A Generalized Approach To Belief Learning In Repeated Games

Christos A. Ioannou *†
University of Southampton

Julian Romero ‡
Purdue University

This draft: Tuesday 25th February, 2014

Abstract

We propose a methodology that is generalizable to a broad class of repeated games in order to facilitate operability of belief learning models with repeated-game strategies. The methodology consists of (1) a generalized repeated-game strategy space, (2) a mapping between histories and repeated-game beliefs, and (3) asynchronous updating of repeated-game strategies. We implement the proposed methodology by building on three proven action learning models. Their predictions with repeated-game strategies are then validated with data from experiments with human subjects in four, symmetric 2×2 games: Prisoner's Dilemma, Battle of the Sexes, Stag-Hunt, and Chicken. The models with repeated-game strategies approximate subjects' behavior substantially better than their respective models with action learning. Additionally, inferred rules of behavior in the experimental data overlap with the rules of behavior predicted.

JEL Classification: C51, C92, C72, D03

Keywords: Adaptive Models, Belief Learning, Repeated-Game Strategies, Finite Automata, Prisoner's Dilemma, Battle of the Sexes, Stag-Hunt, Chicken

*The paper has benefited greatly from the comments of Antonio Cabrales, David J. Cooper, Philippe Jéhiel, Nobuyuki Hanaki, Antonella Ianni, Leonidas Spiliopoulos, Miltos Makris, Spyros Galanis, Thomas Gall, Syngjoo Choi, Jian Tong, Chong Juin-Kuan, Laurent Mathevet, Tim Cason, Jasmina Arifovic, David Gill and Guillaume Fréchette. We would also like to thank the seminar participants at the Purdue University, University of Southampton, Midwest Economic Theory Conference (May 2012), 4^{th} World Congress of the Game Theory Society (July 2012), 11^{th} Conference on Research on Economic Theory & Econometrics (July 2012), European Meeting of the Econometric Society (August 2012), Association of Southern European Economic Theorists Meeting (November 2012), North American Summer Meeting of the Econometric Society (June 2013), and 2^{nd} Winter Workshop in Economic Theory (Southampton 2014). Finally, we are indebted to the editor, Vince Crawford, the advisory editor and three anonymous referees for their detailed and helpful comments, which significantly improved the paper. The usual disclaimer applies.

†Mailing Address: Department of Economics, University of Southampton, Southampton, SO17 1BJ, United Kingdom. Email: c.ioannou@soton.ac.uk

‡Mailing Address: Department of Economics, Krannert School of Management, Purdue University, Lafayette, IN 47907. Email: jnromero@purdue.edu

1 Introduction

The limitations of adaptive models with actions have been well recognized in the literature. For instance, Erev and Roth (1998) note that it will not generally be the case that learning behavior can be analyzed in terms of actions alone (p. 872). Along similar lines, Camerer and Ho (1999) point out that actions "are not always the most natural candidates for the strategies that players learn about" (p. 871). Yet developing models with repeated-game strategies has been inhibited by several obstacles. First, the set of possible strategies in repeated games is infinite (uncountable). Expecting a player to fully explore such an infinite set is therefore unrealistic and impractical. Second, as McKelvey and Palfrey (2001) note, "players face an inference problem going from histories to beliefs about opponents' strategies" in repeated games (p. 25). A player's beliefs about the repeated-game strategy of the opponent become more complex because several different strategies can lead to the same history. Consequently, even though the history of play is publicly observed, a player may not know the precise strategy of the opponent. Third, repeated-game strategies need several periods to be evaluated. Traditionally, action learning models required that the updating of each player's action set occurs synchronously at the end of each period. This is sensible if a player uses actions, but if a player uses repeated-game strategies, then the player ought to play the stage game a number of times before assessing the payoff consequences of the repeated-game strategy chosen.

In this study, we propose a methodology that is generalizable to a broad class of repeated games in order to facilitate operability of belief learning models with repeated-game strategies. The methodology consists of (1) a generalized repeated-game strategy space, (2) a mapping between histories and repeated-game beliefs, and (3) asynchronous updating of repeated-game strategies. The first step in operationalizing the proposed framework is to use generalizable rules, which require a relatively small repeated-game strategy set, but may implicitly encompass a much larger space (see for instance, Stahl's rule learning in Stahl (1999) and Stahl and Haruvy (2012)). A large number of repeated-game strategies is impractical for updating under most existing learning models because the probability of observing any particular strategy in the space is near zero. We propose instead a generalized repeated-game strategy space where players' strategies are implemented by a type of finite automaton, called a *Moore machine* (Moore (1956)); thus, the strategy space only includes a subset of the theoretically large set of possible strategies in repeated games. The second step establishes a mapping between histories and repeated-game beliefs. In particular, a fitness function is proposed, which counts the number of consecutive fits of each candidate strategy of the opponent with the observed action profile sequence, starting from the most recent action profile and going backwards. Beliefs for each candidate strategy of the opponent are derived by normalizing each strategy's respective fitness by the total fitness of all candidate strategies of the opponent. This novel approach solves the inference problem of going from histories to

