environments on the performance and structure of finite automata strategies. Multiple game environments encapsulate strategic situations in which agents compete on varied fronts. Variations of multiple game environments have been studied to explain the emergence of culture (Bednar and Page, 2007), social systems (Miller and Page, 2007), decision making in the presence of reasoning constraints (Samuelson, 2001), and decision making under uncertainty (Gilboa and Schmeidler, 2001). This framework facilitates an understanding of how the multifaceted strategic situation affects the qualities of strategies.

This modeling technique using computer experiments allows an analysis of optimizing behavior without the rigidity required for a formal mathematical model (Holland and Miller, 1991). This approach also has the advantage that one may study more complex and difficult situations than with a strictly mathematical model. Another benefit is the potential to observe emergent behavior.

This paper presents the conceptual and procedural design of the experimental framework and summarizes the technical implementation issues. A brief overview of the experimental approach follows. Discussion subsequently addresses games, strategies, statistics, dynamics, algorithms, and applications.

Experimental Approach

To clarify the language in this paper, an experiment consists of numerous trials, each with a specific environment defined by the set of simultaneously played games and the cognitive cost function. Figure 2.1 illustrates the components of a trial. Each trial consists of a number of independent runs. Each run continues for a number of generations. During each generation, evolutionary dynamics are applied separately to each player's stock of strategies, and then the players' automata match to play every supergame within the defined environment. Each supergame is a repeated 2 x 2 normal form game.

A player or agent in this experimental framework possesses a pool of available strategies. Each strategy is a heuristic that directs how the agent will play against an opponent. The framework allows this pool of strategies to evolve over time in two ways. First, it applies a selection process so that relatively successful strategies will replicate and replace less successful strategies. Second, it applies a genetic algorithm that creates new strategies by combining different parts of existing strategies, similar to chromosomal crossover in a cellular nucleus. These dynamic processes will over time adapt a player's strategy pool to the particular environment in which it is competing. The framework reports a rich set of statistics that measure the performance and describe the structure of the strategies. Thus, a researcher may gain insight into which features of the game-cost environment are driving the changes in the performance and structural statistics as the strategy pool evolves. Chapter 3 will incorporate strategy evolution to pursue these insights.

The framework capitalizes on evolved strategy pools by implementing a facility to store and retrieve experienced strategies. This feature allows a researcher to match agents with distinct histories against each other. In this mode of operation, the framework allows the adaptive processes to be deactivated to focus attention on the role history plays in the multiple game environments. Chapter 4 will utilize this feature to consider the influence of experience.

2.2. Games

Game Selection

This purpose of this framework is to facility the investigation of multiple game environments. It is primarily designed for simple 2 x 2 matrix games because these games are individually well understood[1].

One goal of this research is to compare the performance of strategies in different single game and multiple game environments. To this end, it would be convenient to have a way to measure the similarity between two environments. Rapoport, et al. (1976), Kilgour and Fraser (1988), Walliser (1988), and Parisi (2000) all develop game taxonomies which could assist in the development of a dissimilarity measure. This project incorporates the Rapoport, et al. (1976) taxonomy that restricts the set of games to all ordinally distinct, finitely repeated, 2 x 2 games. These games are characterized by each player having payoffs that include exactly one each of “1,” “2,” “3,” and “4” in the four cells of the payoff matrix[2]. Bednar, et al. (2010) develop an alternative method to compare different games by defining the entropy of outcomes as a measure of behavioral variation in a normal form game.

The framework focuses on these games because the taxonomy allows the similarity between them to be measured precisely. The Rapoport, et al. (1976) taxonomy

[1] One could extend the framework to investigate other types of games with little additional programming.

[2] The computer framework is not restricted to ordinally distinct games and will play any 2 x 2 matrix games. For instance, widening the scope to all ordinal 2 x 2 games, including those not strictly ordinal, would require no additional programming (taxonomized by Kilgour and Fraser (1988)).

contains seventy-eight strategically distinct, strictly ordinal games[3]. Table 2.1 provides this taxonomy in concise form and briefly describes the subdivisions of the taxonomy.

Employing this taxonomy, one can measure the dissimilarity between two games by comparing their phyla, classes, orders, genera, and species. The phyla division, for instance, classifies games as either "no conflict," "complete opposition," or "mixed motive." The measure scores two identical games with a dissimilarity of zero and becomes larger as two games become more dissimilar along the branches of the taxonomy. Because this list of subdivisions becomes increasingly more refined, the measure weights more heavily differences in the more general subdivisions than the more specific subdivisions. Specifically, differences in phyla add five to the dissimilarity measure, differences in classes add four, differences in orders add three, differences in genera add two, and differences in species add one. The sum of all these differences gives the dissimilarity measure.[4] For comparison, the average dissimilarity of any two randomly created games is 8.0.

Other Features

The simulation framework allows the user to control additional features related to the play of games. The features expand the applicability of the framework for further research projects. The features, set by parameters in the input file, include: noisy selection of actions, time discounting of payoffs, and indefinite length games.

---

[3] There are 576 possible numeric configurations using these payoffs. However, eliminating strategically equivalent games reduces the set to 78. Walliser (1988) provides an alternative taxonomy of the 78 games.

[4] Using Rapoport, et al. (1976) notation, if games are both members of phylum M, class E, the dissimilarity measure weighs the subclass (P or p) the same as class. If games are both members of phylum M, class e, order D0, the measure weighs the suborder (e0 or e2) the same as order. The measure ignores competitive pressure since this is irrelevant to automated agents. For games that match one of two genera (f to ft, for example), the measure weighs the genus difference one-half as much as a complete mismatch (i.e., 1 instead of 2).

In addition to specifying which games will be played in an input file, the user can also opt to match players in randomly generated games. Because these could be asymmetric, the user can direct that two matched automata play every game twice, with each player acting in both the row and column roles.

The framework supports two matching paradigms, mean and random matching. Mean matching sweeps all possible pairings of the two players' strategies and games. Each of the row player's automata plays every game with each of the column player's automata. As its name suggests, random matching stochastically selects the automata and game. The program selects each player's automaton from a uniform distribution, and selects the game based on specified probability parameters[5]. The two automata then play the selected game for a set number of rounds. This matching process repeats until it reaches the user-determined number of pairings.

2.3. Strategies

This framework implements strategies as finite automata. Also known as Moore machines, the economics literature contains extensive references to these structures (for example, Aumann (1981) and Miller (1996)). Conceptually, each state of a finite automaton represents a portion of agent *n*'s strategy by specifying the action to take when in the state and the next state to which the strategy should transition after it executes the action. The current state of an automaton is called its *active* state; likewise, an automaton *activates* a state when it plays its prescribed action.

Consider a game environment *G*, composed of a set of $\overline{g}$ 2 x 2 games. Formally, a finite automaton *i* of agent *n* is a machine $M_i = (S_i, S_i^0, \theta_i, f_i, \tau_i)$ where

- S_i is the set of states of size $\overline{s}_n$.
- $S_i^0 \subseteq S_i$ is the set the initial states of size not greater than $\overline{g}$.

[5] With $\overline{g}$ randomly created games, each game is assigned a probability of play equal to $1/(\overline{g})^{-1}$

- $\theta_i(\cdot)$ is a mapping $\theta_i\colon G \mapsto S_i^0$. This initialization function assigns a starting state $s_{i,g}^0$ dependent on the game to be played.
- $f_i(\cdot)$ is a mapping $f_i\colon S_i \mapsto A_n$. This action function prescribes the action $a_n \in A_n$ agent n will play at state s_i.
- $\tau_i(\cdot)$ is a mapping $\tau_i\colon S_i \times A_m \mapsto S_i$. This transition function determines to which state the automaton transitions dependent on the action played by the opponent, agent *m*.

Thus, an automaton[6] corresponds to the conventional meaning of a strategy as a complete description of agent *n*'s plan of action in all possible circumstances that are consistent with agent *n*'s plans. An automaton implementation of a strategy differs subtly from the game theoretic notion of a strategy. The latter requires a complete plan of action for all possible situations, including those not consistent with agent *n*'s plan of action (Rubenstein, 1998, p. 144).

As a simple example, consider the four-state automaton in a two-game environment (Prisoner's Dilemma and Chicken) in Figure 2.2.a. First, note that a complete description of an automaton specifies for each state an assigned action (listed inside of the circle as "0" or "1") and two transition arrows (the upper for an opponent's action of "0" and the lower for an opponent's action of "1"). It also specifies for each game the initial state in which to begin (identified by the thickened arrows). Also, two automata can represent the identical strategy when a they only differ by a reordering of the states; the two automata represented in Figures 2.2.d and 2.2.e provide such an example.

The present approach imposes some symmetry restrictions on the strategies. Players *n* and *m* will have a common number of strategies $(\bar{n} = \bar{m})$, common number of

[6] Hereafter, "automaton" will mean "finite automaton." Another related type of automaton is the cellular automaton (see Wolfram, 2002).

states per strategy $\left(\bar{s}_n = \bar{s}_m = \bar{s}\right)$, and a common set of actions. Attention on 2 x 2 games defines the action space as $A_n = A_m = A = \{0,1\}$[7]. In addition, the particular game being played determines only the initial state in which the automaton begins. Once play commences, an automaton executes actions and transitions independent of the game; actions depend only on the current state and transitions from that state depend only on the opponent's action. Thus, agents play the games simultaneously in the sense that an agent uses a given automaton to play all the games it faces. This automaton remains fixed structurally until it has played all relevant games. It may then be subjected to an evolutionary process which modifies its structure.

This approach, which follows Miller (1996), allows both specificity and reusability in the application of states. For example, Figure 2.2.a shows a four-state strategy that plays two games as if it were two independent strategies. States 1 and 2 are used exclusively for Prisoner's Dilemma, and states 3 and 4 are used exclusively for Chicken. This dichotomous approach could conceivably allow a strategy to target each game with a sub-strategy that works particularly well in that game, but perhaps not so well in others. This specialization may come at a cost, however, if cognitive resources are constrained.

A strategy can economize on these costs by emphasizing reusability, in which it uses a subset of states to play multiple games. Figure 2.2.b provides such an example. Consider this automaton paired with another that always alternates between actions 0 and 1. The automaton in Figure 2.2.b uses state 1 exclusively for Prisoner's Dilemma, it uses state 3 exclusively for Chicken, and it potentially uses states 2 and 4 for both games. Because the transition function, τ_i, is deterministic and not dependent on the game being played, once the path of states played for two different games coincide on the same state

---

[7] The actions (0, 1) correspond to (up, down) for the row player and (left, right) for the column player. The 2 x 2 games employed here are consistent with Rapoport, et al. (1976), and thus the actions have no consistent interpretations as cooperate, defect, etc.

(state 4, for example), the automaton effectively plays the two games similarly for the remaining rounds.

This convergence of paths may limit the development of specificity in automata with just a few states, like those illustrated in Figure 2.2. Increasing an automaton's states enhances its ability to achieve complete or partial state specificity toward games. The number of states can serve as a measure of the complexity of an automaton (Rubenstein, 1998; Samuelson, 2001; Bednar and Page, 2007)[8]. A more complex automaton will be capable of the more complex behavior of state specificity.

A tradeoff of using a complex automaton with many states is the unwieldy size of the set of possible machines, $\{M_i\}$[9]. Because this set grows exponentially with the number of states, it is difficult to develop tractable theoretic results. However, modeling strategies as automata allows a direct measure of the mental cost of using the strategies.

Cognitive Costs

The strategic environment with both multiple games and cognitive costs puts two opposing pressures on an automaton. The multiple games would tend to advantage to an automaton that uses more states (enabling greater specialization), at least in the case of dissimilar games. When one views states (or their use) as reasoning resources, cognitive costs could advantage the automaton that uses fewer states.

The presence of cognitive costs captures the notion of bounded rationality. There are practical reasons why the complexity of a strategy should relate to its associated cost that go beyond the simple paradigm that its uses more cognitive resources. Rubenstein

---

[8] A weakness of this measure is that an automaton needs $x + 1$ states to account for x periods of memory (Bednar and Page, 2007). Rubenstein (1998) develops another measure of complexity based on the maximal order of its states.

[9] A high-end estimate for an automaton with $\bar{s}$ states, $\bar{a}$ possible actions, and playing $\bar{g}$ games is $\bar{a}^{\bar{s}}\bar{s}^{2\bar{s}+\bar{g}}$, which for sixteen states, two actions, and four games is 2^{160}. However, this is a significant overstatement because many of the configurations are strategically equivalent.

(1998, p. 137) argues that "a more complex plan is more likely to break down, is more difficult to learn, and may require more time to be implemented."

There are a number of conceivable ways in which to model cognitive costs. Bednar, et al. (2010) use a game's empirical entropy to measure of the cognitive load induced by game complexity. This framework, rather, develops four methods (and allows the no cost case) that base cost on the implementation or development of a plan, proxied by activation or accessibility of an automaton's states. States can be viewed as cognitive subroutines that contribute to an automaton's overall strategy as well as its cognitive cost. The following paragraphs briefly describe the motivation for each cost specification; the section on statistics explains the specifications of these costs.

Under the first two cost options, it is viewed as costly to maintain and/or utilize a stock of *active* components (that is, states that are actually used or visited). With these two cost measures, cost is increased if a state is activated when the automaton plays the games. Both are based on a binary state indicator function: if a state is used one or more times, then cost increases. They differ on whether the indicator function applies at the single game level or at the game ensemble level.

The third cost option also considers state activation, but departs from the binary indicator function and instead relies on frequencies of state activation. Briefly, this cost measure looks at how state activation frequencies vary from game to game. It assumes that two games that induce very similar patterns of state activation are less cognitively taxing than two other games that induce drastically different state activation patterns.

The fourth cost option (adopted in Chapters 3 and 4 of this dissertation) is distinct from the other three because it bases cost on the *potential* for a state to be used rather than on its actual usage. Here, it is costly for an automaton to keep a state ready to be used—it is cost to *develop* the plan of action rather than to implement it.

Strategy Pools

Players or agents in this framework possess a set (pool, population) of available strategies that it uses to determine its behavior. This set of strategies is the agent's cognitive resources. The size of agent *n*'s set of strategies is $\bar{n}$. There are different ways to interpret an agent's application of its strategy pool.

One way is to view this pool of strategies as its "cognitive toolkit." When the agent faces a certain strategic situation (that is, a particular game), it could select which tool or strategy would be most likely to bring a desirable outcome and then play that strategy. This interpretation is akin to tagging in classifier systems (Holland, 1986). In a multiple game environment, an automaton strategy could become specialized to play a particular game. It is the pool of strategies, then, that encapsulates the agent's ability to play multiple games.

An alternative view, adopted in this framework, is to view an agent's pool of strategies as its "cognitive test kit." Under this paradigm, an agent typically applies all $\bar{n}$ of its available strategies when facing any strategic situation[10], testing its strategies. Over time it will adjust its pool by increasing the presence of those strategies that perform well and decreasing the presence to those that perform poorly (keeping $\bar{n}$ constant). A strategy pool will usually become more homogenous as it repeatedly tests its strategies and adjusts its pool. The entire set of strategies may then be loosely viewed as player's "super strategy." In this view, the ability to play multiple games lies within the automaton itself; even if the pool converges to a single automaton replicated many times, its inherent ability to play multiple games stems from employing distinct initial states for each game.

---

[10] An agent applies all of its strategies in the mean matching paradigm. With random matching, an agent selects which strategy to apply by stochastically drawing from a uniformly distribution. With a sufficiently large number of applications, these two approaches should yield similar results. Section 2.6 provides more details about the matching process.

2.4. Statistics

The computer simulation program developed for this framework generates several statistics that serve both to drive the dynamics within the model to measure game outcomes and strategy characteristics that are related in textual and graphically output. The statistics fall within three broad classes based on whether they describe payoff matrix outcomes, the strategies' performance, or their structure. The key statistic is profit, because a strategy's relative profit influences its likelihood to propagate into the next generation. Profit and the other principle statistics are described briefly below; descriptions of the remaining statistics are found in Appendix 2.1.

Payoff Matrix Outcome Distributions

The program tracks the outcome of each round of each game played and tallies the number of occurrences for each outcome cell in the payoff matrix. This information is output as a fraction of all rounds that play resulted in each payoff matrix cell. For example, 0.50 for the (0, 0) ~ (Up, Left) payoff cell, 0.30 for the (0, 1) ~ (Up, Right) payoff cell, 0.15 for the (1, 0) ~ (Down, Left) payoff cell, and 0.05 for the (1, 1) ~ (Down, Right) payoff cell. This information is outputted at the individual automaton level and at the population aggregate level.

Performance Statistics

Performance statistics capture outcomes of an automaton's play of the set of games. Profit, the main performance statistic, is determined by the difference of two others—average score and total cost.

Score

The program records scores that the automaton strategy earns for each game it plays. The score for each game represents the outcomes against all opponents and over all rounds. The score is expressed as an average per round so that it is comparable to the relevant game payoff matrix.

The program also reports the mean score the automaton strategy earns across all games it plays. The mean score weights each individual game score by the number times the strategy plays it.

For an example, suppose an automaton strategy plays game 1 for six rounds and game 2 for two rounds. It earns payoffs of 1, 2, 3, 4, 4, and 4 in game 1 and earns payoffs of 1 and 3 in game 2. The automaton's reported score for game 1 is 3.0, its score for game 2 is 2.0, and its mean score is 2.75.

Costs

The program calculates an automaton's cost for its play in up to four different ways. Parameters determine which of the four costs the program actually calculates. The program aggregates the separate costs into a total cost measure if more than one type of cost is calculated.

The user can opt to normalize the magnitude of the costs. After the first generation this procedure calculates either a multiplicative or additive factor that equates the average cost incurred across the population of automata strategies to the average score the population earned. Thus, this cost normalization procedure makes the population's mean first round profit—score minus total cost—equal to zero. This factor is then applied to costs incurred in all following generations.[11]

Specifically, with multiplicative cost adjustment, the program calculates a normalization factor $\eta^{\times}$ equal to the strategy population's mean score divided by its mean unadjusted total cost in the first generation. The program then multiplies all total cost measures (in the first and all future generations) by $\eta^{\times}$ to arrive at the normalized total cost statistics. A similar procedure occurs with additive cost adjustment. Here, the

[11] The adjustment procedure can be applied to the row and column populations separately or collectively.

normalization factor η^+ equals the strategy population's mean score minus its mean unadjusted total cost in the first generation. The program then adds η^+ to all total cost measures in all generations to arrive at the normalized total cost statistics.

The four separate measures of cost, explained below, are formulaically combined to construct total cost (before normalization):

$$\text{total cost} = \alpha_0 + \underbrace{\alpha_1 \left(1 + C_1\right)^{\beta_1}}_{\text{Cost 1}} + \underbrace{\frac{1}{\bar{g}} \sum_{g=1}^{\bar{g}} \alpha_2 \left(1 + C_{2,g}\right)^{\beta_2}}_{\text{Cost 2}} + \underbrace{\alpha_3 \left(1 + C_3\right)^{\beta_3}}_{\text{Cost 3}} + \underbrace{\alpha_4 \left(1 + C_4\right)^{\beta_4}}_{\text{Cost 4}} \quad (2.1)$$

The α_i parameter indicates the extent that the i^{th} type of cost is included in total cost. To focus attention on a single type of cost, all but one α_i parameters are set to zero. The calculated statistics C_1, $C_{2,g}$, C_3, and C_4 are based on, respectively, the number of distinct states activated during the play of *all* games, the number of distinct states activated during play of game g, the similarity of state activation frequencies across all games played, and the number of accessible states across all games. These measures are in turn explained more fully below.

Cost 1 and Distinct States Activated in All Games

The first method to model costs (hereafter, "Cost 1") considers it costly to *maintain* a stock of component procedures. This correlates to relating cost to the number of distinct states activated during the play of *all* games. If an agent ever activates a state, it must spend "cognitive capital" to maintain it in its stock of applicable procedures. The model calculates Cost 1 as a function of distinct states activated (*DSA*) to play all games expressed as a fraction of all $\bar{s}$ states.

As an example, consider an eight-state strategy that plays two games. It activates only states 1, 2, and 3 to play Prisoner's Dilemma, it activates only states 2, 3, and 5 to play Chicken, and it never activates states 4, 6, 7, and 8. Its cognitive cost is based on the use of fifty percent (four of eight) of its states. Specifically, letting g index games ($1\ldots\bar{g}$) and s index states ($1\ldots\bar{s}$), the distinct states activated and Cost 1 are:

$$\text{Cost } 1 = \alpha_1 \left(1 + C_1\right)^{\beta_1} \tag{2.2}$$

where

$$C_1 = \frac{DSA}{\bar{s}} = \frac{\sum_{s=1}^{\bar{s}} I\left(s\right)}{\bar{s}}, \text{where } I\left(s\right) = \begin{cases} 1 & \text{if the agent activates state } s \\ & \text{during the play of } \textit{any} \text{ game,} \\ 0 & \text{otherwise} \end{cases} \tag{2.3}$$

Thus, with four distinct states activated, $\alpha_1 = 1$, and $\beta_1 = 2$, Cost 1 for this example is $(1 + 4/8)^2 \cong 2.25$.

Cost 2 and Distinct States Activated Per Game

The second method to model costs ("Cost 2") considers it costly to *utilize* a component procedure. In this light, cognitive costs have some similarity to capacity constraints in the language comprehension literature in which working memory can be construed to include both storage *and operational* functions (Just and Carpenter, 2002). Thus, a strategy's cost increases as the average state activation rate per game played (again expressed as a fraction of all $\bar{s}$ states) increases.

Continuing the preceding example, the automaton activated three of eight states to play Prisoner's Dilemma, and it activated three of eight states to play Chicken, so the mean states activated per game is 3.0. Specifically, Cost 2 for a given game g is

$$\left(\text{Cost } 2\right)_g = \alpha_2 \left(1 + C_{2,g}\right)^{\beta_2} \tag{2.4}$$

where

$$C_{2,g} = \frac{\left(DSA\right)_g}{\bar{s}} = \frac{\sum_{s=1}^{\bar{s}} I_g\left(s\right)}{\bar{s}}, \text{where } I_g\left(s\right) = \begin{cases} 1 & \text{if the agent activates state } s \\ & \text{during the play of game } g, \\ 0 & \text{otherwise} \end{cases} \tag{2.5}$$

The overall Cost 2 for the set of games played is simply the mean of these costs:

$$\text{Cost } 2 = \frac{1}{\bar{g}} \sum_{g=1}^{\bar{g}} (\text{Cost } 2)_g = \frac{1}{\bar{g}} \sum_{g=1}^{\bar{g}} \alpha_2 \left(1 + C_2\right)^{\beta_2} \tag{2.6}$$

Thus, with each game activating three states, $\alpha_2 = 1$, and $\beta_2 = 2$, Cost 2 for this example is $(1 + 3/8)^2 \cong 1.89$, which is less than Cost 1.

Cost 3 and Automaton-Level State Activation Similarity

The third method to model costs ("Cost 3") considers *the manner* in which an automaton activates its states to play different games. With this cost measure, an automaton that plays two games in a similar manner is rewarded (in terms of a low cost), and an automaton that plays two games in a dissimilar manner is penalized (in terms of a high cost). Under Cost 3, an automaton's behavioral homogeneity across the $\bar{g}$ games it plays serves as a proxy for its manner of activation. The degree of similarity of an automaton's state activation frequencies across games played captures the behavioral homogeneity.

The focus on states' activation frequencies rather than binary activation indicators distinguishes Cost 3 from the other two cost measures. State activation similarity for a single state of an automaton is measured as the standard deviation of its activation rate (the percent of rounds during which the agent activated the state) across all games played. The activation similarity for the automaton as a whole is the mean of its states' activation similarities.

Consider automaton *i*, and let $x_{i,g,s}$ indicate the percentage of rounds it activates state *s* when playing game *g*. So for all *g*, $\sum_{s=1}^{\bar{s}} x_{i,g,s} = 100\%$. For each state, the program calculates the standard deviation of the state's activation percentage across all games played. It then takes the mean across all states of these standard deviations to create the automaton-level state activation similarity measure (ALSAS):

$$\text{ALSAS: } \tilde{C}_3 = \mu_s\left(\sigma_g\right) = \frac{1}{\bar{s}} \sum_{s=1}^{\bar{s}} \sqrt{\frac{1}{\bar{g}} \sum_{g=1}^{\bar{g}} \left(x_{i,g,s} - \mu_{i,s}^{x}\right)^2} \tag{2.7}$$

where $\mu_{i,s}^{x} = \frac{1}{\bar{g}} \sum_{g=1}^{\bar{g}} x_{i,g,s}$.

Figure 2.3 provides two contrasting examples. Automaton A on the left of the figure has a much more consistent pattern of state activation across the three games it plays than has automaton B on the right. Correspondingly, automaton A's state activation similarity measure of $\tilde{C}_3 \cong 0.887$ is much lower than automaton B's measure of $\tilde{C}_3 \cong 23.49$.

Cost 3 is based on this measure of automaton-level state activation similarity. The program calculates the Cost 3 statistic as follows:

$$\text{Cost } 3 = \alpha_3 \left(1 + C_3\right)^{\beta_3} = \alpha_3 \left(1 + \frac{\tilde{C}_3}{\gamma_3}\right)^{\beta_3} = \alpha_3 \left(1 + \frac{\mu_s\left(\sigma_g\right)}{\gamma_3}\right)^{\beta_3} \qquad (2.8)$$

The γ_3 parameter in equation (2.8), allows the user to scale $\mu_s\left(\sigma_g\right)$ in a manner comparable to $\bar{s}$ in the Cost 1 and Cost 2 formulas [equations (2.3) and (2.5)]. Continuing the example of Automaton A above with the state activation similarity of 0.887, and with $\alpha_3 = 1$, $\beta_3 = 2$, and $\gamma_3 = 1$, the corresponding Cost 3 is $(1 + 0.887)^2 \cong 3.56$.

Cost 4, Accessible States per Game, and Accessible States in All Games

The fourth method to model costs ("Cost 4") considers *the potential* for a state to be activated during the play of any game. This view considers it costly to have a cognitive state available, or accessible, for use—whether or not it is actually activated is irrelevant. A state is *accessible* in game *g* if, given the starting state for the game, that state may potentially be reached. This case will be true if, beginning from the starting state, there is a sequence of transitions that leads to the accessible state. For example, in Figure 2.2.b, states 1, 2, and 4 are accessible when playing Prisoner's Dilemma, and states 2, 3, and 4 are accessible when playing Chicken.

This accessibility statistic is calculated for each game separately to measure the proportion of states that are accessible in that game. A strategy that has less the one hundred percent accessible states when playing a particular game is effectively less complex than it could potentially be. Such a structure could lead to lower costs. The accessibility in all games statistic simply takes the union of the accessible states across all games. For example, the automaton in Figure 2.2.b has 100 percent if its states accessible in all games. In the four-state automaton of Figure 2.2.c, states 1 and 3 are accessible in Prisoner's Dilemma, states 3 and 4 are accessible in Chicken, and state 2 is not accessible in any game. The corresponding accessibility measures are 0.5 for Prisoner's Dilemma, 0.5 for Chicken (giving an average accessibility of 0.5). The overall accessibility is C_4 = 0.75 since states 1, 3, and 4 are accessible in at least one game (state 2 is not accessible in any game). The accessible states in all games statistic forms the basis for Cost 4:

$$\text{Cost 4} = \alpha_4 \left(1 + C_4\right)^{\beta 4} \text{ with } C_4 = \frac{\sum_{s=1}^{\bar{s}} I(s)}{\bar{s}} \text{ and } I(s) = \begin{cases} 1 & \text{if the state } s \text{ is accessible} \\ & \text{during the play of } \textit{any} \text{ game,} \\ 0 & \text{otherwise} \end{cases} \quad (2.9)$$

With $\alpha_4 = 1$ and $\beta_4 = 2$ the corresponding Cost 4 for the automaton in Figure 2.2.c is $(1 + 3/4)^2 = 3.0625$, while Cost 4 for the automaton in Figure 2.2.b is $(1 + 4/4)^2 = 4.0$, the maximum possible.

Comparing the Cost Measures

The Cost 1 and Cost 2 statistics are both binary measures in the sense that they consider only whether each state has been utilized or not. They differ in the scope of this consideration; for Cost 1 the scope is the entire set of games, and for Cost 2, the scope is single game at a time. When the set of games is a singleton, they are equivalent.

With either of these two costs, a state that is activated but a single round contributes the same to the cost measure as does a state that is activated nearly every round. Cost 3 does not share this feature because, not being a binary measure, it looks at the commonality of state activation frequencies across games. The three examples provided in Figure 2.4 illustrate the differences between these costs measures. Each example contains two simple automata that differ in only one of the three cost statistics.

In Figure 2.4.a, the automata differ only in distinct states activated in all games (Cost 1), in Figure 2.4.b, the automata differ only in distinct states activated per game (Cost 2), and in Figure 2.4.c, the automata differ only in automaton-level state activation similarity (Cost 3).

Cost 4 is also a binary measure in the sense that it considers only whether each state has the *potential* to be utilized (i.e., accessible) or not. Since it is based upon the union of states accessibility in each individual game, it is alike Cost 1 that is based upon the union of states utilized in each individual game. Cost 4 differs from Cost 1 because all accessible states might not be actually utilized; this will depend upon the opponent's sequence of actions.

Profit

The primary performance statistic is profit, defined as a strategy's score net of its total cost. Where strategies compete in multiple, simultaneously played games, the average score all of games played, net of cost, determines profit. The simulation framework bases selection on strategies' profits, so profit may be construed as a strategy's fitness.

Other Performance Statistics

The framework is designed to be capable of addressing a broad set of research questions. For this reason, the simulation program also generates three other performance statistics. The first measures the frequency that action 0 is actually played by the strategy. The other two measure homogeneity of play by considering state activation frequencies. At the automaton level, the program measures the variance of state activation across games played. At the population level, the program reports the variance of state activation across the population for a given game (see Figure 2.5). Appendix 2.1 develops these statistics more thoroughly.

Structural Statistics

Whereas the performance statistics measure (retroactively) how a strategy *actually* activated its states (and received the resulting payoffs and costs), the structural statistics describe how an automaton strategy could have *potentially* activated its states based on its construction. These statistics summarize and describe an automaton's initialization, action, and transition functions, $\theta(\cdot)$, $f(\cdot)$, and $\tau(\cdot)$. Though a strategy's structure certainly influences its behavior, it is not the sole determinant. Its structure interacts with opponents and the environment to determine the specific actions to take. Still, an understanding of a strategy's structure gives insight into potential behavior and economization.

An automaton strategy economizes by using less than its full complement of states when playing the set of games. One key feature of an automaton's structure is the inherent limitation on which states it can potentially use. There may be a subset of states within an automaton out of which no transitions lead. So once play transitions into this subset, the automaton cannot access any states outside of the set. When this subset is a single state, it acts as an attractor because the transitions terminate at the state.

The accessible states in all games statistic is described above and forms the basis for the fourth cost measure. The other key structural statistics are detailed below. Each measures the proportion of accessible states (as opposed to all states) that meet the statistic's criterion since states that are not accessible are irrelevant.

Terminal States

This statistic measures the proportion of accessible states that are terminal states. A state is *terminal* if all transitions from it lead directly to itself. Formally, automaton *i* has a terminal state *k* if $\tau_i\left(s_{i,k}, a_j\right) = s_{i,k}$ for all opponent actions a_j. For example, in Figure 2.2.a state 4 is terminal, as is state 3 in Figure 2.2.c. The presence of a terminal state indicates that the strategy uses a trigger mechanism. This type of strategy responds to an opponent's action by making an irrevocable commitment to continue playing the same action for the remainder of the game.

Counting States

This statistic measures the proportion of accessible states that count. A state *counts* if it is not a terminal state and both transitions lead to the same state. Formally, automaton *i* has a counting state $k \neq \ell$ if $\tau_i\left(s_{i,k}, a_j\right) = s_{i,\ell}$ for all opponent actions a_j. This definition differentiates counting states from terminal states. Counting states allow a strategy to delay a response to an action played by an opponent's strategy. State 3 in Figure 2.2.d provides an example.

Other Structural Statistics

The multiple games experimental framework produces several other statistics that describe the potential capabilities of an automaton strategy that are encoded in its structure. One is the proportion of (accessible) states that play action 0. Note that this is a distinct statistic from action 0 usage frequency described above. The usage frequency measures the percentage of rounds the automaton actually used action 0 during play of the game(s); the proportion of states that play action 0 measures the automaton's capability to play action 0.

Three statistics measure how an automaton reciprocates its opponent's actions. The first statistic is the proportion of states that reciprocate an opponent's 0-actions; that is, those that transition to a state that plays action 0 in response to action 0 being played by the opponent. The second statistic analogously measures action 1 reciprocity. The third statistic measures the proportion of states that reciprocate both action 0 and action 1. These states are labeled Tit-for-Tat states, and an automaton that has one or more of them is incorporating Tit-for-Tat components into its overall strategy.

Two other statistics describe how a strategy begins play. These are the percent of starting states the play action 0 and the number of distinct initial states. For instance, consider the two automata in Figure 2.2.f; the one on the right has two distinct initial states while the other on the left has only a single initial state. Additionally, the program can output a complete description of the automaton. Appendix 2.1 describes these statistics in further detail.

Output of the Statistics

The simulation program allows flexibility in the manner that it reports the statistics. There are two modes in which to execute the computer experiments—graphical user interface (GUI) mode and batch mode. In GUI mode, the statistics are reported visually in time series graphs that are drawn on the computer screen as the program runs. Each statistic the user selects is displayed in a separate window. The GUI mode is useful for initial exploratory experiments and for demonstrations of the experiments. Although the GUI mode does have a limited ability to write the results to output files, the batch mode is better suited for generating and collecting data.

The batch mode is capable of running trials over sweeps of parameters and formatting the data for analysis by common statistical analysis software. The program reports the data on two levels—for the whole pool of automaton strategies that belong to a player and for the individual automata themselves. The former is in essence the population means of the latter. The program writes these different data sets to separate files. Additionally, it writes to separate files for the row player and the column player.

Both modes allow the user to specify how often the data is conveyed. The user specifies the number of generations between data output. The user has an additional optional to create snapshots of the strategy level data at the same frequency as the aggregated data or only during the first and last generations.

Other options allow the user to select which statistics should be included in the output. Most of the statistics described above and in Appendix 2.1 are calculated for each game played[12]. The user can opt for the output to include the game level statistics in addition to the means over all the games played.

Additionally, the program generates a report on the details of the application of the genetic algorithm if desired by the user. Lastly, it saves the evolved automata after the last generation to special serialized files that can be reused in subsequent experiments.

2.5. Dynamics

Evolutionary Dynamics

In this model, the row and column players will each have a finite set of strategies available to use. This set of automata represents a repertoire of possible approaches to a problem. Employing the test kit paradigm explained above, this procedure allows a thorough probing of the relative merits of different approaches. Over time, this set (or population) will evolve, as strategies that performed relatively successfully will come to represent a greater proportion of the population. Conversely, those automata that performed relatively poorly will dwindle. This dynamic captures the simple idea that an agent who has several options available will choose a strategy that performed well in the past.

The model uses a genetic algorithm to supplement this dynamic. Genetic algorithms, first developed by Holland (1975), are a highly efficient method to optimize in environments with high dimensionality, noise, discontinuities and nonlinearities (Miller, 1996). In the context of dynamics modeled with a genetic algorithm, strategies become chromosomes and evolve according to the paradigms of selection, crossover, and

---

[12] Exceptions include the distinct states activated for all games, automaton-level state activation similarity, and distinct initial states statistics. Since the program uses these statistics to derive cost 1, cost 3, total cost and profit, it also cannot report any of these statistics on a per game basis.

mutation borrowed from biology. Agents compete over several generations, and the model applies the genetic algorithm during each generation[13].

Selection, the first subroutine in the evolutionary dynamic, occurs in two steps. First the program ranks the strategies according to their profit in the previous generation and identifies the top $(1-x)\bar{n}$ strategies (where $x \in [0, 1]$ is the crossover rate). The dynamic copies these strategies directly into the new strategy pool. In the second step, the program randomly selects the remaining $(x)\bar{n}$ strategies for the new pool. During this step, a strategy's probability for selection increases with its relative profit. Before adding these randomly selected strategies into the new pool, the dynamic executes the second subroutine, the genetic algorithm.

The implementation of strategies as finite automata makes them amenable to a genetic algorithm dynamic. Conceptually, an automaton is an array of states just as a chromosome is (loosely) an array of nucleotides. The genetic algorithm first applies crossover to pairs of the randomly selected strategies and then applies mutation. Crossover provides an opportunity for an agent to recombine portions of different existing strategies to create novel ones.[14] Mutation allows for additional variation to seep into the strategy pool. The following section on the framework's algorithm contains the specifics of the selection mechanism that the model employs.

2.6. Algorithm

The experimental framework uses the following algorithm to conduct a trial of an experiment. Appendix 2.2 describes the framework's software platform[15] and other technical specifications. Appendix 2.3 details program usage and parameter input. A

---

[13] Since the program applies the dynamics at the beginning of a generation before matching occurs, it skips the dynamics in the first generation.

[14] The framework provides two ways to implement crossover. The first recombines states between pairs of automata. The second translates the automata into bit strings, and then recombines the bit strings (see Miller, 1996). See section 2.6 for the details.

number of independent runs comprise a single trial (see Figure 2.1). Each run of the program applies the four steps outlined below. The program applies one of two substeps when they are labeled with an 'a' or 'b'.

Step 1: Initialize populations. Trial parameters determine the number of automata $\bar{n}$ and $\bar{m}$ in the row and column players' strategy pools and how the populations are initialized. The simulation either creates the populations randomly or loads them from previous trial(s).

Step 1a: *Create population randomly*. The simulation creates finite automata during which it assigns the actions and transitions randomly from a uniform distribution.

Step 1b: *Load population from a previous simulation run*. Alternatively, the population is loaded intact from a saved file. For each strategy, the only the features retained are the finite automaton itself and its ID number (performance records are discarded).[16]

Step 2: Apply dynamics. This simulation models the evolutionary dynamics as described in section 2.5 above to include selection, crossover, and mutation. The program skips this step during the first generation and applies the dynamic independently on the row and column populations.

[15] The simulation is written is Java v1.6 and uses the Repast (Recursive Porous Agent Simulation Toolkit) v3.1 simulation libraries. Repast uses a high quality Mersenne Twister pseudo-random number generators included in the Colt v1.0.2 libraries. Each run used a unique clock-determined seed. Java's platform independence allowed the use of a variety of computers and operating systems to run the simulations. Code is available from the author upon request.

[16] A loaded strategy's automaton will only have initial states designated for games in which it has experience. For each new game it is to play, the program randomly assigns an initial state.

Step 2.1: *Copy top performers*. The simulation ranks the automata based on their profit in the preceding generation[17]. It then adds the best $\left(1-crossover\ rate\right)\overline{n}$ automata to the next generation's strategy pool without any alterations.

Step 2.2: *Add genetically modified strategy pairs*. For the remaining positions in the next generation's strategy pool, the program selects pairs of strategies, modifies them either by bit-wise or state-wise crossover and mutation, and inserts them into the new population.

Step 2.2.1: *Identify parent strategies*. The simulation randomly selects (with replacement) two automata. The probability for selection depends on the relative fitness (profit) of the strategies[18].

Step 2.2.2: *Apply crossover.* The framework encompasses two options for implementing crossover: bit-wise and state-wise. In either case, crossover does not affect at which state(s), depending on the game, the automata begin play. For instance, $\theta_i\left(\text{Prisoner's Dilemma}\right)=4$, means the i^{th} automata initiates in state 4 when playing the the PD game. After crossover, the automaton will still begin in its fourth state, though the characteristics of that state may have been changed by crossover.

Step 2.2.2a: *Apply bit-wise crossover*. The algorithm first converts the automata to bit representations[19], then it applies the crossover genetic operator. The program

[17] In the default setting, the profit statistic uses no time discounting, weighting all rounds equally. The user may specify a time discounting parameter.

[18] The population's profits are normalized so that each automaton *i*'s fitness is $f_i = \left(\pi_i - \mu\right)/\sigma + \alpha$, where π_i is *i*'s profit, μ and σ are the mean profit and standard deviation. The relative performance parameter, α, ensures that automata that perform worse than α standard deviations from the mean cannot be selected since the algorithm reassigns negative fitness scores to zero. As α increases toward infinity, selection depends less on performance and more on chance (Miller, 1996). The program selects the automata by two independent draws from the cumulative distribution of fitness scores (with replacement).

determines randomly a common crossover point and length ℓ for the two binary strings. It swaps the strings' next ℓ bits beyond this crossover point. For this operation, the simulation treats the strings as circular so that if there are fewer than ℓ bits to the end of the strings, it continues the swapping at their beginnings.

Step 2.2.2b: *Apply state-wise crossover*. The algorithm applies the crossover genetic operator directly on the automata. The program determines randomly for the two automata a common crossover point between states and crossover length ℓ. It swaps the automata's next ℓ states beyond the crossover point. For this operation, the simulation treats the automata as circular so that if there are fewer than ℓ states to the end of the automata, it continues the swapping at their beginnings.

Step 2.2.3: *Apply mutation.* The framework applies either bit-wise or state-wise mutation (corresponding with the crossover option) to an automaton after crossover. Top performing automata copied to the new pool in step 2.1 are not subject to mutation.

Step 2.2.3.a: *Apply bit-wise mutation*. For each of the two automata that experienced crossover, the program flips each bit in its string with a parameterized probability. The program converts the bit string back to a conceptual finite automaton and adds it to the pool for the next generation.

Step 2.2.3.b: *Apply state-wise mutation*. For each state *k*, the program makes three independent draws from a uniform distribution. If the first draw is below the mutation probability threshold, the program modifies its action function $f_i(\cdot)$ and assigns the other action as the output of $f_i(s_{i,k})$. If the second draw is below the threshold, the program modifies its transition function $\tau_i(\cdot,\cdot)$ and assigns a randomly determined state

[19] The simulation encapsulates a finite automaton in an Automaton object built from State objects. To maintain procedurally consistency with Miller (1996) during selection, the bit-wise crossover option converts this Automaton object to a bit representation (a string of 0's and 1's). This bit string consists of binary representations of the initial starting state for each of the $\overline{g}$ games followed by representations for each of the $\overline{s}$ states. Three numbers, converted to binary strings, comprise each state's structure: the first is the state's action, and the second and third are the transition states if the opponent plays actions 0 or 1.

as the output of $\tau_i\left(s_{i,k}, a_m = 0\right)$. If the third draw is below the threshold, the program assigns a randomly determined state as the output of $\tau_i\left(s_{i,k}, a_m = 1\right)$. After all states have been (potentially) modified, the program adds the automaton to the next generation's strategy pool.

Step 3: Play Games.

Step 3.1: *Match automata*. The program allows two methods to match the row and column players' automata for competition in the supergames (that is, repeated games)—mean matching and random matching.

Step 3.1a: *Mean matching*. Under this paradigm, each automaton in the row population (agent *n*) pairs with every automaton in the column population (agent *m*) to play all supergames selected for that particular trial. Thus, each automaton in agent *n*'s (*m*'s) pool plays $\overline{m} \cdot \overline{g}$ $\left(\overline{n} \cdot \overline{g}\right)$ supergames in every generation.

Step 3.1b: *Random matching*. Under this paradigm, the user sets the number matches or uses the default setting of $\overline{n} \cdot \overline{m} \cdot \overline{g}$ matches (the total number of pairings with mean matching). For each match, a randomly selected automaton from the row strategy pool competes against a randomly selected automaton from the column pool (draws are independent from a uniform distribution). They play a stochastically determined supergame, where parameters specify the game probabilities.

Step 3.2: *Play supergame(s).* The matched automata then play a repeated game for a parameterized number of rounds[20]. During every round, each automaton plays the action specified by its current state and then transitions to the state dependent on the play of its opponent. To accommodate play of an asymmetric game, an option allows the automata to play the game, switch (row and column) roles, and play it again.

---

[20] Setting the rounds per game parameter to $10*\overline{s}$, where $\overline{s}$ is the number of states, ensures the play extends well beyond the automata's intrinsic memory capacity (Miller, 1996).

In the multiple-game settings, simultaneous play means that an automaton plays each of the selected games before the algorithm applies selection. This ensures that the same automaton is playing the different games (with a possibly distinct starting state for each game). Automata play a game for the full number of specified rounds before switching to a different game[21].

Step 4: Iterate or stop program. The program terminates execution if it has reached the final generation. Otherwise, the algorithm proceeds to the next generation (step 2).

2.7. Applications and Extensions

The overall experimental design of the framework presents two primary types of applications for the investigation of strategic behavior in multiple games settings. One type of application studies how strategies evolve in varied game and cognitive cost settings. The framework provides various measures of strategies' game performance and structural characteristics to assess the tradeoff between specialization and reusability of a strategy's subcomponent states, the appearance of certain heuristic rules (such as trigger strategies), and the relationship between similarity of games in the environment and strategy development. Chapter 3 of this dissertation will address these issues.

An important byproduct of this first type of application is the creation of strategy pools that have adapted to particular multiple game (and cost) settings. The framework saves the actual automata strategies after they complete their evolutionary process. These strategy pools can then serve as the inputs for the other main type of application of this framework.

[21] With noisy actions, the program repeats each pairing of automata five times to minimize the influence of the random variations (see Miller, 1996).

The second primary application investigates the role experience plays in multiple game settings. The investigator may face off agents employing strategy pools that encapsulate disparate experiences to address the multiple game environment issues of specialization, reusability, and game similarity. Chapter 4 of this dissertation will explore these topics.

One can envision interesting, straightforward extensions of this framework. One category of extensions includes matching players with asymmetric capabilities. The asymmetry could exist with the number of automata in each player's strategy pool or the number of states per automata. In the presence of cognitive costs, it may be that having additional resources available turns out to be not a significant advantage, depending on the mix of games the player faces.

Other potential extensions relate to expanding the types of games the framework allows. The core of the framework relies on the taxonomy of strictly ordinal 2 x 2 games provided in Rapoport, et al. (1976). While this taxonomy serves to organize and classify the seventy-eight possible games in a sensible manner, it excludes many other interesting 2 x 2 games. Kilgour and Fraser (1988) develop a taxonomy of all ordinal 2 x 2 games, including those not strictly ordinal (those games in which a player may rank one or more of her payoffs equally), extending the set of games to seven hundred twenty-six. The existing framework is currently capable of examining games within this extended taxonomy. More generally, with some modifications to the programming code, one could employ the framework to handle other classes of games than 2 x 2 games.

Table 2.1: Concise Rapoport, Guyer, and Gordon (1976) taxonomy.

RGG Game #	Phylum	Class	Order	Genus	Stability	Payoffs							
						Up Left Row	**Up Left Col**	**Up Right Row**	**Up Right Col**	**Down Left Row**	**Down Left Col**	**Down Right Row**	**Down Right Col**
1	N	EP	D2	-	SS	4	4	3	3	2	2	1	1
2*	N	EP	D2	-	SS	4	4	3	3	1	2	2	1
3*	N	EP	D2	-	SS	4	4	3	2	2	3	1	1
4	N	EP	D2	-	SS	4	4	3	2	1	3	2	1
5*	N	EP	D2	-	SS	4	4	3	1	1	3	2	2
6*	N	EP	D2	c	WS	4	4	2	3	3	2	1	1
7*	M	EP	D2	-	SS	3	3	4	2	2	4	1	1
8	M	EP	D2	-	SS	3	3	4	2	1	4	2	1
9*	M	EP	D2	-	SS	3	3	4	1	1	4	2	2
10	M	EP	D2	-	SS	2	3	4	2	1	4	3	1
11	Z	EP	D2	-	SS	2	3	4	1	1	4	3	2
12*	M	Ep	D2	-	SS	2	2	4	1	1	4	3	3
13	M	EP	D2	-	SS	3	4	4	2	2	3	1	1
14	M	EP	D2	-	SS	3	4	4	2	1	3	2	1
15	M	EP	D2	-	SS	3	4	4	1	2	3	1	2
16	M	EP	D2	-	SS	3	4	4	1	1	3	2	2
17	M	EP	D2	-	SS	2	4	4	2	1	3	3	1
18	M	EP	D2	-	SS	2	4	4	1	1	3	3	2
19	M	EP	D2	t	WS	3	4	4	3	1	2	2	1
20	M	EP	D2	t	WS	3	4	4	3	2	2	1	1
21	M	EP	D2	t	WS	2	4	4	3	1	2	3	1
22	N	EP	D1	c	WS	4	4	3	3	2	1	1	2
23	N	EP	D1	-	SS	4	4	3	3	1	1	2	2
24	N	EP	D1	c	WS	4	4	3	2	2	1	1	3
25	N	EP	D1	-	SS	4	4	3	2	1	1	2	3
26	N	EP	D1	c	WS	4	4	2	3	3	1	1	2
27	N	EP	D1	c	WS	4	4	2	1	3	2	1	3
28	N	EP	D1	-	SS	4	4	3	1	2	2	1	3
29	N	EP	D1	-	SS	4	4	3	1	1	2	2	3
30	N	EP	D1	c	WS	4	4	2	2	3	1	1	3
31	M	EP	D1	-	SS	3	4	2	2	1	3	4	1
32	M	EP	D1	-	SS	3	4	2	1	1	3	4	2
33	M	EP	D1	-	SS	3	4	1	2	2	3	4	1
34	M	EP	D1	-	SS	3	4	1	1	2	3	4	2
35	M	EP	D1	-	SS	2	4	3	2	1	3	4	1
36	M	EP	D1	-	SS	2	4	3	1	1	3	4	2
37	M	EP	D1	-	SS	3	4	2	3	1	2	4	1
38	M	EP	D1	-	SS	3	4	1	3	2	2	4	1
39	M	EP	D1	t	WS	2	4	3	3	1	2	4	1

Notes:

1. Asterisks indicate symmetric games.
2. Every game's *natural outcome* is the Up Left cell of its payoff matrix.

The natural outcome (NO) defined by Rapoport, et al. (1976) is determined by applying the following conditions in sequence:

a. If a single outcome contains the high payoff for both players (4 for each), then it is the NO.

b. If there are two dominated strategies, then their elimination defines the NO.

c. If there is a single dominated strategy, then after its elimination, the NO is the outcome in which the player with no dominated strategies receives the higher payoff.

d. The NO is the maximin outcome.

3. Phyla include:
   - N – No conflict (see 2.a. above).
   - Z – Complete opposition (constant sum games).
   - M – Mixed motive.
4. Classes include:
   - EP – NO is a Nash equilibrium (NE) and Pareto optimal.
   - Ep – NO is a NE but not Pareto optimal.
   - e – Natural outcome is not a NE.
5. Orders include:
   - D2 – Two dominating strategies.
   - D1 – One dominating strategy.
   - D0 – Zero dominating strategies.

   Phylum M, Class e has two suborders:
   - D0e2 – No dominating strategies, two NE.
   - D0e0 – No dominating strategies, no NE.

Table 2.1: Concise Rapoport, Guyer, and Gordon (1976) taxonomy (continued).

RGG #	Phylum	Class	Order	Genus	Stability	Payoffs							
						Up Left Row	Up Left Col	Up Right Row	Up Right Col	Down Left Row	Down Left Col	Down Right Row	Down Right Col
40	M	EP	D1	-	WS	3	4	4	1	2	2	1	3
41	M	EP	D1	-	SS	3	4	4	1	1	2	2	3
42	M	EP	D1	-	SS	3	3	4	1	2	2	1	4
43	M	EP	D1	-	SS	3	3	4	1	1	2	2	4
44	M	EP	D1	f	WS	2	4	4	1	1	2	3	3
45	Z	EP	D1	-	SS	3	2	4	1	2	3	1	4
46	M	EP	D1	f	WS	3	2	4	1	1	3	2	4
47	M	Ep	D1	f	WS	2	3	4	1	1	2	3	4
48	M	Ep	D1	f	WS	2	2	4	1	1	3	3	4
49	M	EP	D1	ctf	US	3	4	4	3	2	1	1	2
50	M	EP	D1	tf	US	3	4	4	3	1	1	2	2
51	M	EP	D1	ctf	US	3	4	4	2	2	1	1	3
52	M	EP	D1	tf	US	3	4	4	2	1	1	2	3
53	M	EP	D1	ctf	US	3	3	4	2	2	1	1	4
54	M	EP	D1	tf	US	3	3	4	2	1	1	2	4
55	M	EP	D1	tf	US	2	4	4	3	1	1	3	2
56	M	EP	D1	tf	US	2	4	4	2	1	1	3	3
57	M	Ep	D1	tf	US	2	3	4	2	1	1	3	4
58	N	EP	D0	c	WS	4	4	2	3	1	1	3	2
59	N	EP	D0	-	SS	4	4	2	2	1	1	3	3
60*	N	EP	D0	-	SS	4	4	2	1	1	2	3	3
61*	N	EP	D0	c	WS	4	4	1	3	3	1	2	2
62	N	EP	D0	c	WS	4	4	1	2	3	1	2	3
63*	N	EP	D0	c	WS	4	4	1	2	2	1	3	3
64	M	EP	D0	f	WS	3	4	2	1	1	2	4	3
65	M	EP	D0	f	WS	2	4	3	1	1	2	4	3
66*	M	e	D0e2	c	US	3	3	2	4	4	2	1	1
67	M	e	D0e2	c	US	2	3	3	4	4	2	1	1
68*	M	e	D0e2	c	US	2	2	3	4	4	3	1	1
69*	M	e	D0e2	-	US	2	2	4	3	3	4	1	1
70	M	e	D0e0	c	US	3	4	2	1	4	2	1	3
71	M	e	D0e0	c	US	3	3	2	1	4	2	1	4
72	M	e	D0e0	c	US	3	2	2	1	4	3	1	4
73	M	e	D0e0	c	US	2	4	4	1	3	2	1	3
74	M	e	D0e0	c	US	2	4	3	1	4	2	1	3
75	Z	e	D0	c	US	2	3	4	1	3	2	1	4
76	M	e	D0e0	c	US	2	3	3	1	4	2	1	4
77	M	e	D0e0	-	US	2	2	4	1	3	3	1	4
78	M	e	D0e0	c	US	2	2	3	1	4	3	1	4

<u>Notes (continued)</u>:

6. Genera include:

   f – Force vulnerable.
   t – Threat vulnerable.
   c – Competitive pressure.

7. Stability:

   SS – Strongly stable.
   WS – Weakly stable.
   US – Unstable.

A game in which the NO is also a NE is *strongly stable* (SS) if there are no competitive, threat, or force pressures, and it is *weakly stable* (WS) if exactly one of these pressures is present. All other games are *unstable* (US).

8. This payoffs in table here correct two typographical errors in Rapoport, et al. (1976). The first is for Game 17: column payoffs in game 17 as reported there had two 3's and no 2. The second is for Game 30: as reported there, it was identical to game 27.

9. Common games (highlighted in bold type) with toy names are:
   12 ~ Prisoner's Dilemma (PD)
   25 ~ Stag Hunt (SH)
   39 ~ Bluff (BL)
   66 ~ Chicken (CH)
   68 ~ Leader (LD)
   69 ~ Battle of the Sexes (BS)

10. The following games conform with Parisi's (2000) taxonomy:
   1. Pure common interest games ~ 1–11, 13–46, and 49–56.
   2. Battle of the sexes games ~ 58–69.
   3. Prisoners' Dilemma games ~ 12, 47, 48, and 57.
   4. Inessential games ~ 11, 45, 66, 68, 69, and 75.

Figure 2.1: An example of a trial for a two-game environment.

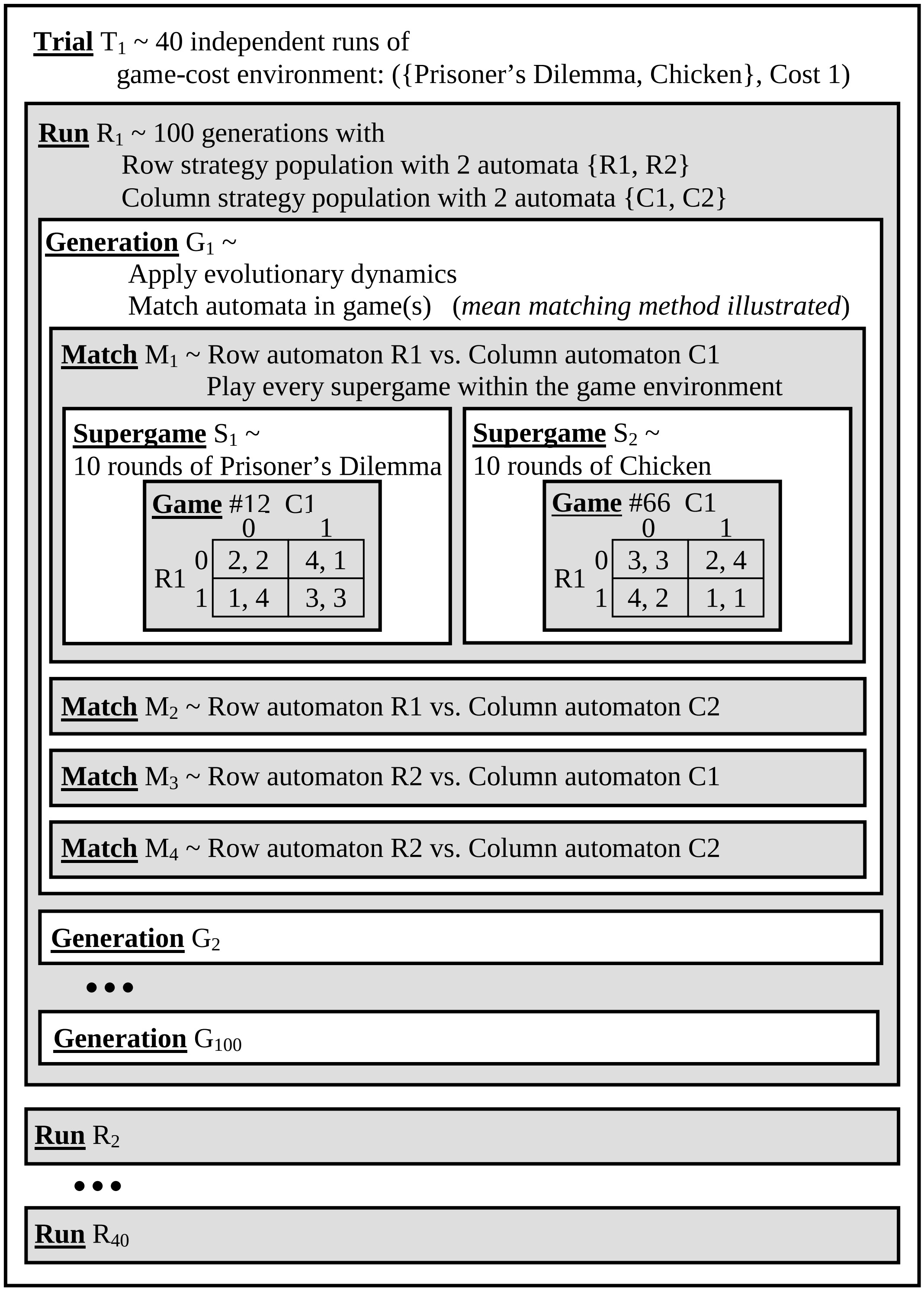
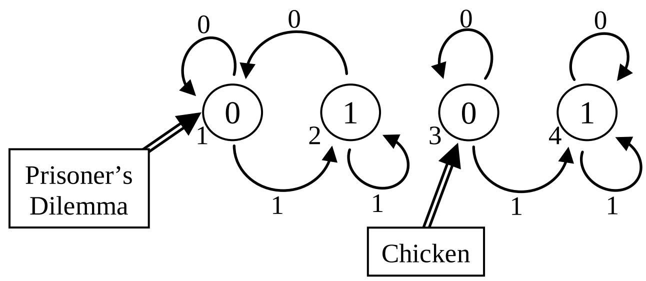


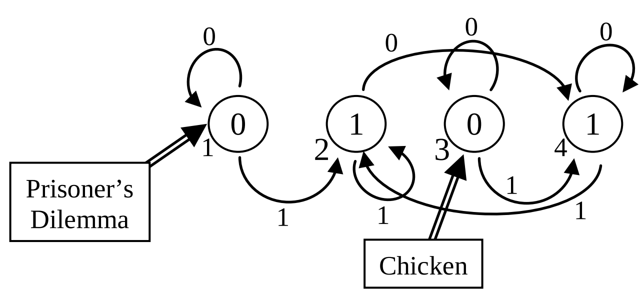
Figure 2.2: Finite automata examples.

Figure 2.2.a: A four-state automaton that specializes all states to the play of a particular game.



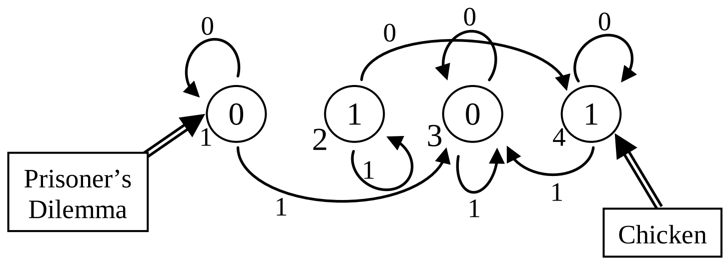
Prisoner's Dilemma potentially uses only states 1 and 2, and Chicken potentially uses only states 3 and 4.

Figure 2.2.b: A four-state automaton that specializes some states to the play of a particular game and reuses other states for both games.



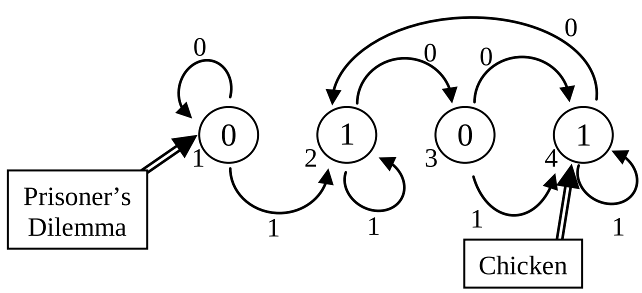
Prisoner's Dilemma exclusively uses states 1, Chicken exclusively uses state 3, but both games potentially use states 2 and 4.

Figure 2.2.c: A four-state automaton that displays limited accessibility in each game and overall.



In Prisoner's Dilemma, states 1 and 3 are accessible. In Chicken, states 3 and 4 are accessible. Accessible states in all games is the union, states 1, 3, and 4—state 2 is not accessible in any game.

Figure 2.2.d: A four-state automaton with a counting state.



State 3 is a counting state because both transition arrows lead to the same state.

Figure 2.2.e: A four-state automaton strategically equivalent to the one in Figure 2.2.d.

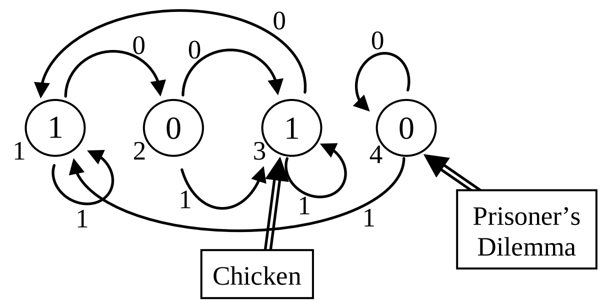


Figure 2.2.f: Two two-state automata that differ only in their initial states.

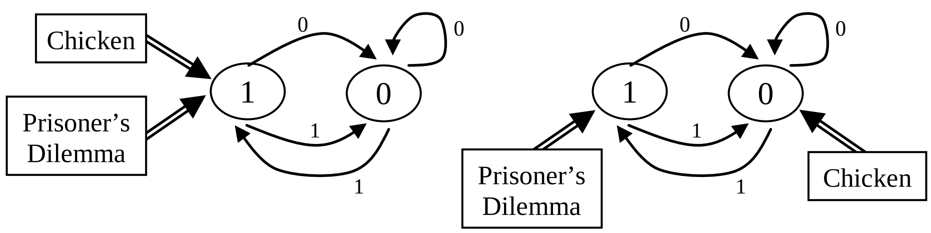


Figure 2.3: An example of the automaton-level state activation similarity statistic.

A player has a pool of two three-state strategies (A and B) and plays three games…

	Percent of rounds the player activates each state					
	Automaton A			Automaton B		
Game 1	40%	10%	50%	65%	30 %	5%
Game 2	40%	11%	49%	0%	50%	50%
Game 3	42%	8%	50%	30%	5%	65%
Standard deviation	0.943	1.247	0.471	26.56	18.41	25.50
Automaton-level state activation similarity (mean standard deviation)	**0.887**			**23.49**		

Figure 2.4: Differentiating cost examples.

Figure 2.4.a: Automata that vary only by Cost 1.

The three-state automata A and B **differ in distinct states activated in all games**, but they have the same distinct states activated per game and nearly the same automaton-level state activation similarity.

	Percent of rounds the player activates each state Automaton A	 Automaton B
Game 1	50% – 49.9% – 0.1%	50% – 50% – 0%
Game 2	100% – 0% – 0%	99.1% – 0.1% – 0%
Game 3	100% – 0% – 0%	100% – 0% – 0%
Distinct states activated in all games	**3**	**2**
Distinct states activated per game	1.67	1.67
Automaton-level state activation similarity	0.157	0.157

Figure 2.4.b: Automata that vary only by Cost 2.

The three-state automata A and B **differ in distinct states used per game**, but they have the same distinct states activated in all games and nearly the same automaton-level state activation similarity.

	Percent of rounds the player activates each state Automaton A	 Automaton B
Game 1	33.3% – 33.3% – 33.3%	33.3% – 33.3% – 33.3%
Game 2	100% – 0% – 0%	99.1% – 0.1% – 0%
Game 3	100% – 0% – 0%	99.9% – 0.1% – 0%
Distinct states activated in all games	3	3
Distinct states activated per game	**1.67**	**2.33**
Automaton-level state activation similarity	0.210	0.209

Figure 2.4.c: Automata that vary only by Cost 3.

The three-state automata A and B **differ in automaton-level state usage similarity**, but they have the same distinct states activated in all games and distinct states activated per game.

	Percent of rounds the player activates each state	
	Automaton A	Automaton B
Game 1	65% – 30% – 5%	65% – 30 % – 5%
Game 2	65% – 30% – 5%	5% – 65% – 30%
Game 3	65% – 30% – 5%	30% – 5% – 65%
Distinct states activated in all games	3	3
Distinct states activated per game	3	3
Automaton-level state activation similarity	**0**	**0.246**

Figure 2.5: An example of the population-level state usage similarity statistic.

A player has a pool of three, three-state strategies (A, B, and C) and plays two games…

	Percent of rounds the player activates each state					
	Game 1			Game 2		
Automaton A	65%	30%	5%	65%	30 %	5%
Automaton B	65%	20%	15%	0%	50%	50%
Automaton C	60%	24%	16%	30%	5%	65%
Standard deviation	2.357	4.110	4.967	26.56	18.41	25.50
Population-level state usage similarity (mean standard deviation)	**3.811**			**23.49**		

Appendix 2.1

Descriptions of Other Statistics

State Activation Frequency

This performance statistic measures the percent of rounds a state is activated during play of a game. State activation frequencies form the basis of automaton-level state activation similarity and Cost 3.

Population-Level State Activation Similarity

This performance measure is analogous to the automaton-level state activation similarity described in section 2.4 above. That statistic measures, for a single *automaton*, its behavioral homogeneity across the $\bar{g}$ games it plays. In contrast, the population-level statistic measures, for a single *game*, an agent's behavioral homogeneity across its strategy pool's $\bar{n}$ automata.

The program calculates the population-level state activation similarity (PLSAS) separately for each game. The program first finds, for each state, the standard deviation of the percent of all rounds the agent activates it (over all automata in the player's strategy pool), and then the program averages the $\bar{s}$ standard deviations. Consider a player with a pool of $\bar{n}$ automata, and let $x_{i,g,s}$ indicate the percent of rounds automaton *i* activates state *s* when playing game *g*. Then the population state activation similarity is

$$\text{PLSAS:} \quad \mu_s\left(\sigma_n\right) = \frac{1}{\bar{s}} \sum_{s=1}^{\bar{s}} \sqrt{\frac{1}{\bar{n}} \sum_{i=1}^{\bar{n}} \left(x_{i,g,s} - \mu_{g,s}'^{x}\right)^2} \qquad (2.10)$$

where $\mu_{g,s}'^{x} = \frac{1}{\bar{n}} \sum_{i=1}^{\bar{n}} x_{i,g,s}$.

Figure 2.5 presents two contrasting examples. The player has a much more consistent pattern of state activation across the three automata in its strategy pool for Game 1 on the left of the figure than it has for Game 2 on the right. Correspondingly, the agent's PLSAS of 3.81 for Game 1 is much lower than its PLSAS of 23.49 for Game 2.

One drawback of this measure is that it does not account for potential strategic equivalence between two structural different automata. The automata displayed in Figures 2.2.d and 2.2.e provide an example of strategic equivalence. Renaming states 1, 2, 3, and 4 of the automaton in Figure 2.2.d to 4, 1, 2, and 3 and sorting the renumbered states from low to high results in the automaton in Figure 2.2.e. The population state activation similarity statistic would incorrectly state the degree of similarity.

Figure 2.5 provides an example of this measurement error. Suppose for game 2[22], automata A and C are strategically equivalent such that states 1, 2, and 3 for A correspond to states 3, 1, and 2 for C. Without considering equivalence, the population state activation similarity statistic is 23.49 for game 2. After correcting for equivalence, the measure falls to either 20.43 (if C is rearranged to match A) or 14.14 (if A is rearranged to match C). In either case, the similarity was underestimated (i.e., the statistic was overestimated), but it is not clear which revision is superior. The measurement error can also work to understate the degree of similarity; once the equivalence between A and C is corrected, the population-level state activation similarity statistic for game 1 would increase.

Action 0 Usage Frequency

This performance statistic reports the frequency (proportion of all rounds) that action 0 was actually played by the automaton in each individual game and as an overall average.

Play Percentage

This performance statistic reports the frequency that each game was played. When the user employs mean matching, these statistics will always be $(\bar{g})^{-1}$ for each

---

[22] Two automata could be strategically equivalent for game 2 and not for game 1 if there existed an equivalence for game 2's initial state but not for game 1's.

game. In random matching, the program selects games stochastically, and the play percentage statistic should approach the frequency parameters set by the user.

Percent 0-Playing States

This structural statistic measures the percent of accessible states that play action 0. This is the potential to play action 0, not the frequency of actual use of action 0. Since there is no consistent application of the labels "cooperate" and "defect" to binary actions in the Rapoport, et al. (1976) taxonomy, one must exercise prudence when comparing this statistic (and the next several that follow) in different game environments.

Reciprocation Measuring Statistics

The program measures the percent 0-reciprocating states and percent 1-reciprocating states. These structural statistics measure the percent of accessible states that reciprocate an opponent's 0 action (1 action) by transitioning to a state that plays 0 (1). They correspond to "retaliatory" and "forgiving" characteristics as in Axelrod (1984) and Casti (1992), though the specific correspondence depends on the interpretation of actions 0 and 1 in a given game. The framework also reports the percent of Tit-for-Tat states, those states that are both a 0-recipricator and 1-recipricator.

Percent 0-Playing Initial States

This structural statistic measures the percent of initial states that play action 0. It corresponds to the "nice" characteristic prescribed in Axelrod (1984) and Casti (1992) (or one hundred percent minus this statistic does depending, again, on specific interpretations).

Distinct Initial States

This structural statistic reports the mean number of distinct starting states for all automata in a player's strategy pool. This number being less than the number of games indicates that the agent is playing some the games in a similar manner from start to finish. Figure 2.2.f illustrates two simple automata that are identical except for their initial states.

Assuming these are the only automata in the player's pool, the distinct initial states measurement is 1.5. The automaton on the left of the figure does not distinguish between Prisoner's Dilemma and Chicken games while the automaton on the right does. The disparity could lead to very different outcomes depending on the opposing automaton against which it is matched. The program also lists the number of strategies that have $1, 2, \ldots, \overline{s}$ distinct initial states.

Appendix 2.2

Technical Specifications of the Simulation Framework

The framework is coded in Java programming language and relies primarily upon the Repast simulation library. It also makes uses a high quality random number generator from the Colt library. These software packages are free for public download at the following websites:

- Java: http://java.sun.com/
- Repast: http://repast.sourceforge.net/
- Colt: http://acs.lbl.gov/software/colt/

The simulation code is compatible with the latest releases of these software packages, which at time of publication are Java version 1.6.0_24-b07, Repast version 3.1, and Colt version 1.2.0.

The simulation code consists 7765 lines of code in three separate packages. All code for these packages is available from the author upon request.

Multigames package. This package is the primary simulation code. It contains the following classes: Strategy, MGGame, GAObject, MGModel, BatchMGModel, GuiMGModel, MGStarter, and GameRGG.

Games package. This package provides abstract classes and interfaces that implement basic game theory structures and concepts. This package is designed primarily for use with either the Repast and Swarm[23] simulation toolkits. The package contains the Game and Playable interfaces and the AbstractGame, AbstractGameMxN, and AbstractGame2x2 classes. This library is also available at www.nd.edu/~jleady.

My_Utilities package. This package provides of the following helper classes: MyIO, MyArrays, MySerializer, MyOpenSeqStatistic, and MyOpenSequenceGraph.

---

[23] The Multigames framework does not make use of the Swarm features of the Games package. The Swarm simulation toolkit is available at http://www.swarm.org/.

Appendix 2.3

Parameter Descriptions for the Multiple Games Simulation Program

Usage

java MGStarter [*-b* | *--batch*] [*inputFile*]

Command line parameters:

-b, --batch: start in Batch mode (default is GUI mode).

inputFile: the parameter file, formatted in accordance with Repast specifications.

Optionally, the GUI and Batch modes may be started individually as

java GuiMGModel [*inputfile*]

java BatchMGModel [*inputfile*]

General Parameters[24] – those marked with an asterisk (*) are ignored in GUI mode.

trial: (int) the administrative trial number

runs*: (int > 0) the number of independent runs of the program to conduct.

generations: (int > 0) number of generations per run.

seedString: (String) the seeds for each individual run. When using the default (random) clock seed, this must be set to "random."

testRun: (boolean) flag to test the program with specific automata specified by the user in a special input file.

Population Parameters

[24] Java parameter types and allowable values are listed in parentheses. The parameter names match the computer code and may differ from the language used throughout Chapter 2.

[row|col]PopSize: (int > 0) number of strategies in [row|column] player's pool. Correspond to $\bar{n}$ and $\bar{m}$.

[row|col]States: (int > 0) number of states in [row|column] player's automata. Correspond to $\bar{s}_n$ and $\bar{s}_m$.

Genetic Algorithm Parameters

relativePerformance: (double) corresponds to α in the fitness formula in footnote 18.

useMillerCrossover: (boolean) if true, use bit-wise crossover and mutation as in Miller (1996). If false, use state-wise crossover and mutation.

[row|col]CrossoverRate: (double [0, 1]) the crossover rate.

[row|col]MaxCrossoverLength: (int [0, [row|col]States]) maximum number of states to crossover.

[row|col]MutationRate: (double [0, 1]) the mutation rate.

Cost Function Parameters

costPar[A0|A1|A2|A3|A4|B1|B2|B3|B4|C3]: (double) determine the cost function, equation (2.1), where A's correspond to α's, B's correspond to β's, and C3 correspond to γ_3.

costAdjustment: (int) determines which type of adjustment, if any, to use (multiplicative or additive) and how to calculate (jointly for both populations or separately for the row and column populations).

Game Parameters

roundsPerGame: (int ≥ 0) how many iterations of each supergame to play during a pairing of strategies. Set to zero for an indefinitely repeated game.

noise: (double [0, 1]) probability the opponent's move is misreported.

discount: (double [0, 1]) the discount factor for a repeated game.

probContinue: (double [0, 1]) the probability for play to continue for another round in a repeated game.

matchType: (String {random, mean}) how to determine pairings of strategies.

matches: (int ≥ 0) the number of pairings per generation for random matchType.

randomRole: (boolean) if true, each pairing of strategies from agent 1 and agent 2 plays the selected game twice, switching the agents' row and column roles between play.

gameString: (String) sets which games to play and how often.

Option 1 (specified games): list the specific games and probability each is played.

Option 2 (random games): randomly create $\bar{g}$ RGG strictly ordinal games[25].

Input/Output Parameters

dataDir: (String) all output is saved to a new folder within dataDir named "tXXX", where XXX is the trial number.

outputFile: (String) name of primary output file(s) with population averages. Program will prepend "row_" or "col_" to this and append the trial number.

outputFrequency: (int > 0) how many generations between writes to the outputFile.

[row|col]Detail*: (int 0, 1, 2, 3) what statistics to calculate and output.

perGame: (boolean) if true, calculates per game stats (where applicable) in addition to the average over all games played.

[25] The method used does not correspond to a random draw from [1, 78], but rather from [1, 576] (see footnote 3).

write[Row|Col]Strategies*: (int 0, 1, 2, 3) different options for when to write the strategy-level statistics.

writeCrossovers*: (boolean) if true, write details of each crossover during selection to the file "crossovers.xls".

outputDigits: (int > 0) number of digits after the decimal point to output.

<u>Serialization Parameters</u>

serialize[Row|Col]Pop: (boolean) if true, the population is saved for future use.

useSerial[Row|Col]Pop: (boolean) if true, the population is not randomly generated and instead is loaded from a previous trial.

[row|col]SerialInputFile: (String) filename for previously saved strategy population.

<u>Display Parameters (for GUI mode)</u>

displayFrequency: (int > 0) how many generations between display updates.

snapshots: (boolean) if true, a snapshot will be made for each graph after the last generation.

graphX: (int [0, 15] with exceptions) a separate parameter for each statistic X. Sets the graphical display options for statistic X: display the row player's mean (over all games) statistic, display the column player's mean statistic, show the statistic on a per-game level, and write the data to a file (there are some exceptions depending on the particular statistic).

Bibliography

Aumann, R. J. 1981. "Survey of Repeated Games." In *Essays in Game Theory and Mathematical Economics in Honor of Oscar Morgenstern*, eds. R. J. Aumann, et al., 11-42. Zurich, Mannheim: Bibliographisches Institut.

Axelrod, R. M. 1984. *The Evolution of Cooperation*. New York: Basic Books.

Bednar, J., Chen, Y., Liu, T. X., Page, S. 2010. "Behavioral Spillovers and Cognitive Load in Multiple Games." http://yanchen.people.si.umich.edu/papers/Multiple_Games_20101101.pdf.

Bednar, J., Page, S. 2007. "Can Games(s) Theory Explain Culture? The Emergence of Cultural Behavior within Multiple Games." *Rationality and Society*, 19: 65–97.

Casti, J. L. 1992. *Reality Rules: II*. New York: John Wiley & Sons, Inc.

Gilboa, I., Schmeidler, D. 2001. *A Theory of Case-Based Decisions*. Cambridge, UK: Cambridge University Press.

Holland, J. H. 1975. *Adaptation in Natural and Artificial Systems*. Ann Arbor, MI: The University of Michigan Press.

Holland, J. H. 1986. *Induction: Processes of Inference, Learning, and Discovery*. Cambridge, MA: The MIT Press.

Holland, J. H., Miller, J. H. 1991. "Artificial Adaptive Agents in Economic Theory." *AEA Papers and Proceedings*, 81(2): 365–370.

Just, M. A., Carpenter, P. A. 2002. "A Capacity Theory of Comprehension: Individual Differences in Working Memory." In *Cognitive Modeling*, eds. T. A. Polk and C. M. Seifert, 131–177. Cambridge, MA: The MIT Press.

Kilgour, D. M., Fraser, N. M. 1988. "A Taxonomy of all Ordinal 2 x 2 Games." *Theory and Decision*, 24: 99–117.

Miller, J. H. 1996. "The Coevolution of Automata in the Repeated Prisoner's Dilemma." *Journal of Economic Behavior and Organization*, 29: 87–112.

Miller, J. H., Page, Scott E. 2007. *Complex Adaptive Systems: An Introduction to Computational Models of Social Life*. Princeton, NJ: Princeton University Press.

Parisi, F. 2000. "The Cost of the Game: A Taxonomy of Social Interactions." *European Journal of Law and Economics*, 9: 99-114.

Rapoport, A., Guyer, M. J., Gordon, D. G. 1976. *The 2 X 2 Game*. Ann Arbor, MI: The University of Michigan Press.

Rubenstein, A. 1998. *Modeling Bounded Rationality*. Cambridge, MA: The MIT Press.

Samuelson, L. 2001. "Analogies, Adaptation, and Anomalies." *Journal of Economic Theory*, 97: 320–366.

Walliser, B. 1988. A Simplified Taxonomy of 2 x 2 Games. *Theory and Decision*, 25: 163–191.

Wolfram, S. 2002. *A New Kind of Science*. Champaign, IL: Wolfram Media.

Chapter 3

Strategy Evolution, Multitasking, and Context Effects in Costly Multiple Game Environments

3.1. Introduction

Multitasking is simply "the performance…of a number of different tasks or jobs concurrently" (Oxford English Dictionary, 2010). Although this term originated in computer science as a synonym for multiprocessing, popular culture has expanded its meaning to describe people's behavioral response to an output-oriented and time-constrained life. Multitasking—especially the business world—has become an increasingly predominate concept in modern life.

Multitasking may also be applied more broadly to the functionality of the firm; not only does a division within a firm multitask, but the firm itself multitasks amongst its divisions as product and service lines become ever more diversified. In these multitasking environments, firms seek to exploit synergies that will allow them to maximize the gains from economies of scope. This research develops a method to model multitasking situations and investigates the conditions that influence whether or not synergies materialize.

The popularity of these paradigms has not translated, however, into voluminous research in the economics field. The traditional game theory literature has centered on analysis of equilibrium concepts, decision-making, and learning in strategic environments described by a single game. By following the general modeling principle of striping the problem to its core features, the focus on single-game settings has contributed enormously to the understanding of economic phenomena.

While the study of single games is appropriate for many situations, the purpose of this study is to investigate how multiple-game environments affect strategy structure and performance. Consider, for example, agents who must interact in distinct settings simultaneously. A supplier of intermediate goods in a competitive market bids for contracts with a variety of firms and organizations. These include other small firms, large firms, government agencies, and perhaps non-profit organizations. A sales representative working for this supplier faces a heterogeneous strategic environment if the success of his sales strategy depends on with whom he is negotiating. He faces a potentially distinct strategic game with each different type of customer. A strategy emphasizing cost savings might be best suited for one type of potential customer while a strategy emphasizing responsiveness and personal attention might be best suited for another. In addition to varying by content, sales strategies could also vary by style: aggressive, friendly, professional, etc. Ideally, the sales agent would want to tailor the strategy's content and style to the recipient.

However, building and maintaining a repertoire of strategies might be costly. In such cases, under what conditions should the sales representative apply the same strategy to different types of customers? Can it be better to develop one strategy that is "good enough" for several situations, and perhaps another for one particular situation? This study takes one step towards answering these questions with attention focused on the features of the strategic environments and their influence on outcomes.

Continuing this example, potential customers are not the only people with whom he interacts in the workplace. He must also deal with existing customers, his manager, peers in his firm and from competing firms, secretarial assistants, human resource representatives, etc. Interactions generally do not occur in isolated capsules. Rather, the sales agent must interact intermittently—multitask—with all of these people throughout the workday.

This paper models multitasking by having agents, who maintain stocks of strategies, compete in two games simultaneously. The model captures agents' limited cognitive abilities by implementing strategies as finite automata and by assessing costs related to the strategies' complexity. This first broad set of computer experiments evolves agents' strategy stocks in defined single and multiple game environments. It also provides the stock of strategies for the subsequent chapter on multiple game experiments. Chapter 4 will consider the role of experience in multiple game settings by pitting agents who have evolved in distinct backgrounds against each other.

One purpose of pursuing this study is to establish the need for a theory capable of addressing the multitasked nature of the world. A satisfactory model of decision making in multiple games played simultaneously must assume limits on the agents' rationality. If it did not, a hyperrational agent could simple play an ensemble of strategies consisting of those strategies with which it would play each game in isolation. No separate theory would be needed. This model assumes that limits on rationality exist and incorporates them into a strategy's structure and cognitive costs.

Several questions that involve these limits concern the strategies used by agents in multiple-game environments. How do strategies used by agents who compete in multiple-game environments differ from those used in single-game environments? Do these differences, if any, appear in a strategy's employment, structure, or both? Upon what conditions do these differences depend?

This model attempts to answer these questions within certain environments. It provides automated agents with experience playing in various environments distinguished by two features. The first is the set of games that agents play. This set may consist of a single two-player normal form game or two of these games at once. The second is the form of costs that agents incur. The model considers two cases, one case without costs and another case in which costs depend on the structure of the strategy.

Attention to strategic environments characterized by multiple games is relatively new in the economics literature, but is beginning to grow. Recent studies look at sequences of different games. LiCalzi (1995) and Gilboa and Schmeidler (1995, 2001) both develop theoretical models of multiple-game situations. In these models, an agent faces a sequence of games and determines a strategy for a given situation based on its resemblance to past encounters. Here, however, agents face games simultaneously, unlike Gilboa and Schmeidler (1995, 2001). Chapter 4 will consider the role of past game experience. Güth (2000a, 2000b) investigates learning in four different multiple-game experiments. He shows how subjects learn to anticipate rule changes as they play repeated sequences of games, which he calls "robust" experiments.

Other studies align more closely with the present one both in the simultaneous nature of interactions and in the employment of finite automata to represent strategies. Samuelson (2001) considers a simultaneous three-game environment in which agents balance gains from more sophisticated strategies against their greater cognitive costs. Unlike the present study, his model does not allow sharing of cognitive resources. Bednar and Page (2007) develop a game theoretic model of simultaneous game settings to explain the emergence of culture. Bednar, et al. (2010) use multiple game settings in a human subject experiments to model behavioral spillovers and find that people may employ heuristics that apply across games.

This study employs artificial adaptive agents in computer experiments to investigate the effects of multiple-game environments on the performance and structure of finite automata strategies. Miller (1996) provides the main theoretical and technical basis for the framework developed in Chapter 2 and extended here. Although that study considered only the single-game Prisoner's Dilemma environment, Miller and Page (2007) consider multiple game environment using finite automata strategies. This modeling technique allows an analysis of optimizing behavior without the rigidity required for a formal mathematical model (Holland and Miller, 1991). This approach also has the advantage that one may study more complex and difficult situations than with a strictly mathematical model. Another advantage is the potential to observe emergent behavior.

Of course, one must exercise some caution when interpreting results produced from this type of modeling. The results are often suggestive rather than conclusive, making this technique a useful complement to other research methods. Like all modeling techniques, one must put careful thought into procedure and parameter selection to generate credible results. The following sections discuss procedural and parametric decisions for the design of the experiments and more specifically for the conduct of the simulations.

The overall experimental design consisted of two phases of computer experiments. One purpose of the first phase of experimental trials, considered in this chapter, is to create populations that have evolved in a particular game setting. A strategy population generated during this first phase is then ready to be matched against another one with a different game playing experience. These pairings of strategy populations with different strategic histories comprise the second phase of trials explored in Chapter 4.

The following section outlines the model and discusses related theoretic considerations. Section 3.2 explains the experimental design and delineates the simulation procedures. Section 3.3 presents the main findings under the natural equilibrium context, and Section 3.4 discusses the results for the alternative context. The last section offers concluding remarks.

3.2. Experimental Design

This project employs the experimental simulation framework developed in Chapter 2. Specific design features for this study are described below.

Game Selection

This study restricts attention to four 2x2, symmetric, ordinally distinct games—Prisoner's Dilemma (PD), Stag Hunt (SH), Chicken (CH), and Battle of the Sexes(BS)[1]—for several reasons. Symmetric games simplifies the analysis since both row and column players face identical situations. Hence, unless otherwise noted, the results reported below relate to the row populations only. Having simple, widely studied individual games allowed the focus to be on the multiple game nature of the strategic environments. Table 3.1 presents the versions of the games employed here. These are consistent with the Rapoport, Guyer, and Gordon (1976) taxonomy (hereafter, RGG).

The automata that the agents use to represent their strategies can be interpreted as collections of mental states. Thus, if the state of the automaton calls for action 0 to be taken, we can interpret this as the agent being in a mental state in which action 0 seems appropriate. For example, in the PD game, if an opponent has defected in the previous period, and the agent's automaton may move to a state that also defects. Formally, I call this 0-reciprocating.

In the multigame setting, an agent can play the same or similar strategies across both games. If we interpret the states of the automata as mental states, then the actions taken in those states and the previous plays that led to those states must have similar interpretations. For example, if 0-reciprocating implies "choosing the natural

[1] Prisoner's Dilemma, Chicken, and Battle of the Sexes are three of the four games in the Casti (1992) typology of mixed-motive games. Leader, the other game in the Casti typology, is not considered here because Chicken and Battle of the Sexes are sufficient to model similar games since Leader has much in common with these two.

equilibrium" in one game, then it should not mean choosing the unnatural equilibrium in another game. Otherwise, the idea of transferring a common strategy or part of a strategy across games becomes nonsensical.

Thus, a core assumption of my formulation is that action 0 has similar if not identical meaning across games. In the games that I consider here, I rely on Rapoport, Guyer, and Gordon (1976) classification of these games. In each game, action 0 denotes the natural symmetric equilibrium—the equilibrium we would expect people to play. Thus the generic labeling of actions simply as 0 and 1 has the advantage that action 0 is always the "natural" action that corresponds to the natural outcome of the game.

This assumption can be justified on the following lines. In the PD, the natural equilibrium is (defect, defect) because defect is a dominant strategy. In the other games, no dominant strategy exists and so Rapoport, Guyer, and Gordon (1976) choose the Pareto Dominant symmetric equilibrium.[2]

This is not the only possible categorization of the strategies. One might alternatively, categorize the actions by whether they are more self serving or other regarding. This is the approach taken by Bednar and Page (2007) and later by Bednar, et al. (2010) in their experimental work. This categorization has strong appeal but it cannot adjudicate between the two actions in the Battle of the Success game. Neither strategy is more other regarding than the other.

Clearly, how the actions are categorized has implications for the strategies that evolve in the model. Different categorizations can lead to very different causal models

---

[2] Formally, the natural outcome (NO) defined by Rapoport, et al. (1976) is determined by applying the following conditions in sequence: (1) if a single outcome contains the high payoff for both players (4 for each), then it is the NO [applies to SH]; (2) if there are two dominated strategies, then their elimination defines the NO [applies to PD]; (3) if there is a single dominated strategy, then after its elimination, the NO is the outcome in which the player with no dominated strategies receives the higher payoff; (4) the NO is the maximin outcome [applies to CH & BS].

(see Fryer and Jackson (2008), Nisbett (2003), and Page (2007) for summaries). More generally, categories are the basis for how we interpret reality and thus underpin many of the statistical assumptions we make about signals (Hong and Page 2009).

To demonstrate the role that diverse categorizations can play in section 3.4, I rerun my model using the Prisoner's Dilemma, Chicken, and Battle of the Sexes games, using the other-regarding / self-regarding (or, equivalently, cooperate / defect) categorization of actions. I now name these $PD^{\#}$, $CH^{\#}$, and $SH^{\#}$ to distinguish them from the games under the RGG natural outcome context. I do not consider Battle of the Sexes for the reason I already mentioned. The natural hypothesis to make is that this different categorization will have implications for how the automata evolve (Page 2007).

When facing multiple game environments, one might expect different types of strategies to be employed in an environment with two very similar games than in one with two quite dissimilar games. Using the RGG taxonomy allows a systematic measurement of the similarity of any two RGG games (see Chapter 2 for details) based on the nature of the payoffs in the games themselves. Table 3.3 presents the dissimilarity scores between these four games, with a larger value indicating greater dissimilarity. This particular set of similar and dissimilar games allows a comparison of the performance and structure of strategies[3].

Chicken and Battle of the Sexes are selected because they are very similar—both have two symmetric Nash equilibria. They differ mainly in the Pareto optimality of the various outcomes. Obtaining an average of the two Nash equilibrium payoffs (perhaps through an alternating strategy) strictly Pareto dominates all other non-Nash equilibrium

---

[3] Samuelson (2001) provides the idea for this configuration of similar and dissimilar games. He investigates a three-game environment with two similar games and one dissimilar game.

outcomes in Battle of the Sexes but only weakly Pareto dominates the greater non-Nash equilibrium outcome in Chicken.

Prisoner's Dilemma is selected because it is very different from the other three games and because of its importance in the literature. Prisoner's Dilemma has a single, dominant strategy, Nash equilibrium. However, because the Nash equilibrium is Pareto dominated by another cooperative outcome, it may be viewed as "difficult" to play. Bednar, et al. (2010) develop entropy as a proxy for the difficulty of a game that uses observed action choices in laboratory experiments. Possible entropy values in these games range from 0 when all outcomes occur in a single cell of the payoff matrix to 2 when all outcomes occur with equal frequency (0.25). The entropy value for PD in the baseline no cost setting is 1.76, indicating that PD is ex post relatively high difficulty (see Table 3.11).[4]

On the other hand, in the parlance of RGG, Stag Hunt is a game of "no conflict" as both players receive their greatest payoffs at the same outcome. The other three games are all "mixed motive" games. Accordingly, Stag Hunt is quite dissimilar from the other three games along the taxonomy divisions. SH has two Nash equilibria; the natural outcome (0, 0) is the payoff dominant equilibrium, while the (1, 1) outcome is the risk dominant equilibrium. Harsanyi (1995) suggests that risk dominance may be the stronger equilibrium pull; however, within the natural outcome payoff context and the baseline no cost setting, the payoff dominant outcome is *ex post* clearly salient: the payoff dominant (0, 0) outcome is reached in 99.3 percent rounds, respectively. The corresponding

---

[4] The lowest-to-highest entropy rankings for the single-game, no cost setting are: SH 0.07, BS 1.03, CH 1.51, and PD 1.76. Entropy is formally defined in Bednar, et al. (2010) as $H(X) = -\sum_x p(x)\log_2 p(x)$, where X is a random variables with probability density function $p(x) = \Pr\{X = x\}$.

entropy value for SH is 0.07, indicating that SH is relatively low difficulty in the no cost setting.

Strategies

Finite automata are used to implement a player's strategy. Each state of a finite automaton represents a portion of an agent's strategy by specifying the action to take when in the state and the next state to which the strategy should transition after it executes the action. This transition depends on the action taken by the player's opponent.

The desire to allow complete or partial state specificity toward games, as described above, leads to a further design consideration. In all computer experiments reported here, both agents *i* and *j* employ automata with sixteen states: $\overline{s}_i = \overline{s}_j = 16 = \overline{s} \;\; \forall \;\; i, j$. The number of states can serve as a measure of the complexity of an automaton (Rubenstein, 1998; Samuelson, 2001; Bednar and Page, 2007)[5]. Since all automata have the same number of states by design, their complexity will not depend on their aggregate structure, but rather on the degree of complexity of sub-structures within the automaton. A sixteen-state automaton is large enough for several independent sub-structures to emerge. For instance, in the two-game settings explored in this paper, a strategy may evolve into two separate sub-automata of eight states each—one specific sub-automaton for each game played. Automaton A in Figure 3.1 provides one such illustrative multiple game strategy. Alternatively, a strategy may use a common subset of strategies to play both games, as shown by Automaton B in Figure 3.1. Though this study uses four games, it only considers only two games at a time; sixteen-state automata allow enough sophistication for four simultaneous game settings to be explored at a future time.

---

[5] One weakness of this measure is that an automaton needs n + 1 states to account for n periods of memory (Bednar and Page, 2007). Rubenstein (1998) develops another measure of complexity based on the maximal order of its states.

Cognitive Costs

The primary performance statistic is profit, defined as a strategy's score net of its cost. The mean per round payoff of the stage game represents the score. Thus, a score is bounded by [1, 4]. When strategies compete in multiple, simultaneously played games, the average score all of games played, net of cost, determines profit.

The presence of cognitive costs captures the notion of bounded rationality. There are several practical reasons why the complexity of a strategy should relate to its associated cost that go beyond the simple paradigm that its uses more cognitive resources. Rubenstein (1998, p. 137) argues that "a more complex plan is more likely to break down, is more difficult to learn, and may require more time to be implemented." There are a number of conceivable ways in which to model cognitive costs. This study incorporates the presence of cognitive cost, but will sometimes contrast the results to the no cost case.

Specifically, referring to the complete framework outlined in Chapter 2, cognitive costs here are modeled by "Cost 4," which are determined by the proportion of states that are accessible in all games:

$$\text{Cost } 4 = (1 + C_4)^2 \text{ with } C_4 = \text{proportion of states accessible across all games}$$

When Automaton A in Figure 3.1 plays Battle of the Sexes, states 1-8 are all accessible; when it plays Prisoner's Dilemma, states 9-16 are all accessible. Automaton A's cognitive cost would then be based on sixteen states being accessible in all games (proportion of overall accessible states is 1.0). Specifically, its cognitive cost would be $(1 + 16/16)^2 = 4.0$. For contrast, consider Automaton B in the same figure. It uses only states 1 and 2 to play both games, so its cognitive cost would be $(1 + 2/16)^2 \cong 1.266$.

Also, the simulation adjusts each strategy's cost in order to normalize combined row and column populations' first generation average profit to zero. This procedure balances the impact of game payoffs and cognitive costs in the determination of profits.

Computer Experiments

The experiment consists of twenty-eight trials, each with a specific game, cost, and context environment. Table 3.4 summarizes these environments by trial. Each trial consists of one hundred independent runs, and each run consists of one thousand generations. Trials 1-20 cover the experiments under the natural strategy context and trials 21-28 cover the alternative cooperate/defect context. During each generation, the row and column players' automata match up to play either one or two repeated games.

During any given trial, two randomly generated populations of thirty finite automaton strategies compete against each other. A selection procedure filters out poorly performing strategies and a genetic algorithm combines successful ones into new strategies. This selection occurs at the beginning of each generation (after the first).

The end result for each trial during this phase is two populations that have adaptively gained extensive experience playing in a selected game and cost setting. For each trial, the simulation repeats this procedure one hundred times, creating many independent sets of populations with a common experience. The simulation saves the automaton structures of the populations after each run for use during the follow-on experiments (see Chapter 4).

Simulation Algorithm

The computer experiments use the following algorithm to conduct the trials. A more detailed description of this algorithm appears in Chapter 2. Table 3.5 summarizes parameter values used throughout the experiments.

Step 1: Initialize strategy populations. Row and column player strategy populations each consist of thirty randomly created automata.

Step 2: Apply selection. This simulation models selection mostly consistent with the experiments reported in Miller (1996) to include crossover rates (0.33333333) and mutation rates (0.005) and the relative fitness parameter (2). The program skips this step during the first generation and applies selection on the row and column populations independently.

Step 2.1: *Copy top performers*. The simulation selects the twenty best performing automata, based on their profit earned in the previous generation (average payoff in all rounds[6] of all games played net of cognitive cost) for the new generation without any alterations.

Step 2.2: *Add five genetically modified strategy pairs*. For the ten remaining positions in the new population, the program selects five pairs of strategies, modifies them, and inserts them into the new population.

Step 2.2.1: *Identify parent strategies*. The simulation randomly selects (with replacement) two automata. The probability for selection depends on the relative fitness (profit) of the strategies. A fitness parameter ensures that automata that perform worse than two standard deviations from the mean cannot be selected.

Step 2.2.2: *Apply state-wise crossover*[7]. The algorithm applies the crossover genetic operator directly on the automata. The program determines randomly for the two automata a common crossover point between states and crossover length ℓ. It swaps the automata's next ℓ states beyond the crossover point.[8]

Step 2.2.3: *Apply state-wise mutation*. For each state *k*, the program makes three independent draws from a uniform distribution. If the first draw is below the mutation probability threshold of 0.005, the program flips the state's action from 0 to 1 (or vice

---

[6] The profit statistic uses no time discounting and weights all rounds equally.

[7] This step is the only difference from the genetic algorithm in Miller (1996), where automata are first represented as bit strings, and then the operation is applied to the bits (as opposed to states).

[8] For this operation, the simulation treats the automata as circular so that if there are fewer than ℓ states to the end of the automata, it continues the swapping at their beginnings.

versa). If the second draw is below 0.005, the program modifies its "if opponent played action 0" transition to a randomly determined state. If the third draw is below 0.005, the program modifies its "if opponent played action 1" transition to a randomly determined state. After all states have been (potentially) modified, the program adds the automaton to the next generation's strategy pool.

Step 3: Play Games.

Step 3.1: *Match automata*. The program employs mean matching to pair automata for competition. Under this paradigm, each of the thirty automata in the row population pairs with each of the thirty automata in the column population to play all games selected for that particular trial. Thus, each automaton plays thirty or sixty repeated games in every generation (depending on whether the trial is for a one-game or two-game setting).

Step 3.2: *Play game(s)*. The matched automata play a repeated game for one hundred sixty rounds[9]. During every round, each automaton plays the action specified by its current state and then transitions to a state dependent on the play of its opponent.

In the two-game settings, simultaneous play means that an automaton plays both of the selected games before the algorithm applies selection. This ensures that the same automaton is playing the different games (with a possible distinct starting state for each game). Automata play a game for the full number of specified rounds before switching to a different game.

Step 4: Iterate or stop program. Terminate program execution if it has reached the last (1000th) generation. Otherwise, increment generation and go to Step 2 (selection).