beliefs about opponents' strategies in a manner consistent with belief learning.[1] The third step accommodates asynchronous updating of repeated-game strategies. For one, a player's strategy set is updated with the completion of a *block of periods*, and not, necessarily, at the end of each period as traditional action learning models require. Furthermore, the probability of updating the strategy set is endogenous and based on the "surprise-triggers-change" regularity identified by Erev and Haruvy (2013).[2] Surprise is defined as the difference between actual and expected (anticipated) payoff. Thus, if a player receives a payoff similar to what is expected, then surprise is low hence the probability of updating the strategy set increases by a relatively small amount. On the other hand, if a player receives a payoff that is drastically different from the one anticipated, then surprise is high hence the probability of updating the strategy set increases by a relative large amount. Henceforth, for brevity, we refer to learning of repeated-game strategies as *strategy learning*. The proposed methodology for strategy learning is indicated in Table 1.

Generalized Repeated-Game Strategy Space We propose a generalized repeated-game strategy space where players strategies are implemented by a type of finite automaton, called a Moore machine (Moore (1956)).
Mapping Between Histories and Repeated-Game Beliefs We propose a fitness function, which counts the number of consecutive fits of each candidate strategy of the opponent with the observed action profile sequence, starting from the most recent action profile and going backwards. Beliefs for each candidate strategy of the opponent are derived by normalizing each strategy's respective fitness by the total number of fitness of all candidate strategies of the opponent.
Asynchronous Updating of Repeated-Game Strategies We propose that the updating of repeated-game strategies is endogenous and based on the "surprise-triggers-change" regularity identified by Erev and Haruvy (2013). Thus, a player's strategy set is updated asynchronously with the completion of a block of periods, and not, necessarily, at the end of each period.

Table 1: Proposed Methodology For Strategy Learning

[1] Alternatively, Hanaki, Sethi, Erev, and Peterhansl (2005) develop a model of learning of repeated-game strategies with standard reinforcement. Reinforcement learning responds only to payoffs obtained by strategies chosen by the player, hence evades the inference problem highlighted above. Yet reinforcement models are most sensible when players do not know the foregone payoffs of unchosen strategies. Several studies show that providing foregone payoff information affects learning, which suggests that players do not simply reinforce chosen strategies (see Mookherjee and Sopher (1994), Rapoport and Erev (1998), Camerer and Ho (1999), Costa-Gomes, Crawford, and Broseta (2001), Nyarko and Schotter (2002) and Van Huyck, Battalio, and Rankin (2007)).

[2] Erev and Haruvy (2013) observed that subjects exhibit a positive relationship (inertia) between recent and current action choices (see also Cooper and Kagel (2003) and Erev and Haruvy (2005)). Yet the probability of terminating the inertia mode increases with surprise; that is, surprise triggers change.

We assess the impact of the proposed methodology by building on three leading action learning models: a self-tuning Experience Weighted Attraction model (Ho, Camerer, and Chong (2007)), a γ-Weighted Beliefs model (Cheung and Friedman (1997)), and an Inertia, Sampling and Weighting model (Erev, Ert, and Roth (2010)). The predictions of the three models with strategy learning are validated with data from the experiments with human subjects of Mathevet and Romero (2012) in four, symmetric 2×2 games: Prisoner's Dilemma, Battle of the Sexes, Stag-Hunt, and Chicken. We use the experimental dataset to also validate the predictions of their respective models with action learning, which enables us to determine the improvement in fit in moving from action learning models to strategy learning ones. Finally, we infer rules of behavior in the experimental dataset and compare them to those predicted by the strategy learning models.

We find that the strategy learning models approximate subjects' behavior substantially better than their respective models with action learning. Furthermore, inferred rules of behavior in the experimental data overlap with the rules of behavior predicted. More specifically, the most prevalent rules of behavior in the experimental dataset in the Prisoner's Dilemma, Stag-Hunt, and Chicken are cooperative rules of behavior "Grim-Trigger" and "Tit-For-Tat." The same two rules emerge as the most prevalent in the simulations. Likewise, in the Battle of the Sexes, the same cooperative rules of behavior implementing alternations that prevail in the experimental dataset, also prevail in the simulations.