---

[9] The paired automata play $10 \cdot \bar{s} = 160$ rounds, where $\bar{s} = 16$ is the number of their states. Using $10 \cdot \bar{s}$ rounds ensures the play extends well beyond the automata's intrinsic memory capacity (Miller, 1996). All agents report and perceive actions accurately; there is no noise.

Application of this algorithm generates strategy populations with specific game-playing experience. This is the first phase of broader research agenda. The second phase (Chapter 4) consists of experiments that match these populations against each other in both familiar and novel multiple game environments. The following section describes and explains the results of the first phase trials.

3.3. Results for the Natural Outcome Context

The results illustrate the performance of strategy populations in terms of profit, score, and cost in the various game strategic environments using the generic action 0 or 1 label consistent with the RGG taxonomy. In the results and discussions that follow, score will normally be preferred to profit as the primary measure of performance. Profit is most useful as the measure to drive the evolution of strategies since it incorporates both interactive outcomes (scores, or equivalently, payoffs) and cognitive costs. When interpreting and comparing outcomes, though, scores are more easily compared to the payoffs in the game matrices themselves, and so provide a clearer picture. Additionally, as expanded on below, the differences in costs from game to game are negligible, and so scores correlate highly with profits.

Structural characteristics of the strategy populations, also presented here, can explain much of the populations' differing performance in the various environments. The results presented below will evaluate the impact of cognitive cost on the performance and structure of strategy populations. Two tables below summarize the single-game results: Table 3.6 for cognitive cost environments and Table 3.7 for no cost environments. In these, a comparison of the fourth data column labeled "distinct states activated in all games" provides a quick snapshot of how costs drive automata to evolve to simpler structures. Overall, facing costs drives the number of activated states from 4.63 to 1.32, a reduction of 71 percent. This is expanded in Result 3.1.2 below. Tables 3.8 and 3.9 summarize the two-game results for cognitive cost and no cost environments, respectively. On the whole in the two-game settings, the number of activated states from 7.49 to 2.09, a reduction of 72 percent, in the presence of cognitive costs. Note that unless otherwise noted, all tables and figures present population results on a per automaton basis. These results are means over the one hundred runs and the thirty automata comprising the player's population in each run.

The results presented in this chapter arise from matching two new, inexperienced populations repeatedly over 1,000 generations. The purpose is to trace the evolution of the strategies, to evaluate their last generation performance and structure, and to create populations that have evolved in a particular game and cost setting. A strategy population generated during this phase of the experiment is then ready to be matched in the second phase against another one with a different game playing experience (see Chapter 4).

In addition to producing experienced populations of strategies, the experiments in this phase provide baseline values for performance and structural statistics. The results show differences between both types of statistics for strategy populations that evolved in different game environments.

Result Group 3.1 – General Results Common in One-Game and Two-Game Settings

Result 3.1.1. *Populations converge from thirty automata with entirely random structures to many copies of one or two functionally distinct automata; in about ninety of one hundred runs of a given cognitive cost trial, all thirty automata activate their states in exactly the same frequencies, and in the other ten runs, twenty-nine automata activate their states exactly the same along with only one uniquely acting automaton.*

The typical population converges rather quickly to a very small set of functionally distinct automata; in other words, redundancy of particular automata within the population of thirty is common. Most automata within a population will use the same states with the same frequencies. For example, Figure 3.2 illustrates a typical automaton strategy (this particular one is row automaton #1 from trial 17, run 1, after evolving for one thousand generations). All twenty-nine other automata in the population are identical to this one. Although a population's automata in most trials do not evolve to thirty identical structural copies, they do mostly evolve to identical operational copies. For example, the full sixteen-state automaton shown in Figure 3.2.a is rather complicated. However, once inaccessible states are removed from consideration (since they can never be activated), this automaton reduces in functionality to simple two-state automaton shown in Figure 3.2.b[10]. Most populations evolve to just one or two functionally equivalent reduced-form automata; some structural differences from automaton to automaton may occur, but these are most often in the inaccessible portion of the automaton, and therefore irrelevant. For instance, suppose row automaton #2 in the automaton #1's population was identical to it except that the right-side transition (if opponent plays action 1) from state 8 leads to state 6 instead of state 7. Then automaton #2 would function identically to #1 since it is impossible to reach state 8 in the first place.

One way to measure this convergence is by considering the standard deviation of a particular state's usage (the percentage of rounds that the automaton spent in this state) across all automata in a population during the play of a game. A low value indicates that each automaton strategy in the population uses that particular state with relatively the same frequency. One can then average the standard deviations across all states (and games in multiple-game environments) to obtain a measure of homogeneity. The statistic that measures this homogeneity is the population-level state activation similarity (PLSAS). Figure 3.3 shows that PLSAS converges rather quickly (by about the 100th generation) to its eventual 1000th generation value of about 0.22 in single game Cost 4

---

[10] The automaton in Figure 3.2.b could still be reduced further the simpleton one-state automaton that plays action 0; however, that reduction did not occur with this particular automaton in the computer experiment (trial 17, run #1, row automaton #1).

environments (the four actual values range from 0.17 to 0.27; see Table 3.6). This value means that averaged over 16 states, the standard deviation across thirty automata of a state's activation frequency is 0.22. An analysis of Battle of the Sexes, which has a PLSAS of 0.22, helps to illustrate what this value means. Of the one hundred sample runs, ninety runs have a PLSAS of 0, eight runs have PLSAS's of 2.23 or 2.24, and two other runs have PLSAS of 2.91 and 0.74. So, in ninety percent of the runs, all thirty automata in a population function in exactly the manner in the sense that they use their states in exactly the same frequencies. The automata in the runs with the non-zero PLSAS's function almost identically. For instance, if an automaton activates only two states (as most did), a PLSAS measure of 2.24 corresponds to twenty-nine of the automata behaving identically (activating the same two states each with the same frequencies) and just one errant automaton activating different states.

A typical population of thirty converges, for instance, to one or two distinct state usage patterns.[11] This homogeneity result is by design. In the selection dynamics, the algorithm copies high performing strategies directly and combines them into new ones for the next generation. The resulting population that emerges from thus experience-gaining portion of the experiment consists of a few strategies tuned for a particular game-cost environment.

Result 3.1.2: *On average, strategies use only 1.32 (of 16) states in cognitive cost settings and 4.63 states in no cost settings, leaving 92 percent and 71 percent, respectively, of their cognitive power idle*. The strategies typically use but a few states in substantial frequencies by the last generation of a run. For example, the automaton strategy in Figure 3.2.a utilizes just two states, 5 and 12. The functionally equivalent automaton has just two states, as shown in the inset, Figure 3.2.b. Several statistics

[11] There may be more truly distinct patterns; however, some minor differences stem from states that are played with very low frequencies.

reported in Tables 3.6 and 3.8 provide support that strategies use very fewer of their available states.

Two performance statistics that demonstrate this result are distinct states activated in all games and mean states activated in per game (these measures are identical in single-game environments). In the one-game, cognitive cost environments, the typical automaton strategy activates about 1.3 (out of 16) states. In two-game, cognitive cost environments, distinct states activated per game is about 1.6 and distinct states activated in all games is about 2.1. Although these strategies all have 16 states available, they usually economize by using just a few states. For example, the automaton in Figure 3.2 activated only one state (12) when it played Prisoner's Dilemma (because its opponents always played action 0) and just two states (5 and 12) when it played Battle of the Sexes. For this automaton, distinct states activated per game is 1.5 and distinct states activated in all games is 2.

These performance statistics are driven by the underlying structure of the automata, and in particular, by the number of accessible states because these determine the cognitive cost. A state *s* is accessible if, given the initial starting state, a series of transitions exists such that state *s* can be potentially reached. Whether an accessible state is actually ever activated depends on the series of actual actions taken by the two competing strategies. Referring to the automaton if Figure 3.2, it has two accessible states (5 and 12) both when playing Prisoner's Dilemma and when playing Battle of the Sexes, so 12.5 percent of its states are accessible.

For the single game environments with cognitive costs, only 9 percent of states are accessible on average. This is true for any of the four games; there is no statistically significant difference between the different games for this statistic. Similar results stem from the multiple game, cognitive cost environments. In these, accessible states are calculated per game and across all games, with the latter serving as the basis for cognitive costs. 12 percent of states are accessible per game and 14 percent are accessible across both games. Again, there is no significant variance for among the six pairings of games for either of these statistics.

Result 3.1.3: *Cognitive costs significantly affect nearly all performance and structural statistics, and the differences are most pronounced in Prisoner's Dilemma and Stag Hunt. In particular, costs reduce the number of states that can be / are activated; structurally, the proportion of accessible states falls (on average in single game settings) from 73 to 9 percent, and performance-wise, the number of distinct states activated falls from 4.63 to 1.32.* The presence of costs significantly affects both the structure and performance of strategies, as shown by a comparison of the statistics in Tables 3.6 and 3.7 (for one-game settings) and Tables 3.8 and 3.9 (for two-game settings).

For instance, a comparison of distributions of payoff matrix outcomes for one-game, cognitive cost environments (Table 3.10) and for one-game, no cost environments (Table 3.11) shows drastically different distributions for both Prisoner's Dilemma and Stag Hunt. For the former, the presence of costs increases the prevalence of the dominant strategy Nash equilibrium (the top left cell) from 34.3 percent to 97.1 percent. Accordingly, payoff efficiency falls from 54.4 to 1.4 percent[12]. Without costs, players are able to coordinate on the Pareto efficient but dominated cooperative outcome (bottom right cell) 43.1 percent of the time, but with costs this outcome almost never occurs. For the later, Stag Hunt shows an opposite flow in efficiency: without costs, players' action lead to the payoff dominant Nash equilibrium (top left cell) 99.3 percent of the time, but with costs, this percentage drops to 83.6 percent as in almost 16 percent of rounds, play gets stuck at the risk dominant Nash equilibrium (bottom right cell). Efficiency falls from 99.3 percent to 83.6 percent.

---

[12] Payoff efficiency E is the percentage of the potential joint payoff above of minimum joint payoff that players receive: $E = \dfrac{\text{actual joint payoffs} - \text{minimum joint payoffs}}{\text{maximum joint payoffs} - \text{minimum joint payoffs}}$.

The payoff matrix distributions outcomes (and payoff efficiencies) for two-game settings that include Prisoner's Dilemma or Stag Hunt follow a similar alteration as the one-game settings when comparing the no cost (Table 3.13) and cognitive cost (Table 3.12) cases.

Table 3.14 shows Welch's t-statistics for differences between all statistics in the one-game, no cost settings and the corresponding statistics in the one-game, cognitive cost settings. For Prisoner's Dilemma and Stag Hunt, all differences are statistically significant, mostly at the one percent significance level. The results are less dramatic for Chicken and Battle of the Sexes, but real differences do exist for the key statistics of proportion of accessible states (which determines cost) and profit as well as many of other the statistics.

Similar results come from the two-game environments, as shown in Table 3.15 that shows Welch's t-statistics for differences between all statistics in the two-game, no cost setting and the corresponding statistics in the two-game, cognitive cost setting. Nearly all performance statistics exhibit statistically significant differences between the no cost and cost settings: profit, distinct states activated (both in all games and per game), and state usage similarity (both automaton-level and population-level). For score and percent of rounds action 0 is played, the results are mixed: costs matter to these statistics always in Prisoner's Dilemma and Stag Hunt games, but only sometimes in Battle of the Sexes games and Chicken games. Many structural statistics are affected by the presence of cognitive costs as well. This is always true for distinct initial starting states, proportion of accessible states (both in all games and per game), proportion of terminal states, and proportion of counting states, and nearly always true for proportion of states that play Tit-for-Tat. For the other structural statistics—the proportion of initial states that play action 0, and the proportions of (regular) states that play action 0, reciprocate action 0, and reciprocate action 1—costs matter for all Prisoner Dilemma games, but only for select other games, depending on the two-game environment.

The disparity between cost and no cost environments for row populations can also be seen visually by comparing Figure 3.4's score histograms on the left-hand side (cognitive cost) to those on the right-hand side (no cost). Figure 3.5 shows similar differences for column populations.

Taken as a whole, this result is critically important because it shows that automata evolve to different structures when facing cognitive costs. Furthermore, these structural differences definitely affect the execution of the automata's strategies. By design, the cognitive costs were to matter so that mental constraints in multiple game settings would have real effects; this result affirms the design of the model. Accordingly, the remainder of the results presented below will focus mostly on those stemming from cognitive cost environments.

<u>Result 3.1.4</u>: *Row and column populations exhibit no differences as expected in the symmetric games, providing a positive validity check for the underlying framework.* Row and column populations performed similarly as expected. Any significant discrepancy would indicate a bias in the simulation programming. Figures 3.4 and 3.5 show the distribution of scores for the row populations and column populations, respectively, in single game settings. A comparison of the graphs show that the score distribution for the row population in Figure 3.4 is the nearly identical to its column population counterpart in Figure 3.5. This robustness result shows that there is no loss of generality by focusing solely of the results for the row population.

Result Group 3.2 – One-Game Environment Results

While this paper focuses on the nature of strategies in multigame environments, single game results are needed to serve as a baseline for comparison. Primary single game results are presented below; others will be presented as needed to contrast or highlight a two game environment result.

Result 3.2.1: *In the no cost settings, outcomes in SH, CH, and BS are all highly payoff efficient (greater than 97.9 percent), while outcomes in PD are modestly efficient at 54.4 percent. The introduction of cognitive costs has mixed effects on efficiency in these games: efficiency rises slightly in CH by 1.8 percentage points, does not change in BS, falls modestly in SH by 15.7 percentage points, and plummets in PD by 52.7 percentage points to near zero efficiency*. See Tables 3.10 and 3. 11. The rationale for these patterns is outlined below in the discussions of profits for each game.

Result 3.2.2: *Profits are greatest in SH, followed by BS, CH, and PD.* Profit is greatest in the game of Stag Hunt (1000th generation average profit of 3.21), followed by Battle of the Sexes (3.04), Chicken (2.62), and lastly, Prisoner's Dilemma (1.54). The differences in the profits are all statistically significant well beyond the 1 percent level (p-values less than 0.0005) except for the difference between the top two, Stag Hunt and Battle of the Sexes. Their difference is only significant at the 10 percent level (p-value 0.058).

Profit is simply score (or payoff) minus cost. Profit differences are driven by similar rankings and significance of differences with scores and nearly no difference in costs (see Table 3.6). The cognitive cost measures are nearly the same in all four single game settings. The only statistically significant (5 percent level) differences in cost are between the highest cost (in Chicken) and the two lowest costs (Battle of the Sexes and Stag Hunt); all other differences in costs are not statistically different. However, since the costs only range from 0.46 to 0.50, the small magnitude of any difference in cost is largely negligible.

Result 3.2.3: *Profits in Prisoner's Dilemma reveal that cognitive costs not only directly impact profit, but also indirectly impact it by making it more difficult for the automata to reach the cooperative outcome*. Cognitive costs push play in Prisoner's Dilemma to the Pareto dominated Nash equilibrium where both players get a payoff of 2 (the top left cell, see payoff matrix in Table 3.1). Table 3.10.a reveals this result strikingly by showing the distribution of payoff matrix outcomes for the 1000th generation. Nearly every round (97.1 percent) play results in the Nash equilibrium supported by the natural action 0. The top left graph in Figure 3.4 supports this by showing the distribution of scores.

Interestingly, without cognitive costs, strategies are more successful in escaping the Nash equilibrium trap: the mode outcome is the cooperative but dominated outcome as shown in Table 3.11.a. Payoff efficiency in this setting is 54.4 percent compared to only 1.4 percent with costs.

The costs force strategies to economize on the complexity of their automata; the mean number of states activated is 5.64 in the no cost setting but only 1.34 in the cognitive cost setting (see Table 3.6). With the simpler automata structures, it seems too difficult to coordinate on the more cooperative outcome (in this game, when each player uses action 1). For instance, in the cognitive cost setting, the action 0 usage frequency is 0.988. Given this value, if both row and column players use this value as a mixed strategy, then the player's expected score would be 2.01.[13] This is about the same as the actual average score in this game, 2.02. Thus players do not seem to be using their strategies to coordinating their actions since their performance is no better than random. A similar effect occurred in the no cost setting—the expected score of a mixed strategy playing the natural strategy action 0 with (observed average) probability 0.452 would be 2.55, and the actual score is 2.53.

The other three games do not exhibit this feature; in these games, the actual scores achieved are much greater than the expected scores. In settings with costs: (actual score) 3.51 > 2.52 (expected score) in Battle of the Sexes, 3.13 > 2.38 in Chicken, and 3.67 > 3.41 in Stag Hunt. There are similar differences in the settings without costs. In these

[13] The mixed strategy with Pr(action 0) = 0.988 in Prisoner's Dilemma yields E[score] = $(0.988)^2(2) + (0.988)(1 - 0.988)(4) + (1 - 0.988)(0.988)(1) + (1 - 0.988)^2(3) = 2.01$.

three games, automata are able to reach some degree of coordination to lift their payoffs above those that would result from random play (further evidence is presented below in the discussion of these games). In Prisoner's Dilemma, however, its single, dominant-strategy Nash equilibrium appears to prevent such coordination in the cost setting.

Result 3.2.4: *Stag Hunt profits are indirectly worsened by the pressures of cognitive costs that lower mean score from 3.98 to 3.67*. Play evolves in Stag Hunt settings to the one of the two Nash equilibria in over 99 percent of the rounds. The cooperative, payoff dominant Nash equilibrium where both players get their greatest payoff of 4 occurs most commonly, in 83.6 percent of rounds, and the risk dominant outcome where both players get a payoff of 2 occurs in 15.9 percent of the rounds (see Table 3.10.b), giving a payoff efficiency of 83.6 percent. The second row of panels in Figure 3.4 shows this information in slightly different manner by showing the distribution of scores. In the no cost environment, payoff dominant equilibrium occurs in over 99 percent of rounds in the last generation, leading to a payoff efficiency of 99.3 percent. When there are cognitive costs, evolutionary paths that initially lead to the Pareto inferior Nash equilibrium are more likely to get stuck there since it is costly to have a complex strategy that could more readily move away from this Nash equilibrium to the other that has the higher payoff.

Again, the presence of the cognitive costs created some coordination difficulties for the strategies, even in this relatively simply game that Rapoport, et al. (1976) classify as game of no conflict. Strategy complexity again helps to explain the complication. The average number states activated falls from 3.59 in the no cost setting to 1.32 in the cognitive costs setting (Table 3.6). A greater presence of terminal states also point to less complex strategies. The percent of terminal states jumps from 3 percent to 75 percent when going from no cost to cognitive cost environments[14].

Result 3.2.5: *Profits in Chicken and Battle of the Sexes follow similar patterns resulting from strategies coordinating on one of the equilibria in 97.6 percent and 99.5 percent of rounds, respectively*. These two games are very similar in nature—their paired dissimilarity score of 1 means that in the RGG taxonomy they have the same phylum, class, order, and genus[15]. Additionally, the taxonomy classifies both as unstable. So it is not surprising that the distributions of outcomes in these two games have a similar pattern: for the cognitive cost case, nearly all of the runs at one of the two Nash equilibria (see Tables 3.10.c and 3.10.d). They also have similar, near-maximal payoff efficiencies.

Another similarity is the tendency to play the cooperative action. When facing cognitive costs in CH, action 0 is played in 44.4 percent of rounds, and in BS, action 1 is played in 48.0 percent of rounds. There is no significant difference between these two values (two-tailed p-value 0.58).

What is remarkable, however, is the efficiency of coordinating on a Nash equilibrium, especially in the Battle of the Sexes. For both games, players' actions result in one or the other equilibria in nearly all rounds (97.6 percent for Chicken and 99.5 percent for Battle of the Sexes). Indeed, the players' average scores in both of these games—regardless of cost situation—are very close to the Pareto optimal scores. In Chicken, the best score that players' could jointly achieve is 3, either by alternating between the two equilibrium payoffs of 2 and 4 or by coordinating on the upper right outcome. The (row) player's average score is 3.13 in the cognitive cost situation and

---

[14] These are actually percentages of accessible states that are terminal, meaning both transitions out of the state directly back to itself.

[15] Chicken and Battle of the Sexes do differ slightly in genus in that the former exhibits "competitive pressure" while the latter does not. However, this pressure is ignored with automated agents that are presumed to lack motives. See Table 2.1 in Chapter 2.

2.96 without costs. Here, out of 100 independent runs, In Battle of the Sexes, the Pareto best average score is 3.5, achieved by alternating between the two equilibrium payoffs of 3 and 4. The (row) player's average score in this game is 3.51 with costs and 3.47 without costs (see Tables 3.6 and 3.7).

The appendix provides graphs that depict the evolution of all of the statistics for one-game environments.

Result Group 3.3 – Two-Game Environment Results

Unless otherwise noted below, the results in this section pertain to environments in which automata simultaneously play two games in the presence of cognitive costs.

Result 3.3.1: *Game pairings affect strategy performance and structure*. Sometimes in a two-game setting the automata employ distinctive sub-automaton strategies for each of the two games, and other times they employ a similar sub-automaton strategy for both games. Chicken and Battle of the Sexes have the lowest dissimilarity value of 1 (Table 3.3), and so one may expect these two games to be played quite comparable manner; in fact, they are. The dissimilarity value of 13 between Prisoner's Dilemma and Stag Hunt may suggest, then, that these two games would be played in quite different manners, but they are not. The dissimilarity score based on the Rapoport, et al. (1976) taxonomy, it turns out, does not well predict how alike two games will be played in multiple game environments. Instead, one can measure directly how the automaton played the two games.

Automaton-level state activation similarity (ALSAS) is a performance statistic that measures how similarly an automaton strategy plays two different games. State activation similarity for a single state of an automaton is measured as the standard deviation of its activation rate (the percent of rounds during which the agent activated the state) across all different types of games played. The ALSAS for the automaton as a whole is the mean of its states' activation similarities (see Chapter 2 for details). The lower the ALSAS, the more similarly the automaton plays two games.[16] ALSAS in the two-game environments are (in order from most to least similar): {CH, BS}, {PD, SH}, {PD, CH}, {SH, CH}, {PD, BS}, and {SH, BS}; the ALSAS values along with other data described below and presented in Table 3.16. The {Chicken, Battle of the Sexes} setting does indeed have a very low ALSAS (5.4), confirming that when an automaton plays these two taxonomy-similar games, it activates its states in very similar proportions. Surprisingly, at least from the high taxonomy dissimilarity score, is that {Prisoner's Dilemma, Stag Hunt} also has a low ALSAS. This discrepancy, though, is well explained by automata's tendencies to play action 0 with high frequency in both of these games as explained above in the single-game results. The other four two-game settings have increasingly greater ALSAS measures, indicating that automata are playing the two games in increasingly distinct manners.

This trend in ALSAS measures is matched by the outcome distribution similarity of both games in a set. Consider, for example, the distributions of outcomes in the {Chicken, Battle of the Sexes} environment. Here, the (0, 0) or (Up, Left) outcome occurred with frequency 1.3 percent in Chicken and 0.2 percent in Battle of the Sexes. The absolute value of the difference is 1.1 percent. Likewise, the absolute values of the difference in frequencies for the other three outcomes are 1.0 percent, 2.0 percent, and 0.0 percent. Summing these absolute differences for all four outcomes and dividing by four results in a mean absolute outcome distribution difference (AODD) of 1.0 percentage points—on average, the percentage of rounds each outcome occurs is only 1.0 percentage point different in Chicken than in Battle of the Sexes. Table 3.16 shows that

[16] The ALSAS measure has a potential shortcoming. The limitation of the ALSAS measure relates to isomorphic equivalences of different automaton strategies. For example, if the numbering of an automaton's states is altered but so too are its transitions to reflect the new numbering, then the new automaton is functionally identical to its original configuration. The ALSAS statistic would fail to recognize the equivalence.

AODD for the other five two-game sets as well; these follow in the same order as the ALSAS measures. The ALSAS measures how similarly an automaton uses its states when playing both games, and the AODD measures how similar an automaton reaches outcomes when playing both games. Since the game sets follow the same order in both measures, these measures taken together reinforce the finding that some games are played similarly and other are not.

A third statistic—distinct initial states (DIS)—less stringently supports the same finding. An automaton will either have one or two DIS; if the initialization function $\theta(\cdot)$ assigns the same state for both games, then DIS is 1, otherwise it is 2. Across the population of thirty automata (and one hundred independent runs), then, the DIS will take on a value in the range [1, 2] for any two-game environment. Table 3.16 also lists the DIS for each two-game set. Again, DIS follows the same ordering as ALSAS and AODD.[17]

Given the evidence, then, that automata play Chicken and Battle of the Sexes in nearly the same way in the {CH, BS} setting, and interesting result that contrasts emerges when one compares the degree of similarity of how automata play Chicken in {PD, CH} setting and Battle of the Sexes in the {PD, BS} setting. At first thought, one may conclude that since Chicken and Battle of the Sexes are played alike, and in these two settings they are matched with the same game, Prisoner's Dilemma, that Chicken and Battle of the Sexes will continue to be played alike. This is not the case. The mean absolute outcome distribution difference between Chicken in {PD, CH} and Battle of the Sexes in {PD, BS} is 9.8 percentage points,[18] which is greater than the 1.0 percentage point difference between the two games in the {CH, BS} environment. The main reason for the increase in the AODD measure when these games are paired with Prisoner's

[17] This is true after accounting for the fact that not all DIS are statistically distinct from each other; see note that accompanies Table 3.16.

[18] $9.8 = (|28.3 - 9.2| + |34.7 - 41.5| + |36.3 - 49.0| + |0.7 - 0.3|)/4$

Dilemma is the difference in the frequency of action 0 play and the resulting likelihood of the (0, 0) outcome. When Chicken is played within {PD, CH}, the (0, 0) outcome occurs in 28.3 percent of rounds, an increase of 27.0 percentage points from the {CH, BS} setting. When Battle of the Sexes is played within {PD, BS}, the (0, 0) outcome occurs in 9.2 percent of rounds, an increase of only 8.9 percentage points.

This difference in the jump in the (0, 0) outcome between Chicken and Battle of the Sexes is explained by considering the payoff for that outcome in each game. In Chicken, both players receive 3 at the (0, 0) outcome, and this is equivalent to the mean payoff of the two Nash equilibria—there is no drop in payoff when switching for a mean of the two equilibria and the (0, 0) outcome. The same is not true for Battle of the Sexes in which there is a drop: the mean of the two equilibria is 3.5, while the (0, 0) outcome only pays 2. Accordingly, in Battle of the Sexes, the higher score received by avoiding the (0, 0) outcome justifies an investment in a more complex strategy that can play this game distinctly how it plays Prisoner's Dilemma. A more complex strategy will mean a greater cognitive cost burden. Cost in the {PD, BS} setting is 0.514, and in the {PD, CH} setting it is only 0.466 (see Table 3.8); the difference between these two values is significantly different from zero (p-value 0.025). The greater scores incurred when playing Battle of the Sexes distinctly in Prisoner's Dilemma in {PD, BS} justifies the added cost; in {PD, CH}, there is no score advantage to playing it distinctly from Prisoner's Dilemma, and so a more complex (and costly) strategy is not justified.

Automata in {PD, BS} environments are more complex than those in {PD, CH} environments and incur higher costs. Cost is based on accessible states in all games; the {PD, BS} setting has a greater percentage of accessible states (14 percent) than the {PD, CH} setting (12 percent, p-value for difference is 0.008). Another structural statistic that correlates to complexity is the percentage of terminal states; the more terminal states, the simpler the structure. Battle of the Sexes with {PD, BS} has 41 percent terminal states[19], and Chicken within {PD, CH} has 54 percent (the difference between these two values being non-zero has a p-value of 0.060). Thus, by several measures, the automaton strategies in {PD, BS} environments are more complex than those in {PD, CH} environments. The added complexity, and corresponding added cost, is worth is for automata in {PD, BS} settings to avoid playing Battle of the Sexes like Prisoner's Dilemma (with a high frequency of action 0) which would result in subpar payoffs.

Result 3.3.2: *Game pairings most greatly affect strategies' scores in Stag Hunt*. The average score that an automaton earns when playing a particular game depends to some degree on which other game it is paired in the environment. Here, scores, and not profits, must be considered because an automaton's profit cannot be calculated for each individual game.[20] The second column of data in Table 3.8 lists the average scores that the thirty automata receive for each game by pairing (averaged over the one hundred independent runs).

As a first comparison of how pairings matter, consider the standard deviation of the scores in a particular game across the three different pairs in which it is a member. For instance, average score in Stag Hunt when paired with Prisoner's Dilemma was 3.99, when paired with Chicken it was 3.41, and when paired with Battle of the Sexes it was 3.73. The standard deviation of these three scores is 0.294. Likewise, the score standard deviation for Battle of the Sexes, Chicken, and Prisoner's Dilemma were 0.088, 0.067, and 0.054, respectively. Even with only two degrees of freedom in each sample, F-tests show that the there are significant difference (at the 10 percent level) between the score

---

[19] These figures are actually the percentage of accessible states that are terminal states, not the percentage of all states that are terminal states.

[20] Costs are determined by the proportion of accessible sates in all games, which is not calculated on a per game basis.

variance within Stag Hunt and the score variances within Prisoner's Dilemma (F-test value 0.0664) and also between Stag Hunt and Chicken (F-test value 0.0978). Figure 3.6 depicts for each game a graph that captures the spread of its scores. All four graphs in the figure have the same range on the score axis—the maximum and minimum score shown all differ by 1.0—to make visual comparisons between graphs meaningful. Clearly, Stag Hunt's scores it its three pairings have the widest spread. The gap in maximum scores in SH is 0.58, while in the other games gaps are 0.08 (PD), 0.10 (CH), and 0.16 (BS).

Differences between the mean Stag Hunt scores cited in the paragraph above are all significant at the 1% level (Table 3.17). The scores in Stag Hunt when it is paired with Prisoner's Dilemma and Chicken are also statistically distinct from scores in a single game Stag Hunt setting (at the 10 percent and 5 percent levels, respectively). A reason why there are drastic differences in automata score performance in Stag Hunt may be due to its very simple nature—recall that the entropy measure for this game is the fairly low value 0.68 in the cognitive cost setting (and 0.07 with no costs). As a game of no conflict, both players receive their greatest payoff at the same outcome when both play the natural strategy action 0. So even though it does have a second Nash equilibrium, it is fairly easy for automata to evolve to usually play action 0. In the single game Stag Hunt environment, they do just that: in 83.6 percent of all rounds, the resulting outcome is the payoff dominant Nash equilibrium that occurs when both players use the action 0. However, when an automaton also must play a second game, the strategic situation is no longer so simple; the presence of the second game can enhance or inhibit a strategy's effectiveness.

<u>Result 3.3.3</u>: *Pairing Prisoner's Dilemma with Stag Hunt creates a strategic complementarity that increases scores in both games relative to the single-game environments.* In some cases, the strategic nature of each game in the environment may correlate to improve performance in one or both games. Such is the case with the

{Prisoner's Dilemma, Stag Hunt} environment. The scores in this environment are 2.05 for Prisoner's Dilemma and 3.99 for Stag Hunt (see Table 3.8), for an average of 3.02. In single game environments, the scores are 2.02 for Prisoner's Dilemma and 3.67 for Stag Hunt (see Table 3.6), for an average of 2.85—all less than in the two-game {PD, SH} setting. This difference in scores between Stag Hunt when paired with Prisoner's Dilemma and when Stag Hunt is played by itself is significant at the 1 percent level (see Table 3.17). So too is the difference between the average of the two scores in the two-game environment and the average of the two one-game environment scores (t-statistic of -4.66, see Figure 3.7). The difference in scores between Prisoner's Dilemma when paired with Stag Hunt and when Prisoner's Dilemma is played by itself is even fairly significant (p-value of 0.120), though not large in magnitude (an improvement to the score of only 0.03).

A comparison of the single game outcome distributions in Table 3.10.a and 3.10.b to those for the {Prisoner's Dilemma, Stag Hunt} setting in Table 3.12.a illustrates why the scores are greater in the two-game setting. For scores in Prisoner's Dilemma, the occurrence of the (0, 0) Nash equilibrium outcome (where both players receive payoff 2) drops from 97.1 percent to 91.4 percent. The other three outcomes pick up the extra 5.7 percentage of outcomes in roughly even proportions, for which the average payoff is 3; thus score rises modestly. At the same time, payoff efficiency increases somewhat from 1.45 percent to 5.51 percent. Stag Hunt's presence causes strategies to use natural action 0 *less* frequently in Prisoner's Dilemma, improving scores in that game as it uses a more cooperative strategy.

Conversely, Prisoner's Dilemma presence causes strategies to use natural strategy action 0 *more* frequently in Stag Hunt (compared to the single-game Stag Hunt setting), also improving scores in that game. For automata playing Stag Hunt, the complication of also having to play Prisoner's Dilemma improves scores outcomes in Stag Hunt because it increases the frequency that the natural action 0 is played from 83.9 to 99.8 percent.

There is a much smaller chance that the strategy will evolve to play the Pareto inferior, risk dominant Nash equilibrium. This bears out in the in the outcome distributions: in the two-game setting with Prisoner's Dilemma, play in Stag Hunt ends in the payoff dominant equilibrium 99.6 percent of rounds and in the risk dominant equilibrium in 0.0 percent of rounds. These frequencies starkly contrast with those from the solo Stag Hunt setting (83.6 percent for payoff dominant equilibrium and 15.9 percent for risk dominant equilibrium). Payoff efficiencies rise from 83.59 percent in the single-game SH setting to 99.6 percent when SH is played as part of the {PD, SH} game ensemble.

Stag Hunt has enough complexity (due to the second Nash equilibrium) that when cognitive resources must be expended to play, a player may get trapped in a suboptimal strategy. Pairing this game with Prisoner's Dilemma creates a positive strategic coupling and improves performance in at least one and possibly both games. In Prisoner's Dilemma, the dominant strategy Nash equilibrium pulls play toward action 0, which improves performance in Stag Hunt.

Investigating a multiple game environment with these strategic complementarities serves as a tangible way to model corporate "synergies." This term, which has been in vogue in the corporate world for some time, is defined by *The Oxford English Dictionary* as "increased effectiveness, achievement, etc., produced as a result of combined action or co-operation." This realization creates several applications for the use of multiple game models, from teamwork models in management to merger and acquisition models to industrial organization.

<u>Result 3.3.4</u>: *Pairing Stag Hunt with Chicken creates a negative strategic coupling that decreases scores in both games relative to the single-game environments*. In other cases, the strategic nature of each game in the environment may correlate to degrade performance in one or both games. This is the case with the {Stag Hunt, Chicken} environment. Payoff efficiencies fall in both games in the ensemble relative to the single-game settings: for SH, from 83.6 percent to 69.6 percent, and for CH, from

99.7 percent to 97.2 percent. This two-game setting also produces distinct differences in scores from the single-game settings. The scores in the coupled environment are 3.41 for Stag Hunt and 2.90 for Chicken (see Table 3.8), for an average of 3.15. In single game environments, the scores are 3.67 for Stag Hunt and 3.13 for Chicken (see Table 3.6), for an average of 3.40—all greater than in the two-game {SH, CH} setting. This difference in scores between Stag Hunt when paired with Chicken and scores when Stag Hunt is played by itself is significant at the 5 percent level (see Table 3.17 and Figure 3.6). Also significant at the 10 percent level is the difference in scores between Chicken when paired with Stag Hunt and scores when Chicken is played by itself. Likewise, the difference between the average of the two scores in the two-game environment and the average of the two one-game environment scores is significant at the 1 percent level (t-statistic of -2.75, see Figure 3.7).

Why is it that pairing the Stag Hunt and Chicken games into the same strategic environment hampers automata's ability to score as well? In the pairing of Prisoner's Dilemma and Stag Hunt, the presence of PD helped strategies play the natural action (0) in SH more often, and so play in paired Stag Hunt avoided the inferior equilibrium more often. In the pairing of Stag Hunt with Chicken, by contrast, the presence of each game pushes the other to play the natural action 0 at an inefficient level. Looking at one-game environment results in Table 3.6, automata play action 0 relatively often in Stag Hunt (frequency of 0.84) and play it relatively seldom in Chicken (0.44). When paired, Chicken's presence decreases the frequency the natural action 0 is played in Stag Hunt to 0.73, while Stag Hunt's presence increases the frequency the natural action 0 is played in Chicken to 0.64 (see Table 3.8). Each game is inhibiting optimal play in the other.

The effect of these different rates of playing action 0 in the paired environment as compared to the individual game environments is reflected in the outcome distributions of Table 3.12.d. When paired with Chicken, play in Stag Hunt ends in the risk dominant/Pareto inferior Nash equilibrium 22.6 percent of the time, up from the 15.9

percent from the Stag Hunt only setting (Table 3.10.b). Additionally, a total of 7.8 percent of rounds end at one of the two non-equilibrium outcomes that are least efficient in terms of payoffs (with an average of 1.5); this frequency, too, is greater than the comparable 0.5 percent for the Stag Hunt single game setting.

Stag Hunt's presence also has significant influence on the distribution of payoff matrix outcomes in Chicken. When played by itself, players in Chicken reach one of the two Nash equilibria in 97.6 percent of rounds (Table 3.10.d). Once Stag Hunt is included in the mix, the figure drops to 71.5 percent (3.12.d). Admittedly, the effect of this shift on score is minimal because most of the shift goes to the (0, 0) outcome cell at which both players receive a payoff of 3, the same as the average of the two equilibria. More damaging to the player's average score is the 12.4 percentage point shift to the (1, 1) outcome cell at which both players receive a payoff of 1; the increase is 2.5 percentage points.

3.4. Results for the Cooperate/Defect Context

This section considers an alternative, and albeit more common in the literature, interpretation of actions available to the players. Here, a player's actions are viewed as other-regarding (cooperate, denoted by "C") or as self-regarding (defect, denoted by "D"). Table 3.2 presents the payoffs and features for the three games under these new action connotations, $PD^{\#}$, $SH^{\#}$, and $CH^{\#}$. BS is excluded in this section because it lacks clear associations of its actions with these contexts. With these payoff matrices, now action 0 corresponds always to action C. Note that the only new configuration is for Prisoner's Dilemma; $SH^{\#}$ and $CH^{\#}$ are identical to SH and CH from Section 3.3. Within the motivating story for each of these three games, C and D take on even more specific meanings, though the other-regarding and self-regarding interpretations will suffice for the analysis that follows.[21]

The purpose of these computer experiments under the alternative contexts is two-fold. First, these will be a robustness check on the results presented in section 3.3. The expectation is that the game ensembles involving PD# will have distinctly different results than those involving PD. This anticipated distinction leads to the second reason for these further experiments: to gain an initial understanding of how and why *context* matters. In multiple game settings, there may not always, or even often, be a common action context to apply across games; this complication becomes more likely as the size of the game ensemble grows.

Other than the modified payoff matrix for Prisoner's Dilemma, the remainder of the experimental design is the same as presented in section 3.2. The additional trials for the computer experiment are detailed in the right-hand side of Table 3.4.

Prisoner's Dilemma is the only game to change between the 0/1 action context and the C/D action context. In a single-game environment, the context or labeling of the actions is trivial; because the PD# game is an isomorphism of PD, strategy performance and structure should be the same after evolving in either game. A comparison of the statistics in from the 0/1 contexts in Tables 3.6 and 3.7 to those from the C/D contexts in Tables 3.18 and 3.19 bears this out (for cognitive cost and no cost settings, respectively). The only differences in the statistics stem from the relabeling of the actions: action 0 in the original experiments was the defect action, and in this alternative context, action 0 is the cooperative action. So, the action 0 usage frequency of 99 percent in the original cognitive cost setting means the cooperative action 1 was used in 1 percent of rounds, and

---

[21] In PD#, action C ~ "confess" and action D ~ "don't confess;" in SH#, action C ~ "high effort" and action D ~ "low effort;" and in CH#, action C ~ "swerve" and action D ~ "straight."

this is consistent with the action C usage frequency of 1 percent in the alternative C/D context experiment. Similarly, in the no cost setting, action 0 (D) usage frequency of 45 percent in the original context is consistent with the action C usage frequency of 55 percent in the alternative C/D context. Likewise, outcome distributions for the single-game $PD^{\#}$ settings are very similar in those for the PD settings: compare the 0/1 context Tables 3.10.a and 3.11.a to the C/D context Tables 3.22.a and 3.23.a.[22]

More interesting are the two-game setting results where context *does* matter. Unless otherwise noted below, these results pertain to environments in which automata simultaneously play two games in the presence of cognitive costs under the alternative C/D context.

Result Group 3.4 – Two-Game Environment Results for C/D Context

Result 3.4.1: *Action contexts affect strategy performance and structure*. The switch in context from the natural outcome paradigm (in which action 0 in every game that leads to its natural outcome) to the other- and self-regarding paradigm essentially transposes the payoffs in PD to those in $PD^{\#}$. In PD, strategies evolve to play the natural action 0 (that corresponds to defect) with high frequency—99 percent when played in isolation (Table 3.6), 95 percent when paired with SH, and 79 percent when paired with CH (see Table 3.8). Now that the Prisoner's Dilemma payoff matrix is transposed in the C/D context, one would expect action D to continue to be used with high frequency, and it is: in the solo $PD^{\#}$ setting with costs, it continues to be used in 99 percent of rounds (Table 3.18). Accordingly, the game ensembles {$PD^{\#}$, $SH^{\#}$} and {$PD^{\#}$, $CH^{\#}$} induce

---

[22] The outcome distribution tables for SH (3.10.b and 3.11.b) are identical to those for $SH^{\#}$ (3.22.b and 3.23.b)—they aggregate the same data—as are the outcome distribution tables for CH (3.10.c and 3.11.c) and $CH^{\#}$ (3.22.c and 3.23.c). Likewise, the rows for $SH^{\#}$ and $CH^{\#}$ in Tables 3.18 and 3.19 present the same data as the SH and CH rows in Tables 3.6 and 3.7. The data are repeated to ease comparisons within the C/D context. The two-game setting statistics and outcome distribution tables for the C/D context (Tables 3.20, 3.21, 3.24, and 3.25) also repeat the data for {SH, CH} as {$SH^{\#}$, $CH^{\#}$}.

quite different results for several statistics. Tables 3.20 (cognitive costs) and 3.21 (no costs) present the automata population means for the performance and structural statistics. A comparison against their counterpart tables from the natural outcome context (3.8 and 3.9) reveals many distinctions. Table 3.26 evaluates these differences by presenting the two-tailed t-statistics that the difference between a given statistic in the natural outcome context and the same statistic in the cooperate/defect context is non-zero.[23]

The first result that stands out—as expected—is the extremely significant differences for all of the action 0 / action C related statistics: actual action 0/C usage frequency, states that play action 0/C, initial states that play this action, and states that reciprocate this action. Because the change of contexts implies lower payoffs for 0/C in $PD^{\#}$ than in PD as this becomes the dominated action, action 0/C is used less in the new context than the original (as indicated by the positive t-statistics). This effect spills over into the tandem games as well: both $SH^{\#}$ and $CH^{\#}$ in the $PD^{\#}$ ensembles show lower potential and actual usage of action 0/C as compared to the natural outcome contexts.

Because $PD^{\#}$ pushes action D, there are now vastly fewer states that play action C, and so the number of states that reciprocate action C (0) drops dramatically. Similar reasoning explains why the number of D-reciprocating states rises (indicated by positive t-statistics) just as dramatically as the context switches from 0/1 to C/D actions. The most interesting of the other significant differences found in Table 3.26 apply to specific game settings are discussed separately below.

---

[23] The payoff matrix for Stag Hunt is the same under either context; the same is true of Chicken. The statistics for $SH^{\#}$, $CH^{\#}$, and {$SH^{\#}$, $CH^{\#}$} in are the same Tables 3.22, 23, 24, and 25 are the same as in natural outcome context tables (indeed, the data come from the same computer experiments)—the data is repeated to aid comparisons within the C/D context. To avoid triviality, these games are not included in Table 3.26.

Result 3.4.2: *The strategic complementarity between Prisoner's Dilemma with Stag Hunt under the natural outcome context (Result 3.3.3) unravels under the cooperate / defect context.* First, note that the joint profit is 2.55 in {PD, SH}, and that is significantly higher than the 1.98 in {PD$^{\#}$, SH$^{\#}$} (two-tailed p-value of 0.000). Costs are nearly identical in the two paradigms (accessible states in all games, upon which costs are based, are very close—12.2 percent and 12.8 percent). So, the distinction in profit originates in score (payoff) differences, but asymmetrically.

Scores in PD$^{\#}$ are actually higher than in PD (2.17 v. 2.05, p-value for non-zero difference is 0.051). In {PD, SH}, action 0 was the natural action in both games. In SH, this action leads to the greatest payoff outcome (4, 4) for both players, and as their strategies develop to play it with high frequency, it supports the selection of the natural outcome with payoffs of (2, 2) in PD, the dominant strategy equilibrium. Now in {PD$^{\#}$, SH$^{\#}$}, the same cooperative action in SH$^{\#}$ pushes play in PD$^{\#}$ away from the dominant strategy equilibrium and towards the Pareto superior (3, 3) outcome. Compare Table 3.12.a to 3.24.a: in Prisoner's Dilemma, the Pareto superior outcome's occurrence increases from 2.5 percent to 11.8 percent as the context of the "up" action changes from 0 to C. Payoff efficiency improves from 5.5 percent to 16.6 percent.

Thus, in the natural outcome context, SH's presence hampers' players ability to escape the Pareto inferior Nash equilibrium, but just the opposite is true in the cooperate / defect context where SH$^{\#}$ presences improves outcomes in PD$^{\#}$. This modest improvement in score from 2.05 to 2.17 going from PD to PD$^{\#}$, however, is dwarfed by the much larger drop in score from 3.99 to 2.75 going from SH to SH$^{\#}$. In the change of contexts from 0/1 to C/D, payoff efficiency in Stag Hunt falls from 99.6 percent to 37.6 percent. The tendency for strategies to evolve to play action D with high frequency in PD$^{\#}$ now spills over in SH$^{\#}$ as a much greater likelihood to reach the Pareto inferior, risk-dominant equilibrium, now reached in a majority of rounds (60.9 percent).

One last item to note about the {Prisoner's Dilemma, Stag Hunt} setting under the two different contexts. Under the natural outcome context, as action 0 is the "natural" strategy is both games, one would expect automata to be more likely to economize on the number of accessible states by having a common initial state than under the cooperate / defect context. This expectation is confirmed in Table 3.26 by the negative t-statistic for number of distinct initial states (p-value 0.98): mean number of distinct initial states in the natural outcome is 1.41 and in the cooperate / defect context is 1.53. An automaton strategy is more likely to start each game with a distinct mental subroutine.

3.5. Remarks

This model of multiple game settings combines a conceptually simple game construct with a rich set of performance and structure statistics. It is an important contribution to our ability to evaluate and understand common strategic situations that involve agents multitasking. It also is useful for investigating potential synergies that do or do not arise as firms or other entities seek to take advantage of economies of scope.

In particular, this paper convinces that cognitive costs do affect performance of strategies and thus adaptation over time. The results of the model show that multiple game environments are in some cases very different from single game ones. Strategic complementarities can arise, such as in the Prisoner's Dilemma and Stag Hunt pairing, where the presence of the second game creates a synergy that improves performance. Sometimes, the coupling effects are negative, like the Stag Hunt and Chicken pairing. In other cases, and multiple game nature of the environment makes little difference, like in the Stag Hunt and Battle of the Sexes coupling. This model can be used to investigate which of these situations would arise in other strategic environments.

This paper also establishes that context of actions is critically important when modeling multiple game environments. If common mental subroutines (proxied here by states) are to be applied across different strategic settings, one must think carefully about what “common” means—the answer to that question has significant implications for strategy structure and performance. It does not seem to require a one-size-fits-all answer, either. The characteristics of the real world phenomena the modeler is investigating may dictate the appropriate context in which to frame the agents’ actions.

Extensions

Some further steps toward that theory would extend this model to new experiments, both on the computer and in the laboratory. An obvious supplement would be to search for more strategic complements and substitutes among broader class of games.

As noted in Section 3.2, the current experiments interpreted simultaneous play to mean that an agent played multiple games in sequence—one complete game and then another—before the model updated its pool of strategies. Simultaneity here means that an unchanged set of automata played each game.

Agents could play multiple games in a different fashion. Instead of playing each of $\bar{g}$ games separately for *r* rounds, strategies could play $\bar{g} * r$ rounds where the game played each round varies either deterministically (as in Güth 2000b) or stochastically. Conceivably, there are cases where this would be the more realistic model. It implies that an agent’s next action in one repeated game depends on its opponent’s current action in a different game. This interpretation of simultaneity may lead to different conclusions.

Another natural extension of this model is to replicate the experiment in the laboratory with human subjects along the lines of Bednar et al. (2010). One could relax the strict assumptions placed on strategic form and modify or eliminate the automaton nature of strategies. Lab experiments would provide further insights to the development of a general theory that computer simulations alone cannot produce.

Perhaps the most interesting direction for further research suggested by this paper, though, deals with the issue of context. A study designed specifically to investigate the effects of context in multiple game environments would be a critical addition to this nascent field.

Table 3.1. Prisoner's Dilemma, Stag Hunt, Chicken, and Battle of the Sexes in the natural outcome context.

For all four games, action 0 corresponds to the natural strategy consistent with the Rapoport, Guyer, and Gordon (1976) taxonomy.

Prisoner's Dilemma (PD)
RGG #12

	0	1
0	**2, 2**	*4, 1*
1	*1, 4*	*3, 3*

Stag Hunt (SH)
RGG #61

	0	1
0	***4,4***	1, 3
1	3, 1	2, 2

Chicken (CH)
RGG #66

	0	1
0	***3, 3***	*2, 4*
1	*4, 2*	1, 1

Battle of the Sexes (BS)
RGG #69

	0	1
0	**2, 2**	*4,3*
1	*3, 4*	1, 1

<u>Legend</u>
Bold type indicates the natural outcome, the (0, 0) of each game[24]
Italic type indicates Pareto optimal outcomes
◯ indicates a Nash equilibrium of the stage game

Table 3.2. Prisoner's Dilemma, Stag Hunt and Chicken in the cooperate/defect context.

For all three games, action C corresponds to the cooperative or other-regarding strategy and action D corresponds to defecting or self-regarding strategy.

PD#

	C	D
C	*3, 3*	*1, 4*
D	*4, 1*	2, 2

SH#

	C	D
C	*4, 4*	1, 3
D	3, 1	2, 2

CH#

	C	D
C	*3, 3*	*2, 4*
D	*4, 2*	1, 1

<u>Legend</u>
Italic type indicates Pareto optimal outcomes
◯ indicates a Nash equilibrium of the stage game

[24] See footnote 2.

Table 3.3. Dissimilarity measures between games under the natural outcome context.

	PD	SH	CH	BS
PD	0	13	12	12
SH	13	0	10	10
CH	12	10	0	1
BS	12	10	1	0

Table 3.4. Game and cost specification of experiment trials.