The layout of this paper adheres to the following plan. In Section 2, we discuss the games and review the related experimental literature. In Section 3, we provide details on the simulations and the goodness-of-fit measure used to compare quantitatively the predictions of the models with laboratory experimental data. In Section 4, we provide the action learning models deployed to implement the proposed methodology. Our methodology is presented in Section 5. In Section 6, we display the results of the computational simulations. Additionally, we infer rules of behavior in the experimental data and compare them to the rules of behavior predicted by the strategy learning models. Finally, in Section 7, we offer concluding remarks and direction for future research.

2 The Games

Our choice of games is not coincidental. We targeted four, symmetric 2×2 games that are simple albeit capture important aspects of everyday experiences, such as cooperation, coordination and reciprocity. In particular, we chose the Prisoner's Dilemma game, the Battle of the Sexes game, the Stag-Hunt game, and the Chicken game. The payoff matrices of the games are illustrated in Figure 1. The payoff matrix of the Prisoner's Dilemma game is indicated in Figure 1(a), where the cooperative action is denoted with the letter "A" and the action of defection is denoted with

	A	B
A	3,3	1,4
B	4,1	2,2

(a) Prisoner's Dilemma

	A	B
A	1,1	2,4
B	4,2	1,1

(b) Battle of the Sexes

	A	B
A	3,3	0,2
B	2,0	1,1

(c) Stag-Hunt

	A	B
A	3,3	1,4
B	4,1	0,0

(d) Chicken

Figure 1: Payoff Matrices

the letter "B." Each player's dominant strategy is to play B. The payoff matrix of the Battle of the Sexes game is indicated in Figure 1(b). In this game, there are two pure-strategy Nash equilibria: (A, B) and (B, A). Each player receives a higher payoff in the equilibrium in which he plays B. Alternating between the two pure-strategy Nash equilibria leads to the only Pareto optimal outcome where the two players receive equal payoffs. The payoff matrix of the Stag-Hunt game is indicated in Figure 1(c). In this game, there are two pure-strategy Nash equilibria: (A, A) and (B, B). However, outcome (A, A) is the payoff-dominant Nash equilibrium. The payoff matrix of the Chicken game is indicated in Figure 1(d). In this game, there is a mixed symmetric Nash equilibrium and two pure-strategy Nash equilibria: (A, B) and (B, A). Furthermore, the mutual conciliation outcome of (A, A) yields higher payoffs than the average payoffs of each player when alternating between the pure-strategy Nash equilibria.

Recently, Mathevet and Romero (2012) conducted laboratory experiments using the payoff matrices displayed in Figure 1. The experimental sessions were run at the Vernon Smith Experimental Economics Laboratory at Purdue University. Pairs were matched in a fixed matching protocol. The continuation probability for an additional period was 0.99 and was common knowledge in all experimental sessions. The authors find remarkable regularity in the data. In the

Prisoner's Dilemma game, pairs predominantly end up at the mutual cooperation outcome. However, there is also a large number of pairs that end up at the mutual defection outcome. In the Battle of the Sexes, most pairs alternate between the two pure-strategy Nash equilibria. The latter outcome promotes efficiency and fainess. In the Stag-Hunt game, pairs predominantly end up at the payoff-dominant Nash equilibrium, whereas in the Chicken game, most pairs coordinate on the mutual conciliation outcome. The experimental data is included in Appendix A. (The relative frequency of each payoff combination over the last 10 periods of game-play in the experiments of Mathevet and Romero (2012) is displayed in Figure 2.)

The findings in Mathevet and Romero (2012) confirm the trends detected by a vast number of much earlier studies (e.g. Rapoport and Chammah (1965) and Rapoport, Guyer, and Gordon (1976)). Yet standard learning algorithms have limited success in capturing the degree of cooperation in the Prisoner's Dilemma game and the alternation between the two pure-strategy Nash equilibria in the Battle of the Sexes game (Arifovic, McKelvey, and Pevnitskaya (2006)). Along similar lines, Erev and Haruvy (2013) point out that "human agents exhibit higher social intelligence and/or sensitivity than assumed by the basic learning models" (p. 61). Hanaki, Sethi, Erev, and Peterhansl (2005) partially resolve the problem by demonstrating that applying a restricted set of repeated-game strategies to a simple reinforcement model of learning can account for some of the trends observed in the experiments. However, several studies show that providing foregone payoff information affects learning, which suggests that players do not simply reinforce chosen strategies. In fact, in certain settings players are more sensitive to foregone than to obtained payoffs (see Grosskopf, Erev, and Yechiam (2006)). An alternative approach could be to assume a mixture of adaptive and sophisticated players. An adaptive player responds to either the payoffs earned or the history of play, but does not anticipate how others are learning. On the other hand, a sophisticated player rationally best responds to his forecasts of all other behaviors.[3] Furthermore, a sophisticated player is either myopic or farsighted. A farsighted player develops multiple-period rather than single-period forecasts of others' behaviors and chooses to "teach" the other players by selecting a strategy scenario that gives him the highest discounted net present value.[4] Yet such teaching models' inability to both execute *and* anticipate sophisticated behaviors, impedes the delivery of cooperation and conciliation in the Prisoner's Dilemma game and the Chicken game, respectively. Take for instance, learning in the Prisoner's Dilemma game. Assume that there exists a population of agents, which consists of sophisticated players and adaptive players á la

[3]The theoretical literature on sophisticated decision makers demonstrates that their presence may lead to Stackelberg payoffs for the sophisticated player (Fudenberg and Levine (1989)), may lead to the risk-dominant equilibrium as long as the sophisticated player is sufficiently patient (Ellison (1997)), or may force cooperation if the sophisticated player has limited foresight (Jéhiel (2001)).

[4]In the generalized model of Camerer, Ho, and Chong (2002) cooperation emerges in p-beauty contests and repeated Trust games when sophisticated players are able to teach their opponents that cooperation is beneficial (see also Chong, Camerer, and Ho (2006) and Hyndman, Ozbay, Schotter, and Ehrblatt (2012)).

Camerer, Ho, and Chong (2002). An adaptive player always chooses to defect, regardless of his belief about the opponent's action, because defection is a strictly dominant action.[5] On the other hand, a sophisticated player is able to anticipate the effect of his own behavior on the actions of his opponent. However, this is not sufficient to drive a sophisticated player paired with an adaptive player to cooperative behavior, because the adaptive player will choose to defect, as defection is always his best response. Consequently, the sophisticated player will also respond with defection thus, the pair will lock themselves into an endless string of defections. Analogous arguments hold for the Chicken game; that is, a teaching model with sophisticated and adaptive players would predict the Nash equilibrium − not, the mutual conciliation outcome.

In this study, we seek to capture the effect of experience on cooperation and coordination by facilitating learning among a subset of repeated-game strategies. In particular, players are assumed to consider strategies that can be represented by finite automata having no more than two states. Such specification of the strategy set improves predictions in two distinct ways. First, it allows for convergence to non-trivial sequences, such as alternations in the Battle of the Sexes game. Second, the richer set of strategies allows sophisticated strategic behavior, which not only incorporates punishments and triggers, but also *anticipation* of punishments and triggers. As a result, the threat of punishment will drive a selfish player to conform to cooperation in the Prisoner's Dilemma game and to conciliation in the Chicken game.

3 Simulations and Goodness of Fit

According to the thought experiment, players play an infinitely-repeated game with *perfect monitoring* and *complete information*. Similar to the framework of Hanaki, Sethi, Erev, and Peterhansl (2005), we allow for two phases of learning. In the pre-experimental phase, pairs of players engage in a lengthy process of learning in a fixed matching protocol. Each simulation is broken up into epochs of 100 periods. The simulation runs until the average epoch payoff of the pair has not changed by more than 0.01 from the previous epoch (in terms of Euclidean distance) in 20 consecutive epochs. Convergence of the average payoff of a pair marks the end of the pre-experimental phase (details on convergence are included in Appendix B).[6] The pre-experimental phase is used to develop the initial attractions that will be used in the experimental phase. In the first period of the experimental phase, players are randomly rematched. Afterwards, pairs stay matched for 100

[5]Cooperation in the Prisoner's Dilemma cannot be taught, even if the adaptive player is using a pure reinforcement learning model. For example, a reinforcement learner matched against a "Tit-For-Tat" strategist will, eventually, have attractions that favor defection.

[6]The maximum number of periods for the pre-experimental stage was set at 50,000.

periods.[7] The application of two phases serves as a partition between the knowledge subjects bring to the experimental laboratory (knowledge accrued from subjects' different experiences, maybe, due to learning transferred from different games or due to introspection) and the actual game-play within the laboratory.