Natural outcome context				**Cooperate/defect context**[25]			
Cognitive Costs		No Costs		Cognitive Costs		No Costs	
Trial	Game(s) Played	Trial	Game(s) Played	Trial	Game(s) Played	Trial	Game(s) Played
1	PD	11	PD	21	$PD^{\#}$	25	$PD^{\#}$
2	SH	12	SH	2	$SH^{\#}$	12	$SH^{\#}$
3	CH	13	CH	3	$CH^{\#}$	13	$CH^{\#}$
4	BS	14	BS				
5	PD & SH	15	PD & SH	22	$PD^{\#}$ & $SH^{\#}$	26	$PD^{\#}$ & $SH^{\#}$
6	PD & CH	16	PD & CH	23	$PD^{\#}$ & $CH^{\#}$	27	$PD^{\#}$ & $CH^{\#}$
7	PD & BS	17	PD & BS				
8	SH & CH	18	SH & CH	8	$SH^{\#}$ & $CH^{\#}$	18	$SH^{\#}$ & $CH^{\#}$
9	SH & BS	19	SH & BS				
10	CH & BS	20	CH & BS				

[25] The payoffs for $SH^{\#}$ and $CH^{\#}$ are identical to SH and CH, so trials 2-3 and 12-13 also represent the single-game trials for $SH^{\#}$ and $CH^{\#}$.

Table 3.5. Key simulation parameters common to all trials.[26]

Type	Parameter	Value	Explanation
General	runs	100	Independent runs per trial
	generations	1,000	Evolutionary generations per run
	popSize[27]	30	Automata in row and column populations
	states[27]	16	States per automaton
Game	rounds	160	Rounds per game
	noise	0	Probability opponent's action is misreported.
	discount	1	Time discount factor
	matchType	mean	Each round, each row population automaton is matched against every column population automaton.
Cost	costParA4	1	Use overall accessible states for cost
	costParB4	2	Cost increases quadraticly
	costParA0,A1,A2,A3	0	Cost 1, Cost 2, and Cost 3 are not used.
	costAdjustment	-1 (common, additive)	The first generation's average profit is normalized to zero.
Genetic Algotrithm	millerCrossover	false	Use state-wise (as opposed to bit-wise) crossover.
	relativePerformance	2	factor in automaton fitness formula
	mutationRate[27]	0.005	Probability or mutation for each state's action, transition to states.
	crossoverRate[27]	0.33333333	Proportion of the population to which crossover is applied.
	maxCrossoverLength[27]	15	Maximum number of states to swap in crossover operation.

[26] See Appendix to Chapter 2 for detailed definitions of these parameters.
[27] Same parameter values for row and column populations.

Table 3.6. Statistics in one-game, cognitive cost settings for the natural outcome context.

Game	Performance statistics: Profit	Score	Cost	Distinct states activated in all games	Population-level state activation similarity	Action 0 usage frequency	Structural statistics: Proportion of states that are… Accessible states	Terminal states	Counting states	0-playing initial states	0-playing states	0-reciprocating states	1-reciprocating states	Tit-for-tat states
PD	1.54	2.02	0.48	1.34	0.24	0.99	0.09	0.68	0.15	0.99	0.99	1.00	0.01	0.01
SH	3.21	3.67	0.46	1.32	0.17	0.84	0.09	0.75	0.12	0.78	0.80	0.81	0.20	0.04
CH	2.62	3.13	0.50	1.36	0.27	0.44	0.09	0.68	0.14	0.43	0.42	0.44	0.61	0.06
BS	3.04	3.51	0.47	1.25	0.22	0.52	0.08	0.78	0.11	0.54	0.53	0.55	0.48	0.05
Overall Mean	2.60	3.08	0.48	1.32	0.22	0.70	0.09	0.72	0.13	0.68	0.69	0.70	0.33	0.04

Means for the 30 automata in row population in the 1000th generation, averaged over 100 independent runs.

Table 3.7. Statistics in one-game, no cost settings for the natural outcome context.

Game	Performance statistics: Profit	Score	Cost	Distinct states activated in all games	Population-level state activation similarity	Action 0 usage frequency	Structural statistics: Proportion of states that are… Accessible states	Terminal states	Counting states	0-playing initial states	0-playing states	0-reciprocating states	1-reciprocating states	Tit-for-tat states
PD	2.53	2.53	n/a	5.64	1.15	0.45	0.75	0.00	0.08	0.73	0.69	0.75	0.35	0.24
SH	3.98	3.98	n/a	3.59	0.67	1.00	0.76	0.03	0.05	0.90	0.64	0.73	0.37	0.26
CH	2.96	2.96	n/a	5.31	1.06	0.54	0.69	0.03	0.09	0.52	0.45	0.47	0.58	0.22
BS	3.47	3.47	n/a	3.95	0.60	0.48	0.74	0.02	0.06	0.48	0.57	0.53	0.39	0.18
Overall Mean	3.24	3.24	n/a	4.63	0.87	0.62	0.73	0.02	0.07	0.66	0.59	0.70	0.42	0.23

Means for the 30 automata in row population in the 1000th generation, averaged over 100 independent runs.

Table 3.8. Statistics in two-game, cognitive cost settings for the natural outcome context.

		Performance statistics:								Structural statistics:									
											Proportion of states that are…								
Games(s)		Profit	Score	Cost	Distinct states activated in all games	Distinct states activated per game	Automaton-level state activation similarity	Population-level state activation similarity	Action 0 usage frequency	Distinct initial states	Accessible states in all games	Accessible states per game	Terminal states	Counting states	0-playing initial states	0-playing states	0-reciprocating states	1-reciprocating states	Tit-for-tat states
{PD, SH}	**PD** in PD-SH	n/a	2.05	n/a	n/a	1.63	n/a	0.28	0.95	n/a	n/a	0.12	0.43	0.22	0.94	0.96	0.98	0.05	0.05
	SH in PD-SH	n/a	3.99	n/a	n/a	1.61	n/a	0.31	1.00	n/a	n/a	0.12	0.43	0.22	0.97	0.96	0.98	0.05	0.05
	Mean	2.55	3.02	0.48	1.84	1.62	8.60	0.29	0.97	1.41	0.12	0.12	0.43	0.22	0.95	0.96	0.98	0.05	0.05
{PD, CH}	**PD** in PD-CH	n/a	2.16	n/a	n/a	1.52	n/a	0.21	0.79	n/a	n/a	0.10	0.54	0.20	0.80	0.79	0.81	0.23	0.08
	CH in PD-CH	n/a	3.00	n/a	n/a	1.51	n/a	0.24	0.63	n/a	n/a	0.11	0.54	0.19	0.53	0.61	0.64	0.40	0.08
	Mean	2.12	2.58	0.47	1.85	1.52	27.5	0.23	0.71	1.47	0.12	0.10	0.54	0.19	0.67	0.70	0.73	0.31	0.08
{PD, BS}	**PD** in PD-BS	n/a	2.08	n/a	n/a	1.65	n/a	0.31	0.84	n/a	n/a	0.12	0.48	0.15	0.92	0.87	0.88	0.15	0.11
	BS in PD-BS	n/a	3.32	n/a	n/a	1.61	n/a	0.25	0.51	n/a	n/a	0.13	0.41	0.16	0.56	0.65	0.67	0.28	0.10
	Mean	2.19	2.70	0.51	2.16	1.63	45.8	0.28	0.67	1.62	0.14	0.12	0.45	0.15	0.74	0.76	0.78	0.22	0.10
{SH, CH}	**SH** in SH-CH	n/a	3.41	n/a	n/a	1.49	n/a	0.21	0.73	n/a	n/a	0.13	0.35	0.18	0.64	0.54	0.58	0.52	0.21
	CH in SH-CH	n/a	2.90	n/a	n/a	1.66	n/a	0.24	0.64	n/a	n/a	0.13	0.34	0.19	0.52	0.52	0.52	0.52	0.18
	Mean	2.63	3.15	0.52	2.17	1.57	45.0	0.22	0.68	1.67	0.15	0.13	0.34	0.19	0.58	0.53	0.55	0.52	0.20
{SH, BS}	**SH** in SH-BS	n/a	3.73	n/a	n/a	1.71	n/a	0.28	0.88	n/a	n/a	0.14	0.33	0.15	0.74	0.69	0.75	0.33	0.16
	BS in SH-BS	n/a	3.48	n/a	n/a	1.68	n/a	0.30	0.48	n/a	n/a	0.15	0.35	0.14	0.51	0.59	0.64	0.38	0.14
	Mean	3.05	3.61	0.55	2.50	1.70	62.5	0.29	0.69	1.74	0.16	0.14	0.34	0.15	0.63	0.64	0.69	0.35	0.15
{CH, BS}	**CH** in CH-BS	n/a	3.02	n/a	n/a	1.77	n/a	0.40	0.49	n/a	n/a	0.13	0.36	0.23	0.44	0.47	0.49	0.55	0.08
	BS in CH-BS	n/a	3.46	n/a	n/a	1.78	n/a	0.37	0.47	n/a	n/a	0.13	0.36	0.23	0.47	0.47	0.49	0.54	0.09
	Mean	2.76	3.24	0.48	1.98	1.78	5.42	0.38	0.48	1.48	0.13	0.13	0.36	0.23	0.46	0.47	0.49	0.55	0.09
Overall Mean		2.55	3.05	0.50	2.09	1.64	32.5	0.28	0.70	1.57	0.14	0.12	0.41	0.19	0.67	0.68	0.70	0.33	0.11

Means for the 30 automata in the row population in the 1000th generation, averaged over 100 independent runs.

Table 3.9. Statistics in two-game, no cost settings for the natural outcome context.

Games(s)		Performance statistics:								Structural statistics:										
											Proportion of states that are…									
		Profit	Score	Cost	Distinct states activated in all games	Distinct states activated per game	Automaton-level state activation similarity	Population-level state activation similarity	Action 0 usage frequency	Distinct initial states	Accessible states in all games	Accessible states per game	Terminal states	Counting states	0-playing initial states	0-playing states	0-reciprocating states	1-reciprocating states	Tit-for-tat states
{PD, SH}	**PD** in PD-SH	n/a	2.60	n/a	n/a	5.14	n/a	1.11	0.40	n/a	n/a	0.75	0.01	0.07	0.60	0.69	0.80	0.36	0.28
	SH in PD-SH	n/a	3.99	n/a	n/a	3.48	n/a	0.97	1.00	n/a	n/a	0.74	0.03	0.07	0.80	0.71	0.81	0.33	0.26
	Mean	3.29	3.30	n/a	7.28	4.31	84.5	1.04	0.70	1.97	0.78	0.74	0.02	0.07	0.70	0.70	0.80	0.34	0.27
{PD, CH}	**PD** in PD-CH	n/a	2.47	n/a	n/a	6.87	n/a	1.82	0.54	n/a	n/a	0.70	0.03	0.08	0.67	0.60	0.67	0.41	0.27
	CH in PD-CH	n/a	3.01	n/a	n/a	7.17	n/a	1.77	0.58	n/a	n/a	0.70	0.04	0.08	0.54	0.55	0.59	0.47	0.26
	Mean	2.74	2.74	n/a	9.52	6.99	52.2	1.80	0.56	1.96	0.77	0.70	0.04	0.09	0.60	0.58	0.63	0.44	0.26
{PD, BS}	**PD** in PD-BS	n/a	2.44	n/a	n/a	5.11	n/a	1.02	0.56	n/a	n/a	0.65	0.07	0.07	0.73	0.70	0.76	0.33	0.22
	BS in PD-BS	n/a	3.45	n/a	n/a	4.38	n/a	0.64	0.50	n/a	n/a	0.73	0.02	0.07	0.55	0.64	0.68	0.38	0.23
	Mean	2.95	2.95	n/a	7.20	4.74	58.8	0.83	0.53	1.95	0.77	0.69	0.05	0.07	0.64	0.67	0.72	0.36	0.22
{SH, CH}	**SH** in SH-CH	n/a	3.98	n/a	n/a	3.18	n/a	0.86	0.99	n/a	n/a	0.76	0.02	0.05	0.84	0.58	0.65	0.48	0.30
	CH in SH-CH	n/a	2.90	n/a	n/a	5.78	n/a	1.65	0.77	n/a	n/a	0.75	0.01	0.06	0.53	0.55	0.61	0.48	0.28
	Mean	3.44	3.44	n/a	7.35	4.48	63.5	1.25	0.88	1.98	0.80	0.75	0.02	0.06	0.69	0.57	0.63	0.48	0.29
{SH, BS}	**SH** in SH-BS	n/a	3.98	n/a	n/a	3.35	n/a	0.57	0.99	n/a	n/a	0.71	0.03	0.07	0.77	0.65	0.69	0.33	0.20
	BS in SH-BS	n/a	3.49	n/a	n/a	4.63	n/a	0.79	0.53	n/a	n/a	0.73	0.03	0.06	0.53	0.64	0.68	0.34	0.20
	Mean	3.74	3.74	n/a	6.81	3.99	89.7	0.68	0.76	1.93	0.77	0.72	0.03	0.07	0.65	0.65	0.68	0.33	0.20
{CH, BS}	**CH** in CH-BS	n/a	2.93	n/a	n/a	4.93	n/a	1.18	0.53	n/a	n/a	0.65	0.05	0.08	0.51	0.47	0.45	0.52	0.15
	BS in CH-BS	n/a	3.53	n/a	n/a	4.48	n/a	0.91	0.55	n/a	n/a	0.67	0.04	0.07	0.52	0.53	0.52	0.46	0.16
	Mean	3.23	3.23	n/a	6.78	4.70	33.1	1.05	0.54	1.95	0.73	0.66	0.04	0.08	0.51	0.50	0.48	0.49	0.16
Overall Mean		3.23	3.23	n/a	7.49	4.87	62.7	1.11	0.66	1.96	0.77	0.71	0.03	0.07	0.63	0.61	0.66	0.41	0.23

Means for the 30 automata in the row population in the 1000th generation, averaged over 100 independent runs.

Table 3.10. Outcome distributions in one-game, cognitive cost settings for the natural outcome context.

Table 3.10.a.
Outcomes in Prisoner's Dilemma

PD		Column player **0**	Column player **1**
Row player	**0**	97.1%	1.7%
Row player	**1**	1.2%	0.0%

efficiency: 1.4%
entropy: 0.22

Table 3.10.b.
Outcomes in Stag Hunt

SH		Column player **0**	Column player **1**
Row player	**0**	83.6%	0.3%
Row player	**1**	0.2%	15.9%

efficiency: 83.6%
entropy: 0.68

Table 3.10.c.
Outcomes in Chicken

CH		Column player **0**	Column player **1**
Row player	**0**	2.2%	42.2%
Row player	**1**	55.4%	0.3%

efficiency: 99.7%
entropy: 1.14

Table 3.10.d.
Outcomes in Battle of the Sexes

BS		Column player **0**	Column player **1**
Row player	**0**	0.3%	51.7%
Row player	**1**	47.8%	0.2%

efficiency: 99.7%
entropy: 1.04

Distributions are from the 1000th generation.

Table 3.11. Outcome distributions in one-game, no cost settings for the natural outcome context.

Table 3.11.a.
Outcomes in Prisoner's Dilemma

PD		Column player **0**	Column player **1**
Row player	**0**	34.3%	10.9%
	1	11.7%	43.1%

efficiency: 54.4%
entropy: 1.76

Table 3.11.b.
Outcomes in Stag Hunt

SH		Column player **0**	Column player **1**
Row player	**0**	99.3%	0.3%
	1	0.2%	0.2%

efficiency: 99.3%
entropy: 0.07

Table 3.11.c.
Outcomes in Chicken

CH		Column player **0**	Column player **1**
Row player	**0**	11.0%	43.2%
	1	43.7%	2.1%

efficiency: 97.9%
entropy: 1.51

Table 3.11.d.
Outcomes in Battle of the Sexes

BS		Column player **0**	Column player **1**
Row player	**0**	0.3%	47.4%
	1	52.2%	0.1%

efficiency: 99.7%
entropy: 1.03

Distributions are from the 1000th generation.

Table 3.12. Outcome distributions in two-game, cognitive cost settings for the natural outcome context.

Table 3.12.a.
Two-game set: **{PD, SH}**

Outcomes in **PD** within {PD, SH}

Row Player \ Column player	0	1
0	91.4%	3.1%
1	3.0%	2.5%

efficiency: 5.5%
entropy: 0.56

Outcomes in **SH** within {PD, SH}

Row Player \ Column player	0	1
0	99.6%	0.2%
1	0.2%	0.0%

efficiency: 99.6%
entropy: 0.04

Table 3.12.b.
Two-game set: **{PD, CH}**

Outcomes in **PD** within {PD, CH}

Row Player \ Column player	0	1
0	66.3%	12.9%
1	15.3%	5.5%

efficiency: 19.6%
entropy: 1.42

Outcomes in **CH** within {PD, CH}

Row Player \ Column player	0	1
0	28.3%	34.7%
1	36.3%	0.7%

efficiency: 99.3%
entropy: 1.62

Distributions are from the 1000th generation.

Table 3.12. Outcome distributions in two-game, cognitive cost settings for the natural outcome context (continued).

Table 3.12.c.
Two-game set: **{PD, BS}**

Outcomes in **PD** within {PD, BS}

Row Player \ Column player	**0**	**1**
0	76.5%	7.7%
1	11.5%	4.3%

efficiency: 13.9%
entropy: 1.13

Outcomes in **BS** within {PD, BS}

Row Player \ Column player	**0**	**1**
0	9.2%	41.5%
1	49.0%	0.3%

efficiency: 94.2%
entropy: 1.37

Table 3.12.d.
Two-game set: **{SH, CH}**

Outcomes in **SH** within {SH, CH}

Row Player \ Column player	**0**	**1**
0	69.6%	3.1%
1	4.7%	22.6%

efficiency: 69.6%
entropy: 1.21

Outcomes in **CH** within {SH, CH}

Row Player \ Column player	**0**	**1**
0	25.6%	38.1%
1	33.4%	2.8%

efficiency: 97.2%
entropy: 1.71

Distributions are from the 1000th generation.

Table 3.12. Outcome distributions in two-game, cognitive cost settings for the natural outcome context (continued).

Table 3.12.e.
Two-game set: **{SH, BS}**

Outcomes in **SH** within {SH, BS}

Row Player	Column player 0	Column player 1
0	86.8%	1.2%
1	1.0%	11.0%

efficiency: 86.8%
entropy: 0.67

Outcomes in **BS** within {SH, BS}

Row Player	Column player 0	Column player 1
0	0.5%	48.7%
1	50.7%	0.1%

efficiency: 99.6%
entropy: 1.05

Table 3.12.f.
Two-game set: **{CH, BS}**

Outcomes in **CH** within {CH, BS}

Row Player	Column player 0	Column player 1
0	1.3%	47.9%
1	50.5%	0.3%

efficiency: 99.7%
entropy: 1.11

Outcomes in **BS** within {CH, BS}

Row Player	Column player 0	Column player 1
0	0.2%	46.9%
1	52.5%	0.3%

efficiency: 99.5%
entropy: 1.05

Distributions are from the 1000th generation.

Table 3.13. Outcome distributions in two-game, no cost settings for the natural outcome context.

Table 3.13.a.
Two-game set: **{PD, SH}**

Outcomes in **PD** within {PD, SH}

Row Player \ Column player	0	1
0	22.0%	18.0%
1	17.9%	42.1%

efficiency: 60.0%
entropy: 1.90

Outcomes in **SH** within {PD, SH}

Row Player \ Column player	0	1
0	99.3%	0.2%
1	0.2%	0.2%

efficiency: 99.3%
entropy: 0.07

Table 3.13.b.
Two-game set: **{PD, CH}**

Outcomes in **PD** within {PD, CH}

Row Player \ Column player	0	1
0	29.6%	24.1%
1	23.8%	22.5%

efficiency: 46.5%
entropy: 1.99

Outcomes in **CH** within {PD, CH}

Row Player \ Column player	0	1
0	21.2%	36.6%
1	40.5%	1.7%

efficiency: 98.3%
entropy: 1.63

Distributions are from the 1000th generation.

Table 3.13. Outcome distributions in two-game, no cost settings for the natural outcome context (continued).

Table 3.13.c.
Two-game set: **{PD, BS}**

Outcomes in **PD** within {PD, BS}

Row Player \ Column player	**0**	**1**
0	33.9%	21.9%
1	22.0%	22.2%

efficiency: 44.1%
entropy: 1.97

Outcomes in **BS** within {PD, BS}

Row Player \ Column player	**0**	**1**
0	2.1%	47.4%
1	50.3%	0.1%

efficiency: 98.6%
entropy: 1.14

Table 3.13.d.
Two-game set: **{SH, CH}**

Outcomes in **SH** within {SH, CH}

Row Player \ Column player	**0**	**1**
0	98.8%	0.3%
1	0.2%	0.7%

efficiency: 98.8%
entropy: 0.11

Outcomes in **CH** within {SH, CH}

Row Player \ Column player	**0**	**1**
0	47.3%	30.1%
1	21.8%	0.8%

efficiency: 99.2%
entropy: 1.57

Distributions are from the 1000th generation.

Table 3.13. Outcome distributions in two-game, no cost settings
for the natural outcome context (continued).

Table 3.13.e.
Two-game set: **{SH, BS}**

Outcomes in **SH** within {SH, BS}

Row Player \ Column player	0	1
0	99.2%	0.3%
1	0.3%	0.3%

efficiency: 99.2%
entropy: 0.08

Outcomes in **BS** within {SH, BS}

Row Player \ Column player	0	1
0	1.6%	51.3%
1	47.0%	0.2%

efficiency: 98.9%
entropy: 1.12

Table 3.13.f.
Two-game set: **{CH, BS}**

Outcomes in **CH** within {CH, BS}

Row Player \ Column player	0	1
0	2.5%	51.9%
1	44.7%	0.9%

efficiency: 98.6%
entropy: 1.20

Outcomes in **BS** within {CH, BS}

Row Player \ Column player	0	1
0	4.2%	49.2%
1	45.2%	1.4%

efficiency: 99.2%
entropy: 1.30

Distributions are from the 1000th generation.

Table 3.14. t-statistics for differences between cognitive cost and no cost statistics for one-game settings for the natural outcome context.

Game	Performance statistics: Profit	Score	Cost	Distinct states activated in all games	Population-level state activation similarity	Action 0 usage frequency	Structural statistics: Accessible states	Terminal states	Counting states	0-playing initial states	0-playing states	0-reciprocating states	1-reciprocating states	Tit-for-tat states
PD	20.9 ***	10.95 ***	n/a	14.2 ***	4.82 ***	-13.1 ***	41.1 ***	-14.4 ***	-2.31 ***	-5.82 ***	-20.2 ***	-15.5 ***	16.3 ***	15.6 ***
SH	10.5 ***	4.32 ***	n/a	8.89 ***	3.40 ***	4.39 ***	40.6 ***	-15.8 ***	-2.59 **	2.34 **	-4.07 ***	-2.07 **	4.18 ***	11.6 ***
CH	3.00 ***	-1.45	n/a	14.4 ***	4.71 ***	1.68 *	28.5 ***	-13.3 ***	-1.77 *	1.29	0.49	0.69	-0.54	6.89 ***
BS	6.65 ***	-0.65	n/a	11.1 ***	2.96 ***	-0.66	40.5 ***	-17.7 ***	-1.88 *	-0.84	0.8	-0.28	-1.92 *	6.26 ***

Welch's t-statistics for difference of statistics between No Cost and Cognitive Cost environments in the 1000th generation. Asterisks indicate significance for two-tailed tests that (statistic with no cost – statistic with cognitive cost) differences are non-zero: *** 1% significance, ** 5% significance, * 10% significance.

Table 3.15. t-statistics for differences between cognitive cost and no cost statistics for two-game settings for the natural outcome context.

		Performance statistics:								Structural statistics:									
											Proportion of states that are…								
Games(s)		Profit	Score	Cost	Distinct states activated in all games	Distinct states activated per game	Automaton-level state activation similarity	Population-level state activation similarity	Action 0 usage frequency	Distinct initial states	Accessible states in all games	Accessible states per game	Terminal states	Counting states	0-playing initial states	0-playing states	0-reciprocating states	1-reciprocating states	Tit-for-tat states
{PD, SH}	**PD** in PD-SH	n/a	12.6 ***	n/a	n/a	11.5 ***	4.24 ***	4.24 ***	-14 ***	n/a	n/a	35.9 ***	-8.8 ***	-4.7 ***	-6.3 ***	-15.3 ***	-12.8 ***	12.2 ***	9.97 ***
	SH in PD-SH	n/a	-2.0 **	n/a	n/a	8.78 ***	3.94 ***	3.94 ***	-2.1 **	n/a	n/a	32.3 ***	-7.9 ***	-4.7 ***	-3.7 ***	-14.4 ***	-12.8 ***	11.5 ***	9.48 ***
	Mean or **Joint**	30.7 ***	12.5 ***	n/a	19.6 ***	14.1 ***	5.24 ***	5.24 ***	-14 ***	10.9 ***	43.7 ***	36.6 ***	-8.4 ***	-4.7 ***	-7.2 ***	-15.3 ***	-13.0 ***	12.0 ***	9.93 ***
{PD, CH}	**PD** in PD-CH	n/a	3.35 ***	n/a	n/a	15.2 ***	7.65 ***	7.65 ***	-5.2 ***	n/a	n/a	28.0 ***	-9.9 ***	-3.6 ***	-2.2 **	-5.3 ***	-3.9 ***	4.40 ***	6.74 ***
	CH in PD-CH	n/a	0.03	n/a	n/a	16.8 ***	7.57 ***	7.57 ***	-1.0	n/a	n/a	28.1 ***	-9.6 ***	-3.3 ***	0.03	-1.3	-1.0	1.53	6.56 ***
	Mean or **Joint**	13.3 ***	3.33 ***	n/a	26.0 ***	19.6 ***	9.04 ***	9.04 ***	-3.6 ***	9.18 ***	44.5 ***	32.2 ***	-10.4 ***	-3.6 ***	-1.3	-3.5 ***	-2.6 **	3.17 ***	6.75 ***
{PD, BS}	**PD** in PD-BS	n/a	6.38 ***	n/a	n/a	11.1 ***	3.98 ***	3.98 ***	-6.1 ***	n/a	n/a	20.4 ***	-7.5 ***	-3.2 ***	-3.6 ***	-6.3 ***	-4.5 ***	4.95 ***	3.94 ***
	BS in PD-BS	n/a	2.08 **	n/a	n/a	10.2 ***	2.74 ***	2.74 ***	-0.2	n/a	n/a	28.0 ***	-8.4 ***	-3.2 ***	-0.2	-0.4	0.25	2.17 **	4.79 ***
	Mean or **Joint**	14.8 ***	5.42 ***	n/a	16.5 ***	12.9 ***	4.62 ***	4.62 ***	-3.9 ***	6.15 ***	34.7 ***	27.8 ***	-8.3 ***	-3.3 ***	-2.2 **	-3.4 ***	-2.0 **	3.79 ***	4.46 ***
{SH, CH}	**SH** in SH-CH	n/a	6.30 ***	n/a	n/a	7.35 ***	4.04 ***	4.04 ***	6.21 ***	n/a	n/a	37.8 ***	-7.2 ***	-4.1 ***	3.31 ***	1.10	1.55	-1.0	2.84 ***
	CH in SH-CH	n/a	0.05	n/a	n/a	13.8 ***	6.55 ***	6.55 ***	2.62 ***	n/a	n/a	32.0 ***	-7.2 ***	-4.1 ***	0.27	0.80	1.90 *	-0.9	3.16 ***
	Mean or **Joint**	11.4 ***	4.03 ***	n/a	18.7 ***	15.0 ***	7.24 ***	7.24 ***	5.53 ***	6.28 ***	46.4 ***	39.6 ***	-7.6 ***	-4.4 ***	2.35 **	1.06	1.87 *	-1.0	3.08 ***
{SH, BS}	**SH** in SH-BS	n/a	3.83 ***	n/a	n/a	7.08 ***	2.08 **	2.08 **	3.81 ***	n/a	n/a	29.4 ***	-6.7 ***	-3.3 ***	0.53	-1.1	-1.3	0.02	1.27
	BS in SH-BS	n/a	0.22	n/a	n/a	10.8 ***	3.33 ***	3.33 ***	0.58	n/a	n/a	29.8 ***	-6.8 ***	-3.0 ***	0.25	1.19	0.86	-0.9	2.07 **
	Mean or **Joint**	13.1 ***	2.82 ***	n/a	16.6 ***	13.0 ***	3.69 ***	3.69 ***	2.08 **	3.80 ***	36.6 ***	32.3 ***	-7.3 ***	-3.4 ***	0.54	0.10	-0.2	-0.5	1.72 *
{CH, BS}	**CH** in CH-BS	n/a	-0.7	n/a	n/a	10.1 ***	3.88 ***	3.88 ***	0.67	n/a	n/a	22.3 ***	-6.3 ***	-4.6 ***	0.94	-0.1	-0.7	-0.5	2.53 **
	BS in CH-BS	n/a	1.14	n/a	n/a	10.1 ***	3.09 ***	3.09 ***	1.31	n/a	n/a	25.2 ***	-6.3 ***	-5.0 ***	0.69	1.11	0.50	-1.6	2.57 **
	Mean or **Joint**	10.9 ***	-.02	n/a	15.5 ***	10.2 ***	4.00 ***	4.00 ***	1.03	8.51 ***	37.4 ***	27.2 ***	-6.5 ***	-4.8 ***	0.97	0.53	-0.1	-1.1	2.60 **

See note for Table 3.14. *** 1% significance, ** 5% significance, * 10% significance.

Table 3.16. Similarity of play across both games in two-game, cognitive cost settings for the natural outcome context.

Game set	Automaton-level state activation similarity (ALSAS)	Absolute outcome distribution difference (AODD)	Distinct initial states (DIS)
{CH, BS}	5.4	1.0%	1.48
{PD, SH}	8.6	4.1%	1.41
{PD, CH}	27.5	21.4%	1.47
{SH, CH}	45.0	31.8%	1.67
{PD, BS}	45.8	35.6%	1.62
{SH, BS}	62.5	48.6%	1.74

For ALSAS and DIS, the differences between values in the same block are not statistically different from zero at the 10 percent significance level. For ALSAS, the differences in separate blocks are all significant at the 1 percent level; for DIS, the differences in separate blocks are significant at the 1, 5, or 10 percent level.

Table 3.17. t-statistics for differences in a game's scores by pairing for the natural outcome context.

Prisoner's Dilemma (PD) Scores

	{**PD**, CH}	{**PD**, BS}	{**PD**}
{**PD**, SH}	1.412	0.551	-1.563
{**PD**, CH}	-	-0.942	-1.910*
{**PD**, BS}	-	-	-1.347

Stag Hunt (SH) Scores

	{**SH**, CH}	{**SH**, BS}	{**SH**}
{**SH**, PD}	-6.528***	-3.955***	-4.440***
{**SH**, CH}	-	2.951***	2.297**
{**SH**, BS}	-	-	-0.645

Chicken (CH) Scores

	{**CH**, SH}	{**CH**, BS}	{**CH**}
{**CH**, PD}	-0.948	0.137	1.020
{**CH**, SH}	-	0.992	1.852*
{**CH**, BS}	-	-	0.810

Battle of the Sexes (SH) Scores

	{**BS**, SH}	{**BS**, CH}	{**BS**}
{**BS**, PD}	2.244***	1.955**	2.662***
{**BS**, SH}	-	-0.290	0.487
{**BS**, CH}	-	-	0.767

The t-statistics are for (game's score in top margin set – game's in left margin set) is not equal to zero. For cognitive cost environments during the 1000th generation.

Table 3.18. Statistics in one-game, cognitive cost settings
for the cooperate/defect context.

Game	Performance statistics: Profit	Score	Cost	Distinct states activated in all games	Population-level state activation similarity	Action C usage frequency	Structural statistics: Proportion of states that are… Accessible states	Terminal states	Counting states	C-playing initial states	C-playing states	C-reciprocating states	D-reciprocating states	Tit-for-tat states
PD#	1.54	2.00	0.47	1.24	0.15	0.01	0.08	0.79	0.10	0.01	0.01	0.0	1.00	0.01
SH#	3.21	3.67	0.46	1.32	0.17	0.84	0.09	0.75	0.12	0.78	0.80	0.81	0.20	0.04
CH#	2.62	3.13	0.50	1.36	0.27	0.44	0.09	0.68	0.14	0.43	0.42	0.44	0.61	0.06
Overall Mean	2.46	2.93	0.48	1.31	0.20	0.43	0.09	0.74	0.12	0.41	0.41	0.42	0.60	0.04

Means for the 30 automata in row population in the 1000th generation, averaged over 100 independent runs.

Table 3.19. Statistics in one-game, no cost settings
for the cooperate/defect context.

Game	Performance statistics: Profit	Score	Cost	Distinct states activated in all games	Population-level state activation similarity	Action C usage frequency	Structural statistics: Proportion of states that are… Accessible states	Terminal states	Counting states	C-playing initial states	C-playing states	C-reciprocating states	D-reciprocating states	Tit-for-tat states
PD#	2.52	2.52	n/a	5.96	1.36	0.55	0.71	0.01	0.07	0.35	0.31	0.34	0.76	0.24
SH#	3.98	3.98	n/a	3.59	0.67	1.00	0.76	0.03	0.05	0.90	0.64	0.73	0.37	0.26
CH#	2.96	2.96	n/a	5.31	1.06	0.54	0.69	0.03	0.09	0.52	0.45	0.47	0.58	0.22
Overall Mean	3.15	3.15	n/a	4.95	1.03	0.70	0.72	0.02	0.07	0.59	0.47	0.51	0.57	0.24

Means for the 30 automata in row population in the 1000th generation, averaged over 100 independent runs.

Table 3.20. Statistics in two-game, cognitive cost settings
for the cooperate/defect context.

Games(s)		Performance statistics: Profit	Score	Cost	Distinct states activated in all games	Distinct states activated per game	Automaton-level state activation similarity	Population-level state activation similarity	Action C usage frequency	Structural statistics: Distinct initial states	Proportion of states that are… Accessible states in all games	Accessible states per game	Terminal states	Counting states	C-playing initial states	C-playing states	C-reciprocating states	D-reciprocating states	Tit-for-tat states
{PD$^\#$, SH$^\#$}	**PD$^\#$** in PD$^\#$-SH$^\#$	n/a	2.17	n/a	n/a	1.61	n/a	0.30	0.17	n/a	n/a	0.11	0.49	0.16	0.11	0.13	0.15	0.90	0.08
	SH$^\#$ in PD$^\#$-SH$^\#$	n/a	2.75	n/a	n/a	1.54	n/a	0.38	0.38	n/a	n/a	0.13	0.42	0.16	0.33	0.22	0.25	0.87	0.15
	Mean	1.98	2.46	0.48	1.95	1.58	26.7	0.34	0.27	1.53	0.13	0.12	0.46	0.16	0.22	0.17	0.20	0.89	0.11
{PD$^\#$, CH$^\#$}	**PD$^\#$** in PD$^\#$-CH$^\#$	n/a	2.14	n/a	n/a	1.69	n/a	0.22	0.18	n/a	n/a	0.12	0.42	0.23	0.10	0.15	0.15	0.84	0.06
	CH$^\#$ in PD$^\#$-CH$^\#$	n/a	2.97	n/a	n/a	1.65	n/a	0.22	0.43	n/a	n/a	0.12	0.40	0.23	0.41	0.34	0.31	0.66	0.06
	Mean	2.05	2.56	0.51	2.08	1.67	35.2	0.22	0.30	1.62	0.14	0.12	0.41	0.23	0.26	0.24	0.23	0.75	0.06
{SH$^\#$, CH$^\#$}	**SH$^\#$** in SH$^\#$-CH$^\#$	n/a	3.41	n/a	n/a	1.49	n/a	0.21	0.73	n/a	n/a	0.13	0.35	0.18	0.64	0.54	0.58	0.52	0.21
	CH$^\#$ in SH$^\#$-CH$^\#$	n/a	2.90	n/a	n/a	1.66	n/a	0.24	0.64	n/a	n/a	0.13	0.34	0.19	0.52	0.52	0.52	0.52	0.18
	Mean	2.63	3.15	0.52	2.17	1.57	45.0	0.22	0.68	1.67	0.15	0.13	0.34	0.19	0.58	0.53	0.55	0.52	0.20
Overall Mean		2.22	2.72	0.50	2.07	1.61	35.6	0.26	0.42	1.61	0.14	0.12	0.40	0.19	0.35	0.32	0.33	0.72	0.12

Means for the 30 automata in the row population in the 1000th generation, averaged over 100 independent runs.

Table 3.21. Statistics in two-game, no cost settings
for the cooperate/defect context.

Games(s)		Performance statistics:								Structural statistics:									
											Proportion of states that are…								
		Profit	Score	Cost	Distinct states activated in all games	Distinct states activated per game	Automaton-level state activation similarity	Population-level state activation similarity	Action C usage frequency	Distinct initial states	Accessible states in all games	Accessible states per game	Terminal states	Counting states	C-playing initial states	C-playing states	C-reciprocating states	D-reciprocating states	Tit-for-tat states
{PD$^{\#}$, SH$^{\#}$}	**PD$^{\#}$** in PD$^{\#}$-SH$^{\#}$	n/a	2.87	n/a	n/a	5.76	n/a	1.05	0.85	n/a	n/a	0.72	0.03	0.07	0.34	0.41	0.47	0.65	0.29
	SH$^{\#}$ in PD$^{\#}$-SH$^{\#}$	n/a	3.95	n/a	n/a	2.94	n/a	0.54	0.98	n/a	n/a	0.72	0.01	0.07	0.76	0.44	0.50	0.63	0.30
	Mean	3.41	3.41	n/a	6.49	4.35	28.7	0.80	0.91	1.95	0.76	0.72	0.02	0.07	0.55	0.43	0.48	0.64	0.30
{PD$^{\#}$, CH$^{\#}$}	**PD$^{\#}$** in PD$^{\#}$-CH$^{\#}$	n/a	2.66	n/a	n/a	6.68	n/a	1.42	0.62	n/a	n/a	0.72	0.02	0.07	0.37	0.38	0.42	0.68	0.26
	CH$^{\#}$ in PD$^{\#}$-CH$^{\#}$	n/a	2.94	n/a	n/a	5.39	n/a	1.18	0.58	n/a	n/a	0.70	0.01	0.08	0.49	0.39	0.43	0.67	0.26
	Mean	2.80	2.80	n/a	8.10	6.04	41.3	1.30	0.60	1.92	0.75	0.71	0.01	0.07	0.43	0.39	0.42	0.67	0.26
{SH$^{\#}$, CH$^{\#}$}	**SH$^{\#}$** in SH$^{\#}$-CH$^{\#}$	n/a	3.98	n/a	n/a	3.18	n/a	0.86	0.99	n/a	n/a	0.76	0.02	0.05	0.84	0.58	0.65	0.48	0.30
	CH$^{\#}$ in SH$^{\#}$-CH$^{\#}$	n/a	2.90	n/a	n/a	5.78	n/a	1.65	0.77	n/a	n/a	0.75	0.01	0.06	0.53	0.55	0.61	0.48	0.28
	Mean	3.44	3.44	n/a	7.35	4.48	63.5	1.25	0.88	1.98	0.80	0.75	0.02	0.06	0.69	0.57	0.63	0.48	0.29
Overall Mean		3.22	3.22	n/a	7.31	4.96	44.5	1.12	0.80	1.95	0.77	0.73	0.02	0.07	0.56	0.46	0.51	0.60	0.28

Means for the 30 automata in the row population in the 1000th generation, averaged over 100 independent runs.

Table 3.22. Outcome distributions in one-game, cognitive cost settings for the cooperate/defect context.

Table 3.22.a. Prisoner's Dilemma#

PD#	C	D
C	0.0%	0.8%
D	0.6%	98.7%

efficiency: 0.5%
entropy: 0.12

Table 3.22.b. Stag Hunt#

SH#	C	D
C	83.6%	0.3%
D	0.2%	15.9%

efficiency: 83.6%
entropy: 0.68

Table 3.22.c. Chicken#

CH#	C	D
C	2.2%	42.2%
D	55.4%	0.3%

efficiency: 99.7%
entropy: 1.14

Distributions are from the 1000th generation.

Table 3.23. Outcome distributions in one-game, no cost settings for the cooperate/defect context.

Table 3.23.a. Prisoner's Dilemma#

PD#	C	D
C	40.5%	14.1%
D	12.8%	32.6%

efficiency: 54.0%
entropy: 1.83

Table 3.23.b. Stag Hunt#

SH#	C	D
C	99.3%	0.3%
D	0.2%	0.2%

efficiency: 99.3%
entropy: 0.07

Table 3.23.c. Chicken#

CH#	C	D
C	11.0%	43.2%
D	43.7%	2.1%

efficiency: 97.9%
entropy: 1.51

Distributions are from the 1000th generation.

Table 3.24. Outcome distributions in two-game, cognitive cost settings for the cooperate/defect context.

Table 3.24.a.

Games: {PD#, SH#}

PD#	C	D
C	11.8%	4.8%
D	4.8%	78.6%

efficiency: 16.6%
entropy: 1.06

SH#	C	D
C	37.6%	0.7%
D	0.9%	60.9%

efficiency: 37.6%
entropy: 1.08

Table 3.24.b.

Games: {PD#, CH#}

PD#	C	D
C	5.5%	12.7%
D	10.8%	71.0%

efficiency: 17.3%
entropy: 1.31

CH#	C	D
C	6.5%	36.1%
D	49.4%	8.0%

efficiency: 92.0%
entropy: 1.58

Table 3.24.c.

Games: {SH#, CH#}

SH#	C	D
C	69.6%	3.1%
D	4.7%	22.6%

efficiency: 69.6%
entropy: 1.21

CH#	C	D
C	25.6%	38.1%
D	33.4%	2.8%

efficiency: 97.2%
entropy: 1.71

Table 3.25. Outcome distributions in two-game, no cost settings for the cooperate/defect context.

Table 3.25.a.

Games: {PD#, SH#}

PD#	C	D
C	80.2%	4.4%
D	5.4%	10.1%

efficiency: 85.0%
entropy: 1.01

SH#	C	D
C	97.6%	0.7%
D	0.2%	1.5%

efficiency: 97.6%
entropy: 0.19

Table 3.25.b.

Games: {PD#, CH#}

PD#	C	D
C	45.4%	16.1%
D	18.3%	20.1%

efficiency: 62.7%
entropy: 1.86

CH#	C	D
C	27.3%	30.6%
D	36.4%	5.7%

efficiency: 94.3%
entropy: 1.80

Table 3.25.c.

Games: {SH#, CH#}

SH#	C	D
C	98.8%	0.3%
D	0.2%	0.7%

efficiency: 98.8%
entropy: 0.11

CH#	C	D
C	47.3%	30.1%
D	21.8%	0.8%

efficiency: 99.2%
entropy: 1.57

Distributions are from the 1000th generation.

Table 3.26. t-statistics for differences between statistics in the natural outcome context and the cooperate/defect context.

Games(s)		Performance statistics:								Structural statistics:									
											Proportion of states that are…								
		Profit	Score	Cost	Distinct states activated in all games	Distinct states activated per game	Automaton-level state activation similarity	Population-level state activation similarity	Action 0/C usage freq.	Distinct initial states	Accessible states in all games	Accessible states per game	Terminal states	Counting states	0/C-playing initial states	0/C-playing states	0/C-reciprocating states	1/D-reciprocating states	Tit-for-tat states
PD	**PD**	0.08	1.26	1.08	1.39	1.39	n/a	1.06	120 ***	n/a	1.64	1.64	-1.80 *	1.23	68.9 ***	129 ***	193 ***	-131 ***	0.65
{PD, SH}	**PD** in PD-SH	n/a	-1.97 *	n/a	n/a	0.20	n/a	-0.20	19.4 ***	n/a	n/a	0.90	-0.90	1.36	21.0 ***	28.3 ***	25.9 ***	-28.3 ***	-1.08
	SH in PD-SH	n/a	13.1 ***	n/a	n/a	0.73	n/a	-0.53	12.9 ***	n/a	n/a	-0.58	0.11	1.46	12.7 ***	22.3 ***	21.2 ***	-25.6 ***	-3.80 ***
	Mean or **Joint**	9.47 ***	8.86 ***	-0.30	-0.94	0.51	-3.69 ***	-0.44	18.3 ***	-1.66 *	-0.76	0.15	-0.41	1.42	20.9 ***	26.9 ***	24.8 ***	-27.7 ***	-2.68 ***
{PD, CH}	**PD** in PD-CH	n/a	0.17	n/a	n/a	-1.69 *	n/a	-0.03	12.3 ***	n/a	n/a	-1.89 *	1.77 *	-0.59	14.1 ***	16.1 ***	15.6 ***	-13.2 ***	0.84
	CH in PD-CH	n/a	0.24	n/a	n/a	-1.42	n/a	0.20	3.24 ***	n/a	n/a	-2.15 **	2.02 **	-1.00	1.80 *	4.80 ***	5.57 ***	-4.37 ***	0.81
	Mean or **Joint**	0.81	0.31	-2.00 **	-2.17 **	-1.66 *	-1.28	0.11	8.39 ***	-2.12 **	-2.11 **	-2.08 **	1.96 *	-0.82	8.52 ***	11.1 ***	11.3 ***	-9.30 ***	0.84

For the 1000th generation of cognitive cost settings. Asterisks indicate significance for two-tailed tests that the differences (statistic in 0/1 context – statistic in C/D context) are non-zero: *** 1% significance, ** 5% significance, * 10% significance.

Figure 3.1. Example sixteen-state automata that play two games.

Automaton #7 uses eight separate states to play each game independently.
Automaton #13 uses the same two states to play tit-for-tat strategy in both games.

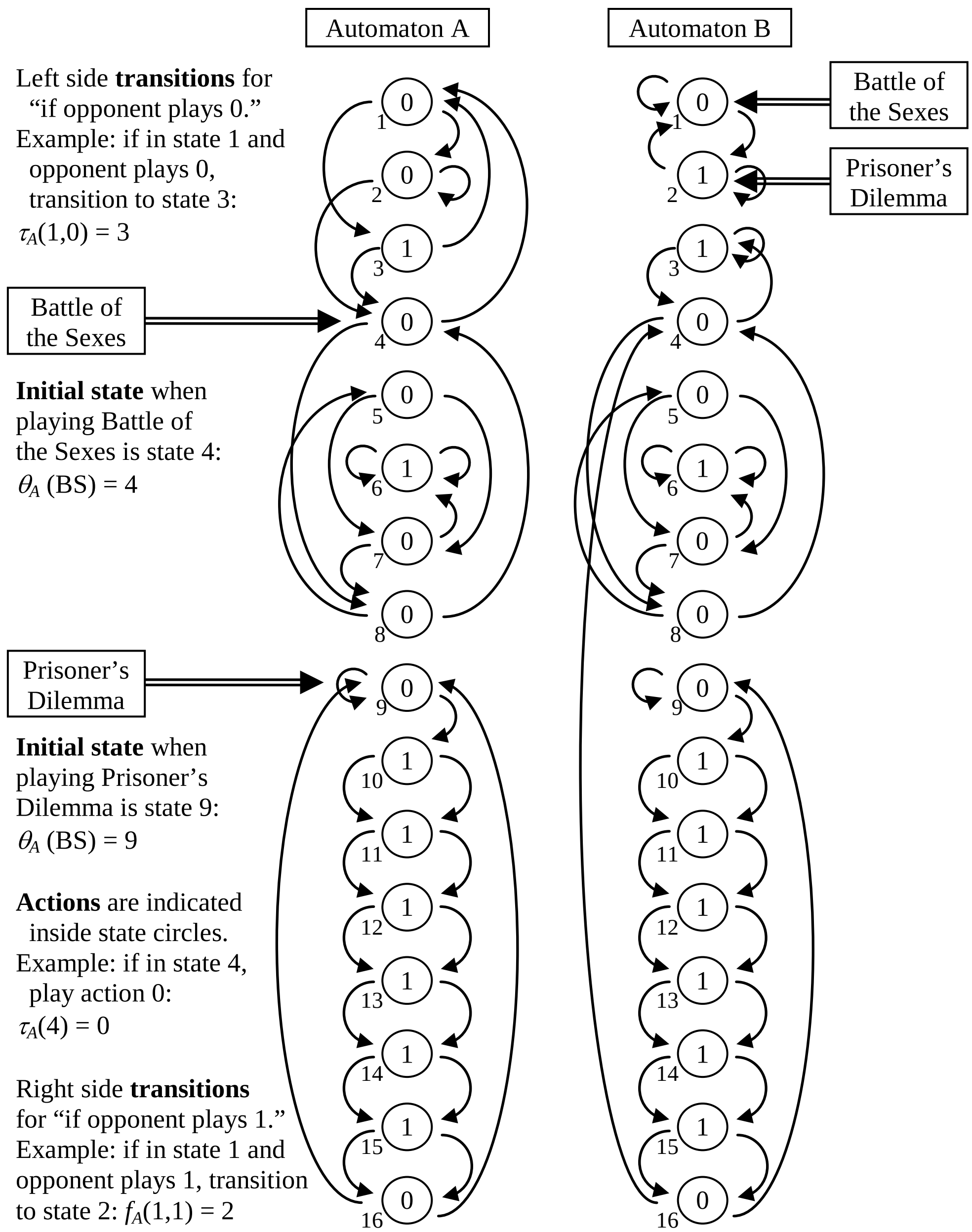


Figure 3.2. Example sixteen-state automaton that plays two games and its reduced form.