We consider a simple goodness-of-fit measure to determine how far the predictions of a given model are from the experimental data. In particular, we compare the average payoffs over the last 10 periods of the experimental phase to the average payoffs over the last 10 periods of the experimental data. To calculate the measure, we first discretize the set of possible payoffs by using the following transformation:

$$D(\pi) = \varepsilon \left\lfloor \frac{\pi}{\varepsilon} \right\rceil,$$

where π is the payoff, ε is the accuracy of the discretization and $D(\pi)$ denotes the transformed payoff. Note that the symbolic function $\lfloor \cdot \rceil$ rounds the fraction to the nearest integer. For example, if $\varepsilon = 0.5$, then the payoff pair $(\pi_1, \pi_2) = (2.2, 3.7)$ would be transformed to $(D(\pi_1), D(\pi_2)) = (2, 3.5)$. We then construct a vector (one for each model) consisting of the relative frequency of each of the transformed payoffs given some ε. We do the same for the experimental data. To determine how far the predictions of each model are from the experimental data, we calculate the Euclidean distance between the specific model's vector and the vector of the experimental data. If the predictions match the experimental data perfectly, then the distance will have a value of 0.[8] The maximum value of distance is $\sqrt{2}$ for each game. This value is attained if only one payoff is predicted by the model, only one payoff is observed in the experiment, and the two payoffs are different. Crucially, for a given model and discretization parameter ε, we define the *best goodness of fit* model as the one whose parameter values minimize the sum of Euclidean distances across the four games studied.

[7] Recall that in the experiments of Mathevet and Romero (2012), whose dataset we use to validate the predictions of the models, subjects were instructed that the continuation probability for an additional period was 0.99; in expectation, the length of game-play is 100 periods, which matches the length of the computational simulations in the experimental phase.

[8] There are many standard statistical procedures for comparing models. For example, the Mean Squared Distance (MSD) criterion calculates the average squared distance between the predicted choice proportion and the observed choice proportion in the relevant game. The lower the MSD, the closer the choice predictions of the model to the observed behaviors. Our measure is similar to the MSD criterion albeit the MSD criterion requires discrete payoff choices; thus, we first need to discretize the set of possible payoffs before using the Euclidean metric. Another popular one is the Akaike Information Criterion, which penalizes theories with more degrees of freedom. Given that the comparison in this study is *within* models and not, across models, the simple criterion chosen suffices.

4 Action Learning

We implement the proposed methodology by building on three proven action learning models: a self-tuning Experience Weighted Attraction model (Ho, Camerer, and Chong (2007)), a γ-Weighted Beliefs model (Cheung and Friedman (1997)), and an Inertia, Sampling and Weighting model (Erev, Ert, and Roth (2010)).[9] Henceforth, for brevity, we refer to the three models by their acronym: STEWA, γ-WB and I-SAW, respectively. In Table 2, we summarize the basic characteristics of each model, describe the parameters used in each model and highlight the modifications implemented to facilitate operability of the models with repeated-game strategies (a detailed description of each model can be found in Appendix C).

The STEWA model builds on the Experience Weighted Attraction model of Camerer and Ho (1999) to address criticisms that the prototype model had "too" many free parameters. The STEWA model fixes instead some parameters at plausible values and replaces others with functions of experience that self-tune. The γ-WB model is a simple belief learning model that includes Cournot best-response (Cournot (1960)) and fictitious play (Brown (1951)) as special cases. Finally, I-SAW is an instance-based model that captures behavioral regularities that have been observed in studies of individual decisions from experience and Market Entry games.[10] In each period, a player enters one of three response modes (exploration, inertia and exploitation) with certain probabilities. The exploration mode enables a player to experiment with different actions, whereas the inertia mode repeats a player's most recent action. Finally, in the exploitation mode, a player calculates a weighted average of payoffs (based on formulated beliefs) to select the action with the highest value. All three models assume reliance on previous experiences, which has been documented to capture subjects' behavior better than models, such as reinforcement learning that do not assume memory of and/or reliance on specific experiences (see Erev, Ert, and Roth (2010)).

In Figure 2, the predictions of the three action learning models are validated with the experimental data from Mathevet and Romero (2012). We display the plots for the parameters that led to the best goodness of fit across all games.[11] The prevalent outcome in the experimental data

[9] We raise the bar of assessment high enough by using leading models, which have been quite successful in documenting subjects' behavior. For instance, Chmura, Goerg, and Selten (2012) noted recently that "the good performance of the self-tuning EWA on the individual level is remarkable" (p. 60). Moreover, the γ-WB model was documented to track players' behavior well across a multitude of games and information conditions (see Cheung and Friedman (1997)). Finally, I-SAW was the best baseline in the Market Entry Prediction Competition of Erev, Ert, and Roth (2010).