Figure 3.2.a: Row automaton #1 in its entirety

(from trial #17, run #1, 1000th generation)

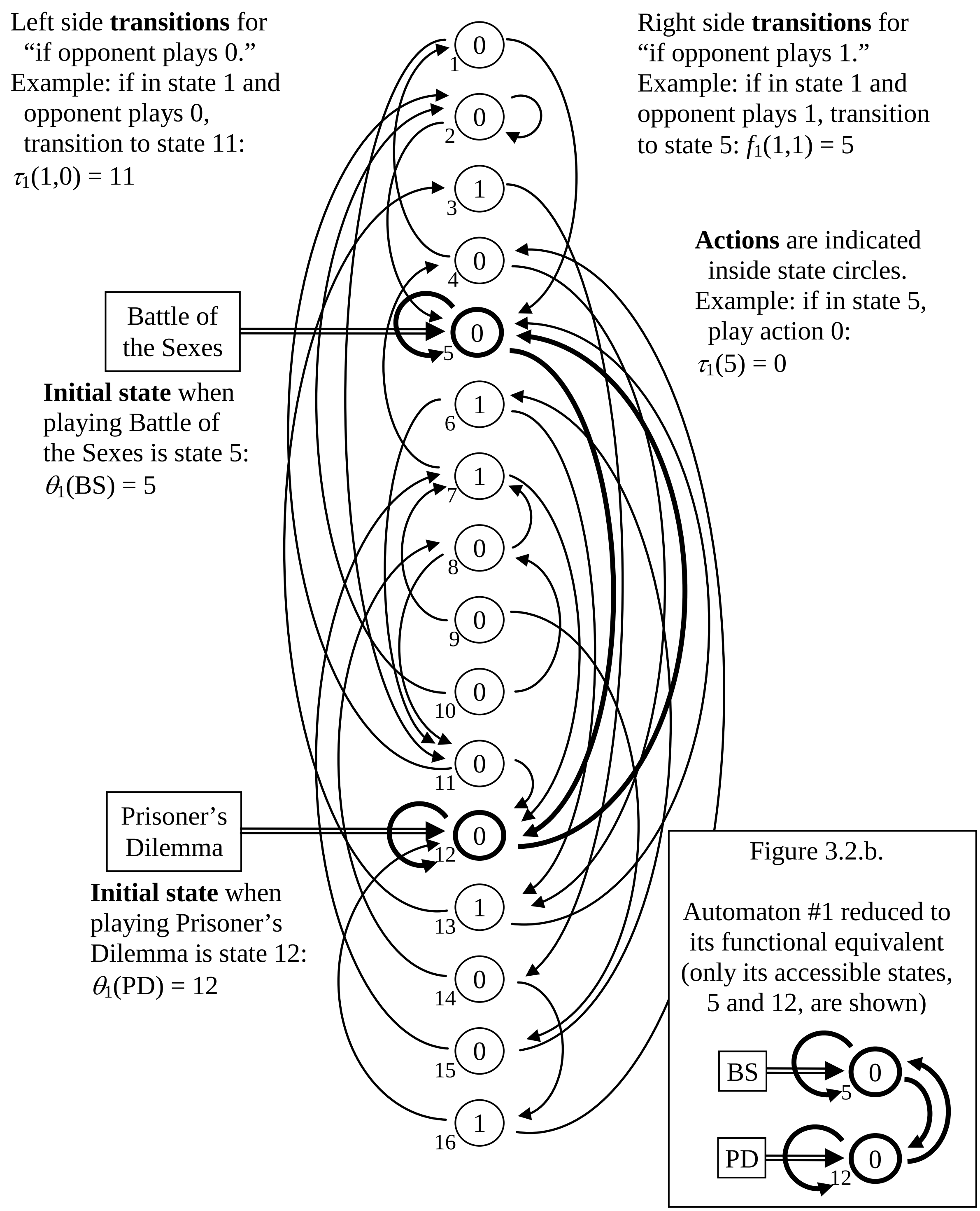


Figure 3.3. Evolution of population-level state activation similarity (PLSAS) in one-game, cost 4 environments.

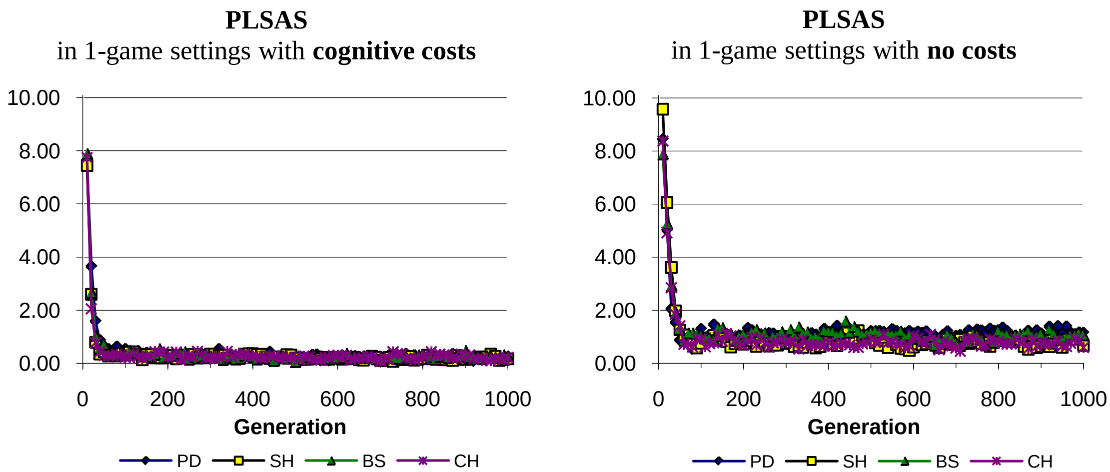


Figure 3.4. Distribution of 1000th generation scores of row populations across 100 runs in single game environments.

Cognitive cost environments

No cost environments

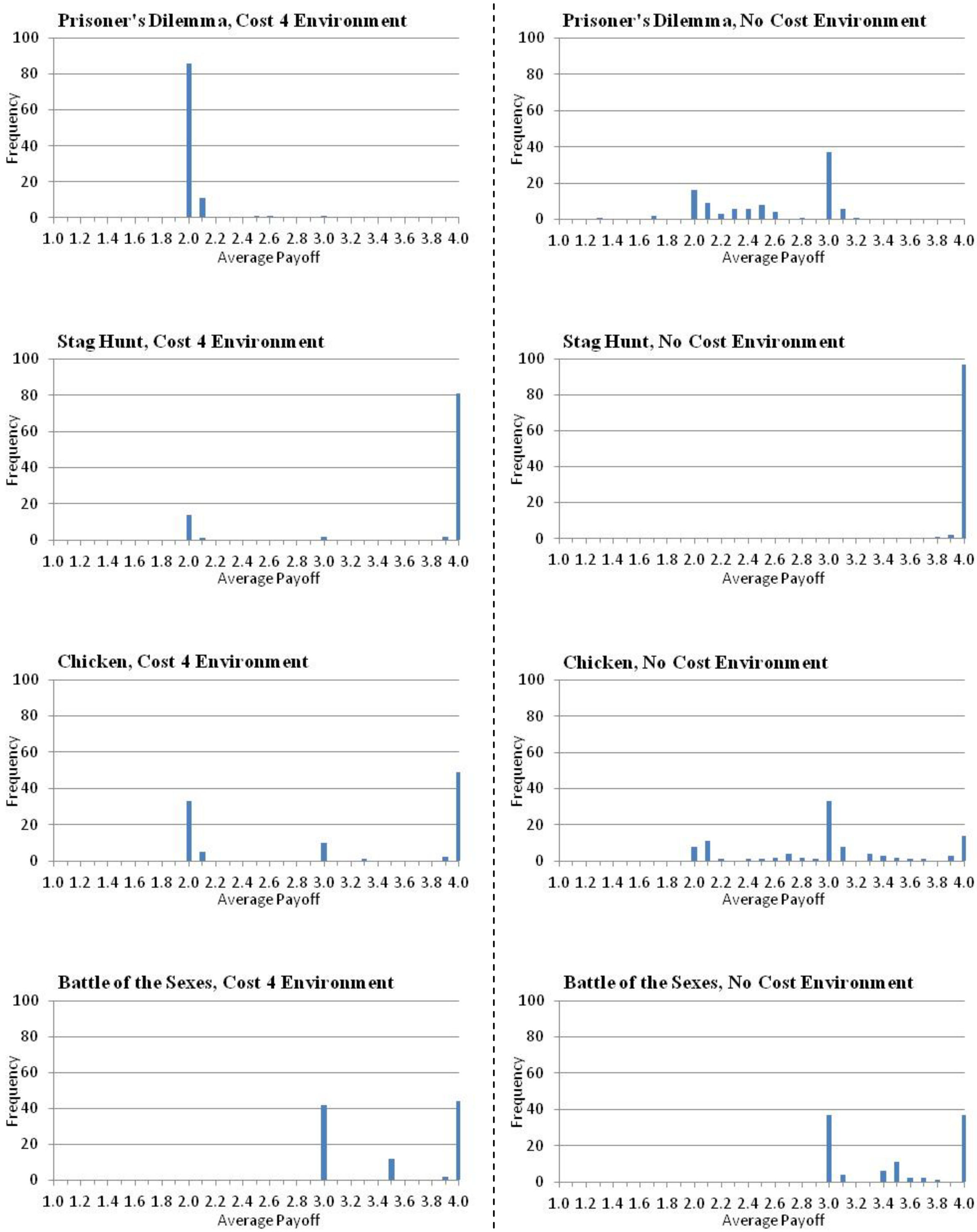


Figure 3.5. Distribution of 1000th generation scores of column populations across 100 runs in single game environments.

Cognitive cost environments | No cost environments

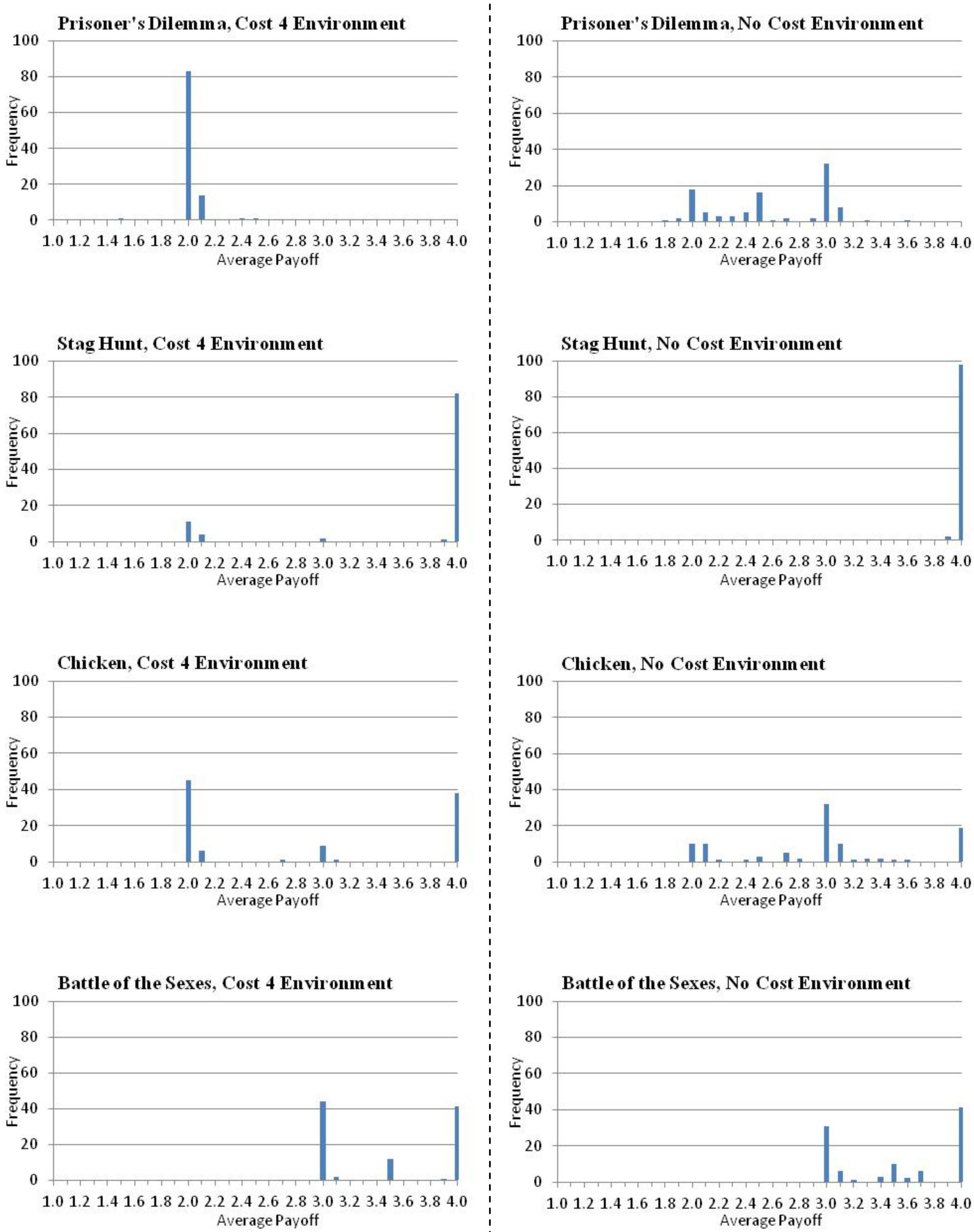


Figure 3.6. Average scores in two-game and one-game cognitive cost settings.

The range of payoffs on all vertical axes is 1.00 to ease comparisons between graphs.

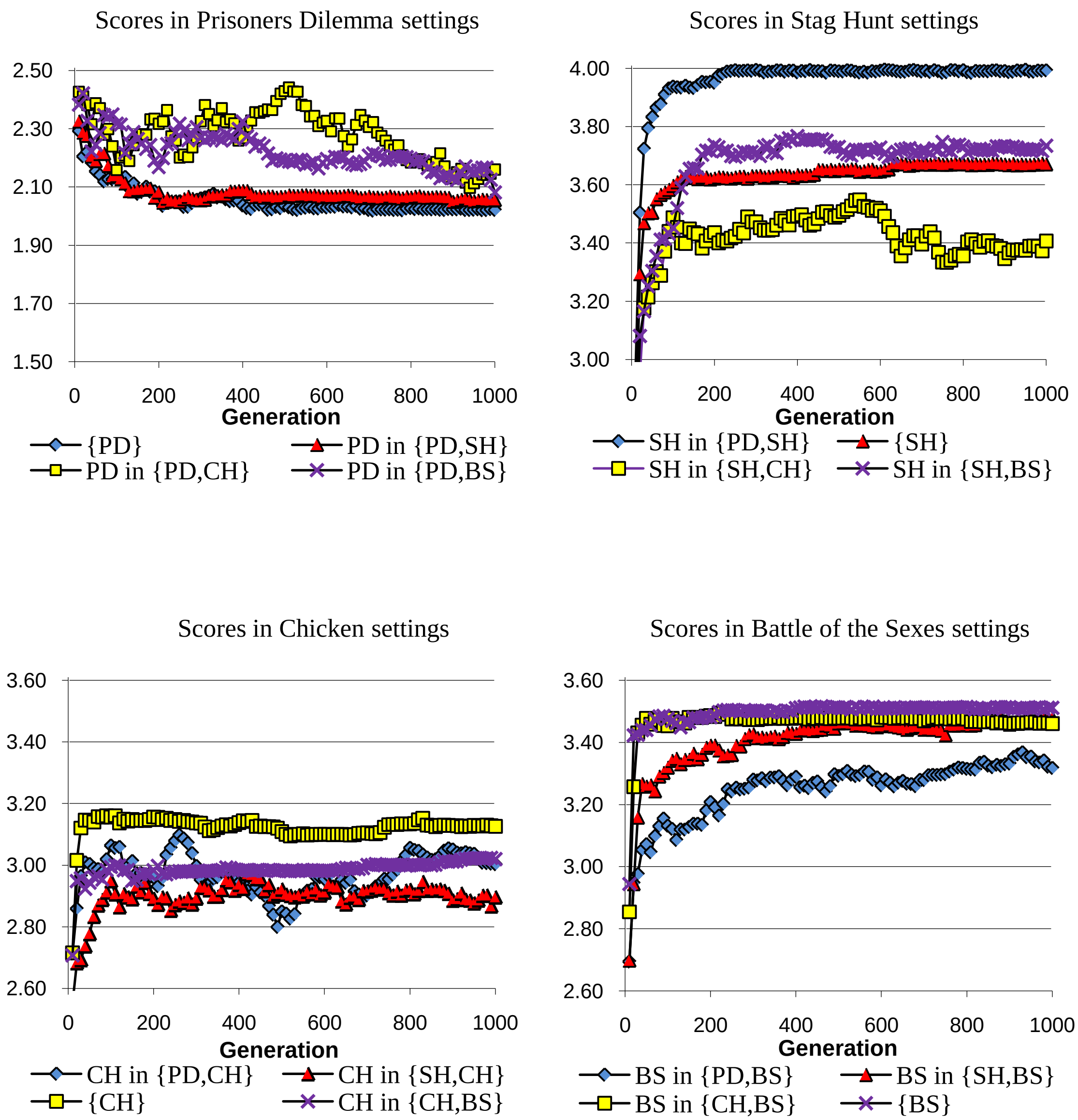
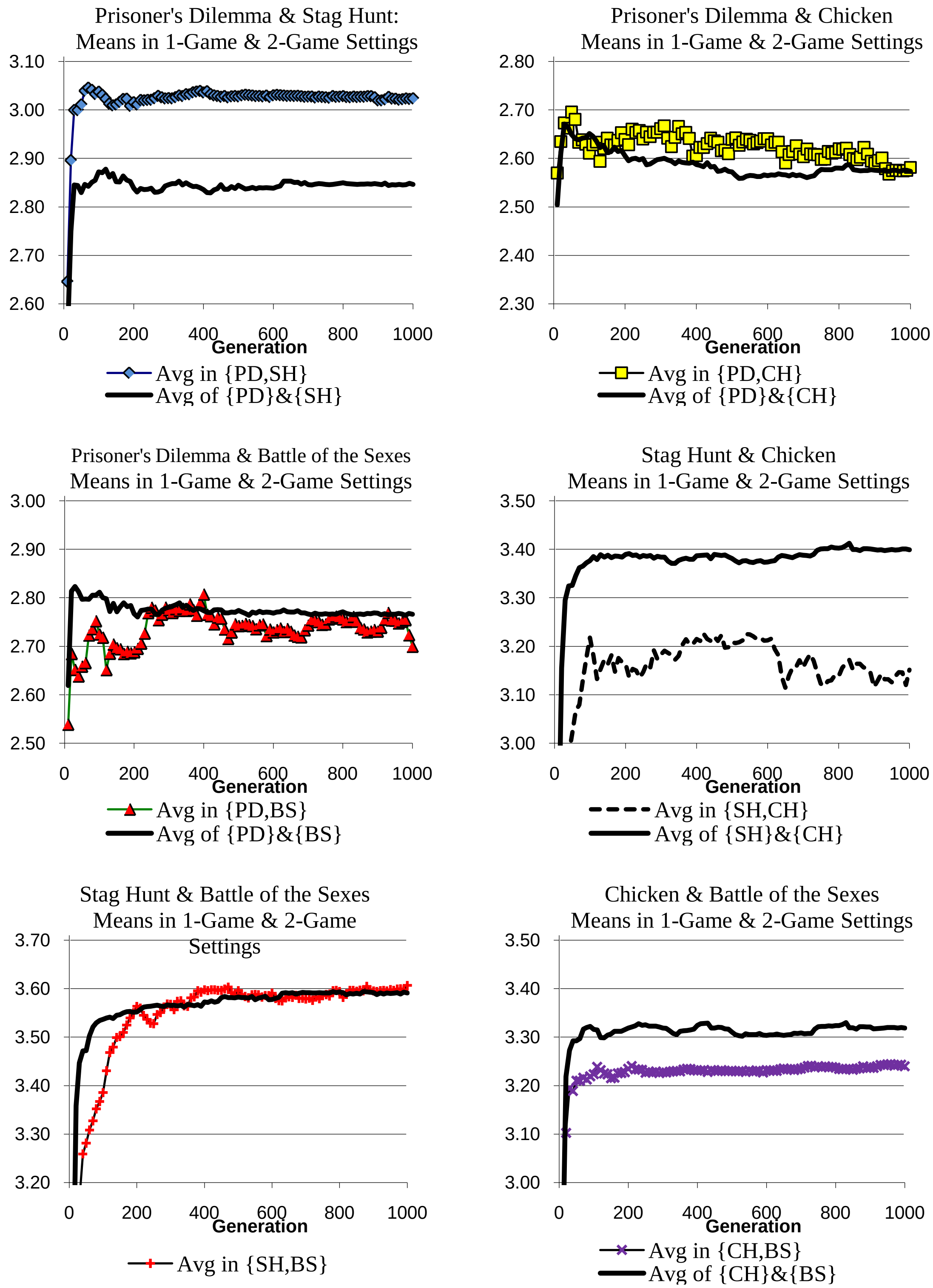


Figure 3.7. Mean score of two one-game environments compared to mean score in a two-game environment with cognitive costs.

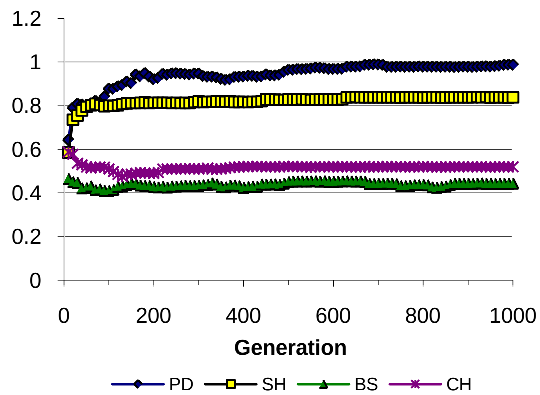
The range of payoffs on all vertical axes is 0.50 to ease comparisons between graphs.



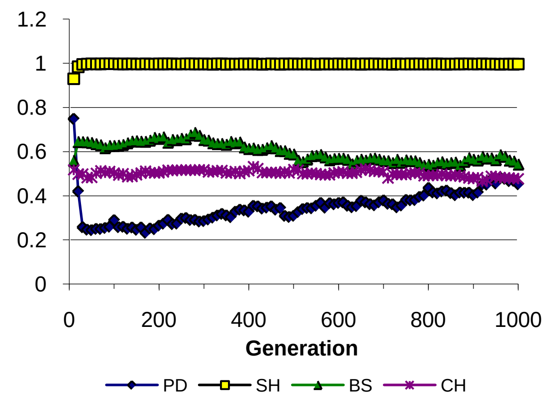
Appendix

Figure 3A.1. Evolution of statistics in one-game, cognitive cost environments.

Score
in 1-game settings with **cognitive costs**

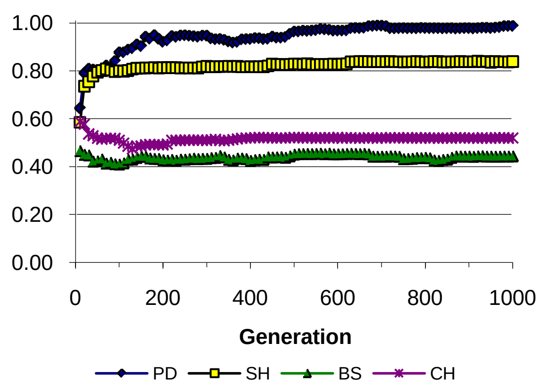


Score
in 1-game settings with **no cost**



In the absence of costs, score is identical to profit; the score graph above is identical the profit below. The redundancy is intentional to facilitate comparison to the graphs in the left column.

Profit
in 1-game settings with **cognitive costs**



Profit
in 1-game settings with **no cost**

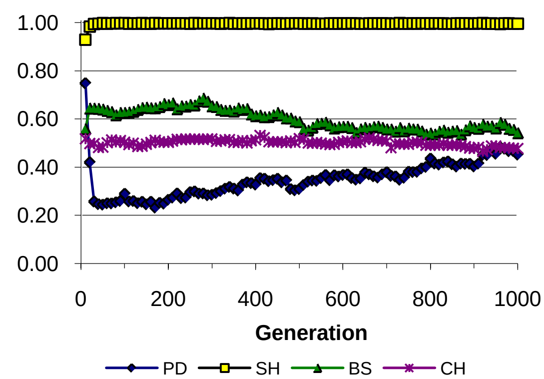
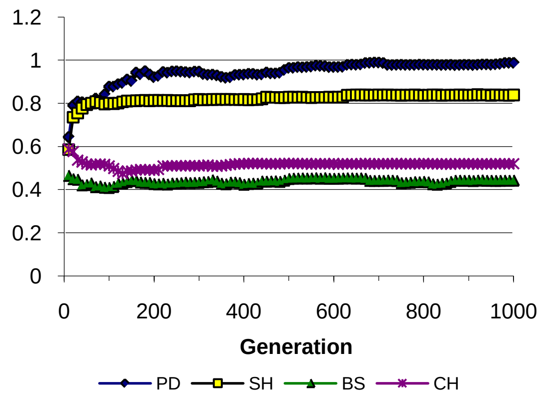
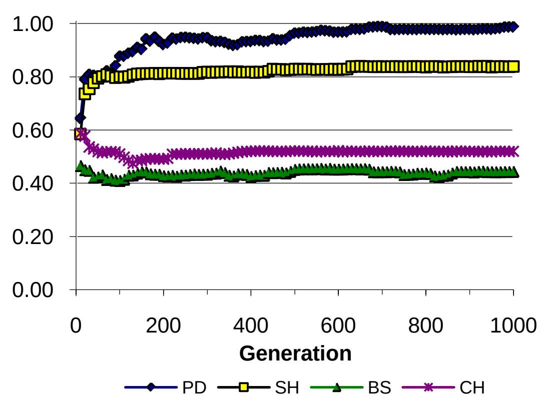


Figure 3A.1. Evolution of statistics in one-game, cost 4 environments (continued).

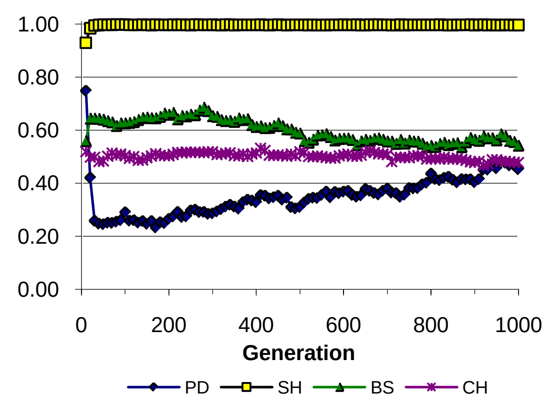
Cost
in 1-game settings with **cognitive costs**



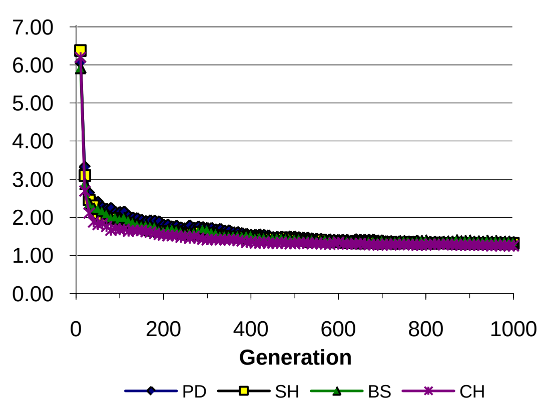
Action 0 usage frequency
in 1-game settings with **cognitive costs**



Action 0 usage frequency
in 1-game settings with **no costs**



Distinct states activated in all games
in 1-game settings with **cognitive costs**



Distinct states activated in all games
in 1-game settings with **no costs**

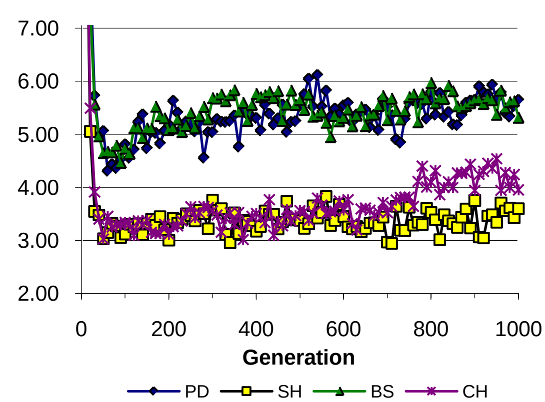


Figure 3A.1. Evolution of statistics in one-game, cost 4 environments (continued).

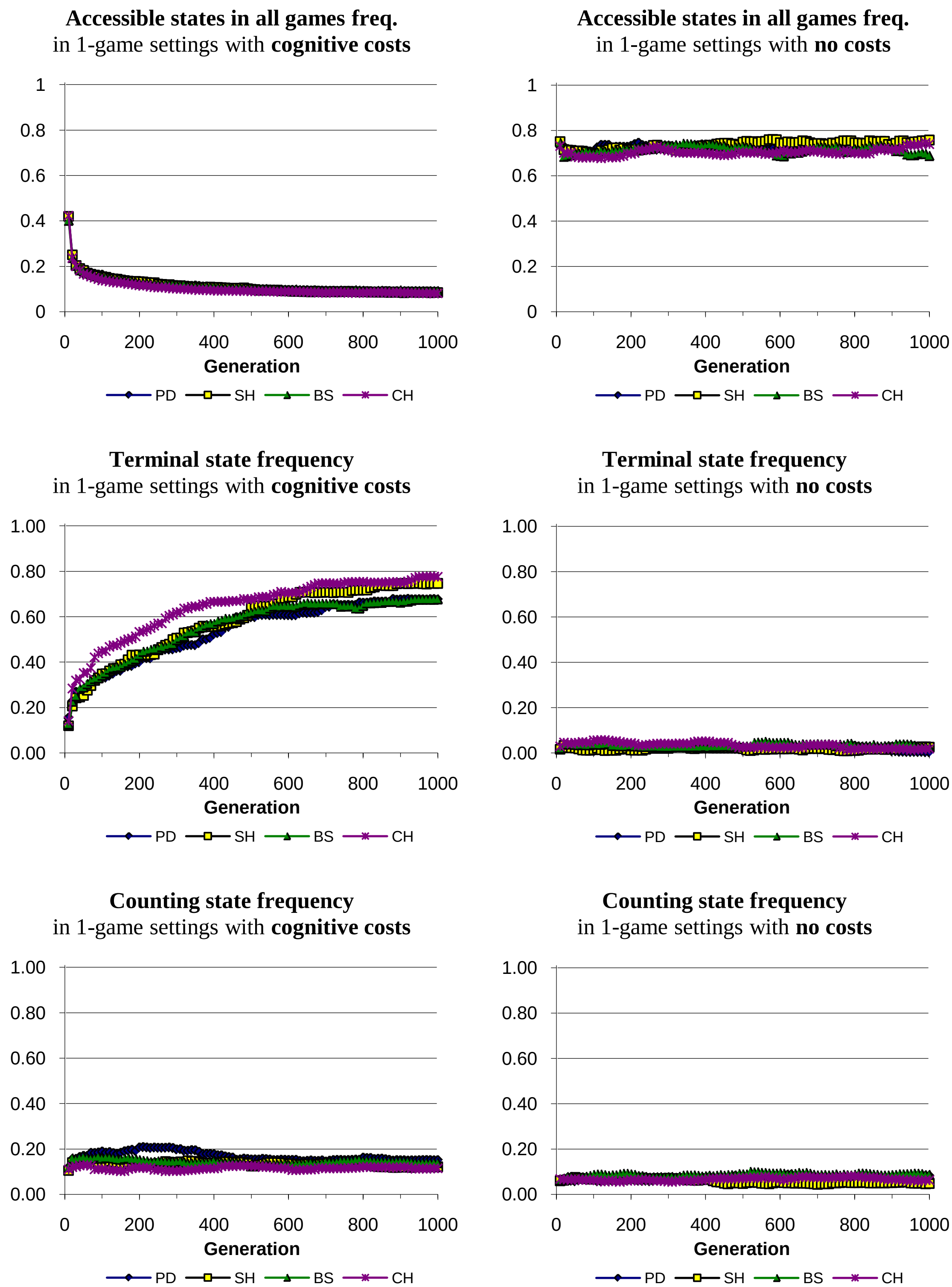


Figure 3A.1. Evolution of statistics in one-game, cost 4 environments (continued).

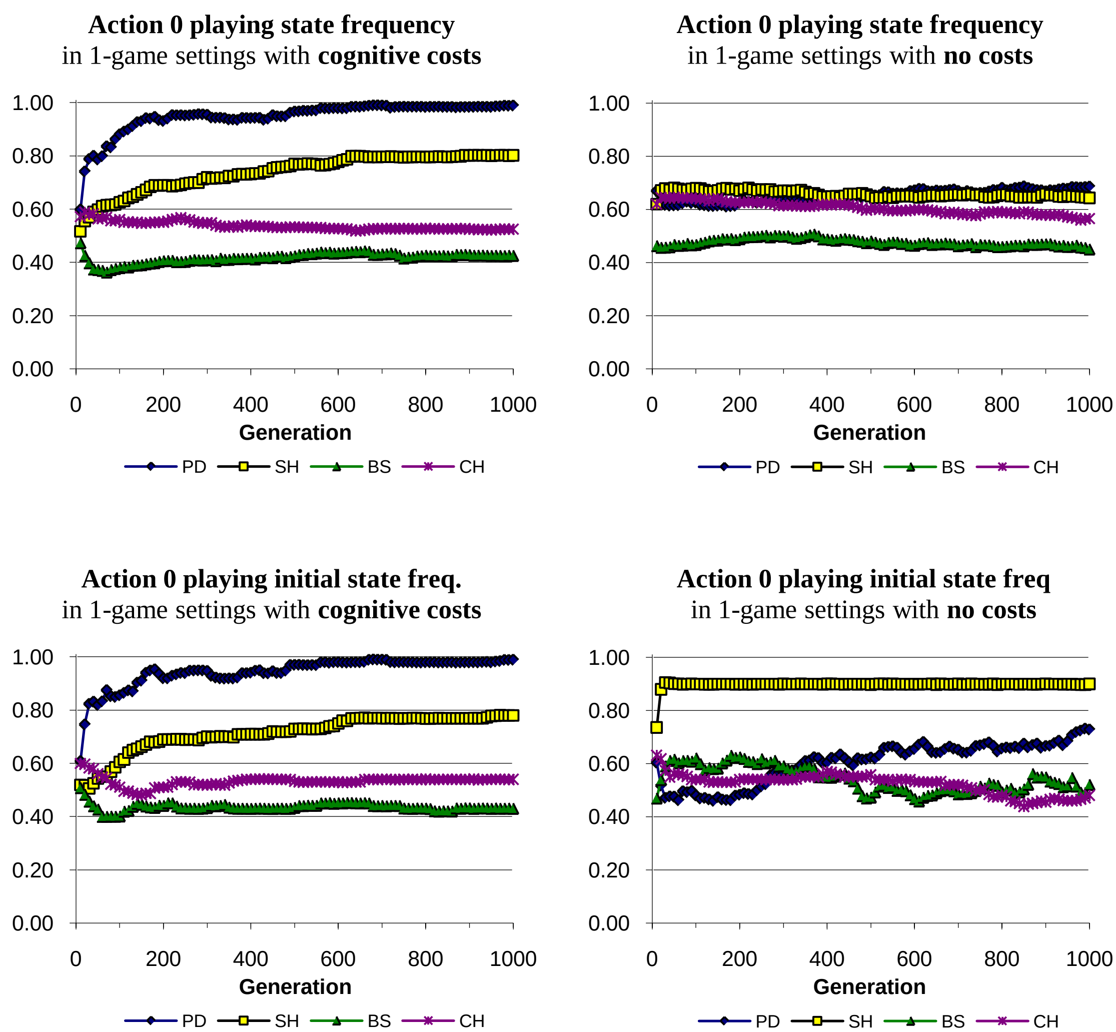
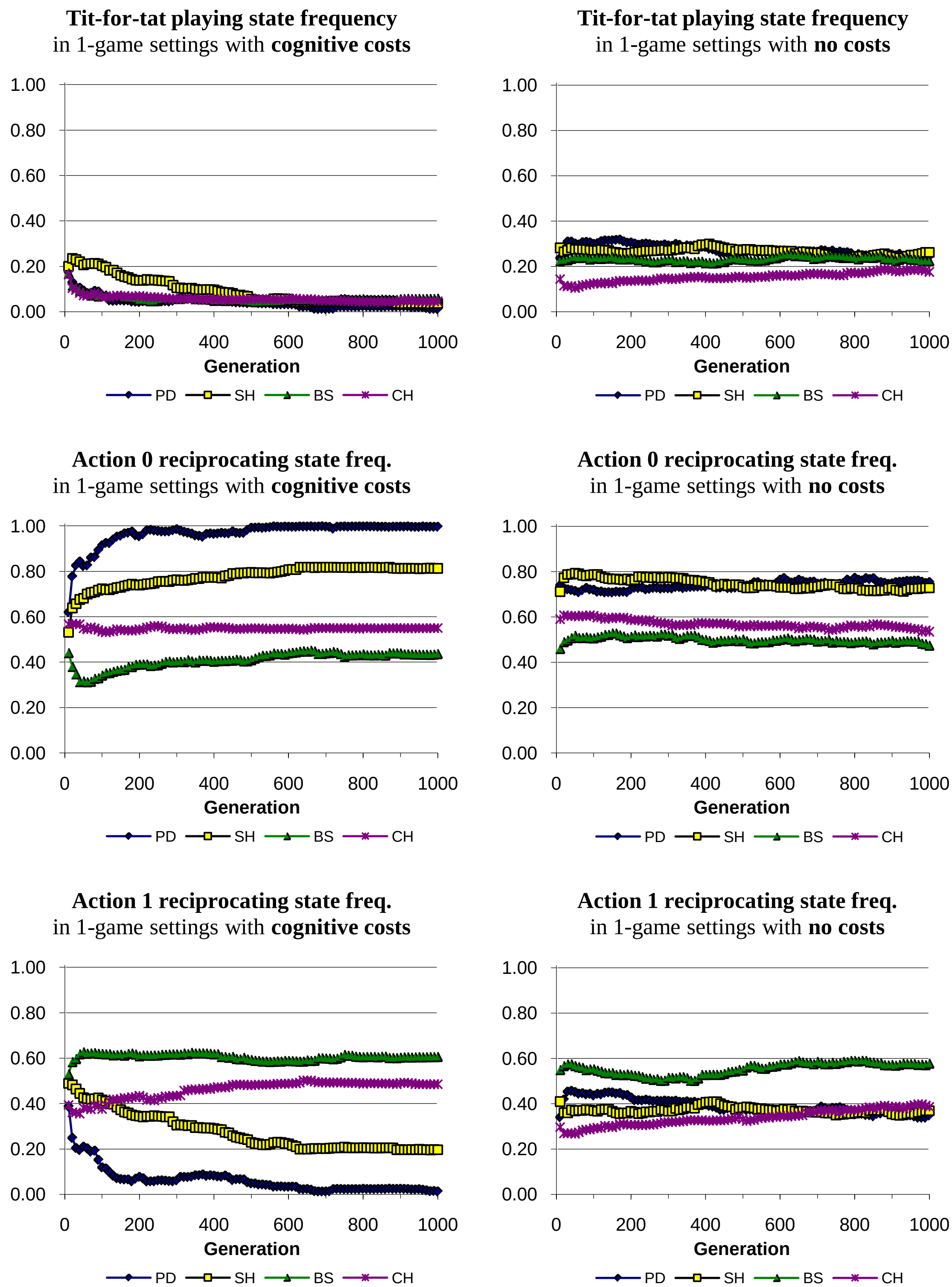


Figure 3A.1. Evolution of statistics in one-game, cost 4 environments (continued).



Bibliography

Bednar, J., Chen, Y., Liu, T. X., Page, S. 2010. "Behavioral Spillovers and Cognitive Load in Multiple Games." http://yanchen.people.si.umich.edu/papers/Multiple_Games_20101101.pdf.

Bednar, J., Page, S. 2007. "Can Games(s) Theory Explain Culture? The Emergence of Cultural Behavior within Multiple Games." *Rationality and Society*, 19: 65–97.

Casti, J. L. (1992). *Reality Rules II*. New York: John Wiley & Sons, Inc.

Fryer, R., Jackson M. 2008. "A Categorical Model of Cognition and Biased Decision Making." *The B.E. Journal of Theoretical Economics*, 8(1) (Contributions), Article 6.

Gilboa, I., Schmeidler, D. 1995. "Case-Based Decision Theory." *The Quarterly Journal of Economics*, 110(3): 605-639.

Gilboa, I., Schmeidler, D. 2001. *A Theory of Case-Based Decisions*. Cambridge, UK: Cambridge University Press.

Güth, W. 2000a. "Boundedly Rational Decision Emergence: A General Perspective and Some Selective Illustrations." *Journal of Economic Psychology*, 21: 433-458.

Güth, W. 2000b. "Robust Learning Experiments: Evidence for Learning and Deliberation." Humboldt-Universität Discussion Paper (Economic Series) No. 165.

Harsanyi, J. 1995. "A new theory of equilibrium selection for games with complete information." *Games and Economic Behavior*, 8(1): 91-122.

Holland, J. H., Miller, J. H. 1991. "Artificial Adaptive Agents in Economic Theory." *AEA Papers and Proceedings*, 81(2): 365 – 370.

Hong L., Page S. 2009. "Interpreted and Generated Signals." *Journal of Economic Theory*, 144(5): 2174-2196.

LiCalzi, M. 1995. "Fictitious Play by Cases." *Games and Economic Behavior*, 11: 64-89.

Miller, J. H. 1996. "The Coevolution of Automata in the Repeated Prisoner's Dilemma." *Journal of Economic Behavior and Organization*, 29: 87-112.

Miller, J. H., Page, S. E. 2007. *Complex Adaptive Systems: An Introduction to Computational Models of Social Life*. Princeton, NJ: Princeton University Press.

Multitasking. (2010, November). In *The Oxford English Dictionary*. Retrieved from <http://www.oed.com.proxy.library.nd.edu/Entry/123632>; accessed 13 March 2011.

Nisbett, R. 2003. *The Geography of Thought: How Asians and Westerners Think Differently...and Why*. New York: Free Press.

Page, S. 2007. *The Difference: How the Power of Diversity Creates Better Firms, Schools, Groups, and Societies*. Princeton, NJ: Princeton University Press.

Rapoport, A., Guyer, M. J., Gordon, D. G. 1976. *The 2 X 2 Game*. Ann Arbor, MI: The University of Michigan Press.

Rubinstein, A. 1998. *Modeling Bounded Rationality*. Cambridge, MA: MIT Press.

Samuelson, L. 2001. “Analogies, Adaptation, and Anomalies.” *Journal of Economic Theory*, 97: 320-366.

Synergy. (2010, November). In *The Oxford English Dictionary*. Retrieved from <http://www.oed.com.proxy.library.nd.edu/Entry/196482>; accessed 13 March 2011.

Chapter 4

The Role of Strategy Experience in Costly Multiple Game Environments

4.1. Introduction

People face incredibly complex strategic situations every day. The complexity can arise from many sources, and perhaps one of most perplexing is variations in the strategic environment itself. We must interact in numerous and distinct strategic settings simultaneously. Multiple game ensembles can be constructed to model such situations.

One natural question that arises in these models is how past learning affects success or failure in the multiple game environments. Can a strategy that proved successful in a single-game environment continue its effectiveness once it is applied to an ensemble of games? How is this effectiveness dependent upon the specific game history of the player, her opponent, and the nature of the multiple game environment itself?

This paper attempts to resolve these questions. Each player begins with experience in the sense that she has strategies which have evolved in a specific one-game or two-game environment. Now she must play two games simultaneously against an opponent who has a different experience set. Her strategies' effectiveness in these multiple game sets are assessed along several lines including payoffs, profits, and distributions of matrix outcomes, among others.

This investigation into the role of experience is a novel contribution to the literature of multiple game settings. Others have investigated the simultaneous play of multiple games using finite automata to model strategies: Samuelson (2001) compares the benefits and costs of greater strategy sophistication, Miller and Page (2007) develop a basic theory and methodology, Bednar and Page (2007) apply analytical and computer experiment results to explain the emergence of culture, and Bednar, et al. (2010) fit human subject choices from a laboratory experiment to automaton strategies and show how behavioral spillovers and cognitive load spread across contexts.

Many others have considered the role of histories on subsequent performance in the context of games, and specifically when an agent's past differs strategically from its present. Holland (1995) develops classifier systems in which agents use tags to add in recognition. Similarly, Gilboa and Schmeidler (1995, 2001) model multiple game situations theoretically as sequences of games for which an agent must determine a current strategy based on previous encounters. Güth (2000a, 2000b) also models learning in different sequential multiple-game environments and shows how subjects learn to anticipate rule changes as they play repeated of games. The sequential nature of the games differentiates these models from the one presented here that emphasizes simultaneous play.

4.2. Experimental Design

This project employs the computational experiment framework developed in Chapter 2. It also extends the experiments detailed in Chapter 3. One of the purposes of the experiments described in that chapter was to create experienced populations of automaton strategies for future use. The experiments in this chapter make uses of these. There are two primary differences in the current project from the previous one.

First, strategies here have experience playing in different game environments. Specifically, each player's strategy pool has evolved in one of ten possible game environments. There are four single game environments: Prisoner's Dilemma {PD}, Stag Hunt {SH}, Chicken {CH}, and Battle of the Sexes {BS}; and the six dual game environments: {PD, SH}, {PD, CH}, {PD, BS}, {SH, CH}, {SH, BS}, and {CH, BS}. All game environments considered in this chapter involve cognitive costs. This game environment summarizes the experience of the player.

Second, the purpose here is to gauge how a player's own and its opponent's experiences influence outcomes and performance. Since the strategies already have experience, they compete against each other in a single generation; the model takes a snapshot of their current performance given their history. These experiments make no further use of the evolutionary toolset provided in the framework.

The section below describes specific design features, focusing mainly on the differences between the approaches used in Chapter 2.

Games and Experience

This project continues with the same four games utilized in Chapter 2: Prisoner's Dilemma (PD), Stag Hunt (SH), Chicken (CH), and Battle of the Sexes (BS). The versions of these games have ordinally ranked payoffs and are consistent with the Rapoport, et al. (1976) taxonomy; see Table 4.1. Since strategies, at least subcomponents of them, will be applied in different games, a particular generic action (up, down; right, left; etc.) must be defined consistently across games. The main results of this chapter consider the case in which the row player's up action (and the column player's left action) is her natural action—the one that leads to the Rapoport natural outcome. Both players' available actions are generically labeled "0" and "1" so that the payoff matrices match those in the taxonomy. Using this labeling system, action 0 (up or left) always corresponds to the action that leads to the *natural equilibrium* of the game[1]. An

---

[1] The natural outcome (NO) defined by Rapoport, et al. (1976) is determined by applying the following conditions in sequence: (1) if a single outcome contains the high payoff for both players (4 for each), then it is the NO (applies to Stag Hunt); (2) if there are two dominated strategies, then their elimination defines the NO (applies to Prisoner's Dilemma); (3) if there is a single dominated strategy, then after its elimination, the NO is

automaton strategy can be viewed as a collection of mental states. Under this context in multiple game settings, a state that call for action 0 to be played means that when in this state, play the action that leads to the natural outcome of that game. Thus the context of the mental state points action towards a particular *type of outcome*. Note that because the game matrices coincide with the taxonomy, neither action 0 nor action 1 consistently correlate to a cooperating or defecting action.

Actions will have these alternative contexts in Section 4.4—the alternative case in which the row player's up action (and the column player's left action) is her other-regarding or cooperate action and her down action (and the column player's right action) is her self-regarding or defect action. In that section, action 0 will always correspond to the other-regarding action (cooperate) and action 1 will always mean the self-regarding action (defect). The context of the mental state now translates as a particular *motivation* (self or collective interest) for an action. Table 4.2 presents the payoffs for Prisoner's Dilemma, Stag Hunt, and Chicken under the alternative cooperate/defect context (these games are denoted $PD^{\#}$, $SH^{\#}$, and $CH^{\#}$).

The types of experience considered fall into two categories, those with experience playing a single game and those with experience playing a binary set of games. Thus, a player's experience comes from set {{PD},{SH}, {CH}, {SH}, {PD & SH}, {PD & CH}, {PD & BS}, {SH & CH}, {SH & BS}, {CH & BS}}. Table 4.3 outlines the trial configurations. These experiments consider only the situation in which agents face cognitive costs. The costs are modeled in mostly the same way as in Chapter 3's

---

the outcome in which the player with no dominated strategies receives the higher payoff; (4) the NO is the maximin outcome (applies to Chicken and Battle of the Sexes).

experiments, which are the "Cost 4" specification defined in the framework of Chapter 2. The only difference is that in the present experiments, costs are never adjusted.[2]

The first set of experiments matches strategies that have experience with only a single game against each other in the two-game environment game set built by the union of their individual experience sets. There are six such trials. The interest in these configurations is to look for game experiences that are either broadly applicable to other game settings or narrowly confined to their own game.

The second set of experiments matches strategies that have experience playing two simultaneous games against those that have experience in just one the two games. They play the same two-game environment game set as the first player's experience. There are twelve of these trials. There are two reasons for these trials.

One is to learn whether an opponent's experience matters. For instance, suppose a player with PD&SH-experience plays first an opponent with PD-experience and then separately an opponent with SH experience. How do the two-game experience player's outcomes depend upon the opponent's experience?

The other is to compare a wider breadth of experience to a narrow one. Continuing the example above, now the objective is to compare the PD&SH-experienced player's outcomes directly to its one-game experience opponents. Since they are playing in a two-game environment, one would expect that the strategy that evolved in this environment to outperform its opponents who have, in a sense, less experience.

The third set of experiments investigates the role of action context. Section 4.4 provides a sample of the effects context by contrasting the results of the {PD, SH} and {$PD^{\#}$, $SH^{\#}$} settings as well as {PD, CH} and {$PD^{\#}$, $CH^{\#}$} settings.

---

[2] In the Chapter 3 experiments, an additive cost factor was determined so that the first generation's average profits across both row and column players were zero. This factor was then applied to costs in all subsequent generations.

Strategies

Sixteen-state finite automata are used to implement a player's strategy. Each state of a finite automaton represents a portion of an agent's strategy by specifying the action to take when in the state and the next state to which the strategy should transition after it executes the action. This transition depends on the action taken by the player's opponent. An automaton also specifies the initial state for a given game. This is an important aspect in these games in which some players lack experience in a game because this implies that they do not a have an initial state assigned for that game. Implications of this inexperience will be explained below.

Computer Experiments and Simulation Algorithm

The experiment consists of eighteen trials, each with a specific two-game environment and experience sets of its players. Each trial consists of 100 independent runs, and each run consists a single generation. During each generation, the row and column players' automata match up to play 160 rounds of the first game and then 160 rounds of the second game.

The computer experiments use the following algorithm to conduct the trials. A more detailed description of this algorithm appears in Chapter 2; specifications particular to this set of experiments are highlighted here. Table 4.4 summarizes parameter values used throughout the experiments.

Step 1: Initialize strategy populations. Row and column player strategy populations each consist of thirty automata that have evolved in a specific game-cost environment. Each automaton *i*'s structure—its initialization mappings $\theta_i(\cdot)$, action mappings $f_i(\cdot)$, and transition mappings $\tau_i(\cdot)$—have been maintained from previous experiments, detailed in Chapter 3. An automaton's experience and its effect on how the automaton executes its strategy is embedded in the automaton's structure, so no performance data (such as histories of actions and profits) is carried forward. If an automaton did not have experience playing a particular game, then the algorithm randomly assigned its initialization mapping for that game. This way an automaton has a defined state at which to start for every game.

Step 2: Play Games.[3]

Step 2.1: *Match automata*. The program employs mean matching to pair automata for competition. Under this paradigm, each of the thirty automata in the row population pairs with each of the thirty automata in the column population to play both games selected for that particular trial. Thus, each automaton plays sixty repeated games (thirty repetitions of each repeated game).

Step 2.2: *Play game(s)*. The matched automata play a repeated game for 160 rounds[4]. During every round, each automaton plays the action specified by its current state and then transitions to a state dependent on the play of its opponent.

Step 3: Stop program. There is no need to iterate to the program since strategies are not evolving.

4.3. Results

Result Group 4.1 – One-Game Experience Populations in Two-Game Settings

In this group of experiments (trials 1-6), both the row and column players have experience in only a single game, but they are matched in the binary game setting that is the union of their experience.

---

[3] Step 2 as presented in Chapters 2 and 3 was selection. Strategies are fixed in this set of experiments, so selection is not applied.

[4] The paired automata play $10 \cdot \bar{s} = 160$ rounds, where $\bar{s} = 16$ is the number of their states. Using $10 \cdot \bar{s}$ rounds ensures the play extends well beyond the automata's intrinsic memory capacity (Miller, 1996). All agents report and perceive actions accurately; there is no noise.

Result 4.1.1. *Prisoner's Dilemma experience leads always to the greatest profit and almost always to the greatest score.* In the three two-game environments that include the Prisoner's Dilemma game ({PD, SH}, {PD, CH}, and {PD, BS}), the Prisoner's Dilemma experienced strategy population always earns significantly greater profit than the other population with Stag Hunt, Chicken, or Battle of the Sexes experience (p-values less than 0.00005). See Tables 4.5.a., b., and c. Because of the method in which costs are measured, profit can only measured across the average performance in the two-game environment and not on a per game basis.[5] Breaking profit into its score (the payoffs from the game matrices) and cost components, allows a more refined look at what happens.

Similar to the Chapter 3 no-experience simulations, costs do not significantly vary between row and column players (who have different experience sets) or between different game environments; see Table 4.6. Thus, the differences in profits are driven by differences in scores.

The strategy population with PD-experience outscored its opponents when playing Prisoner's Dilemma, as shown in the left-hand columns of Tables 4.7.a, b, and c. Moreover, PD-experience even outscores SH-experience and BS-experience when playing Stag Hunt and Battle of the Sexes, respectively (see the middle columns on the table). The CH-experienced population was the only one that is able to beat a PD-experienced population, and then only when playing Chicken. Overall, the average score across both games is always significantly greater for the PD-experienced population, even compared to the CH-experienced population (see right-hand columns of Tables 4.7.a, b, and c).

---

[5] This is because profit is score net of cost and cost is determined by the number of accessible states in all games, which is not calculated on a per game basis.

An analysis of the tendency to play action 0 coupled with the effects of inexperience explain these strong results. First I will establish that action 0 is used more frequently by PD-experienced strategies and then explain the structural characteristics of these strategies that lead to the high usage of action 0.

Tables 4.8.a, b, and c. show that in these multiple game settings, PD-experience induces the high frequency of the use of action 0 in Prisoner's Dilemma games (at least 99.0 percent of rounds no matter what experience the opponent has), in the other three games (about 72 percent of rounds) and on average in each of the three two-game sets (about 86 percent of rounds). These percentages are all significantly greater than the opponent's use of action 0 except for the in the Stag Hunt game against a SH-experienced player.

Before evaluating the impact of this high frequency of action 0 usage by PD-experienced populations on scores, first consider the structural features of PD-experienced strategies that lead them to play action 0 so often. The first structural statistics that provides support is the percentage of *initial* states that play action 0. When playing Prisoner's Dilemma, the population with PD-experience always has significantly more initial states that play action 0 than its SH-, CH-, or BS-experienced opponent (p-values of 0.0000). The magnitude of the PD-experienced population's excess of action 0 playing initial states over its opponents is so great that the same result holds when considering the average over both games in the two-game set. See Tables 4.9.a, b, and c.

Expanding the view beyond just initial states to consider the number of *accessible* states that play action 0, we see a similar effect: PD-experienced populations have significantly more action 0 playing accessible states than opponents with other experience when playing PD and when considering the average of both games in the two-game set, and even when playing Chicken and Battle of the Sexes. Stag Hunt-experience is the only population that PD-experience does not have more 0-playing accessible states and then only when playing Stag Hunt. See Tables 4.10.a, b, and c.

Another structural feature of PD-experienced automata is their extremely high frequency of action 0 reciprocating states: those states that transition to a state that plays action 0 in response to the opponent's play of action 0. Table 4.11 shows that this frequency was 99.6 percent for a PD-experienced strategy when playing Prisoner's Dilemma in a multigame setting (regardless of its opponent's experiences) and about 64 percent when it played any of the other three games (again, regardless of its opponent's experiences). In all cases, this frequency of action 0 reciprocation is significantly greater than the opponent with a different experience base.

These structural characteristics to play action 0—both in initial states and accessible states—translate to the high frequencies that the PD-experienced strategies actually employ action 0 as described above. It is this feature of PD-experienced strategies in multiple game settings that enables their high scores.

First, consider why a PD-experienced row player outscores a BS-experienced player in the Battle of the Sexes game within the {PD, BS} environment. Table 4.7.c shows the scores are 2.758 for PD-experience compared to 2.570 for BS-experience (the p-value for the difference being distinct from zero is 0.0139).

Action 0 is the natural strategy action for both of these games, and the experience of each player leads to different propensities to play action 0. The PD-experience strategy evolved in a PD setting to use action 0 in nearly all rounds (99 percent, see Chapter 3, Table 3.6). Then in the multiple game setting, the PD-experienced player, of course, lacks expertise in Battle of the Sexes, and its initial state for BS is randomly assigned. It still uses action 0 in a relatively high frequency (72.9 percent), but not as high a frequency as it did in the single Prisoner's Dilemma game environment.

The BS-experience strategy evolved in a BS setting to use action 0 in about half of the rounds (52 percent; see Chapter 3, Table 3.6). Since the BS-experience player tailored its strategy to play Battle of the Sexes, it continues to use action 0 with about the same frequency (54.2 percent, see Table 4.8.c) in the multiple game setting as in the single game setting.

The player's different experiences lead to quite different tendencies to use action 0: 72.9 percent versus 54.2 percent. It is the *asymmetry* of these action 0 frequencies that lead to PD-experience's greater score. Because the BS-experienced column player is more likely to use action 1 (100 – 54.2 = 45.8 percent) than was the PD-experienced row player (100 – 72.9 = 27.1 percent), the outcome lies in the (0, 1) payoff matrix cell more than twice as often than the (1, 0) cell. The bottom panel of Table 4.12.c shows that play ended in the (0, 1) cell 35.1 percent of the time compared to only 16.3 percent of the time for the (1, 0) cell. Thus, the row player with PD-experience was more than twice as likely to receive a payoff of 4 than was the column player with BS-experience. The different experience led to the disparity of payoffs between these two types of experience in Battle of the Sexes. In this particular case, Prisoner's Dilemma experience translates better to a Battle of the Sexes game than vice versa. The advantage lies in the outcome asymmetry that is created. Asymmetric transference is not uncommon in real situations. Using a sports analogy, soccer players often can integrate successful into the long distance running sport of cross country; the soccer player's experience has prepared her well for the long distance running which demands stamina. The reverse integration may likely less successful; a cross country runner would likely lack the necessary ball handling skills and tactics necessary for success in soccer.

Next, consider the Prisoner's Dilemma and Stag Hunt multiple game setting. When learning to play Prisoner's Dilemma by itself, strategies developed a very strong tendency to play the natural action 0; by the end of their evolution, this action is utilized in 99 percent of rounds (see Chapter 3, Table 3.6). The corresponding (0, 0) outcome, the Nash equilibrium, occurred in 97.1 percent of rounds (see Chapter 3, Table 3.10.a). When learning to play Stag Hunt by itself, strategies developed a less intense inclination to play the action 0; this action was utilized in only 84 percent of rounds (see again Table

3.6). In the single game setting of Stag Hunt, the (0, 0) outcome, the Pareto superior Nash equilibrium, occurred in 83.6 percent of rounds, while the (1, 1) outcome, the Pareto inferior Nash equilibrium, occurred in 15.9 percent of rounds (see Table 3.10.b).

These learned action 0 tendencies are hard-wired into the automaton structures (in their high frequency of initial and accessible states that play action 0 as described above) and lead to non-symmetric outcome distributions when the experienced players are matched together. Compare the top and bottom panels of Table 4.12.a. that show the distribution of payoff matrix outcomes for Prisoner's Dilemma and Stag Hunt, respectively, in the two-game environment that matches these two types of experienced players.

Starting with the Prisoner's Dilemma game, the PD-experienced row players used action 0 in 99.3 percent of the rounds (see Table 4.8.a), while the SH-experienced column players used action 0 in 67.1 percent of rounds. Given their usage of action 0 (and shown in Table 4.12.a), two-thirds of the time play results in the (0, 0) outcome and both get a payoff of 2. However, roughly one-third of the time, play results in the (0, 1) outcome where the PD-experienced row player gets the 4 payoff while the SH-experienced column player gets the 1 payoff (see Table 4.12.a.). The SH-experienced column player does not have a set of rules developed specifically for Prisoner's Dilemma, and so its randomly assigned initial state for PD causes it to activate a state that plays action 1 in this game more than would otherwise (action 0 frequency drops from 84 to 67 percent, so action 1 frequency increases from 16 to 33 percent). These deviations from action 0 to action 1 are caused by the SH-experienced column player's *lack of experience in Prisoner's Dilemma*. The deviations help the PD-experienced row player by giving it its greatest payoff (4) and harm the SH-experienced column player by giving it its least payoff (1). Intuitively, one would not expect an SH-experienced player to do better than the PD-experienced player in Prisoner's Dilemma.

Counterintuitively, even when playing Stag Hunt, the row population's experience in Prisoner's Dilemma prepared it to outscore the SH-experienced column player. The middle two columns of Table 4.7.a. show the two populations' scores when playing Stag Hunt; the PD-experienced row player earns a score of 3.222 on average, while the SH-experienced column player earns only 3.053. This difference is fairly significant with a p-value of 0.1657.

The bottom panel of Table 4.12.a. shows the distribution of outcomes in the Stag Hunt game that resulted when these two experienced players are matched in this game. The difference in the players' scores came from the asymmetrical outcome percentages in the (0, 1) cell, 15.3 percent, and the (1, 0) cell, 23.7 percent. In 15.3 percent of rounds that resulted in the (0, 1) outcome, the PD-experienced row player received a payoff of 1 and the SH-experienced column player received a payoff of 3, so the SH-experienced player did better. However, in the greater percentage—23.7 percent—of rounds that resulted in the (1, 0) outcome, the PD-experienced row player received a payoff of 3 and the SH-experienced column player received a payoff of 1, so the PD-experienced player outscores the SH-experienced player overall.

Unlike Prisoner's Dilemma, in Stag Hunt a row player sticking with the natural action 0 while the column player switches and plays action 1 incurs a more severe penalty (row's payoff falls from 4 to 1) for the natural action row player than for the column player (whose payoff falls from 4 to 3). The PD-experienced player is more likely to deviate in Stag Hunt precisely because it lacks experience in this game. While it evolved to almost always play action 0, this tendency would also make it almost always play the natural strategy in SH. Specifically, The PD-experienced strategy's crucial structural characteristic that directs it at which state to start play in Stag Hunt is simply a random guess among its sixteen states; it has had no experience to fine tune this initial position. Given that the strategy evolved to use less than two of its sixteen states (see Chapter 3, Table 3.6), fourteen of its states have not been subjected to evolutionary pressure and so will have a more random structure. Accordingly, the newly assigned initial state for Stag Hunt has a reasonable chance to start at action 1. Its lack of experience causes it to switch from action 0 to action 1 to its benefit and to its SH-experienced opponent's detriment.

Result 4.1.2. *Stag Hunt experience handicaps the player in the game of Stag Hunt. The player with SH-experience receives lower payoffs than its PD-, CH-, and BS-experienced opponents even when playing Stag Hunt.* The lack-of-experience shortcoming was a liability in the Prisoner's Dilemma; in Stag Hunt it turns out to be an asset. PD-experience outscores a SH-experienced player in a Stag Hunt game as explained above. Similar arguments can be constructed to explain why a player with experience in either Chicken or Battle of the Sexes outscores a SH-experienced player in the game of Stag Hunt. The cause of a SH-experienced strategy's lower score is the other strategy's random initial state for Stag Hunt (since in lacks experience in this game) and the resulting asymmetries of action 0 and 1 usage among the two players.

A player with Stag Hunt experience loses on its home turf. PD-experience automata outscore SH-experience automata when playing Stag Hunt (3.222 > 3.053), CH-experience automata outscore SH-experience automata when playing Stag Hunt (3.126 > 2.553), and BS-experience automata outscore SH-experience automata when playing Stag Hunt (3.138 > 2.581). The two-tailed p-values are 0.1567, 0.0000, and 0.0000, respectively. See Tables 4.7.a, d, and e.

The lack of cost differentiation between the multiple game environments implies a similar effect with profit. In the three environments that involve a SH-experienced strategy population, that population always earns significantly less profit than the other populations with Prisoner's Dilemma, Chicken, or Battle of the Sexes experience (p-values are 0.0000, 0.0001, and 0.0606). See Tables 4.5.a, d, and e.

Result 4.1.3. *Inexperience always increases the prevalence of Tit-for-Tat components of strategies with the average increase15.9 percent.* Inexperience in a particular game always causes an automaton to have more accessible states that play the Tit-for-Tat strategy than its experienced opponent in that game. These Tit-for-Tat states reciprocate the opponent's play of action 0 by transitioning to a state that plays action 0 and reciprocate the opponent's play of action 1 by transitioning to a state that plays action 1. Whenever a player with experience in game *X* plays game *Y* with a player that has experience in game *Y*, the *X*-experienced player's automaton strategies will have more Tit-for-Tat states than the *Y*-experienced player's automaton strategies. This is a very robust result—it holds for all twelve possible combinations of games formed by substituting PD, SH, CH, and BS in for *X* and *Y* above, and the two-tailed p-values for differences are all less than 0.0001. Tables 4.13.a-f list the frequencies of (accessible) states that play Tit-for-Tat. The right-hand columns of this table provide the frequencies differentials between the inexperienced and experienced strategy populations for each game within the two-game environments: all twelve differentials are positive (indicating the inexperienced player is using a greater frequency) and range from 11.1 percent to 20.6 percent, with the mean increase being 15.9 percent.

Chapter 3 showed that in single game environments in which players face cognitive costs, a very small percentage of accessible states played Tit-for-Tat. This was true for all four games, ranging from 1% for Prisoner's Dilemma to 6% for Chicken. Tit-for-Tat components were not successful in those environments.

The same experiments in Chapter 3 also showed that the sixteen-state automata evolve their structures such that they actually only activate very few states (on average, about 1.3 states in single game settings and about 2.1 states in multiple game settings). Thus the sixteen-state automaton is usually isomorphically equivalent to a one- or two-state automaton (Miller and Page, 2007). So when a player faces a novel game not found in its experience set, there is a high probability that its newly assigned initial state for that game will fall outside of the small subset of states it has evolved to use in its strategy. If the new initial state lies outside the experienced game's accessible set that that is previously used, then that new initial state is likely to have a more random appearing structure—since it was not involved in determining the success of the strategy, it would not have been subjected to evolutionary pressure to change in any particular way. The same would be true of most other states to which the new initial state transitioned. Thus, when playing a game in which a player has no previous experience, the states that are activated are likely to have components (actions and transitions) that have no regularity to their structures since they have not been subjected to evolutionary pressures.

Any randomly created state would have 25 percent probability of playing Tit-for-Tat.[6] Table 4.13. shows that the inexperienced player in each pair had about 20 percent of its accessible states play Tit-for-Tat. This makes sense; usually, that randomly assigned initial state has a 25 percent chance of playing Tit-for-Tat, but occasionally the random assignment would fall within the automaton's accessible set (for the game in which it has experience), and then have a very small probability of playing Tit-for-Tat.

<u>Result 4.1.4</u>. *Inexperience does not significantly influence several strategy statistics including cost, distinct states activated in all games, distinct initial states, automaton-level state activation similarity, and accessible states across all games.* See Tables 4.A.1, 2, 3, and 4 in the appendix for the statistic's values and their (small) t-statistics that the differences between the experienced and inexperienced player are distinct from zero and correspondingly large p-values. For the most of the other statistics, there are very predictable experienced-based distinctions on the per game basis, but these cancel each other out across the two-game set resulting in no difference on average that is based on experience. These statistics include mean states activated per game, population-level state activation similarity, mean accessible states per game, and

---

[6] There is a 0.5 probability that state *m*'s transition function, in response to the opponent's action 0, assigns a state *n* that plays action 0, and an independent 0.5 probability that state *m*'s transition function, in response to the opponent's action 1, assigns a state *o* that plays action 1.

proportion of terminal states (see Tables 4.A.5, 6, 7, and 8). The randomly assignment of the initial state for the new game for which a player lacks experience drives this observation. The effect of this random assignment is to likely start the automaton's active state along a segment of the automata where transitions are more likely to be random. Since most automata have evolved to use only a few states (see Result 3.1.2 in Chapter 3), then it is probable that the randomly assigned state will not belong to this subset of states. The majority portion of the automaton that is not used has not been subjected to as much adaptive pressure through selection and the genetic operators, and thus is more likely to have a random structure. This randomness directly impacts the statistics listed above in predictable ways.

Result Group 4.2 – Two-Game Experience v. One-Game Experience in Two-Game Settings

In this group of experiments (trials 7-18), the row player has experience playing in a binary game setting, column player has experience playing in a single game setting. The first result compares the two players directly. The subsequent results compare players who faced a common opponent.

Result 4.2.1: *Broader experience always leads to significantly greater profit, with the two-game experienced player earning a profit premium over the one-game experienced player ranging from 1.26 to 2.09 with a mean of 1.53.* For a reference point to understand how large this profit advantage is, the maximum possible profit is 2.87.[7] Whenever the row player that had experience playing both of the games it faced was matched against the column player that had experience playing only one of the two games, the row player always received the greatest profit. This was true for all twelve

---

[7] This maximum profit stems from a strategy that receives a score of 4 from the payoff matrix that has only one state accessible in both games (that is, the same state accessible for both games), incurring a cost of $(1 + 1/16)^2 = 1.13$.

possible combinations of a player with two-game experience matched against a player game experience. Tables 4.14.a and b through 4.19.a and b present the mean profits as well as the t-statistics that the difference in profit between the two players is non-zero. Eleven of the corresponding p-values are less than 0.0001, and the other is only 0.0469 (for {PD, SH}-experience matched against PD-experience in the Prisoner's Dilemma game). This result is not surprising on an intuitive level. It is easily explained on a structural level by the random assignment of the initial state for the new game that the one-game experienced column player had to then play.

Result 4.2.2: *Prisoner's Dilemma experience continues to dominate the other types of experience in terms of profit; against the same two-game experienced opponent PD-experience earns profit premiums of 0.124, 0.188, and 0.300 compared to SH-, CH-, and BS-experience, respectively.* See Tables 4.14.c, 4.15.c, and 4.16.c. This result is similar to and resonates with Result 4.1.1. In the three two-game settings that include Prisoner's Dilemma ({PD, SH}, {PD, CH}, and {PD, BS}), there are six possible pairings. The row player is always the one with the experience in the two-game set. The column player has experience in only one of the two games. For instance, trial 7 matched the {PD, SH}-experienced row player against the PD-experienced column player, and trial 8 matched the {PD, SH}-experienced row player against the SH-experienced column player. The profit received by the column player with PD-experience from trial 7 can be meaningfully compared to the profit received by the column player with SH-experience from trial 8 because both face the same opponent in the same two-game setting. In this comparison, the PD-experienced player has the greater profit (0.057 > -0.067, p-value 0.0671). The same is true for the other two-game environments that combine Prisoner's Dilemma with Chicken (-0.234 > -0.422, p-value 0.0310) and with Battle of the Sexes (-0.351 > -0.651, p-value 0.0014).

The argument to illustrate why a player with Prisoner Dilemma experience bests a player with Stag Hunt experience parallels the one presented in Result 4.1.1—

inexperience means a random assignment of an initial state, and this leads to asymmetric outcomes that favor the PD-experienced strategy. The {PD, SH}-experienced population evolved to use natural action 0 in Prisoner's Dilemma with frequency 0.95 and to use natural action 0 in Stag Hunt with frequency 1.00 (Chapter 3, Table 3.8). Then when facing an opponent with either PD experience or SH experience, it continues to almost always use action 0 in either game with frequency 0.99, so the result is almost always in the top row of the relevant payoff matrix. When playing PD against this {PD, SH}-experienced opponent, the PD-experienced player that evolved to play action 0 with frequency 0.99, still uses this high frequency, and so play results in the Pareto-dominated (0, 0) equilibrium in 99.4 percent of rounds (Table 4.20), giving the PD-experienced column player a payoff close to 2 (2.002, see Table 4.21). The SH-experienced player uses action 0 less frequently for two reasons. First, it lacks expertise in PD, and its random initial state leads to lower use of action 0 since there is roughly a 14-in-16 chance that the newly assigned initial state will fall in the non-optimized portion of the automaton (where action 0 and 1 are both equally likely). Second, even for the 2-in-16 chance when the new initial state does fall within in evolved structure, its tendency to play action 0 is less: it evolved to play the action 0 with a relatively lower frequency (0.84) than the PD-experienced player (0.99). Even though the context of action 0 changes for the SH-experienced strategy from cooperative to self-serving, these two reasons coupled results in the SH-experienced players using action with frequency 0.667. Accordingly, it receives a payoff of 2.0 in two-thirds of rounds at the (0, 0) outcome and only 1.0 in the remaining one-third of rounds at the (0, 1), giving it a payoff of 1.687 ($\cong$ 2/3*2.0 + 1/3*1.0). Thus, when playing Prisoner's Dilemma against a {PD, SH}-experienced opponent, the PD-experienced player outscores the SH-experienced player by 0.315.

Now compare how these two players fare when playing Stag Hunt against the {PD, SH}-experienced opponent. Although action 0 led to the sub-optimal equilibrium in Prisoner’s Dilemma, now it leads to the Pareto dominant equilibrium in Stag Hunt. A closer look at usage frequencies explains why the PD-experienced player still does relatively well in Stag Hunt.

The SH-experienced player uses the natural action 0 almost as frequently (0.804) as it did when it evolved in the single game setting (0.84). Now it is the PD-experienced player that gets a randomly assigned initial state. Since the {PD, SH}-experienced row player is almost always playing action 0, the players are best served by also playing action 0 all the time to reach the payoff dominant equilibrium (0, 0) to receive a payoff of 4. While the lack of experience does cause the PD-experienced player to deviate more often to action 1 (frequency 0.282) than the SH-experienced players (0.196) for a similar reason as the preceding paragraph, this is less problematic. Compared to the SH-experienced player’s deviates when playing Prisoner’s Dilemma, the PD-experienced player in Stag Hunt deviates absolutely and relatively less. Absolutely less because its action 1 frequency or 0.282 in Stag Hunt is less than the SH-experienced players action 1 frequency of 0.333 in Prisoner’s Dilemma. Relatively less because the difference between action 1 usage in Prisoner’s Dilemma is 0.329 (0.333 – 0.004), while the difference between action 1 usage in Stag Hunt is 0.086 (0.282 – 0.196). So there are fewer deviations from the equilibrium action by PD-experience in Stag Hunt than by SH-experience in Prisoner’s Dilemma, and the deviations incur a penalty of a one point lower payoff in both games.

All this results in SH-experienced player outscoring the PD-experienced player in Stag Hunt by only 0.103 (3.792 – 3.689, see Table 4.21). Recalling that in the opposite situation, PD-experienced player outscores the SH-experienced player by 0.315, and that there is no significant difference between cost (across both games) in either situation, we see that on the whole, the PD-experienced player earns more profit than the SH-experienced player.

Similar asymmetric outcome arguments can be constructed to show why PD-experienced player earns higher profit against the {PD, CH}-experienced and {PD, BS}-experienced than do CH-experienced and BS-experienced players, respectively. Noticeably, the other three pairings that do not involve Prisoner's Dilemma ({SH, CH}, {SH, BS}, and {CH, BS}) show no significant difference between the profits earned by players with different single-game experiences.

<u>Result 4.2.3</u>: *An opponent's experience matters: a {X, PD}-experienced player earns* less *profit when facing a PD-experienced player than when facing an X-experienced player for* $X \in$ *{SH, CH, BS}; the profit premiums are -0.457, -0.199, and -0.524, respectively (see Tables 4.14.d, 4.15.d, and 4.16.d). Alternatively, a {X, SH}-experienced player earns* more *profit when facing a SH-experienced player than when facing an X-experienced player for* $X \in$ *{PD, CH, BS}; the profit premiums are 0.457, 0.535, and 0.255, respectively (see Tables 4.14.d, 4.17.d, and 4.18.d).*

A player that has two-game experience receives lower profit when its opponent has (single game) Prisoner's Dilemma experience than when its opponent has (single game) experience in one of the other three games. For example, take the {PD, SH}-experienced row player; against the PD-experienced column opponent, the row player earned a profit of 1.313 (Table 4.14.a). Matched against a SH-experienced column player, the row player earned a profit of 1.770 (Table 4.14.b). The t-statistic for the difference is 7.174 and the corresponding p-value is less than 0.0001 (Table 4.14.d). Similarly significant differences are established in Table 4.15 (for {PD, CH}) and Table 4.16 (for {PD, BS}).

Just the opposite is true when the opponent has Stag Hunt experience. A player that has two-game experience receives more profit when its opponent has Stag Hunt experience than when its opponent has experience in one of the other three games. The case for the {PD, SH}-experienced player is established in preceding paragraph and Table 4.14. Table 4.17 presents an analogous case for the {SH, CH}-experienced player, and Table 4.18 does the same for the {SH, BS}-experienced player. Again, the arguments for these results are quite similar to the one detailed above in Result 4.2.2.

4.4. Cooperate/Defect Context

This section considers an alternative interpretation of actions available to the players. Here, a player's actions are viewed as other-regarding (or cooperate), denoted by "C," or as self-regarding (or defect), denoted by "D." Table 4.2 presents the payoffs and features for the three games under these new action connotations, $PD^{\#}$, $SH^{\#}$, and $CH^{\#}$. BS is excluded in this section because it lacks clear associations of its actions with these contexts. With these payoff matrices, now action 0 corresponds always to action C. Note that the only new payoff configuration in for Prisoner's Dilemma; $SH^{\#}$ and $CH^{\#}$ are identical to SH and CH from Section 4.3.

Result 4.4.1. *Prisoner's Dilemma experience always leads to the greatest profit and the greatest score.* The natural outcome context produced a similar result (Result 4.1.1); even though the change of contexts effectively transposed the payoffs in PD to those in $PD^{\#}$, experience in this game still is a boon. In the both two-game environments that include the Prisoner's Dilemma game, {$PD^{\#}$, $SH^{\#}$} and {$PD^{\#}$, $CH^{\#}$}, the $PD^{\#}$-experienced strategy population always earns significantly greater profit than the other population with either $SH^{\#}$- or $CH^{\#}$- experience (p-values less than 0.00005). See Table 4.22. Again, costs do not vary much from game to game, so the profit differences are driven by scores.

The strategy population with PD-experience achieves a greater score than its opponents when playing $PD^{\#}$, as shown in the left-hand columns of Table 4.23 (two-tailed p-values of 0.0000). Moreover, $PD^{\#}$-experience even outscores $SH^{\#}$-experience and $CH^{\#}$-experience when playing $SH^{\#}$ and $CH^{\#}$ (see the middle columns on the table, p-values are 0.0000 and 0.0750, respectively). This is an even stronger result than under the natural outcome context, because in that context the CH-experienced population outscored the PD-experienced population when playing Chicken.

To explain this phenomenon, consider the outcome distributions of single-game experienced players for the cooperate/defect context in Table 4.28. The basic arguments follow the lines presented in the support for Result 4.1.1 that match (a) the evolved action C usage frequency from Chapter 3, Table 3.18 for the player with experience in the game being played with (b) a closer-to-random action C usage frequency for the player with no experience in the game because she has a randomly assigned initial state. These action C usage frequencies result in outcome distributions whose asymmetries always favor the PD$^{\#}$-experienced player.

For example, consider when the players with PD$^{\#}$-experience and SH$^{\#}$-experience square off to play SH$^{\#}$. Table 3.18 shows that the player with SH$^{\#}$-experience evolved to play action C with frequency 0.84. Since she is still playing SH$^{\#}$, she can be expected to continue to play action C with roughly this frequency. And she does; in SH$^{\#}$ within the {PD$^{\#}$, SH$^{\#}$} setting, she plays action C with frequency 0.805 (inferred from Table 4.28.a, bottom panel). The PD$^{\#}$-experience player evolved to play action C with frequency 0.01; but this player is now playing an unfamiliar game, and so does not have an efficiently assigned initial state. Since she starts playing at a random state, she does not play action C with such a low frequency—her actual frequency of action C usage is now 0.278. The particular action C usage frequencies by the two players lead to the asymmetric outcomes. The (D, C) outcome is reached in 57.2 percent of rounds at which the PD$^{\#}$-experience player gets a payoff of 3 while the SH$^{\#}$-experience player gets a payoff of only 1. The other off-diagonal outcome cell at which these payoffs are inverted only occurs in 4.5 percent of rounds, and both players receive the same payoff at the on-diagonal payoffs. Thus, the PD$^{\#}$-experienced player earns the greater score.

Supporting this result further is a comparison of the profits of the PD$^{\#}$-experienced to those of the SH$^{\#}$-experienced player when each is matched against a player with experience in the two-game {$PD^{\#}$, $SH^{\#}$} setting. Trials 21 and 22 of the experiments matched the row player with experience in both $PD^{\#}$ and $SH^{\#}$ with the column player who had experience in only $PD^{\#}$ and $SH^{\#}$, respectively. Since both single-game experience players are facing the same opponent, we can meaningfully compare their profits and scores.

Looking first at profits, we see the same effect: Table 4.24 shows that the $PD^{\#}$-experienced player earns a higher profit (-0.670) than the $SH^{\#}$-experienced player (-0.945) when playing this common {$PD^{\#}$, $SH^{\#}$}-experienced opponent (p-value for the difference is 0.0173). Similarly, Table 4.25 shows that the $PD^{\#}$-experienced player earns a higher profit (-0.603) than the $CH^{\#}$-experienced player (-0.753) when playing a common {$PD^{\#}$, $CH^{\#}$}-experienced opponent, although the significance of the difference in profits is borderline (p-value is 0.1784).

Turning attention to scores in the two-game experience versus one-game experience trials at least does not contradict the $PD^{\#}$-experience superiority result. Table 4.26 shows that when the {$PD^{\#}$, $SH^{\#}$}-experienced row player matches against the $PD^{\#}$-experienced column player in $PD^{\#}$, the column player earns a payoff of 2.204; this is significantly greater than the payoff earned by the $SH^{\#}$-experienced column player in $PD^{\#}$ (1.662, p-value for difference is 0.0000).

When the same column players match against the {$PD^{\#}$, $SH^{\#}$}-experienced row player in $SH^{\#}$, the $PD^{\#}$-experienced player no longer earns a greater payoff—there is no statistically difference between the two column player’s payoffs (two-tailed p-value of 0.7296, Table 4.26.c).

Trials 23 and 24 of the experiments matched the row player with experience in both $PD^{\#}$ and $CH^{\#}$ with the column player who had experience in only $PD^{\#}$ and $CH^{\#}$, respectively. These trials produce qualitatively the same results described in the preceding two paragraphs—see Table 4.27.

Thus, experience in Prisoner's Dilemma trumps experience in both Stag Hunt and Chicken (especially in profits, and to a lesser extent in payoffs) for the cooperate/defect context and for the natural outcome context. This result is surprising; I had conjectured that the transposition of the payoffs from PD to $PD^{\#}$ would negate Prisoner's Dilemma experience effectiveness in the natural outcome context described Result 4.1.1. Instead, experience in Prisoner's Dilemma proves advantageous in either context.

4.5. Remarks

The preceding results provide strong evidence of the utility of experience in Prisoner Dilemma like settings. Strategies developed in such settings perform well in multiple game environments and not only in other Prisoner Dilemma games, but the other games as well. This in true in both the natural outcome and the cooperate/defect contexts. Opponents do worse, too, when they face a player with PD-experience. This result may provide a template for training regimes. It may also suggest an effective catch-all strategy—when in doubt as to how to proceed, play as if you face a Prisoner's Dilemma.

On the other hand, experience in Stag Hunt was detrimental to future success both in other games and even in Stag Hunt itself. Stag Hunt is the "easy" game. This suggests, then, that strategies developed in low-stress, no-conflict environments will not fare well when utilized in strategically diverse settings.

The results indicate more broadly that strategic complementarities form in multiple game environments, and experience is crucial to determining whether those complementarities will be positive or negative. This framework and these results propose an approach to model specific strategic interdependencies desired in a model. The specific situation desired can be tailored by appropriate combinations of games into a multiple game environment.

Extensions

Several natural extensions that consider other types of multiple game environments related to the present work are readily evident. First, there some ways one could expand the analysis of the four games used in the present paper. Here, the matching of past experience is limited to the cases where both players have experience in a single game or where one player has experience in two games while her opponent has experience in a single game. Instead, one could investigate the case where both players have two-game experience. For instance, match a player with {PD, CH}-experience against another with {SH, CH}-experience, and then compare their performance along several dimensions: in familiar games, in unfamiliar games, in the common CH game, and in the entire set. Another expansion to consider would be to use the same four games, but also consider three-game and the four-game settings.

Along these lines, one could also widen the set of available games beyond the four employed here. The individual players would have distinctive experience with different sets of games that may or may not include a common game. These models would allow us to see whether some games have a consistently larger impact on future play than others. Additionally, these models with expanded game sets would enable us to more thoroughly explore the issues of context, such as if the experience of playing a particular game, say the PD, depends on the context within which it is played.

Using agent based modeling allows the large number of possible combinations of sets of games discussed in these extensions to be reasonable managed. Ideally, one could initially explore these environments using agent based modeling and then use laboratory experiments and analytic techniques to delve deeper into those combinations that produce the most interesting results.

Table 4.1. Prisoner's Dilemma, Stag Hunt, Chicken, and Battle of the Sexes
for the natural outcome context.

For all four games, action 0 corresponds to the natural strategy
consistent with the Rapoport, Guyer, and Gordon (1976) taxonomy.

Prisoner's Dilemma (PD)
RGG #12

	0	1
0	***2, 2***	*4, 1*
1	*1, 4*	*3, 3*

Stag Hunt (SH)
RGG #61

	0	1
0	***4,4***	1, 3
1	3, 1	2, 2

Chicken (CH)
RGG #66

	0	1
0	***3, 3***	*2, 4*
1	*4, 2*	1, 1

Battle of the Sexes (BS)
RGG #69

	0	1
0	**2, 2**	*4,3*
1	*3, 4*	1, 1

Legend
Bold type indicates the natural outcome, the (0, 0) of each game[8]
Italic type indicates Pareto optimal outcomes
indicates a Nash equilibrium of the stage game

Table 4.2. Prisoner's Dilemma, Stag Hunt, and Chicken
for the cooperate/defect context.

For all three games, action C corresponds to the cooperative or other-regarding strategy
and action D corresponds to defecting or self-regarding strategy.

PD#

	C	D
C	*3, 3*	*1, 4*
D	*4, 1*	2, 2

SH#

	C	D
C	*4, 4*	1, 3
D	3, 1	2, 2

CH#

	C	D
C	*3, 3*	*2, 4*
D	*4, 2*	1, 1

Legend
Italic type indicates Pareto optimal outcomes
indicates a Nash equilibrium of the stage game

[8] See footnote 1.

Table 4.3. Games and experience specification of experiment trials.

Trial	Games Played	Row Player's Experience	Column Player's Experience
One-game v. one-game experience trials for the natural outcome context			
1	PD & SH	PD	SH
2	PD & CH	PD	CH
3	PD & BS	PD	BS
4	SH & CH	SH	CH
5	SH & BS	SH	BS
6	CH & BS	CH	BS
Two-game v. one-game experience trials for the natural outcome context			
7	PD & SH	PD & SH	PD
8	PD & SH	PD & SH	SH
9	PD & CH	PD & CH	PD
10	PD & CH	PD & CH	CH
11	PD & BS	PD & BS	PD
12	PD & BS	PD & BS	BS
13	SH & CH	SH & CH	SH
14	SH & CH	SH & CH	CH
15	SH & BS	SH & BS	SH
16	SH & BS	SH & BS	CH
17	CH & BS	CH & BS	CH
18	CH & BS	CH & BS	BS
Trials for the cooperate/defect context			
19	$PD^{\#}$ & $SH^{\#}$	$PD^{\#}$	$SH^{\#}$
20	$PD^{\#}$ & $CH^{\#}$	$PD^{\#}$	$CH^{\#}$
21	$PD^{\#}$ & $SH^{\#}$	$PD^{\#}$ & $SH^{\#}$	$PD^{\#}$
22	$PD^{\#}$ & $SH^{\#}$	$PD^{\#}$ & $SH^{\#}$	$SH^{\#}$
23	$PD^{\#}$ & $CH^{\#}$	$PD^{\#}$ & $CH^{\#}$	$PD^{\#}$
24	$PD^{\#}$ & $CH^{\#}$	$PD^{\#}$ & $CH^{\#}$	$CH^{\#}$

Legend: PD, $PD^{\#}$ = Prisoner's Dilemma
SH, $BS^{\#}$ = Stag Hunt
CH, $CH^{\#}$ = Chicken
BS = Battle of the Sexes

Table 4.4. Key simulation parameters common to all trials.[9]

Type	Parameter	Value	Explanation
General	runs	100	Independent runs per trial
	generations	1	No evolution of strategies
	popSize[10]	30	Automata in row and column populations
	states[10]	16	States per automaton
Game	rounds	160	Rounds per game
	noise	0	No misreporting of opponents' actions
	discount	1	Time discount factor
	matchType	mean	Each round, each row population automaton is matched against every column population automaton
Cost	costParA4	1	Accessible states in any game determine cost
	costParB4	2	Cost increases quadratically
	costParA0,A1,A2,A3	0	Cost 1, Cost 2, and Cost 3 are not used
	costAdjustment	0	No cost adjustment

[9] See Appendix to Chapter 2 for detailed definitions of these parameters.

[10] Same parameter values for row and column populations.

Table 4.5. Profits in settings where both players have experience in a single game for the natural outcome context.

Welch's t-statistics are for (Column profit – Row profit) ≠ 0.

Table 4.5.a.
Games played are Prisoner's Dilemma (PD) and Stag Hunt (SH)
(Trial 1)

In game:	Mean of PD & SH	
Player:	Row	Column
Experience:	PD	SH
Profit:	0.165	-0.411
t-statisitic:	-6.829	
p-value:	0.0000	

Table 4.5.b.
Games played are Prisoner's Dilemma (PD) and Chicken (CH)
(Trial 2)

In game:	Mean of PD & CH	
Player:	Row	Column
Experience:	PD	CH
Profit:	-0.102	-0.488
t-statisitic:	-4.961	
p-value:	0.0000	

Table 4.5.c.
Games played are Prisoner's Dilemma (PD) and Battle of the Sexes (BS)
(Trial 3)

In game:	Mean of PD & BS	
Player:	Row	Column
Experience:	PD	BS
Profit:	0.073	-0.785
t-statisitic:	-9.347	
p-value:	0.0000	

Table 4.5. Profits in settings where both players have experience in a single game for the natural outcome context (continued).

Welch's t-statistics are for (Column profit – Row profit) ≠ 0.

Table 4.5.d.
Games played are Stag Hunt (SH) and Chicken (CH)
(Trial 4)

In game:	Mean of SH & CH	
Player:	Row	Column
Experience:	SH	CH
Profit:	-0.341	0.136
t-statisitic:	4.05	
p-value:	0.0001	

Table 4.5.e.
Games played are Stag Hunt (SH) and Battle of the Sexes (BS)
(Trial 5)

In game:	Mean of SH & BS	
Player:	Row	Column
Experience:	SH	BS
Profit:	-0.216	-0.043
t-statisitic:	1.888	
p-value:	0.0606	

Table 4.5.f.
Games played are Chicken (CH) and Battle of the Sexes (BS)
(Trial 6)

In game:	Mean of CH & BS	
Player:	Row	Column
Experience:	CH	BS
Profit:	-0.297	-0.440
t-statisitic:	-1.409	
p-value:	0.1605	

Table 4.6. Cost in settings where both players have experience in a single game for the natural outcome context.

Welch's t-statistics are for (Column cost – Row cost) ≠ 0.

Table 4.6.a.
Games played are Prisoner's Dilemma (PD) and Stag Hunt (SH)
(Trial 1)

Cost in game:	Mean of PD & SH	
Player:	Row	Column
Experience:	PD	SH
Cost:	2.772	2.780
t-statisitic:	0.141	
p-value:	0.8883	

Table 4.6.b.
Games played are Prisoner's Dilemma (PD) and Chicken (CH)
(Trial 2)

Cost in game:	Mean of PD & CH	
Player:	Row	Column
Experience:	PD	CH
Cost:	2.787	2.757
t-statisitic:	-0.475	
p-value:	0.6357	

Table 4.6.c.
Games played are Prisoner's Dilemma (PD) and Battle of the Sexes (BS)
(Trial 3)

Cost in game:	Mean of PD & BS	
Player:	Row	Column
Experience:	PD	BS
Cost:	2.771	2.842
t-statisitic:	1.131	
p-value:	0.2595	

Table 4.6. Cost in two-game environments where both players have experience in a single game for the natural outcome context (continued).

Welch's t-statistics are for (Column cost – Row cost) ≠ 0.

Table 4.6.d.
Games played are Stag Hunt (SH) and Chicken (CH)
(Trial 4)

Cost in game:	Mean of SH & CH	
Player:	Row	Column
Experience:	SH	CH
Cost:	2.774	2.768
t-statisitic:	-0.090	
p-value:	0.9283	

Table 4.6.e.
Games played are Stag Hunt (SH) and Battle of the Sexes (BS)
(Trial 5)

Cost in game:	Mean of SH & BS	
Player:	Row	Column
Experience:	SH	BS
Cost:	2.778	2.833
t-statisitic:	0.909	
p-value:	0.3647	

Table 4.6.f.
Games played are Chicken (CH) and Battle of the Sexes (BS)
(Trial 6)

Cost in game:	Mean of CH & BS	
Player:	Row	Column
Experience:	CH	BS
Cost:	2.788	2.857
t-statisitic:	1.085	
p-value:	0.2794	

Table 4.7. Scores in settings where both players have experience in a single game for the natural outcome context.

Welch's t-statistics are for (Column score – Row score) ≠ 0.

Table 4.7.a.
Games played are Prisoner's Dilemma (PD) and Stag Hunt (SH)
(Trial 1)

In game:	PD		SH		Mean of PD & SH	
Player:	Row	Column	Row	Column	Row	Column
Experience:	PD	SH	PD	SH	PD	SH
Score:	2.651	1.684	3.222	3.053	2.937	2.369
t-statisitic:	-15.871		-1.422		-10.021	
p-value:	0.0000		0.1567		0.0000	

Table 4.7.b.
Games played are Prisoner's Dilemma (PD) and Chicken (CH)
(Trial 2)

In game:	PD		CH		Mean of PD & CH	
Player:	Row	Column	Row	Column	Row	Column
Experience:	PD	CH	PD	CH	PD	CH
Score:	2.973	1.528	2.397	3.010	2.685	2.269
t-statisitic:	-21.211		6.297		-9.611	
p-value:	0.0000		0.0000		0.0000	

Table 4.7.c.
Games played are Prisoner's Dilemma (PD) and Battle of the Sexes (BS)
(Trial 3)

In game:	PD		BS		Mean of PD & BS	
Player:	Row	Column	Row	Column	Row	Column
Experience:	PD	BS	PD	BS	PD	BS
Score:	2.931	1.545	2.758	2.570	2.845	2.058
t-statisitic:	-19.824		-2.486		-13.130	
p-value:	0.0000		0.0139		0.0000	

Table 4.7. Scores in settings where both players have experience in a single game for the natural outcome context (continued).

Welch's t-statistics are for (Column score – Row score) ≠ 0.

Table 4.7.d.
Games played are Stag Hunt (SH) and Chicken (CH)
(Trial 4)

In game:	SH		CH		Mean of SH & CH	
Player:	Row	Column	Row	Column	Row	Column
Experience:	SH	CH	SH	CH	SH	CH
Score:	2.553	3.126	2.312	2.682	2.433	2.904
t-statisitic:	4.807		3.277		4.675	
p-value:	0.0000		0.0013		0.0000	

Table 4.7.e.
Games played are Stag Hunt (SH) and Battle of the Sexes (BS)
(Trial 5)

In game:	SH		BS		Mean of SH & BS	
Player:	Row	Column	Row	Column	Row	Column
Experience:	SH	BS	SH	BS	SH	BS
Score:	2.581	3.138	2.543	2.443	2.562	2.790
t-statisitic:	4.613		-1.246		3.515	
p-value:	0.0000		0.2141		0.0005	

Table 4.7.f.
Games played are Chicken (CH) and Battle of the Sexes (BS)
(Trial 6)

In game:	CH		BS		Mean of CH & BS	
Player:	Row	Column	Row	Column	Row	Column
Experience:	CH	BS	CH	BS	CH	BS
Score:	2.521	2.339	2.460	2.494	2.491	2.417
t-statisitic:	-1.479		0.361		-0.903	
p-value:	0.1407		0.7182		0.3677	

Table 4.8. Action 0 usage frequencies in settings where both players have experience in a single game for the natural outcome context.

Welch's t-statistics are for (Column frequency – Row frequency) ≠ 0.

Table 4.8.a: PD experience v. SH experience in {PD, SH} two-game setting.

Action 0 usage frequency in game:	PD		SH	
Player:	Row	Column	Row	Column
Experience:	PD	SH	PD	SH
Frequency:	0.993	0.671	0.721	0.806
t-statisitic:	-11.793		1.864	
p-value:	0.0000		0.0640	

Table 4.8.b: PD experience v. CH experience in {PD, CH} two-game setting.

Action 0 usage frequency in game:	PD		CH	
Player:	Row	Column	Row	Column
Experience:	PD	CH	PD	CH
Frequency:	0.990	0.509	0.734	0.427
t-statisitic:	-15.784		-5.770	
p-value:	0.0000		0.0000	

Table 4.8.c: PD experience v. BS experience in {PD, BS} two-game setting.

Action 0 usage frequency in game:	PD		BS	
Player:	Row	Column	Row	Column
Experience:	PD	BS	PD	BS
Frequency:	0.993	0.531	0.729	0.542
t-statisitic:	-14.831		-3.594	
p-value:	0.0000		0.0005	

Table 4.8. Action 0 usage frequencies in settings where both players have experience in a single game for the natural outcome context (continued).

Welch's t-statistics are for (Column frequency – Row frequency) ≠ 0.

Table 4.8.d: SH experience v. CH experience in {PD, SH} two-game setting.

Action 0 usage frequency in game:	SH		CH	
Player:	Row	Column	Row	Column
Experience:	SH	CH	SH	CH
Frequency:	0.804	0.518	0.604	0.420
t-statisitic:	-5.921		-3.343	
p-value:	0.0000		0.0010	

Table 4.8.e: SH experience v. BS experience in {PD, CH} two-game setting.

Action 0 usage frequency in game:	SH		BS	
Player:	Row	Column	Row	Column
Experience:	SH	BS	SH	BS
Frequency:	0.802	0.524	0.631	0.531
t-statisitic:	-5.662		-1.850	
p-value:	0.0000		0.0662	

Table 4.8.f: CH experience v. BS experience in {PD, BS} two-game setting.

Action 0 usage frequency in game:	CH		BS	
Player:	Row	Column	Row	Column
Experience:	CH	BS	CH	BS
Frequency:	0.421	0.511	0.496	0.529
t-statisitic:	1.619		0.615	
p-value:	0.1073		0.5395	

Table 4.9. Frequencies of initial states that play action 0 in settings where both players have experience in a single game for the natural outcome context.

Welch's t-statistics are for (Column frequency – Row frequency) ≠ 0.

Table 4.9.a.
Games played are Prisoner's Dilemma (PD) and Stag Hunt (SH)
(Trial 1)

In game:	PD		SH		Mean of PD & SH	
Player:	Row	Column	Row	Column	Row	Column
Experience:	PD	SH	PD	SH	PD	SH
Initial Action 0 Freq.:	0.989	0.532	0.556	0.780	0.772	0.656
t-statisitic:	-25.950		5.096		-4.740	
p-value:	0.0000		0.0000		0.0000	

Table 4.9.b.
Games played are Prisoner's Dilemma (PD) and Chicken (CH)
(Trial 2)

In game:	PD		CH		Mean of PD & CH	
Player:	Row	Column	Row	Column	Row	Column
Experience:	PD	CH	PD	CH	PD	CH
Initial Action 0 Freq.:	0.989	0.496	0.569	0.430	0.779	0.463
t-statisitic:	-26.926		-2.702		-10.671	
p-value:	0.0000		0.0079		0.0000	

Table 4.9.c.
Games played are Prisoner's Dilemma (PD) and Battle of the Sexes (BS)
(Trial 3)

In game:	PD		BS		Mean of PD & BS	
Player:	Row	Column	Row	Column	Row	Column
Experience:	PD	BS	PD	BS	PD	BS
Initial Action 0 Freq.:	0.989	0.499	0.571	0.540	0.780	0.520
t-statisitic:	-25.971		-0.603		-8.631	
p-value:	0.0000		0.5476		0.0000	

Table 4.9. Frequencies of initial states that play action 0 in settings where both players have experience in a single game for the natural outcome context (continued).

Welch's t-statistics are for (Column frequency – Row frequency) ≠ 0.

Table 4.9.d.
Games played are Stag Hunt (SH) and Chicken (CH)
(Trial 4)

In game:	SH		CH		Mean of SH & CH	
Player:	Row	Column	Row	Column	Row	Column
Experience:	SH	CH	SH	CH	SH	CH
Initial Action 0 Freq.:	0.780	0.495	0.513	0.430	0.646	0.462
t-statisitic:	-6.448		-1.634		-5.010	
p-value:	0.0000		0.1051		0.0000	

Table 4.9.e.
Games played are Stag Hunt (SH) and Battle of the Sexes (BS)
(Trial 5)

In game:	SH		BS		Mean of SH & BS	
Player:	Row	Column	Row	Column	Row	Column
Experience:	SH	BS	SH	BS	SH	BS
Initial Action 0 Freq.:	0.780	0.497	0.521	0.540	0.650	0.519
t-statisitic:	-6.274		0.359		-3.563	
p-value:	0.0000		0.7201		0.0005	

Table 4.9.f.
Games played are Chicken (CH) and Battle of the Sexes (BS)
(Trial 6)

In game:	CH		BS		Mean of CH & BS	
Player:	Row	Column	Row	Column	Row	Column
Experience:	CH	BS	CH	BS	CH	BS
Initial Action 0 Freq.:	0.430	0.508	0.482	0.540	0.456	0.524
t-statisitic:	1.513		1.112		1.700	
p-value:	0.1328		0.2686		0.0907	

Table 4.10. Frequencies of accessible states that play action 0 in settings where both players have experience in a single game for the natural outcome context.