[10] The behavioral regularities documented include: the payoff variability effect, high sensitivity to forgone payoffs, underweighting of rare events, strong inertia and surprise-triggers-change, the very recent effect, individual differences (see Erev, Ert, and Roth (2010) and Erev and Haruvy (2013)).

[11] To determine the parameters of the model that fit the data best, we ran a grid search over different parameter values. The parameters for each of the action learning models are described in Table 2. For the single parameter λ in the action STEWA model, we performed a grid search over $\lambda \in \{0.5, 1, 1.5, 2, 2.5, 3, 3.5, 4, 4.5, 5, 5.5, 6\}$. The action γ-WB model has two parameters: γ and λ. We performed a grid search over $\gamma \in \{0.1, 0.3, 0.5, 0.7, 0.9\}$ and $\lambda \in \{1, 1.5, 2, 2.5, 3, 3.5, 4, 4.5, 5\}$. Finally, the action I-SAW model has six parameters: $\varepsilon, \pi, \omega, \rho, p_A$, and μ. To

	Self-Tuning Experience Weighted Attraction	**γ-Weighted Beliefs**	**Inertia, Sampling and Weighting**
Reference	Ho, Camerer, and Chong (2007)	Cheung and Friedman (1997)	Erev, Ert, and Roth (2010)
Acronym	STEWA	γ-WB	I-SAW
Description	This model is a hybridized workhorse of adaptive learning combining elements of belief learning and reinforcement learning. Each action in a player's action set has an attaraction, which is updated at the end of each period. An action is selected via the logistic choice function, where actions with higher attractions are disproportionately more likely to be selected.	Initially, a player's beliefs on the opponent's actions are formulated by discounting previous play with parameter γ. Once the beliefs are formulated, the expected payoff of each potential action is calculated. Finally, a player selects an action via the logistic choice function, where actions yielding higher expected payoffs are disproportionately more likely to be selected.	This model is an instance-based model, which allows for three response modes: exploration, inertia and exploitation. In each period, a player enters one of the modes with certain probabilities. In the exploration mode, an action is randomly chosen. In the inertia mode, the last action is repeated with some probability that depends on surprise. Finally, in the exploitation mode, a player calculates a weighted average of payoffs (based on formulated beliefs) to select the action with the highest value.
Parameters	• λ - logistic function parameter that determines sensitivity when choosing from attractions.	• λ - logistic function parameter that determines sensitivity when choosing from attractions. • γ - discounting parameter for previous actions when determining beliefs.	• p_A - probability of choosing action A when exploring. • ε_i - probability that an agent explores in a given period. • π_i - used to determine the probability that an agent has inertia in a given period. • μ_i - number of samples taken by an agent from the history when determining the sample mean of the estimated subjective value. • ρ_i - the probability that an agent takes his opponent's last action when determining the sample mean of the estimated subjective value. • ω_i - the fraction of weight put on the grand mean in the estimated subjective value.
Proposed Modifications	• Replace actions with strategies. • Use pre-experimental stage to develop initial attractions.	• Replace actions with strategies. • Use pre-experimental stage to develop initial attractions.	• Replace actions with strategies. • Use pre-experimental stage to develop initial attractions. • Endogenize probability of inertia, which reduces model by one parameter (π). • Assume that a player chooses randomly over all strategies during exploration, which reduces model by one parameter (p_A).