Welch's t-statistics are for (Column frequency – Row frequency) ≠ 0.

Table 4.10.a.
Games played are Prisoner's Dilemma (PD) and Stag Hunt (SH)
(Trial 1)

In game:	PD		SH		Mean of PD & SH	
Player:	Row	Column	Row	Column	Row	Column
Experience:	PD	SH	PD	SH	PD	SH
Accessible Action 0 Freq.:	0.989	0.564	0.621	0.802	0.805	0.683
t-statisitic:	-29.002		4.666		-5.242	
p-value:	0.0000		0.0000		0.0000	

Table 4.10.b.
Games played are Prisoner's Dilemma (PD) and Chicken (CH)
(Trial 2)

In game:	PD		CH		Mean of PD & CH	
Player:	Row	Column	Row	Column	Row	Column
Experience:	PD	CH	PD	CH	PD	CH
Accessible Action 0 Freq.:	0.989	0.507	0.621	0.425	0.805	0.466
t-statisitic:	-28.822		-4.076		-11.415	
p-value:	0.0000		0.0001		0.0000	

Table 4.10.c.
Games played are Prisoner's Dilemma (PD) and Battle of the Sexes (BS)
(Trial 3)

In game:	PD		BS		Mean of PD & BS	
Player:	Row	Column	Row	Column	Row	Column
Experience:	PD	BS	PD	BS	PD	BS
Accessible Action 0 Freq.:	0.989	0.500	0.625	0.525	0.807	0.512
t-statisitic:	-27.918		-2.066		-9.805	
p-value:	0.0000		0.0411		0.0000	

Table 4.10. Frequencies of accessible states that play action 0 in settings where both players have experience in a single game for the natural outcome context (continued).

Welch's t-statistics are for (Column frequency – Row frequency) ≠ 0.

Table 4.10.d.
Games played are Stag Hunt (SH) and Chicken (CH)
(Trial 4)

In game:	SH		CH		Mean of SH & CH	
Player:	Row	Column	Row	Column	Row	Column
Experience:	SH	CH	SH	CH	SH	CH
Accessible Action 0 Freq.:	0.802	0.506	0.566	0.425	0.684	0.465
t-statisitic:	-7.439		-2.936		-6.039	
p-value:	0.0000		0.0040		0.0000	

Table 4.10.e.
Games played are Stag Hunt (SH) and Battle of the Sexes (BS)
(Trial 5)

In game:	SH		BS		Mean of SH & BS	
Player:	Row	Column	Row	Column	Row	Column
Experience:	SH	BS	SH	BS	SH	BS
Accessible Action 0 Freq.:	0.802	0.499	0.564	0.525	0.683	0.512
t-statisitic:	-7.578		-0.811		-4.682	
p-value:	0.0000		0.4193		0.0000	

Table 4.10.f.
Games played are Chicken (CH) and Battle of the Sexes (BS)
(Trial 6)

In game:	CH		BS		Mean of CH & BS	
Player:	Row	Column	Row	Column	Row	Column
Experience:	CH	BS	CH	BS	CH	BS
Accessible Action 0 Freq.:	0.425	0.499	0.509	0.525	0.467	0.512
t-statisitic:	1.521		0.326		1.109	
p-value:	0.1309		0.7451		0.2688	

Table 4.11. Frequencies of states that reciprocate action 0 in settings where both players have experience in a single game for the natural outcome context.

Welch's t-statistics are for (Column frequency – Row frequency) ≠ 0.

Table 4.11.a: PD experience v. SH experience in {PD, SH} two-game setting.

0-reciprocating states in game:	PD		SH		Mean of PD & SH	
Player:	Row	Column	Row	Column	Row	Column
Experience:	PD	SH	PD	SH	PD	SH
Frequency:	0.996	0.604	0.644	0.813	0.820	0.708
t-statisitic:	-21.531		4.168		-4.270	
p-value:	0.0000		0.0001		0.0478	

Welch's t-statistics are for (Column Frequency – Row Frequency) ≠ 0

Table 4.11.b: PD experience v. CH experience in {PD, CH} two-game setting.

0-reciprocating states in game:	PD		CH		Mean of PD & CH	
Player:	Row	Column	Row	Column	Row	Column
Experience:	PD	CH	PD	CH	PD	CH
Frequency:	0.996	0.505	0.644	0.437	0.820	0.471
t-statisitic:	-23.541		-4.159		-10.670	
p-value:	0.0000		0.0001		0.0000	

Welch's t-statistics are for (Column Frequency – Row Frequency) ≠ 0

Table 4.11.c: PD experience v. BS experience in {PD, BS} two-game setting.

0-reciprocating states in game:	PD		BS		Mean of PD & BS	
Player:	Row	Column	Row	Column	Row	Column
Experience:	PD	BS	PD	BS	PD	BS
Frequency:	0.996	0.510	0.649	0.550	0.823	0.530
t-statisitic:	-22.377		-1.953		-8.899	
p-value:	0.0000		0.0531		0.0000	

Welch's t-statistics are for (Column Frequency – Row Frequency) ≠ 0

Table 4.11. Frequencies of states that reciprocate action 0 in settings where both players have experience in a single game for the natural outcome context (continued).

Welch's t-statistics are for (Column frequency – Row frequency) ≠ 0.

Table 4.11.d: SH experience v. CH experience in {SH, CH} two-game setting.

0-reciprocating states in game:	SH		CH		Mean of SH & CH	
Player:	Row	Column	Row	Column	Row	Column
Experience:	SH	CH	SH	CH	SH	CH
Frequency:	0.813	0.503	0.605	0.437	0.709	0.470
t-statisitic:	-7.233		-3.332		-5.918	
p-value:	0.0000		0.0011		0.0000	

Welch's t-statistics are for (Column Frequency – Row Frequency) ≠ 0

Table 4.11.e: SH experience v. BS experience in {SH, BS} two-game setting.

0-reciprocating states in game:	SH		BS		Mean of SH & BS	
Player:	Row	Column	Row	Column	Row	Column
Experience:	SH	BS	SH	BS	SH	BS
Frequency:	0.813	0.510	0.603	0.550	0.708	0.530
t-statisitic:	-7.028		-1.032		-4.391	
p-value:	0.0000		0.3039		0.0000	

Welch's t-statistics are for (Column Frequency – Row Frequency) ≠ 0

Table 4.11.f: CH experience v. BS experience in {CH, BS} two-game setting.

0-reciprocating states in game:	CH		BS		Mean of CH & BS	
Player:	Row	Column	Row	Column	Row	Column
Experience:	CH	BS	CH	BS	CH	BS
Frequency:	0.437	0.508	0.508	0.550	0.472	0.529
t-statisitic:	1.375		0.801		1.265	
p-value:	0.1715		0.4244		0.2073	

Welch's t-statistics are for (Column Frequency – Row Frequency) ≠ 0

Table 4.12. Outcome distributions in settings where both players have experience in a single game for the natural outcome context.

Table 4.12.a.
Two-game set: **{PD, SH}**

Outcomes in **PD** within {PD, SH}		Column's experience: **SH** 0	**1**
Row's experience: **PD**	**0**	66.6%	32.7%
	1	0.5%	0.2%

efficiency: 16.8%
entropy: 0.97

Outcomes in **SH** within {PD, SH}		Column's experience: **SH** 0	**1**
Row's experience: **PD**	**0**	56.9%	15.3%
	1	23.7%	4.2%

efficiency: 56.9%
entropy: 1.56

Table 4.12.b.
Two-game set: **{PD, CH}**

Outcomes in **PD** within {PD, CH}		Column's experience: **CH** 0	**1**
Row's experience: **PD**	**0**	50.3%	48.8%
	1	0.6%	0.4%

efficiency: 25.1%
entropy: 1.08

Outcomes in **CH** within {PD, CH}		Column's experience: **CH** 0	**1**
Row's experience: **PD**	**0**	30.9%	42.5%
	1	11.8%	14.8%

efficiency: 85.2%
entropy: 1.82

Table 4.12. Outcome distributions in settings where both players have experience in a single game for the natural outcome context (continued).

Table 4.12.c.
Two-game set: **{PD, BS}**

Outcomes in **PD** within {PD, BS}		Column's experience: **BS** 0	Column's experience: **BS** 1
Row's experience: **PD**	**0**	52.6%	46.7%
	1	0.5%	0.2%

efficiency: 23.8%
entropy: 1.06

Outcomes in **BS** within {PD, BS}		Column's experience: **BS** 0	Column's experience: **BS** 1
Row's experience: **PD**	**0**	37.8%	35.1%
	1	16.3%	10.7%

efficiency: 66.6%
entropy: 1.83

Table 4.12.d.
Two-game set: **{SH, CH}**

Outcomes in **SH** within {SH, CH}		Column's experience: **CH** 0	Column's experience: **CH** 1
Row's experience: **SH**	**0**	42.0%	38.4%
	1	9.8%	9.8%

efficiency: 42.0%
entropy: 1.71

Outcomes in **CH** within {SH, CH}		Column's experience: **CH** 0	Column's experience: **CH** 1
Row's experience: **SH**	**0**	27.6%	32.9%
	1	14.4%	25.1%

efficiency: 74.9%
entropy: 1.94

Table 4.12. Outcome distributions in settings where both players have experience in a single game for the natural outcome context (continued).

Table 4.12.e.
Two-game set: **{SH, BS}**

Outcomes in **SH** within {SH, BS}

Row's experience: **SH**	Column's experience: **BS** **0**	**1**
0	43.0%	37.3%
1	9.4%	10.3%

efficiency: 43.0%
entropy: 1.71

Outcomes in **BS** within {SH, BS}

Row's experience: **SH**	Column's experience: **BS** **0**	**1**
0	35.3%	27.8%
1	17.8%	19.1%

efficiency: 59.7%
entropy: 1.94

Table 4.12.f.
Two-game set: **{CH, BS}**

Outcomes in **CH** within {CH, BS}

Row's experience: **CH**	Column's experience: **BS** **0**	**1**
0	21.7%	20.4%
1	29.5%	28.5%

efficiency: 71.5%
entropy: 1.98

Outcomes in **BS** within {CH, BS}

Row's experience: **CH**	Column's experience: **BS** **0**	**1**
0	27.1%	22.4%
1	25.8%	24.6%

efficiency: 59.1%
entropy: 2.00

Table 4.13. Frequencies of states that play Tit-for-Tat in settings where both players have experience in a single game for the natural outcome context.

Welch's t-statistics are for (Column frequency – Row frequency) ≠ 0.

Table 4.13.a.
Games played are Prisoner's Dilemma (PD) and Stag Hunt (SH)
(Trial 1)

In game:	PD		SH		TFT frequency difference (Inexp. - Exper.)	
Player:	Row	Column	Row	Column		
Experience:	PD	SH	PD	SH		
Tit-for-Tat frequency:	0.012	0.218	0.169	0.037	in PD	in SH
t-statisitic:	15.237		-8.393		+0.206	+0.132
p-value:	0.0000		0.0000			

Table 4.13.b.
Games played are Prisoner's Dilemma (PD) and Chicken (CH)
(Trial 2)

In game:	PD		CH		TFT frequency difference (Inexp. - Exper.)	
Player:	Row	Column	Row	Column		
Experience:	PD	CH	PD	CH		
Tit-for-Tat frequency:	0.012	0.205	0.170	0.059	in PD	in CH
t-statisitic:	13.460		-5.414		+0.193	+0.111
p-value:	0.0000		0.0000			

Table 4.13.c.
Games played are Prisoner's Dilemma (PD) and Battle of the Sexes (BS)
(Trial 3)

In game:	PD		BS		TFT frequency difference (Inexp. - Exper.)	
Player:	Row	Column	Row	Column		
Experience:	PD	BS	PD	BS		
Tit-for-Tat frequency:	0.012	0.199	0.169	0.046	in PD	in BS
t-statisitic:	13.260		-6.612		+0.188	+0.123
p-value:	0.0000		0.0000			

Table 4.13. Frequencies of states that play Tit-for-Tat in settings where both players have experience in a single game for the natural outcome context (continued).

Welch's t-statistics are for (Column frequency – Row frequency) ≠ 0.

Table 4.13.d.
Games played are Stag Hunt (SH) and Chicken (CH)
(Trial 4)

In game:	SH		CH		TFT frequency difference (Inexp. - Exper.)	
Player:	Row	Column	Row	Column		
Experience:	SH	CH	SH	CH		
Tit-for-Tat frequency:	0.037	0.203	0.217	0.059	in SH	in CH
t-statisitic:	9.471		-7.305		+0.166	+0.157
p-value:	0.0000		0.0000			

Table 4.13.e.
Games played are Stag Hunt (SH) and Battle of the Sexes (BS)
(Trial 5)

In game:	SH		BS		TFT frequency difference (Inexp. - Exper.)	
Player:	Row	Column	Row	Column		
Experience:	SH	BS	SH	BS		
Tit-for-Tat frequency:	0.037	0.198	0.217	0.046	in SH	in BS
t-statisitic:	9.400		-8.785		+0.161	+0.171
p-value:	0.0000		0.0000			

Table 4.13.f.
Games played are Chicken (CH) and Battle of the Sexes (BS)
(Trial 6)

In game:	CH		BS		TFT frequency difference (Inexp. - Exper.)	
Player:	Row	Column	Row	Column		
Experience:	CH	BS	CH	BS		
Tit-for-Tat frequency:	0.059	0.204	0.206	0.046	in CH	in BS
t-statisitic:	6.581		-7.947		+0.145	+0.159
p-value:	0.0000		0.0000			

Table 4.14. Profits in settings when row player has {PD, SH}-experience & column player has either PD-experience or SH-experience for the natural outcome context.

t-statistics are for (Column profit – Row profit) ≠ 0 or (Pop. 2 profit – Pop. 1 profit) ≠ 0

Table 4.14.a: {PD, SH} experience v. {PD} experience in {PD, SH} two-game setting.

Games:	Both PD & SH	
Player:	Row	Column
Experience:	PD&SH	PD
Profit:	1.313	0.057
t-statisitic:	-22.347	
p-value:	0.0469	

Table 4.14.b: {PD, SH} experience v. {SH} experience in {PD, SH} two-game setting.

Games:	Both PD & SH	
Player:	Row	Column
Experience:	PD&SH	SH
Profit:	1.770	-0.067
t-statisitic:	-24.979	
p-value:	0.0000	

Table 4.14.c: Differences in Profit between {PD} experience and {SH} experience when facing the same {PD, SH} experience opponent.

Games:	Both PD & SH
Pop. 1 Experience (opponent's experience):	PD (PD&SH)
Pop. 2 Experience (opponent's experience):	SH (PD&SH)
Population 2 profit - Population 1 profit:	-0.124
t-statisitic:	-1.841
p-value:	0.0671

Table 4.14.d: Differences in Profit between {PD, SH} experience when opponent. has {PD} experience and when opponent has {SH} experience.

Games:	Mean of PD & SH
Pop. 1 Experience (opponent's experience):	PD&SH (v PD)
Pop. 2 Experience (opponent's experience):	PD&SH (v SH)
Population 2 profit - Population 1 profit:	0.457
t-statisitic:	7.174
p-value:	0.0000

Table 4.15. Profits in settings when row population has {PD, CH}-experience & column population has either PD-experience or CH-experience for the natural outcome context.

t-statistics are for (Column profit – Row profit) ≠ 0 or (Pop. 2 profit – Pop. 1 profit) ≠ 0

Table 4.15.a: {PD, CH} experience v. {PD} experience in {PD, CH} two-game setting.

Games:	Both PD & CH	
Player:	Row	Column
Experience:	PD&CH	PD
Profit:	1.124	-0.234
t-statisitic:	-24.006	
p-value:	0.0000	

Table 4.15.b: {PD, CH} experience v. {CH} experience in {PD, CH} two-game setting.

Games:	Both PD & CH	
Player:	Row	Column
Experience:	PD&CH	CH
Profit:	1.323	-0.422
t-statisitic:	-20.893	
p-value:	0.0000	

Table 4.15.c: Differences in Profit between {PD} experience and {CH} experience when facing the same {PD, CH} experience opponent.

Games:	Both PD & CH
Pop. 1 Experience (opponent's experience):	PD (PD&CH)
Pop. 2 Experience (opponent's experience):	CH (PD&CH)
Population 2 profit - Population 1 profit:	-0.188
t-statisitic:	-2.175
p-value:	0.0310

Table 4.15.d: Differences in Profit between {PD, CH} experience when opponent has {PD} experience and when opponent has{CH} experience.

Games:	Both PD & CH
Pop. 1 Experience (opponent's experience):	PD&CH (v PD)
Pop. 2 Experience (opponent's experience):	PD&CH (v CH)
Population 2 profit - Population 1 profit:	0.199
t-statisitic:	3.839
p-value:	0.0002

Table 4.16. Profits in settings when row population has {PD, BS}-experience & column population has either PD-experience or BS-experience for the natural outcome context.

t-statistics are for (Column profit – Row profit) ≠ 0 or (Pop. 2 profit – Pop. 1 profit) ≠ 0

Table 4.16.a: {PD, BS} experience v. {PD} experience in {PD, BS} two-game setting.

Games:	Both PD & BS	
Player:	Row	Column
Experience:	PD&BS	PD
Profit:	0.920	-0.351
t-statisitic:	-19.168	
p-value:	0.0000	

Table 4.16.b: {PD, BS} experience v. {BS} experience in {PD, BS} two-game setting.

Games:	Both PD & BS	
Player:	Row	Column
Experience:	PD&BS	BS
Profit:	1.444	-0.651
t-statisitic:	-21.289	
p-value:	0.0000	

Table 4.16.c: Differences in Profit between {PD} experience and {BS} experience when facing the same {PD, BS} experience opponent.

Games:	Both PD & BS
Pop. 1 Experience (opponent's experience):	PD (PD&BS)
Pop. 2 Experience (opponent's experience):	BS (PD&BS)
Population 2 profit - Population 1 profit:	-0.300
t-statisitic:	-3.236
p-value:	0.0014

Table 4.16.d: Differences in Profit between {PD, BS} experience when opponent has {PD} experience and when opponent has{BS} experience.

Games:	Both PD & BS
Pop. 1 Experience (opponent's experience):	PD&BS (v PD)
Pop. 2 Experience (opponent's experience):	PD&BS (v BS)
Population 2 profit - Population 1 profit:	0.524
t-statisitic:	7.075
p-value:	0.0000

Table 4.17. Profits in settings when row population has {SH, CH}-experience & column population has either SH-experience or CH-experience for the natural outcome context.

t-statistics are for (Column profit – Row profit) ≠ 0 or (Pop. 2 profit – Pop. 1 profit) ≠ 0

Table 4.17.a: {SH, CH} experience v. {SH} experience in {SH, CH} two-game setting.

Games:	Both SH & CH	
Player:	Row	Column
Experience:	SH&CH	SH
Profit:	1.678	-0.101
t-statisitic:	-15.269	
p-value:	0.0470	

Table 4.17.b: {SH, CH} experience v. {CH} experience in {SH, CH} two-game setting.

Games:	Both SH & CH	
Player:	Row	Column
Experience:	SH&CH	CH
Profit:	1.144	-0.163
t-statisitic:	-11.103	
p-value:	0.0000	

Table 4.17.c: Differences in Profit between {SH} experience and {CH} experience when facing the same {SH, CH} experience opponent.

Games:	Both SH & CH
Pop. 1 Experience (opponent's experience):	SH (SH&CH)
Pop. 2 Experience (opponent's experience):	CH (SH&CH)
Population 2 profit - Population 1 profit:	-0.062
t-statisitic:	-0.472
p-value:	0.6372

Table 4.17.d: Differences in Profit between {SH, CH} experience when opponent has {SH} experience and when opponent has{CH} experience.

Games:	Both SH & CH
Pop. 1 Experience (opponent's experience):	SH&CH (v SH)
Pop. 2 Experience (opponent's experience):	SH&CH (v CH)
Population 2 profit - Population 1 profit:	-0.535
t-statisitic:	-5.298
p-value:	0.0000

Table 4.18. Profits in settings when row population has {SH, BS}-experience & column population has either SH-experience or BS-experience for the natural outcome context.

t-statistics are for (Column profit – Row profit) ≠ 0 or (Pop. 2 profit – Pop. 1 profit) ≠ 0

Table 4.18.a: {SH, BS} experience v. {SH} experience in {SH, BS} two-game setting.

Games:	Both SH & BS	
Player:	Row	Column
Experience:	SH&BS	SH
Profit:	1.524	0.103
t-statisitic:	-16.306	
p-value:	0.0000	

Table 4.18.b: {SH, BS} experience v. {BS} experience in {SH, BS} two-game setting.

Games:	Both SH & BS	
Player:	Row	Column
Experience:	SH&BS	BS
Profit:	1.269	-0.027
t-statisitic:	-13.797	
p-value:	0.0000	

Table 4.18.c: Differences in Profit between {SH} experience and {BS} experience when facing the same {SH, BS} experience opponent.

Games:	Both SH & BS
Pop. 1 Experience (opponent's experience):	SH (SH&BS)
Pop. 2 Experience (opponent's experience):	BS (SH&BS)
Population 2 profit - Population 1 profit:	-0.130
t-statisitic:	-1.249
p-value:	0.2133

Table 4.18.d: Differences in Profit between {SH, BS} experience when opponent has {SH} experience and when opponent has{BS} experience.

Games:	Both SH & BS
Pop. 1 Experience (opponent's experience):	SH&BS (v SH)
Pop. 2 Experience (opponent's experience):	SH&BS (v BS)
Population 2 profit - Population 1 profit:	-0.255
t-statisitic:	-3.409
p-value:	0.0008

Table 4.19. Profits in settings when row population has {CH, BS}-experience & column population has either CH-experience or BS-experience for the natural outcome context.

t-statistics are for (Column profit – Row profit) ≠ 0 or (Pop. 2 profit – Pop. 1 profit) ≠ 0

Table 4.19.a: {CH, BS} experience v. {CH} experience in {CH, BS} two-game setting.

Games:	Both CH & BS	
Player:	Row	Column
Experience:	CH&BS	CH
Profit:	1.096	-0.331
t-statisitic:	-12.542	
p-value:	0.0000	

Table 4.19.b: {CH, BS} experience v. {BS} experience in {CH, BS} two-game setting.

Games:	Both CH & BS	
Player:	Row	Column
Experience:	CH&BS	BS
Profit:	1.214	-0.393
t-statisitic:	-14.113	
p-value:	0.0000	

Table 4.19.c: Differences in Profit between {CH} experience and {BS} experience when facing the same {CH, BS} experience opponent.

Games:	Both CH & BS
Pop. 1 Experience (opponent's experience):	CH (CH&BS)
Pop. 2 Experience (opponent's experience):	BS (CH&BS)
Population 2 profit - Population 1 profit:	-0.062
t-statisitic:	-0.506
p-value:	0.6136

Table 4.19.d: Differences in Profit between {CH, BS} experience when opponent has {CH} experience and when opponent has{BS} experience.

Games:	Both CH & BS
Pop. 1 Experience (opponent's experience):	CH&BS (v CH)
Pop. 2 Experience (opponent's experience):	CH&BS (v BS)
Population 2 profit - Population 1 profit:	0.117
t-statisitic:	1.124
p-value:	0.2625

Table 4.20. Outcome distributions in settings where the row player has {PD, SH}-experience & the column player has either PD- or SH-experience for the natural outcome context.

Table 4.20.a.
Two-game set: **{PD, SH}**
Experience: **{PD, SH}** v **{PD}**

Outcomes in **PD** within {PD, SH}		Column's experience: **PD** 0	1
Row's experience: **{PD, SH}**	**0**	99.4%	0.4%
	1	0.3%	0.0%

efficiency: 0.3%
entropy: 0.06

Outcomes in **SH** within {PD, SH}		Column's experience: **PD** 0	1
Row's experience: **{PD, SH}**	**0**	71.0%	27.6%
	1	0.8%	0.6%

efficiency: 71.0%
entropy: 0.96

Table 4.20.b.
Two-game set: **{PD, SH}**
Experience: **{PD, SH}** v **{SH}**

Outcomes in **PD** within {PD, SH}		Column's experience: **SH** 0	1
Row's experience: **{PD, SH}**	**0**	66.0%	33.0%
	1	0.7%	0.3%

efficiency: 17.1%
entropy: 1.00

Outcomes in **SH** within {PD, SH}		Column's experience: **SH** 0	1
Row's experience: **{PD, SH}**	**0**	80.2%	19.1%
	1	0.2%	0.5%

efficiency: 80.2%
entropy: 0.77

Table 4.21. Scores in settings when row population has {PD, SH}-experience & column population has either PD-experience or SH-experience for the natural outcome context.

t-statistics are for (Column profit – Row profit) ≠ 0 or (Pop. 2 profit – Pop. 1 profit) ≠ 0

Table 4.21.a: {PD, SH} experience v. {PD} experience in {PD, SH} two-game setting.

Score in game:	PD		SH		Mean of PD & SH	
Player:	Row	Column	Row	Column	Row	Column
Experience:	PD&SH	PD	PD&SH	PD	PD&SH	PD
Score:	2.004	2.002	3.152	3.689	2.578	2.846
t-statisitic:	-0.818		6.768		6.760	
p-value:	0.4142		0.0000		0.0477	

Table 4.21.b: {PD, SH} experience v. {SH} experience in {PD, SH} two-game setting.

Score in game:	PD		SH		Mean of PD & SH	
Player:	Row	Column	Row	Column	Row	Column
Experience:	PD&SH	SH	PD&SH	SH	PD&SH	SH
Score:	2.656	1.687	3.415	3.792	3.036	2.740
t-statisitic:	-15.808		3.059		-5.063	
p-value:	0.0000		0.0027		0.0000	

Table 4.21.c: Differences in Score between {PD} experience and {SH} experience when facing the same {PD, SH} experience opponent.

Game played:	PD	SH	Mean of PD&SH
Pop. 1 Experience (opponent's experience):	PD (PD&SH)	PD (PD&SH)	PD (PD&SH)
Pop. 2 Experience (opponent's experience):	SH (PD&SH)	SH (PD&SH)	SH (PD&SH)
t-statisitic:	-11.080	2.129	-3.318
p-value:	0.0000	0.0347	0.0012

Table 4.21.d: Differences in Score between {PD, SH} experience when opponent has {PD} experience and when opponent has{SH} experience.

Game played:	PD	SH	Mean of PD&SH
Pop. 1 Experience (opponent's experience):	PD&SH (v PD)	PD&SH (v PD)	PD&SH (v PD)
Pop. 2 Experience (opponent's experience):	PD&SH (v SH)	PD&SH (v SH)	PD&SH (v SH)
t-statisitic:	11.984	1.902	7.266
p-value:	0.0000	0.0589	0.0000

Table 4.22. Profits in settings where both players have experience in a single game for the cooperate / defect context.

Welch's t-statistics are for (Column score – Row score) ≠ 0.

Table 4.22.a. Profit when games played are $PD^{\#}$ and $SH^{\#}$.

In game:	Jointly in $PD^{\#}$ & $SH^{\#}$	
Player:	Row	Column
Experience:	$PD^{\#}$	$SH^{\#}$
Profit:	0.302	-1.114
t-statisitic:	-17.312	
p-value:	0.0000	

Table 4.22.b. Profit when games played are $PD^{\#}$ and $CH^{\#}$.

In game:	Jointly in $PD^{\#}$ & $CH^{\#}$	
Player:	Row	Column
Experience:	$PD^{\#}$	$CH^{\#}$
Profit:	-0.130	-0.998
t-statisitic:	-8.490	
p-value:	0.0000	

Table 4.23. Scores in settings where both players have experience in a single game for the cooperate / defect context.

Welch's t-statistics are for (Column score – Row score) ≠ 0.

Table 4.23.a. Score in {$PD^{\#}$, $SH^{\#}$}.

In game:	$PD^{\#}$		$SH^{\#}$		Mean of $PD^{\#}$ & $SH^{\#}$	
Player:	Row	Column	Row	Column	Row	Column
Experience:	$PD^{\#}$	$SH^{\#}$	$PD^{\#}$	$SH^{\#}$	$PD^{\#}$	$SH^{\#}$
Score:	3.162	1.427	2.993	1.940	3.078	1.684
t-statisitic:	-30.112		-12.204		-24.655	
p-value:	0.0000		0.0000		0.0470	

Table 4.23.b. Score in {$PD^{\#}$, $CH^{\#}$}.

In game:	$PD^{\#}$		$CH^{\#}$		Mean of $PD^{\#}$ & $CH^{\#}$	
Player:	Row	Column	Row	Column	Row	Column
Experience:	$PD^{\#}$	$CH^{\#}$	$PD^{\#}$	$CH^{\#}$	$PD^{\#}$	$CH^{\#}$
Score:	2.967	1.528	2.286	2.052	2.627	1.790
t-statisitic:	-23.717		-1.793		-10.184	
p-value:	0.0000		0.0750		0.0000	

Table 4.24. Profits in settings where row player has {$PD^{\#}$, $SH^{\#}$}-experience and column player has either $PD^{\#}$-experience or $SH^{\#}$-experience for the cooperate/defect context

t-statistics are for (Column profit – Row profit) ≠ 0 or (Pop. 2 profit – Pop. 1 profit) ≠ 0

Table 4.24.a: {$PD^{\#}$, $SH^{\#}$} experience v. {$PD^{\#}$} experience in {$PD^{\#}$, $SH^{\#}$} two-game setting.

Games:	Both $PD^{\#}$ & $SH^{\#}$	
Player:	Row	Column
Experience:	$PD^{\#}$&$SH^{\#}$	$PD^{\#}$
Profit:	0.795	-0.670
t-statisitic:	-21.806	
p-value:	0.0475	

Table 4.24.b: {$PD^{\#}$, $SH^{\#}$} experience v. {$SH^{\#}$} experience in {$PD^{\#}$, $SH^{\#}$} two-game setting.

Games:	Both $PD^{\#}$ & $SH^{\#}$	
Player:	Row	Column
Experience:	$PD^{\#}$&$SH^{\#}$	$SH^{\#}$
Profit:	1.799	-0.945
t-statisitic:	-25.624	
p-value:	0.0000	

Table 4.24.c: Differences in Profit between {$PD^{\#}$} experience and {$SH^{\#}$} experience when facing the same {$PD^{\#}$, $SH^{\#}$} experience opponent.

Games:	Both $PD^{\#}$ & $SH^{\#}$
Pop. 1 Experience (opponent's experience):	$PD^{\#}$ ($PD^{\#}$&$SH^{\#}$)
Pop. 2 Experience (opponent's experience):	$SH^{\#}$ ($PD^{\#}$&$SH^{\#}$)
t-statisitic:	-2.403
p-value:	0.0173

Table 4.24.d: Differences in Profit between {$PD^{\#}$, $SH^{\#}$} experience when opponent has {$PD^{\#}$} experience and when opponent has {$SH^{\#}$} experience.

Games:	Mean of $PD^{\#}$ & $SH^{\#}$
Pop. 1 Experience (opponent's experience):	$PD^{\#}$&$SH^{\#}$ (v $PD^{\#}$)
Pop. 2 Experience (opponent's experience):	18.506
t-statisitic:	0.0000

Table 4.25. Profits in settings where row player has {PD$^{\#}$, CH$^{\#}$}-experience and column player has either PD$^{\#}$-experience or CH$^{\#}$-experience for the cooperate/defect context

t-statistics are for (Column profit – Row profit) ≠ 0 or (Pop. 2 profit – Pop. 1 profit) ≠ 0

Table 4.25.a: {PD$^{\#}$, CH$^{\#}$} experience v. {PD$^{\#}$} experience in {PD$^{\#}$, CH$^{\#}$} two-game setting.

Games:	Both PD$^{\#}$ & CH$^{\#}$	
Player:	Row	Column
Experience:	PD$^{\#}$&CH$^{\#}$	PD$^{\#}$
Profit:	0.627	-0.603
t-statisitic:	-14.670	
p-value:	0.0000	

Table 4.25.b: {PD$^{\#}$, CH$^{\#}$} experience v. {CH$^{\#}$} experience in {PD$^{\#}$, CH$^{\#}$} two-game setting.

Games:	Both PD$^{\#}$ & CH$^{\#}$	
Player:	Row	Column
Experience:	PD$^{\#}$&SH$^{\#}$	CH$^{\#}$
Profit:	1.247	-0.753
t-statisitic:	-17.668	
p-value:	0.0000	

Table 4.25.c: Differences in Profit between {PD$^{\#}$} experience and {CH$^{\#}$} experience when facing the same {PD$^{\#}$, CH$^{\#}$} experience opponent.

Games:	Both PD$^{\#}$ & CH$^{\#}$
Pop. 1 Experience (opponent's experience):	PD$^{\#}$ (PD$^{\#}$&CH$^{\#}$)
Pop. 2 Experience (opponent's experience):	CH$^{\#}$ (PD$^{\#}$&CH$^{\#}$)
t-statisitic:	-1.351
p-value:	0.1784

Table 4.25.d: Differences in Profit between {PD$^{\#}$, CH$^{\#}$} experience when opponent has {PD$^{\#}$} experience and when opponent has {CH$^{\#}$} experience.

Games:	Mean of PD$^{\#}$ & CH$^{\#}$
Pop. 1 Experience (opponent's experience):	PD$^{\#}$&CH$^{\#}$ (v PD$^{\#}$)
Pop. 2 Experience (opponent's experience):	7.139
t-statisitic:	0.0000

Table 4.26. Scores in settings where row player has {$PD^{\#}$, $SH^{\#}$}-experience and column player has either $PD^{\#}$-experience or $SH^{\#}$-experience for the cooperate/defect context.

t-statistics are for (Column profit – Row profit) ≠ 0 or (Pop. 2 profit – Pop. 1 profit) ≠ 0

.

Score in game:	$PD^{\#}$		$SH^{\#}$		$PD^{\#}$&$SH^{\#}$ mean	
Player:	Row	Column	Row	Column	Row	Column
Experience:	$PD^{\#}$&$SH^{\#}$	$PD^{\#}$	$PD^{\#}$&$SH^{\#}$	$PD^{\#}$	$PD^{\#}$&$SH^{\#}$	$PD^{\#}$
Score:	1.902	2.204	2.249	1.982	2.075	2.093
t-statisitic:	5.325		-4.171		0.331	
p-value:	0.0000		0.0000		0.0474	

Table 4.26.b: {$PD^{\#}$, $SH^{\#}$} experience v. {$SH^{\#}$} experience in {$PD^{\#}$, $SH^{\#}$} two-game setting.

Score in game:	$PD^{\#}$		$SH^{\#}$		$PD^{\#}$&$SH^{\#}$ mean	
Player:	Row	Column	Row	Column	Row	Column
Experience:	$PD^{\#}$&$SH^{\#}$	$SH^{\#}$	$PD^{\#}$&$SH^{\#}$	$SH^{\#}$	$PD^{\#}$&$SH^{\#}$	$SH^{\#}$
Score:	3.096	1.662	3.063	2.029	3.080	1.846
t-statisitic:	-18.870		-7.222		-13.223	
p-value:	0.0000		0.0000		0.0000	

Table 4.26.c: Differences in Score between {$PD^{\#}$} experience and {$SH^{\#}$} experience when facing the same {$PD^{\#}$, $SH^{\#}$} experience opponent.

Game played:	$PD^{\#}$	$SH^{\#}$	$PD^{\#}$&$SH^{\#}$ mean
Pop. 1 Experience (opponent's experience):	$PD^{\#}$ ($PD^{\#}$&$SH^{\#}$)	$PD^{\#}$ ($PD^{\#}$&$SH^{\#}$)	$PD^{\#}$ ($PD^{\#}$&$SH^{\#}$)
Pop. 2 Experience (opponent's experience):	$SH^{\#}$ ($PD^{\#}$&$SH^{\#}$)	$SH^{\#}$ ($PD^{\#}$&$SH^{\#}$)	$SH^{\#}$ ($PD^{\#}$&$SH^{\#}$)
t-statisitic:	-7.105	0.346	-2.635
p-value:	0.0000	0.7296	0.0092

Table 4.26.d: Differences in Score between {PD#, SH#} experience when opponent has {$PD^{\#}$} experience and when opponent has{$SH^{\#}$} experience.

Game played:	$PD^{\#}$	$SH^{\#}$	$PD^{\#}$&$SH^{\#}$ mean
Pop. 1 Experience (opponent's experience):	$PD^{\#}$&$SH^{\#}$ (v $PD^{\#}$)	$PD^{\#}$&$SH^{\#}$ (v $PD^{\#}$)	$PD^{\#}$&$SH^{\#}$ (v $PD^{\#}$)
Pop. 2 Experience (opponent's experience):	$PD^{\#}$&$SH^{\#}$ (v $SH^{\#}$)	$PD^{\#}$&$SH^{\#}$ (v $SH^{\#}$)	$PD^{\#}$&$SH^{\#}$ (v $SH^{\#}$)
t-statisitic:	21.182	10.494	19.032
p-value:	0.0000	0.0000	0.0000

Table 4.27. Scores in settings where row player has {PD$^{\#}$, CH$^{\#}$}-experience and column player has either PD$^{\#}$-experience or CH$^{\#}$-experience for the cooperate/defect context.

t-statistics are for (Column profit – Row profit) ≠ 0 or (Pop. 2 profit – Pop. 1 profit) ≠ 0

Table 4.27.a: {PD$^{\#}$, CH$^{\#}$} experience v. {PD$^{\#}$} experience in {PD$^{\#}$, CH$^{\#}$} two-game setting.

Score in game:	PD$^{\#}$		CH$^{\#}$		PD$^{\#}$&CH$^{\#}$ mean	
Player:	Row	Column	Row	Column	Row	Column
Experience:	PD$^{\#}$&CH$^{\#}$	PD$^{\#}$	PD$^{\#}$&CH$^{\#}$	PD$^{\#}$	PD$^{\#}$&CH$^{\#}$	PD$^{\#}$
Score:	1.873	2.262	1.966	2.075	1.919	2.168
t-statisitic:	6.186		0.899		3.301	
p-value:	0.0000		0.3698		0.0012	

Table 4.27.b: {PD$^{\#}$, CH$^{\#}$} experience v. {CH$^{\#}$} experience in {PD$^{\#}$, CH$^{\#}$} two-game setting.

Score in game:	PD$^{\#}$		CH$^{\#}$		PD$^{\#}$&CH$^{\#}$ mean	
Player:	Row	Column	Row	Column	Row	Column
Experience:	PD$^{\#}$&CH$^{\#}$	CH$^{\#}$	PD$^{\#}$&CH$^{\#}$	CH$^{\#}$	PD$^{\#}$&CH$^{\#}$	CH$^{\#}$
Score:	2.791	1.833	2.289	2.185	2.540	2.009
t-statisitic:	-10.946		-0.652		-5.092	
p-value:	0.0000		0.5150		0.0000	

Table 4.27.c: Differences in Score between {PD$^{\#}$} experience and {CH$^{\#}$} experience when facing the same {PD$^{\#}$, CH$^{\#}$} experience opponent.

Game played:	PD$^{\#}$	CH$^{\#}$	PD$^{\#}$&CH$^{\#}$ mean
Pop. 1 Experience (opponent's experience):	PD$^{\#}$ (PD$^{\#}$&CH$^{\#}$)	PD$^{\#}$ (PD$^{\#}$&CH$^{\#}$)	PD$^{\#}$ (PD$^{\#}$&CH$^{\#}$)
Pop. 2 Experience (opponent's experience):	CH$^{\#}$ (PD$^{\#}$&CH$^{\#}$)	CH$^{\#}$ (PD$^{\#}$&CH$^{\#}$)	CH$^{\#}$ (PD$^{\#}$&CH$^{\#}$)
t-statisitic:	-5.086	0.746	-1.623
p-value:	0.0000	0.4566	0.1061

Table 4.27.d: Differences in Score between {PD$^{\#}$, CH$^{\#}$} experience when opponent has {PD$^{\#}$} experience and when opponent has{CH$^{\#}$} experience.

Game played:	PD$^{\#}$	CH$^{\#}$	PD$^{\#}$&CH$^{\#}$ mean
Pop. 1 Experience (opponent's experience):	PD$^{\#}$&CH$^{\#}$ (v PD$^{\#}$)	PD$^{\#}$&CH$^{\#}$ (v PD$^{\#}$)	PD$^{\#}$&CH$^{\#}$ (v PD$^{\#}$)
Pop. 2 Experience (opponent's experience):	PD$^{\#}$&CH$^{\#}$ (v CH$^{\#}$)	PD$^{\#}$&CH$^{\#}$ (v CH$^{\#}$)	PD$^{\#}$&CH$^{\#}$ (v CH$^{\#}$)
t-statisitic:	13.689	2.401	7.462
p-value:	0.0000	0.0175	0.0000

Table 4.28. Outcome distributions in settings where both players have experience in a single game in the cooperate/defect context.

Table 4.28.a.
Two-game set: **{PD#, SH#}**

Outcomes in **PD#** within {PD#, SH#}

Row's experience: **PD#**		Column's experience: **SH#** **C**	**D**
	C	0.4%	0.1%
	D	58.0%	41.5%

efficiency: 29.5%
entropy: 1.03

Outcomes in **SH#** within {PD#, SH#}

Row's experience: **PD#**		Column's experience: **SH#** **C**	**D**
	C	23.3%	4.5%
	D	57.2%	15.0%

efficiency: 23.3%
entropy: 1.56

Table 4.28.b.
Two-game set: **{PD#, CH#}**

Outcomes in **PD#** within {PD#, CH#}

Row's experience: **PD#**		Column's experience: **CH#** **C**	**D**
	C	0.6%	0.1%
	D	48.1%	51.1%

efficiency: 24.8%
entropy: 1.06

Outcomes in **CH#** within {PD#, CH#}

Row's experience: **PD#**		Column's experience: **CH#** **C**	**D**
	C	11.0%	17.8%
	D	29.6%	41.6%

efficiency: 58.4%
entropy: 1.84

Table 4.29. Outcome distributions in settings where the row player has {PD#, SH#}-experience & the column player has either PD#- or SH#-experience for the cooperate/defect context.

Table 4.29.a.

Two-game set: **{PD#, SH#}**

Experience: **{PD#, SH#}** v **{PD#}**

Outcomes in **PD#** within {PD#, SH#}

Row's experience: **{PD#, SH#}**	Column's experience: **PD#** C	D
C	0.0%	10.3%
D	0.3%	89.4%

efficiency: 5.3%
entropy: 0.51

Outcomes in **SH#** within {PD#, SH#}

Row's experience: **{PD#, SH#}**	Column's experience: **PD#** C	D
C	5.8%	9.1%
D	22.4%	62.8%

efficiency: 5.8%
entropy: 1.46

Table 4.29.b.

Two-game set: **{PD#, SH#}**

Experience: **{PD#, SH#}** v **{SH#}**

Outcomes in **PD#** within {PD#, SH#}

Row's experience: **{PD#, SH#}**	Column's experience: **SH#** C	D
C	12.1%	2.0%
D	49.7%	36.2%

efficiency: 37.9%
entropy: 1.51

Outcomes in **SH#** within {PD#, SH#}

Row's experience: **{PD#, SH#}**	Column's experience: **SH#** C	D
C	27.3%	2.4%
D	54.1%	16.2%

efficiency: 27.3%
entropy: 1.55

Table 4.30. Outcome distributions in settings where the row player has {PD#, CH#}-experience & the column player has either PD#- or CH#-experience for the cooperate/defect context.

Table 4.30.a.

Two-game set: **{PD#, CH#}**

Experience: **{PD#, CH#}** v **{PD#}**

Outcomes in **PD#** within {PD#, CH#}		Column's experience: **PD#** C	**D**
Row's experience: **{PD#, CH#}**	**C**	0.0%	13.2%
	D	0.2%	86.5%

efficiency: 6.8%
entropy: 0.59

Outcomes in **CH#** within {PD#, CH#}		Column's experience: **PD#** C	**D**
Row's experience: **{PD#, CH#}**	**C**	8.8%	23.8%
	D	18.4%	49.0%

efficiency: 51.0%
entropy: 1.76

Table 4.30.b.

Two-game set: **{PD#, CH#}**

Experience: **{PD#, CH#}** v **{CH#}**

Outcomes in **PD#** within {PD#, CH#}		Column's experience: **CH#** C	**D**
Row's experience: **{PD#, CH#}**	**C**	7.4%	7.8%
	D	39.8%	45.0%

efficiency: 31.2%
entropy: 1.61

Outcomes in **CH#** within {PD#, CH#}		Column's experience: **CH#** C	**D**
Row's experience: **{PD#, CH#}**	**C**	12.9%	21.9%
	D	27.1%	38.2%

efficiency: 61.8%
entropy: 1.90

Appendix

Table 4.A.1. Distinct states activated (DSA) in both games in settings where both players have experience in a single game for the natural outcome context.

Welch's t-statistics are for (Column statistic – Row statistic) ≠ 0.

Table 4.A.1.a: PD experience v. SH experience in {PD, SH} setting.

Games:	Both PD & SH	
Player:	Row	Column
Experience:	PD	SH
DSA in both:	4.969	4.739
t-statisitic:	-1.301	
p-value:	0.1948	

Table 4.A.1.b: PD experience v. CH experience in {PD, CH} setting.

Games:	Both PD & CH	
Player:	Row	Column
Experience:	PD	CH
DSA in both:	4.968	4.927
t-statisitic:	-0.226	
p-value:	0.8218	

Table 4.A.1.c: PD experience v. BS experience in {PD, BS} setting.

Games:	Both PD & BS	
Player:	Row	Column
Experience:	PD	BS
DSA in both:	5.027	4.813
t-statisitic:	-1.164	
p-value:	0.2459	

Table 4.A.1.d: SH experience v. CH experience in {SH, CH} setting.

Games:	Mean of SH & CH	
Player:	Row	Column
Experience:	SH	CH
DSA in both:	5.091	5.025
t-statisitic:	-0.357	
p-value:	0.7214	

Table 4.A.1.e: SH experience v. BS experience in {SH, BS} setting.

Games:	Mean of SH & BS	
Player:	Row	Column
Experience:	SH	BS
DSA in both:	4.995	4.926
t-statisitic:	-0.364	
p-value:	0.7163	

Table 4.A.1.f: CH experience v. BS experience in {CH, BS} setting.

Games:	Mean of CH & BS	
Player:	Row	Column
Experience:	CH	BS
DSA in both:	5.138	5.076
t-statisitic:	-0.313	
p-value:	0.7546	

Table 4.A.2. Distinct initial states (DIS) in settings where both players have experience in a single game for the natural outcome context.

Welch's t-statistics are for (Column statistic – Row statistic) ≠ 0.

Table 4.A.2.a: PD experience v. SH experience in {PD, SH} setting.

Games:	Both PD & SH	
Player:	Row	Column
Experience:	PD	SH
DIS:	4.969	4.739
t-statisitic:	-1.301	
p-value:	0.1948	

Table 4.A.2.b: PD experience v. CH experience in {PD, CH} setting.

Games:	Both PD & CH	
Player:	Row	Column
Experience:	PD	CH
DIS:	4.968	4.927
t-statisitic:	-0.226	
p-value:	0.8218	

Table 4.A.2.c: PD experience v. BS experience in {PD, BS} setting.

Games:	Both PD & BS	
Player:	Row	Column
Experience:	PD	BS
DIS:	5.027	4.813
t-statisitic:	-1.164	
p-value:	0.2459	

Table 4.A.2.d: SH experience v. CH experience in {SH, CH} setting.

Games:	Mean of SH & CH	
Player:	Row	Column
Experience:	SH	CH
DIS:	5.091	5.025
t-statisitic:	-0.357	
p-value:	0.7214	

Table 4.A.2.e: SH experience v. BS experience in {SH, BS} setting.

Games:	Mean of SH & BS	
Player:	Row	Column
Experience:	SH	BS
DIS:	4.995	4.926
t-statisitic:	-0.364	
p-value:	0.7163	

Table 4.A.2.f: CH experience v. BS experience in {CH, BS} setting.

Games:	Mean of CH & BS	
Player:	Row	Column
Experience:	CH	BS
DIS:	5.138	5.076
t-statisitic:	-0.313	
p-value:	0.7546	

Table 4.A.3. Automaton-level state activation similarity (ALSAS) in settings where both players have experience in a single game for the natural outcome context.

Welch's t-statistics are for (Column statistic – Row statistic) ≠ 0.

Table 4.A.3.a: PD experience v. SH experience in {PD, SH} setting.

Games:	Both PD & SH	
Player:	Row	Column
Experience:	PD	SH
ALSAS:	60.224	59.001
t-statisitic:	-0.280	
p-value:	0.7795	

Table 4.A.3.b: PD experience v. CH experience in {PD, CH} setting.

Games:	Both PD & CH	
Player:	Row	Column
Experience:	PD	CH
ALSAS:	60.774	62.382
t-statisitic:	0.385	
p-value:	0.7009	

Table 4.A.3.c: PD experience v. BS experience in {PD, BS} setting.

Games:	Both PD & BS	
Player:	Row	Column
Experience:	PD	BS
ALSAS:	62.108	64.883
t-statisitic:	0.677	
p-value:	0.4994	

Table 4.A.3.d: SH experience v. CH experience in {SH, CH} setting.

Games:	Mean of SH & CH	
Player:	Row	Column
Experience:	SH	CH
ALSAS:	59.079	62.389
t-statisitic:	0.781	
p-value:	0.4359	

Table 4.A.3.e: SH experience v. BS experience in {SH, BS} setting.

Games:	Mean of SH & BS	
Player:	Row	Column
Experience:	SH	BS
ALSAS:	63.408	63.760
t-statisitic:	0.083	
p-value:	0.9339	

Table 4.A.3.f: CH experience v. BS experience in {CH, BS} setting.

Games:	Mean of CH & BS	
Player:	Row	Column
Experience:	CH	BS
ALSAS:	61.927	64.038
t-statisitic:	0.540	
p-value:	0.5897	

Table 4.A.4. Frequencies of accessible states across both games in settings where both players have experience in a single game for the natural outcome context.

Welch's t-statistics are for (Column statistic – Row statistic) ≠ 0.

Table 4.A.4.a: PD experience v. SH experience in {PD, SH} setting.

Games:	Both PD & SH	
Player:	Row	Column
Experience:	PD	SH
Accessible:	0.645	0.648
t-statisitic:	0.150	
p-value:	0.8809	

Table 4.A.4.b: PD experience v. CH experience in {PD, CH} setting.

Games:	Both PD & CH	
Player:	Row	Column
Experience:	PD	CH
Accessible:	0.651	0.640
t-statisitic:	-0.567	
p-value:	0.5713	

Table 4.A.4.c: PD experience v. BS experience in {PD, BS} setting.