Table 2: Models

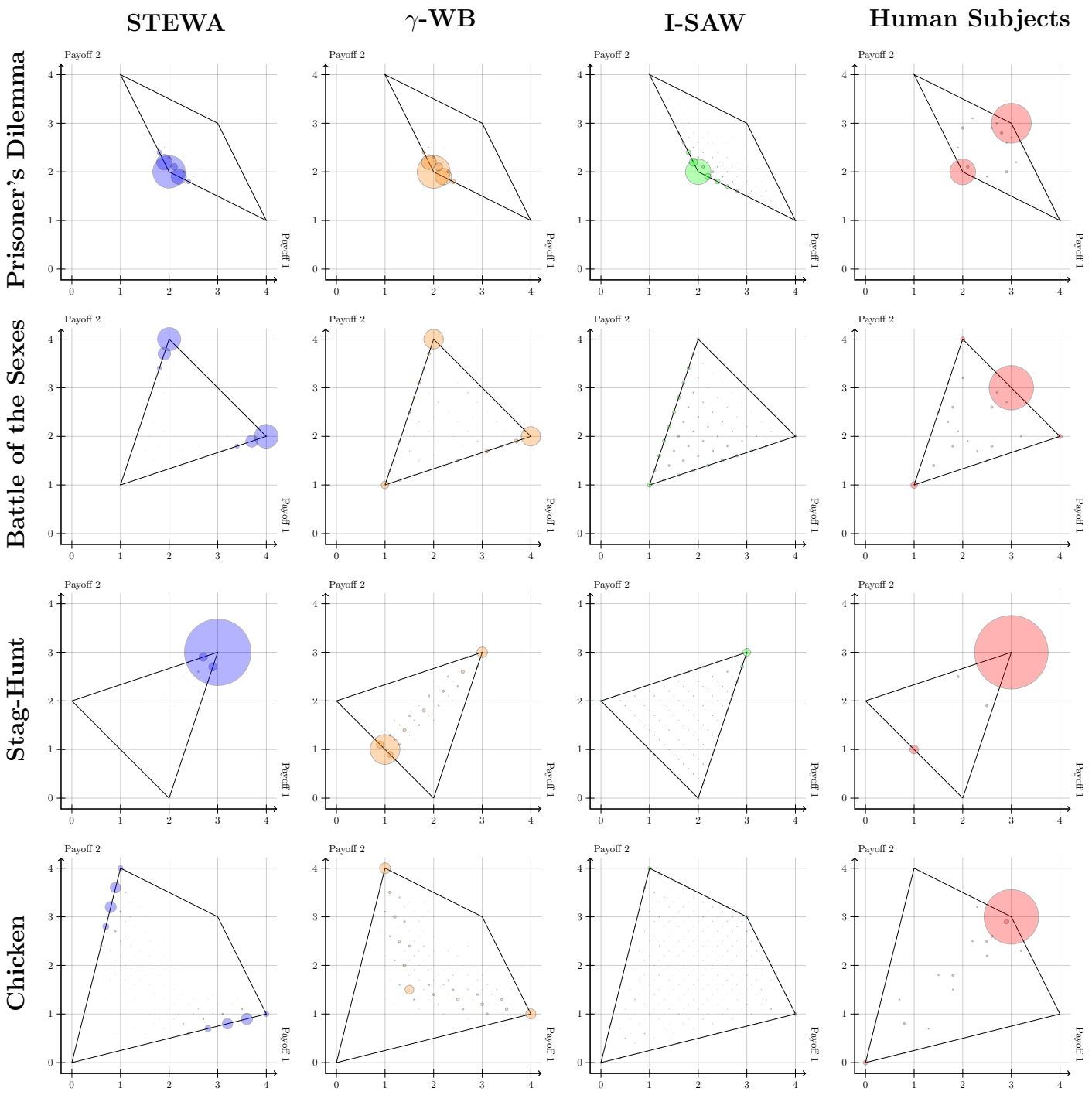


Figure 2: Action Learning Models in the Experimental Phase & Experimental Data

Notes: We validate the predictions of the three action learning models in the experimental phase with human data from the experiments of Mathevet and Romero (2012). The computational simulations and the experimental results with human subjects show the relative frequency of each payoff combination over the last 10 periods of game-play. The relative frequency of a payoff combination is denoted by a circle located on the coordinates that correspond to that combination; the larger the circle, the higher the (relative) frequency of that combination. We also display the set of feasible payoffs.

in the Prisoner's Dilemma game is for subjects to mutually cooperate, albeit there is also a large number of subjects who end up at the mutual defection outcome. The action learning models in Figure 2 only predict mutual defection. Furthermore, in the Battle of the Sexes game, the prevalent outcome in the experimental data is alternations between the two pure-strategy Nash equilibria. A small number of subjects end up at one of the two symmetric pure-strategy Nash equilibria and a few subjects end up at the $(1.0, 1.0)$ payoff. The action learning STEWA and γ-WB models only predict convergence to one of the two symmetric pure-strategy Nash equilibria. The I-SAW model with action learning predicts neither convergence to one of the two pure-strategy symmetric Nash equilibria nor alternations between the two pure-strategy Nash equilibria. In the Stag-Hunt game, the prevalent outcome is for subjects to converge to the payoff-dominant Nash equilibrium. Figure 2 shows that only the action learning STEWA is able to predict clearly that pairs will converge to such outcome. The other two action learning models predict that only a small number of pairs will end up at the payoff-dominant Nash equilibrium. Finally, none of the action learning models in Figure 2 is able to predict that a significant number of pairs will converge to the mutual conciliation outcome in the Chicken game. (The goodness-of-fit results for the experimental phase of the three action learning models are shown in Table 3.)