Games:	Both PD & BS	
Player:	Row	Column
Experience:	PD	BS
Accessible:	0.645	0.666
t-statisitic:	1.113	
p-value:	0.2672	

Table 4.A.4.d: SH experience v. CH experience in {SH, CH} setting.

Games:	Mean of SH & CH	
Player:	Row	Column
Experience:	SH	CH
Accessible:	0.646	0.644
t-statisitic:	-0.118	
p-value:	0.9061	

Table 4.A.4.e: SH experience v. BS experience in {SH, BS} setting.

Games:	Mean of SH & BS	
Player:	Row	Column
Experience:	SH	BS
Accessible:	0.647	0.663
t-statisitic:	0.849	
p-value:	0.3968	

Table 4.A.4.f: CH experience v. BS experience in {CH, BS} setting.

Games:	Mean of CH & BS	
Player:	Row	Column
Experience:	CH	BS
Accessible:	0.650	0.672
t-statisitic:	1.104	
p-value:	0.2709	

Table 4.A.5. Mean states activated in settings where both players have experience in a single game for the natural outcome context.

Welch's t-statistics are for (Column statistic – Row statistic) ≠ 0.

Table 4.A.5.a: PD experience v. SH experience in {PD, SH} setting.

Game(s):	PD		SH		Both PD & SH	
Player:	Row	Column	Row	Column	Row	Column
Experience:	PD	SH	PD	SH	PD	SH
Mean states activated:	1.407	4.019	4.249	1.354	2.828	2.686
t-statisitic:	20.193		-18.672		-1.398	
p-value:	0.0000		0.0000		0.0469	

Table 4.A.5.b: PD experience v. CH experience in {PD, CH} setting.

Game(s):	PD		CH		Both PD & CH	
Player:	Row	Column	Row	Column	Row	Column
Experience:	PD	CH	PD	CH	PD	CH
Mean states activated:	1.407	4.116	4.232	1.452	2.820	2.784
t-statisitic:	18.576		-19.290		-0.350	
p-value:	0.0000		0.0000		0.7267	

Table 4.A.5.c: PD experience v. BS experience in {PD, BS} setting.

Game(s):	PD		BS		Both PD & BS	
Player:	Row	Column	Row	Column	Row	Column
Experience:	PD	BS	PD	BS	PD	BS
Mean states activated:	1.407	4.054	4.242	1.296	2.825	2.675
t-statisitic:	19.984		-19.444		-1.506	
p-value:	0.0000		0.0000		0.1337	

Table 4.A.5. Mean states activated in settings where both players have experience in a single game for the natural outcome context (continued).

Welch's t-statistics are for (Column statistic – Row statistic) ≠ 0.

Table 4.A.5.d: SH experience v. CH experience in {PD, SH} setting.

Game(s):	SH		CH		Both SH & CH	
Player:	Row	Column	Row	Column	Row	Column
Experience:	SH	CH	SH	CH	SH	CH
Mean states activated:	1.357	4.207	4.404	1.456	2.881	2.832
t-statisitic:	18.685		-21.005		-0.479	
p-value:	0.0000		0.0000		0.6328	

Table 4.A.5.e: SH experience v. BS experience in {PD, CH} setting.

Game(s):	SH		BS		Both SH & BS	
Player:	Row	Column	Row	Column	Row	Column
Experience:	SH	BS	SH	BS	SH	BS
Mean states activated:	1.361	4.180	4.253	1.297	2.807	2.739
t-statisitic:	19.417		-20.610		-0.666	
p-value:	0.0000		0.0000		0.5060	

Table 4.A.5.f: CH experience v. BS experience in {PD, BS} setting.

Game(s):	CH		BS		Both CH & BS	
Player:	Row	Column	Row	Column	Row	Column
Experience:	CH	BS	CH	BS	CH	BS
Mean states activated:	1.454	4.337	4.333	1.292	2.894	2.814
t-statisitic:	17.397		-19.587		-0.764	
p-value:	0.0000		0.0000		0.4457	

Table 4.A.6. Population-level state activation similarity (PLSAS) in settings where both players have experience in a single game for the natural outcome context.

Welch's t-statistics are for (Column statistic – Row statistic) ≠ 0.

Table 4.A.6.a: PD experience v. SH experience in {PD, SH} setting.

Game(s):	PD		SH		Both PD & SH	
Player:	Row	Column	Row	Column	Row	Column
Experience:	PD	SH	PD	SH	PD	SH
PLSAS:	0.259	7.314	7.098	0.257	3.678	3.786
t-statisitic:	25.185		-21.451		0.499	
p-value:	0.0000		0.0000		0.0469	

Table 4.A.6.b: PD experience v. CH experience in {PD, CH} setting.

Game(s):	PD		CH		Both PD & CH	
Player:	Row	Column	Row	Column	Row	Column
Experience:	PD	CH	PD	CH	PD	CH
PLSAS:	0.260	7.366	6.865	0.311	3.562	3.838
t-statisitic:	25.446		-20.085		1.276	
p-value:	0.0000		0.0000		0.2036	

Table 4.A.6.c: PD experience v. BS experience in {PD, BS} setting.

Game(s):	PD		BS		Both PD & BS	
Player:	Row	Column	Row	Column	Row	Column
Experience:	PD	BS	PD	BS	PD	BS
PLSAS:	0.258	7.292	7.376	0.228	3.817	3.760
t-statisitic:	24.969		-28.865		-0.305	
p-value:	0.0000		0.0000		0.7604	

Table 4.A.6. Population-level state activation similarity (PLSAS) in settings where both players have experience in a single game for the natural outcome context (continued).

Welch's t-statistics are for (Column statistic – Row statistic) ≠ 0.

Table 4.A.6.d: SH experience v. CH experience in {PD, SH} setting.

Game(s):	SH		CH		Both SH & CH	
Player:	Row	Column	Row	Column	Row	Column
Experience:	SH	CH	SH	CH	SH	CH
PLSAS:	0.208	7.137	6.839	0.325	3.524	3.731
t-statisitic:	25.688		-21.985		1.008	
p-value:	0.0000		0.0000		0.3148	

Table 4.A.6.e: SH experience v. BS experience in {PD, CH} setting.

Game(s):	SH		BS		Both SH & BS	
Player:	Row	Column	Row	Column	Row	Column
Experience:	SH	BS	SH	BS	SH	BS
PLSAS:	0.263	6.978	7.248	0.257	3.756	3.617
t-statisitic:	21.387		-27.669		-0.685	
p-value:	0.0000		0.0000		0.4944	

Table 4.A.6.f: CH experience v. BS experience in {PD, BS} setting.

Game(s):	CH		BS		Both CH & BS	
Player:	Row	Column	Row	Column	Row	Column
Experience:	CH	BS	CH	BS	CH	BS
PLSAS:	0.259	7.418	7.319	0.252	3.789	3.835
t-statisitic:	26.310		-27.045		0.237	
p-value:	0.0000		0.0000		0.8126	

Table 4.A.7. Frequencies of accessible states per game in settings where both players have experience in a single game for the natural outcome context.

Welch's t-statistics are for (Column statistic – Row statistic) ≠ 0.

Table 4.A.7.a: PD experience v. SH experience in {PD, SH} setting.

Game(s):	PD		SH		Both PD & SH	
Player:	Row	Column	Row	Column	Row	Column
Experience:	PD	SH	PD	SH	PD	SH
Accessible states:	0.089	0.638	0.638	0.085	0.363	0.362
t-statisitic:	41.335		-41.417		-0.161	
p-value:	0.0000		0.0000		0.0469	

Table 4.A.7.b: PD experience v. CH experience in {PD, CH} setting.

Game(s):	PD		CH		Both PD & CH	
Player:	Row	Column	Row	Column	Row	Column
Experience:	PD	CH	PD	CH	PD	CH
Accessible states:	0.089	0.632	0.644	0.092	0.366	0.362
t-statisitic:	37.595		-39.286		-0.413	
p-value:	0.0000		0.0000		0.6798	

Table 4.A.7.c: PD experience v. BS experience in {PD, BS} setting.

Game(s):	PD		BS		Both PD & BS	
Player:	Row	Column	Row	Column	Row	Column
Experience:	PD	BS	PD	BS	PD	BS
Accessible states:	0.089	0.655	0.638	0.082	0.363	0.368
t-statisitic:	40.337		-40.545		0.506	
p-value:	0.0000		0.0000		0.6135	

Table 4.A.7. Frequencies of accessible states per game in settings where both players have experience in a single game for the natural outcome context (continued).

Welch's t-statistics are for (Column statistic – Row statistic) ≠ 0.

Table 4.A.7.d: SH experience v. CH experience in {PD, SH} setting.

Game(s):	SH		CH		Both SH & CH	
Player:	Row	Column	Row	Column	Row	Column
Experience:	SH	CH	SH	CH	SH	CH
Accessible states:	0.085	0.636	0.637	0.092	0.361	0.364
t-statisitic:	39.440		-39.367		0.335	
p-value:	0.0000		0.0000		0.7383	

Table 4.A.7.e: SH experience v. BS experience in {PD, CH} setting.

Game(s):	SH		BS		Both SH & BS	
Player:	Row	Column	Row	Column	Row	Column
Experience:	SH	BS	SH	BS	SH	BS
Accessible states:	0.085	0.651	0.637	0.082	0.361	0.366
t-statisitic:	38.743		-43.285		0.556	
p-value:	0.0000		0.0000		0.5792	

Table 4.A.7.f: CH experience v. BS experience in {PD, BS} setting.

Game(s):	CH		BS		Both CH & BS	
Player:	Row	Column	Row	Column	Row	Column
Experience:	CH	BS	CH	BS	CH	BS
Accessible states:	0.092	0.660	0.642	0.082	0.367	0.371
t-statisitic:	40.248		-39.086		0.401	
p-value:	0.0000		0.0000		0.6889	

Table 4.A.8. Frequencies of terminal states in settings where both players have experience in a single game for the natural outcome context.

Welch's t-statistics are for (Column statistic – Row statistic) ≠ 0.

Table 4.A.8.a: PD experience v. SH experience in {PD, SH} setting.

Game(s):	PD		SH		Both PD & SH	
Player:	Row	Column	Row	Column	Row	Column
Experience:	PD	SH	PD	SH	PD	SH
Terminal states:	0.676	0.119	0.106	0.747	0.391	0.433
t-statisitic:	-11.733		14.449		1.126	
p-value:	0.0000		0.0000		0.0469	

Table 4.A.8.b: PD experience v. CH experience in {PD, CH} setting.

Game(s):	PD		CH		Both PD & CH	
Player:	Row	Column	Row	Column	Row	Column
Experience:	PD	CH	PD	CH	PD	CH
Terminal states:	0.676	0.107	0.104	0.678	0.390	0.393
t-statisitic:	-11.992		12.084		0.062	
p-value:	0.0000		0.0000		0.9508	

Table 4.A.8.c: PD experience v. BS experience in {PD, BS} setting.

Game(s):	PD		BS		Both PD & BS	
Player:	Row	Column	Row	Column	Row	Column
Experience:	PD	BS	PD	BS	PD	BS
Terminal states:	0.676	0.122	0.108	0.776	0.392	0.449
t-statisitic:	-11.642		15.713		1.573	
p-value:	0.0000		0.0000		0.1173	

Table 4.A.8. Frequencies of terminal states in settings where both players have experience in a single game for the natural outcome context (continued).

Welch's t-statistics are for (Column statistic – Row statistic) $\neq 0$.

Table 4.A.8.d: SH experience v. CH experience in {PD, SH} setting.

Game(s):	SH		CH		Both SH & CH	
Player:	Row	Column	Row	Column	Row	Column
Experience:	SH	CH	SH	CH	SH	CH
Terminal states:	0.747	0.105	0.118	0.678	0.432	0.391
t-statisitic:	-14.534		11.763		-1.111	
p-value:	0.0000		0.0000		0.2678	

Table 4.A.8.e: SH experience v. BS experience in {PD, CH} setting.

Game(s):	SH		BS		Both SH & BS	
Player:	Row	Column	Row	Column	Row	Column
Experience:	SH	BS	SH	BS	SH	BS
Terminal states:	0.747	0.123	0.123	0.776	0.435	0.449
t-statisitic:	-14.085		15.371		0.411	
p-value:	0.0000		0.0000		0.6817	

Table 4.A.8.f: CH experience v. BS experience in {PD, BS} setting.

Game(s):	CH		BS		Both CH & BS	
Player:	Row	Column	Row	Column	Row	Column
Experience:	CH	BS	CH	BS	CH	BS
Terminal states:	0.678	0.117	0.099	0.776	0.388	0.446
t-statisitic:	-11.826		16.029		1.622	
p-value:	0.0000		0.0000		0.1065	

Table 4.A.9. Frequencies of counting states in settings where both players have experience in a single game for the natural outcome context.

Welch's t-statistics are for (Column statistic – Row statistic) ≠ 0.

Table 4.A.9.a: PD experience v. SH experience in {PD, SH} setting.

Game(s):	PD		SH		Both PD & SH	
Player:	Row	Column	Row	Column	Row	Column
Experience:	PD	SH	PD	SH	PD	SH
Counting states:	0.151	0.101	0.101	0.122	0.126	0.111
t-statisitic:	-1.597		0.684		-0.543	
p-value:	0.1126		0.4953		0.0469	

Table 4.A.9.b: PD experience v. CH experience in {PD, CH} setting.

Game(s):	PD		CH		Both PD & CH	
Player:	Row	Column	Row	Column	Row	Column
Experience:	PD	CH	PD	CH	PD	CH
Counting states:	0.151	0.106	0.098	0.138	0.124	0.122
t-statisitic:	-1.467		1.428		-0.076	
p-value:	0.1450		0.1557		0.9392	

Table 4.A.9.c: PD experience v. BS experience in {PD, BS} setting.

Game(s):	PD		BS		Both PD & BS	
Player:	Row	Column	Row	Column	Row	Column
Experience:	PD	BS	PD	BS	PD	BS
Counting states:	0.151	0.093	0.101	0.115	0.126	0.104
t-statisitic:	-1.939		0.473		-0.852	
p-value:	0.0549		0.6371		0.3951	

Table 4.A.9. Frequencies of counting states in settings where both players have experience in a single game for the natural outcome context (continued).

Welch's t-statistics are for (Column statistic – Row statistic) $\neq 0$.

Table 4.A.9.d: SH experience v. CH experience in {PD, SH} setting.

Game(s):	SH		CH		Both SH & CH	
Player:	Row	Column	Row	Column	Row	Column
Experience:	SH	CH	SH	CH	SH	CH
Counting states:	0.122	0.104	0.100	0.138	0.111	0.121
t-statisitic:	-0.605		1.310		0.395	
p-value:	0.5463		0.1923		0.6931	

Table 4.A.9.e: SH experience v. BS experience in {PD, CH} setting.

Game(s):	SH		BS		Both SH & BS	
Player:	Row	Column	Row	Column	Row	Column
Experience:	SH	BS	SH	BS	SH	BS
Counting states:	0.122	0.093	0.097	0.115	0.109	0.104
t-statisitic:	-0.975		0.610		-0.207	
p-value:	0.3314		0.5427		0.8363	

Table 4.A.9.f: CH experience v. BS experience in {PD, BS} setting.

Game(s):	CH		BS		Both CH & BS	
Player:	Row	Column	Row	Column	Row	Column
Experience:	CH	BS	CH	BS	CH	BS
Counting states:	0.138	0.089	0.106	0.115	0.122	0.102
t-statisitic:	-1.783		0.301		-0.835	
p-value:	0.0770		0.7637		0.4048	

Table 4.A.10. Frequencies of states that reciprocate action 1 in settings where both players have experience in a single game for the natural outcome context.

Welch's t-statistics are for (Column statistic – Row statistic) ≠ 0.

Table 4.A.10.a: PD experience v. SH experience in {PD, SH} setting.

Game(s):	PD		SH		Both PD & SH	
Player:	Row	Column	Row	Column	Row	Column
Experience:	PD	SH	PD	SH	PD	SH
1-Reciprocating states:	0.014	0.420	0.329	0.197	0.172	0.308
t-statisitic:	22.178		-3.285		5.480	
p-value:	0.0000		0.0013		0.0475	

Table 4.A.10.b: PD experience v. CH experience in {PD, CH} setting.

Game(s):	PD		CH		Both PD & CH	
Player:	Row	Column	Row	Column	Row	Column
Experience:	PD	CH	PD	CH	PD	CH
1-Reciprocating states:	0.014	0.516	0.331	0.606	0.172	0.561
t-statisitic:	23.828		5.512		12.044	
p-value:	0.0000		0.0000		0.0000	

Table 4.A.10.c: PD experience v. BS experience in {PD, BS} setting.

Game(s):	PD		BS		Both PD & BS	
Player:	Row	Column	Row	Column	Row	Column
Experience:	PD	BS	PD	BS	PD	BS
1-Reciprocating states:	0.014	0.494	0.326	0.485	0.170	0.490
t-statisitic:	21.771		3.160		9.602	
p-value:	0.0000		0.0020		0.0000	

Table 4.A.10. Frequencies of states that reciprocate action 1 in settings where both players have experience in a single game for the natural outcome context (continued).

Welch's t-statistics are for (Column statistic – Row statistic) ≠ 0.

Table 4.A.10.d: SH experience v. CH experience in {PD, SH} setting.

Game(s):	SH		CH		Both SH & CH	
Player:	Row	Column	Row	Column	Row	Column
Experience:	SH	CH	SH	CH	SH	CH
1-Reciprocating states:	0.197	0.513	0.415	0.606	0.306	0.560
t-statisitic:	7.567		3.817		6.614	
p-value:	0.0000		0.0002		0.0000	

Table 4.A.10.e: SH experience v. BS experience in {PD, CH} setting.

Game(s):	SH		BS		Both SH & BS	
Player:	Row	Column	Row	Column	Row	Column
Experience:	SH	BS	SH	BS	SH	BS
1-Reciprocating states:	0.197	0.494	0.418	0.485	0.307	0.489
t-statisitic:	7.019		1.328		4.609	
p-value:	0.0000		0.1866		0.0000	

.

Game(s):	CH		BS		Both CH & BS	
Player:	Row	Column	Row	Column	Row	Column
Experience:	CH	BS	CH	BS	CH	BS
1-Reciprocating states:	0.606	0.498	0.512	0.485	0.559	0.491
t-statisitic:	-2.108		-0.535		-1.539	
p-value:	0.0368		0.5938		0.1254	

Table 4.A.11. Outcome distributions in settings where the row player has {PD, CH}-experience & the column player has either PD- or CH-experience for the natural outcome context.

Table 4.A.11.a.
Two-game set: **{PD, CH}**
Experience: **{PD, CH}** v **{PD}**

Outcomes in **PD** within {PD, CH}

Row's experience: **{PD, CH}**	Column's experience: **PD** **0**	**1**
0	82.3%	0.2%
1	16.9%	0.5%

efficiency: 9.1%
entropy: 0.73

Outcomes in **CH** within {PD, CH}

Row's experience: **{PD, CH}**	Column's experience: **PD** **0**	**1**
0	43.4%	18.4%
1	29.2%	9.1%

efficiency: 90.9%
entropy: 1.80

Table 4.A.11.b.
Two-game set: **{PD, CH}**
Experience: **{PD, CH}** v **{CH}**

Outcomes in **PD** within {PD, CH}

Row's experience: **{PD, CH}**	Column's experience: **CH** **0**	**1**
0	40.1%	39.1%
1	8.8%	11.9%

efficiency: 35.9%
entropy: 1.73

Outcomes in **CH** within {PD, CH}

Row's experience: **{PD, CH}**	Column's experience: **CH** **0**	**1**
0	26.5%	36.6%
1	15.2%	21.7%

efficiency: 78.3%
entropy: 1.93

Table 4.A.12. Outcome distributions in settings where the row player has {PD, BS}-experience & the column player has either PD- or BS-experience for the natural outcome context.

Table 4.A.12.a.
Two-game set: **{PD, BS}**
Experience: **{PD, BS}** v **{PD}**

Outcomes in **PD** within {PD, BS}

Row's experience: **{PD, BS}**	Column's experience: **PD** 0	**1**
0	90.1%	0.5%
1	9.3%	0.1%

efficiency: 5.0%
entropy: 0.50

Outcomes in **BS** within {PD, BS}

Row's experience: **{PD, BS}**	Column's experience: **PD** 0	**1**
0	50.6%	18.7%
1	23.7%	7.0%

efficiency: 62.7%
entropy: 1.71

Table 4.A.12.b.
Two-game set: **{PD, BS}**
Experience: **{PD, BS}** v **{BS}**

Outcomes in **PD** within {PD, BS}

Row's experience: **{PD, BS}**	Column's experience: **BS** 0	**1**
0	47.0%	40.7%
1	5.9%	6.4%

efficiency: 29.7%
entropy: 1.54

Outcomes in **BS** within {PD, BS}

Row's experience: **{PD, BS}**	Column's experience: **BS** 0	**1**
0	34.1%	31.7%
1	19.9%	14.3%

efficiency: 65.2%
entropy: 1.92

Table 4.A.13. Outcome distributions in settings where the row player has {SH, CH}-experience & the column player has either SH- or CH-experience for the natural outcome context.

Table 4.A.13.a.
Two-game set: **{SH, CH}**
Experience: **{SH, CH}** v **{SH}**

Outcomes in **SH** within {SH, CH}

Row's experience: **{SH, CH}**		Column's experience: **SH** 0	1
	0	55.0%	6.1%
	1	25.3%	13.6%

efficiency: 55.0%
entropy: 1.61

Outcomes in **CH** within {SH, CH}

Row's experience: **{SH, CH}**		Column's experience: **SH** 0	1
	0	34.6%	16.1%
	1	28.6%	20.6%

efficiency: 79.4%
entropy: 1.94

Table 4.A.13.b.
Two-game set: **{SH, CH}**
Experience: **{SH, CH}** v **{CH}**

Outcomes in **SH** within {SH, CH}

Row's experience: **{SH, CH}**		Column's experience: **CH** 0	1
	0	28.4%	26.3%
	1	22.0%	23.3%

efficiency: 28.4%
entropy: 1.99

Outcomes in **CH** within {SH, CH}

Row's experience: **{SH, CH}**		Column's experience: **CH** 0	1
	0	15.4%	33.8%
	1	25.4%	25.4%

efficiency: 74.6%
entropy: 1.95

Table 4.A.14. Outcome distributions in settings where the row player has {SH, BS}-experience & the column player has either SH- or BS-experience for the natural outcome context..

Table 4.A.14.a.
Two-game set: **{SH, BS}**
Experience: **{SH, BS}** v **{SH}**

Outcomes in **SH** within {SH, BS}

Row's experience: **{SH, BS}**	Column's experience: **SH** 0	Column's experience: **SH** 1
0	67.2%	14.3%
1	13.6%	4.9%

efficiency: 67.2%
entropy: 1.39

Outcomes in **BS** within {SH, BS}

Row's experience: **{SH, BS}**	Column's experience: **SH** 0	Column's experience: **SH** 1
0	44.9%	20.4%
1	18.7%	16.0%

efficiency: 57.1%
entropy: 1.86

Table 4.A.14.b.
Two-game set: **{SH, BS}**
Experience: **{SH, BS}** v **{BS}**

Outcomes in **SH** within {SH, BS}

Row's experience: **{SH, BS}**	Column's experience: **BS** 0	Column's experience: **BS** 1
0	38.8%	34.3%
1	14.3%	12.6%

efficiency: 38.8%
entropy: 1.84

Outcomes in **BS** within {SH, BS}

Row's experience: **{SH, BS}**	Column's experience: **BS** 0	Column's experience: **BS** 1
0	30.1%	30.3%
1	23.8%	15.8%

efficiency: 66.1%
entropy: 1.96

Table 4.A.15. Outcome distributions in settings where the row player has {CH, BS}-experience & the column player has either CH- or BS-experience for the natural outcome context..

Table 4.A.15.a.
Two-game set: **{CH, BS}**
Experience: **{CH, BS}** v **{CH}**

Outcomes in **CH** within {CH, BS}

Row's experience: **{CH, BS}**	Column's experience: **CH** 0	1
0	22.6%	25.7%
1	19.3%	32.4%

efficiency: 67.6%
entropy: 1.97

Outcomes in **BS** within {CH, BS}

Row's experience: **{CH, BS}**	Column's experience: **CH** 0	1
0	24.8%	24.6%
1	24.1%	26.5%

efficiency: 58.6%
entropy: 2.00

Table 4.A.15.b.
Two-game set: **{CH, BS}**
Experience: **{CH, BS}** v **{BS}**

Outcomes in **CH** within {CH, BS}

Row's experience: **{CH, BS}**	Column's experience: **BS** 0	1
0	25.8%	21.4%
1	27.5%	25.2%

efficiency: 74.8%
entropy: 1.99

Outcomes in **BS** within {CH, BS}

Row's experience: **{CH, BS}**	Column's experience: **BS** 0	1
0	27.8%	21.1%
1	26.2%	25.0%

efficiency: 58.3%
entropy: 1.99

Table 4.A.16. Scores in settings when row population has {PD, CH}-experience and column population has either PD-experience or CH-experience for the natural outcome context.

Table 4.A.16.a: {PD, CH} experience v. {PD} experience in {PD, CH} two-game setting

Score in game:	PD		CH		Mean of PD & CH	
Player:	Row	Column	Row	Column	Row	Column
Experience:	PD&CH	PD	PD&CH	PD	PD&CH	PD
Score:	1.841	2.341	2.926	2.710	2.383	2.526
t-statisitic:	6.838		-2.468		3.284	
p-value:	0.0000		0.0145		0.0012	

Welch's t-statistics are for (Column Score – Row Score) ≠ 0

Table 4.A.16.b: {PD, CH} experience v. {CH} experience in {PD, CH} two-game setting.

Score in game:	PD		CH		Mean of PD & CH	
Player:	Row	Column	Row	Column	Row	Column
Experience:	PD&CH	CH	PD&CH	CH	PD&CH	CH
Score:	2.813	1.905	2.352	2.779	2.583	2.342
t-statisitic:	-9.051		3.048		-3.847	
p-value:	0.0000		0.0026		0.0002	

Welch's t-statistics are for (Column Score – Row Score) ≠ 0

Table 4.A.16.c: Differences in Score between {PD} experience and {CH} experience when facing the same {PD, CH} experience opponent.

Game played:	PD	CH	Mean of PD&CH
Pop. 1 Experience (opponent's experience):	PD (PD&CH)	PD (PD&CH)	PD (PD&CH)
Pop. 2 Experience (opponent's experience):	CH (PD&CH)	CH (PD&CH)	CH (PD&CH)
t-statisitic:	-4.471	0.545	-3.196
p-value:	0.0000	0.5864	0.0016

Welch's t-statistics are for (Pop. 2 Score – Pop. 1 Score) ≠ 0

Table 4.A.16.d: Differences in Score between {PD, CH} experience when opponent has {PD} experience and when opponent has{CH} experience.

Game played:	PD	CH	Mean of PD&CH
Pop. 1 Experience (opponent's experience):	PD&CH (v PD)	PD&CH (v PD)	PD&CH (v PD)
Pop. 2 Experience (opponent's experience):	PD&CH (v CH)	PD&CH (v CH)	PD&CH (v CH)
t-statisitic:	12.663	-5.391	3.995
p-value:	0.0000	0.0000	0.0001

Welch's t-statistics are for (Pop. 2 Score – Pop. 1 Score) ≠ 0

Table 4.A.17. Scores in settings when row population has {PD, BS}-experience and column population has either PD-experience or BS-experience for the natural outcome context.

Table 4.A.17.a: {PD, BS} experience v. {PD} experience in {PD, BS} two-game setting.

Score in game:	PD		BS		Mean of PD & BS	
Player:	Row	Column	Row	Column	Row	Column
Experience:	PD&BS	PD	PD&BS	PD	PD&BS	PD
Score:	1.918	2.182	2.542	2.591	2.230	2.387
t-statisitic:	5.049		0.746		3.477	
p-value:	0.0000		0.4564		0.0006	

Welch's t-statistics are for (Column Score – Row Score) ≠ 0

Table 4.A.17.b: {PD, BS} experience v. {BS} experience in {PD, BS} two-game setting.

Score in game:	PD		BS		Mean of PD & BS	
Player:	Row	Column	Row	Column	Row	Column
Experience:	PD&BS	BS	PD&BS	BS	PD&BS	BS
Score:	2.818	1.776	2.690	2.571	2.754	2.174
t-statisitic:	-12.938		-0.885		-7.058	
p-value:	0.0000		0.3772		0.0000	

Welch's t-statistics are for (Column Score – Row Score) ≠ 0

Table 4.A.17.c: Differences in Score between {PD} experience and {BS} experience when facing the same {PD, BS} experience opponent.

Game played:	PD	BS	Mean of PD&BS
Pop. 1 Experience (opponent's experience):	PD (PD&BS)	PD (PD&BS)	PD (PD&BS)
Pop. 2 Experience (opponent's experience):	BS (PD&BS)	BS (PD&BS)	BS (PD&BS)
t-statisitic:	-5.777	-0.196	-3.347
p-value:	0.0000	0.8452	0.0010

Welch's t-statistics are for (Pop. 2 Score – Pop. 1 Score) ≠ 0

Table 4.A.17.d: Differences in Score between {PD, BS} experience when opponent has {PD} experience and when opponent has{BS} experience.

Game played:	PD	BS	Mean of PD&BS
Pop. 1 Experience (opponent's experience):	PD&BS (v PD)	PD&BS (v PD)	PD&BS (v PD)
Pop. 2 Experience (opponent's experience):	PD&BS (v BS)	PD&BS (v BS)	PD&BS (v BS)
t-statisitic:	13.752	1.372	7.608
p-value:	0.0000	0.1723	0.0000

Welch's t-statistics are for (Pop. 2 Score – Pop. 1 Score) ≠ 0

Table 4.A.18. Scores in settings when row population has {SH, CH}-experience and column population has either SH-experience or CH-experience for the natural outcome context.

Table 4.A.18.a: {SH, CH} experience v. {SH} experience in {SH, CH} two-game setting.

Score in game:	SH		CH		Mean of SH & CH	
Player:	Row	Column	Row	Column	Row	Column
Experience:	SH&CH	SH	SH&CH	SH	SH&CH	SH
Score:	3.292	2.908	2.712	2.463	3.002	2.686
t-statisitic:	-2.495		-2.650		-2.989	
p-value:	0.0135		0.0087		0.0470	

Welch's t-statistics are for (Column Score – Row Score) $\neq 0$

Table 4.A.18.b: {SH, CH} experience v. {CH} experience in {SH, CH} two-game setting.

Score in game:	SH		CH		Mean of SH & CH	
Player:	Row	Column	Row	Column	Row	Column
Experience:	SH&CH	CH	SH&CH	CH	SH&CH	CH
Score:	2.526	2.611	2.409	2.576	2.467	2.594
t-statisitic:	0.809		1.186		1.231	
p-value:	0.4198		0.2370		0.2197	

Welch's t-statistics are for (Column Score – Row Score) $\neq 0$

Table 4.A.18.c: Differences in Score between {SH} experience and {CH} experience when facing the same {SH, CH} experience opponent.

Game played:	SH	CH	Mean of SH&CH
Pop. 1 Experience (opponent's experience):	SH (SH&CH)	SH (SH&CH)	SH (SH&CH)
Pop. 2 Experience (opponent's experience):	CH (SH&CH)	CH (SH&CH)	CH (SH&CH)
t-statisitic:	-1.967	0.953	-0.815
p-value:	0.0507	0.3420	0.4162

Welch's t-statistics are for (Pop. 2 Score – Pop. 1 Score) $\neq 0$

Table 4.A.18.d: Differences in Score between {SH, CH} experience when opponent has {SH} experience and when opponent has{CH} experience.

Game played:	SH	CH	Mean of SH&CH
Pop. 1 Experience (opponent's experience):	SH&CH (v SH)	SH&CH (v SH)	SH&CH (v SH)
Pop. 2 Experience (opponent's experience):	SH&CH (v CH)	SH&CH (v CH)	SH&CH (v CH)
t-statisitic:	-7.000	-2.504	-5.617
p-value:	0.0000	0.0133	0.0000

Welch's t-statistics are for (Pop. 2 Score – Pop. 1 Score) $\neq 0$

Table 4.A.19. Scores in settings when row population has {SH, BS}-experience and column population has either SH-experience or BS-experience for the natural outcome context.

Table 4.A.19.a: {SH, BS} experience v. {SH} experience in {SH, BS} two-game setting.

Score in game:	SH		BS		Mean of SH & BS	
Player:	Row	Column	Row	Column	Row	Column
Experience:	SH&BS	SH	SH&BS	SH	SH&BS	SH
Score:	3.337	3.349	2.435	2.418	2.886	2.884
t-statisitic:	0.084		-0.239		-0.036	
p-value:	0.9333		0.8116		0.9715	

Welch's t-statistics are for (Column Score – Row Score) ≠ 0

Table 4.A.19.b: {SH, BS} experience v. {BS} experience in {SH, BS} two-game setting.

Score in game:	SH		BS		Mean of SH & BS	
Player:	Row	Column	Row	Column	Row	Column
Experience:	SH&BS	BS	SH&BS	BS	SH&BS	BS
Score:	2.576	2.976	2.686	2.622	2.631	2.799
t-statisitic:	3.356		-0.481		2.196	
p-value:	0.0009		0.6312		0.0293	

Welch's t-statistics are for (Column Score – Row Score) ≠ 0

Table 4.A.19.c: Differences in Score between {SH} experience and {BS} experience when facing the same {SH, BS} experience opponent.

Game played:	SH	BS	Mean of SH&BS
Pop. 1 Experience (opponent's experience):	SH (SH&BS)	SH (SH&BS)	SH (SH&BS)
Pop. 2 Experience (opponent's experience):	BS (SH&BS)	BS (SH&BS)	BS (SH&BS)
t-statisitic:	-2.859	1.961	-1.090
p-value:	0.0047	0.0518	0.2769

Welch's t-statistics are for (Pop. 2 Score – Pop. 1 Score) ≠ 0

Table 4.A.19.d: Differences in Score between {SH, BS} experience when opponent has {SH} experience and when opponent has{BS} experience.

Game played:	SH	BS	Mean of SH&BS
Pop. 1 Experience (opponent's experience):	SH&BS (v SH)	SH&BS (v SH)	SH&BS (v SH)
Pop. 2 Experience (opponent's experience):	SH&BS (v BS)	SH&BS (v BS)	SH&BS (v BS)
t-statisitic:	-5.536	2.249	-3.532
p-value:	0.0000	0.0259	0.0005

Welch's t-statistics are for (Pop. 2 Score – Pop. 1 Score) ≠ 0

Table 4.A.20. Scores in settings when row population has {CH, BS}-experience and column population has either CH-experience or BS-experience for the natural outcome context.

Table 4.A.20.a: {CH, BS} experience v. {CH} experience in {CH, BS} two-game setting.

Score in game:	CH		BS		Mean of CH & BS	
Player:	Row	Column	Row	Column	Row	Column
Experience:	CH&BS	CH	CH&BS	CH	CH&BS	CH
Score:	2.288	2.415	2.468	2.462	2.378	2.439
t-statisitic:	0.817		-0.056		0.591	
p-value:	0.4151		0.9557		0.5551	

Welch's t-statistics are for (Column Score – Row Score) ≠ 0

Table 4.A.20.b: {CH, BS} experience v. {BS} experience in {CH, BS} two-game setting.

Score in game:	CH		BS		Mean of CH & BS	
Player:	Row	Column	Row	Column	Row	Column
Experience:	CH&BS	BS	CH&BS	BS	CH&BS	BS
Score:	2.557	2.435	2.433	2.484	2.495	2.459
t-statisitic:	-1.042		0.348		-0.341	
p-value:	0.2987		0.7279		0.7335	

Welch's t-statistics are for (Column Score – Row Score) ≠ 0

Table 4.A.20.c: Differences in Score between {CH} experience and {BS} experience when facing the same {CH, BS} experience opponent.

Game played:	CH	BS	Mean of CH&BS
Pop. 1 Experience (opponent's experience):	CH (CH&BS)	CH (CH&BS)	CH (CH&BS)
Pop. 2 Experience (opponent's experience):	BS (CH&BS)	BS (CH&BS)	BS (CH&BS)
t-statisitic:	0.130	0.178	0.193
p-value:	0.8966	0.8590	0.8470

Welch's t-statistics are for (Pop. 2 Score – Pop. 1 Score) ≠ 0

Table 4.A.20.d: Differences in Score between {CH, BS} experience when opponent has {CH} experience and when opponent has{BS} experience.

Game played:	CH	BS	Mean of CH&BS
Pop. 1 Experience (opponent's experience):	CH&BS (v CH)	CH&BS (v CH)	CH&BS (v CH)
Pop. 2 Experience (opponent's experience):	CH&BS (v BS)	CH&BS (v BS)	CH&BS (v BS)
t-statisitic:	2.112	-0.274	1.150
p-value:	0.0361	0.7846	0.2514

Welch's t-statistics are for (Pop. 2 Score – Pop. 1 Score) ≠ 0

Table 4.A.21. Action 0 usage frequencies in settings when row player has {PD, SH}-experience & column player has either PD-experience or SH-experience for the natural outcome context.

t-statistics are for (Column profit – Row profit) ≠ 0 or (Pop. 2 profit – Pop. 1 profit) ≠ 0

Table 4.A.21.a: {PD, SH}-experience v. PD-experience in {PD, SH} two-game setting.

Action 0 Frequency in game:	PD		SH	
Player:	Row	Column	Row	Column
Experience:	PD&SH	PD	PD&SH	PD
Action 0 connotative experience:	S & C	S	S & C	n/a
Action 0 Frequency:	0.997	0.996	0.987	0.718
t-statisitic:	-0.609		-10.550	
p-value:	0.5434		0.0000	

Table 4.A.21.b: {PD, SH}-experience v. SH-experience in {PD, SH} two-game setting.

Action 0 Frequency in game:	PD		SH	
Player:	Row	Column	Row	Column
Experience:	PD&SH	SH	PD&SH	SH
Action 0 connotative experience:	S & C	n/a	S & C	C
Action 0 Frequency:	0.990	0.667	0.992	0.804
t-statisitic:	-11.913		-4.775	
p-value:	0.0000		0.0000	

Table 4.A.21.c: Differences in Action 0 Frequency for strategy populations with different one-game experiences facing the same opponent.

Game played:	PD	SH
Pop. 1 Experience (opponent's experience):	PD (PD&SH)	PD (PD&SH)
Pop. 2 Experience (opponent's experience):	SH (PD&SH)	SH (PD&SH)
t-statisitic:	-12.222	1.851
p-value:	0.0000	0.0659

Table 4.A.21.d: Differences in Action 0 Frequency for strategy populations with the same two-game experience facing different opponents.

Game played:	PD	SH
Pop. 1 Experience (opponent's experience):	PD&SH (v PD)	PD&SH (v PD)
Pop. 2 Experience (opponent's experience):	PD&SH (v SH)	PD&SH (v SH)
t-statisitic:	-2.207	0.813
p-value:	0.0294	0.4174

Table 4.A.22. Action 0 usage frequencies in settings when row player has {PD, CH}-experience & column player has either PD-experience or CH-experience for the natural outcome context.

t-statistics are for (Column profit – Row profit) ≠ 0 or (Pop. 2 profit – Pop. 1 profit) ≠ 0

Table 4.A.22.a: {PD, CH}-experience v. PD-experience in {PD, CH} two-game setting.

Action 0 Frequency in game:	PD		CH	
Player:	Row	Column	Row	Column
Experience:	PD&CH	PD	PD&CH	PD
Action 0 connotative experience:	S & C	S	S & C	n/a
Action 0 Frequency:	0.826	0.992	0.618	0.725
t-statisitic:	4.949		2.166	
p-value:	0.0000		0.0318	

Table 4.A.22.b: {PD, CH}-experience v. CH-experience in {PD, CH} two-game setting.

Action 0 Frequency in game:	PD		CH	
Player:	Row	Column	Row	Column
Experience:	PD&CH	CH	PD&CH	CH
Action 0 connotative experience:	S & C	n/a	S & C	C
Action 0 Frequency:	0.793	0.490	0.631	0.417
t-statisitic:	-6.500		-3.320	
p-value:	0.0000		0.0011	

Table 4.A.22.c: Differences in Action 0 Frequency for strategy populations with different one-game experiences facing the same opponent.

Game played:	PD	CH
Pop. 1 Experience (opponent's experience):	PD (PD&CH)	PD (PD&CH)
Pop. 2 Experience (opponent's experience):	CH (PD&CH)	CH (PD&CH)
t-statisitic:	-15.538	-5.768
p-value:	0.0000	0.0000

Table 4.A.22.d: Differences in Action 0 Frequency for strategy populations with the same two-game experience facing different opponents.

Game played:	PD	CH
Pop. 1 Experience (opponent's experience):	PD&CH (v PD)	PD&CH (v PD)
Pop. 2 Experience (opponent's experience):	PD&CH (v CH)	PD&CH (v CH)
t-statisitic:	-0.695	0.215
p-value:	0.4880	0.8299

Table 4.A.23. Action 0 usage frequencies in settings when row player has {PD, BS}-experience & column player has either PD-experience or BS-experience for the natural outcome context.

t-statistics are for (Column profit – Row profit) ≠ 0 or (Pop. 2 profit – Pop. 1 profit) ≠ 0

Table 4.A.23.a: {PD, BS}-experience v. PD-experience in {PD, BS} two-game setting.

Action 0 Frequency in game:	PD		BS	
Player:	Row	Column	Row	Column
Experience:	PD&BS	PD	PD&BS	PD
Action 0 connotative experience:	S	S	S	n/a
Action 0 Frequency:	0.906	0.994	0.693	0.743
t-statisitic:	3.718		1.163	
p-value:	0.0003		0.2467	

Table 4.A.23.b: {PD, BS}-experience v. BS-experience in {PD, BS} two-game setting.

Action 0 Frequency in game:	PD		BS	
Player:	Row	Column	Row	Column
Experience:	PD&BS	BS	PD&BS	BS
Action 0 connotative experience:	S	n/a	S	S
Action 0 Frequency:	0.876	0.529	0.658	0.539
t-statisitic:	-9.286		-1.831	
p-value:	0.0000		0.0686	

Table 4.A.23.c: Differences in Action 0 Frequency for strategy populations with different one-game experiences facing the same opponent.

Game played:	PD	BS
Pop. 1 Experience (opponent's experience):	PD (PD&BS)	PD (PD&BS)
Pop. 2 Experience (opponent's experience):	BS (PD&BS)	BS (PD&BS)
t-statisitic:	-15.748	-3.865
p-value:	0.0000	0.0002

Table 4.A.23.d: Differences in Action 0 Frequency for strategy populations with the same two-game experience facing different opponents.

Game played:	PD	BS
Pop. 1 Experience (opponent's experience):	PD&BS (v PD)	PD&BS (v PD)
Pop. 2 Experience (opponent's experience):	PD&BS (v BS)	PD&BS (v BS)
t-statisitic:	-0.900	-0.616
p-value:	0.3690	0.5384

Table 4.A.24. Action 0 usage frequencies in settings when row player has {SH, CH}-experience and column player has either SH-experience or CH-experience for the natural outcome context.

t-statistics are for (Column profit – Row profit) ≠ 0 or (Pop. 2 profit – Pop. 1 profit) ≠ 0

Table 4.A.24.a: {SH, CH}-experience v. SH-experience in {SH, CH} two-game setting.

Action 0 Frequency in game:	SH		CH	
Player:	Row	Column	Row	Column
Experience:	SH&CH	SH	SH&CH	SH
Action 0 connotative experience:	C	C	C	n/a
Action 0 Frequency:	0.611	0.803	0.508	0.632
t-statisitic:	3.198		2.538	
p-value:	0.0016		0.0121	

Table 4.A.24.b: {SH, CH}-experience v. CH-experience in {SH, CH} two-game setting.

Action 0 Frequency in game:	SH		CH	
Player:	Row	Column	Row	Column
Experience:	SH&CH	CH	SH&CH	CH
Action 0 connotative experience:	C	n/a	C	C
Action 0 Frequency:	0.547	0.504	0.492	0.408
t-statisitic:	-0.892		-1.327	
p-value:	0.3738		0.1860	

Table 4.A.24.c: Differences in Action 0 Frequency for strategy populations with different one-game experiences facing the same opponent.

Game played:	SH	CH
Pop. 1 Experience (opponent's experience):	SH (SH&CH)	SH (SH&CH)
Pop. 2 Experience (opponent's experience):	CH (SH&CH)	CH (SH&CH)
t-statisitic:	-6.431	-4.155
p-value:	0.0000	0.0001

Table 4.A.24.d: Differences in Action 0 Frequency for strategy populations with the same two-game experience facing different opponents.

Game played:	SH	CH
Pop. 1 Experience (opponent's experience):	SH&CH (v SH)	SH&CH (v SH)
Pop. 2 Experience (opponent's experience):	SH&CH (v CH)	SH&CH (v CH)
t-statisitic:	-1.054	-0.264
p-value:	0.2933	0.7918

Table 4.A.25. Action 0 usage frequencies in settings when row player has {SH, BS}-experience & column player has either SH-experience or BS-experience for the natural outcome context.

t-statistics are for (Column profit – Row profit) ≠ 0 or (Pop. 2 profit – Pop. 1 profit) ≠ 0

Table 4.A.25.a: {SH, BS}-experience v. SH-experience in {SH, BS} two-game setting.

Action 0 Frequency in game:	SH		BS	
Player:	Row	Column	Row	Column
Experience:	SH&BS	SH	SH&BS	SH
Action 0 connotative experience:	S & C	C	S & C	n/a
Action 0 Frequency:	0.814	0.808	0.653	0.636
t-statisitic:	-0.116		-0.366	
p-value:	0.9076		0.7148	

Table 4.A.25.b: {SH, BS}-experience v. BS-experience in {SH, BS} two-game setting.

Action 0 Frequency in game:	SH		BS	
Player:	Row	Column	Row	Column
Experience:	SH&BS	BS	SH&BS	BS
Action 0 connotative experience:	S & C	n/a	S & C	S
Action 0 Frequency:	0.731	0.531	0.604	0.540
t-statisitic:	-4.009		-0.965	
p-value:	0.0001		0.3359	

Table 4.A.25.c: Differences in Action 0 Frequency for strategy populations with different one-game experiences facing the same opponent.

Game played:	SH	BS
Pop. 1 Experience (opponent's experience):	SH (SH&BS)	SH (SH&BS)
Pop. 2 Experience (opponent's experience):	BS (SH&BS)	BS (SH&BS)
t-statisitic:	-5.426	-1.738
p-value:	0.0000	0.0842

Table 4.A.25.d: Differences in Action 0 Frequency for strategy populations with the same two-game experience facing different opponents.

Game played:	SH	BS
Pop. 1 Experience (opponent's experience):	SH&BS (v SH)	SH&BS (v SH)
Pop. 2 Experience (opponent's experience):	SH&BS (v BS)	SH&BS (v BS)
t-statisitic:	-1.599	-0.822
p-value:	0.1115	0.4123

Table 4.A.26. Action 0 usage frequencies in settings when row player has {CH, BS}-experience & column player has either CH-experience or BS-experience for the natural outcome context.

t-statistics are for (Column profit – Row profit) ≠ 0 or (Pop. 2 profit – Pop. 1 profit) ≠ 0

Table 4.A.26.a: {CH, BS}-experience v. CH-experience in {CH, BS} two-game setting.

Action 0 Frequency in game:	CH		BS	
Player:	Row	Column	Row	Column
Experience:	CH&BS	CH	CH&BS	CH
Action 0 connotative experience:	S & C	C	S & C	n/a
Action 0 Frequency:	0.483	0.419	0.494	0.488
t-statisitic:	-0.941		-0.099	
p-value:	0.3476		0.9216	

Table 4.A.26.b: {CH, BS}-experience v. BS-experience in {CH, BS} two-game setting.

Action 0 Frequency in game:	CH		BS	
Player:	Row	Column	Row	Column
Experience:	CH&BS	BS	CH&BS	BS
Action 0 connotative experience:	S & C	n/a	S & C	S
Action 0 Frequency:	0.472	0.534	0.488	0.539
t-statisitic:	1.105		0.758	
p-value:	0.2708		0.4493	

Table 4.A.26.c: Differences in Action 0 Frequency for strategy populations with different one-game experiences facing the same opponent.

Game played:	CH	BS
Pop. 1 Experience (opponent's experience):	CH (CH&BS)	CH (CH&BS)
Pop. 2 Experience (opponent's experience):	BS (CH&BS)	BS (CH&BS)
t-statisitic:	2.045	0.920
p-value:	0.0424	0.3590

Table 4.A.26.d: Differences in Action 0 Frequency for strategy populations with the same two-game experience facing different opponents.

Game played:	CH	BS
Pop. 1 Experience (opponent's experience):	CH&BS (v CH)	CH&BS (v CH)
Pop. 2 Experience (opponent's experience):	CH&BS (v BS)	CH&BS (v BS)

Bibliography

Bednar, J., Chen, Y., Liu, T. X., Page, S. 2010. "Behavioral Spillovers and Cognitive Load in Multiple Games." http://yanchen.people.si.umich.edu/papers/Multiple_Games_20101101.pdf.

Bednar, J., Page, S. 2007. "Can Games(s) Theory Explain Culture? The Emergence of Cultural Behavior within Multiple Games." *Rationality and Society*, 19: 65–97.

Gilboa, I., Schmeidler, D. 1995. "Case-Based Decision Theory." *The Quarterly Journal of Economics*, 110(3): 605-639.

Gilboa, I., Schmeidler, D. 2001. *A Theory of Case-Based Decisions*. Cambridge, UK: Cambridge University Press.

Güth, W. 2000a. "Boundedly Rational Decision Emergence: A General Perspective and Some Selective Illustrations." *Journal of Economic Psychology*, 21: 433-458.

Güth, W. 2000b. "Robust Learning Experiments: Evidence for Learning and Deliberation." Humboldt-Universität Discussion Paper (Economic Series) No. 165.

Holland, J. H. 1995. *Hidden Order: How Adaptation Builds Complexity*. Reading, Massachusetts: Addison-Wesley.

Miller, J. H. 1996. The coevolution of automata in the repeated prisoner's dilemma. *Journal of Economic Behavior and Organization 29*, 87-112.

Miller, J. H., Page, S. E. 2007. *Complex Adaptive Systems: An Introduction to Computational Models of Social Life*. Princeton, NJ: Princeton University Press.

Rapoport, A., Guyer, M. J., Gordon, D. G. 1976. *The 2 X 2 Game*. Ann Arbor, MI: The University of Michigan Press.

Samuelson, L. 2001. "Analogies, Adaptation, and Anomalies." *Journal of Economic Theory*, 97: 320-366.