5 Strategy Learning

To simplify exposition, we start with some notation. The stage game is represented in standard strategic (normal) form. The set of players is denoted by $I = \{1, ..., n\}$. Each player $i \in I$ has an *action set* denoted by $\mathcal{A}_i$. An *action profile* $a = (a_i, a_{-i})$ consists of the action of player i, and the actions of the other players denoted by $a_{-i} = (a_1, ..., a_{i-1}, a_{i+1}, ..., a_n) \in \mathcal{A}_{-i}$. In addition, each player i has a real-valued, stage-game, payoff function $g_i : \mathcal{A} \to \mathbb{R}$, which maps every action profile $a \in \mathcal{A}$ into a payoff for i, where $\mathcal{A}$ denotes the cartesian product of the action spaces $\mathcal{A}_i$, written as $\mathcal{A} \equiv \underset{i=1}{\overset{I}{\times}} \mathcal{A}_i$. In the infinitely-repeated game with *perfect monitoring*, the stage game in each time period $t = 0, 1, ...$ is played with the action profile chosen in period t publicly observed at the end of that period. The *history* of play at time t is denoted by $h^t = (a^0, ..., a^{t-1}) \in \mathcal{A}^t$, where $a^r = (a_1^r, ..., a_n^r)$ denotes the actions taken in period r. The set of histories is given by

$$\mathcal{H} = \bigcup_{t=0}^{\infty} \mathcal{A}^t,$$

find the best goodness of fit for the action I-SAW model, we conducted a grid search over the suggested parameter range in Erev, Ert, and Roth (2010). The parameters searched over were $\varepsilon \in \{0.2, 0.24, 0.3\}$, $\pi \in \{0.4, 0.5, 0.6\}$, $\omega \in \{0.6, 0.7, 0.8\}$, $\rho \in \{0.1, 0.2, 0.3\}$, $p_A \in \{0.4, 0.5, 0.6\}$ and $\mu = 3$. The parameters that maximized goodness of fit were $\lambda = 1.5$ for STEWA; $\gamma = 0.1$ and $\lambda = 3$ for γ-WB; and $\varepsilon = 0.2, \pi = 0.6, \omega = 0.8, \rho = 0.1, p_A = 0.6$ and $\mu = 3$ for I-SAW.

where we define the initial history to the null set $\mathcal{A}^0 = \{\emptyset\}$. A *strategy* $s_i \in S_i$ for player i is, then a function $s_i : \mathcal{H} \to \mathcal{A}_i$, where the strategy space of i consists of K_i discrete strategies; that is, $S_i = \{s_i^1, s_i^2, ..., s_i^{K_i}\}$. Furthermore, denote a strategy combination of the n players except i by $s_{-i} = (s_1, ..., s_{i-1}, s_{i+1}, ..., s_n)$. Each player i has a payoff function $\pi_i : S \to \mathbb{R}$, which maps every strategy profile $s = (s_i, s_{-i}) \in S$ into a payoff for i, where S denotes the cartesian product of the strategy spaces S_i. Finally, player i's payoff in period t is denoted as $\pi_i(s_i(t), s_{-i}(t))$.

5.1 Generalized Repeated-Game Strategy Space

Our motivation to use generalizable rules is twofold. First, as highlighted in the Introduction, a large number of strategies is impractical for updating under most existing learning models. Instead, generalizable rules require a relatively small strategy set, which may implicitly encompass a larger space (see for instance, Stahl's rule learning). Second, we desire to reflect elements of bounded rationality and complexity as envisioned by Simon (1947). Bounded rationality suggests that a player may not consider all feasible strategies, but limit himself instead to less-complex strategies. We thus propose a generalized repeated-game strategy space where players' strategies are implemented by a type of finite automaton.[12] The specific type of finite automaton used here is a Moore machine. A *Moore machine* for player i, M_i, in a repeated game $G = (I, \{\mathcal{A}_i\}_{i \in I}, \{g_i\}_{i \in I})$ is a four-tuple $(Q_i,\ q_i^0,\ f_i,\ \tau_i)$ where Q_i is a finite set of internal states of which q_i^0 is specified to be the initial state, $f_i : Q_i \to \mathcal{A}_i$ is an output function that assigns an action to every state, and $\tau_i : Q_i \times \mathcal{A}_{-i} \to Q_i$ is the transition function that assigns a state to every two-tuple of state and other player's action. This approach reduces the set of theoretically possible strategies to a manageable size. To see this, note that the machines make state transitions only in response to the actions of their opponents, but not to their own actions.[13] In Figure 3, we depict a player's strategy set, which consists of one-state and two-state automata. A more detailed exposition on finite automata can be found in Appendix D.

[12] Using finite automata as the carriers of agents' strategies was first suggested by Aumann (1981). The first application originated in the work of Neyman (1985) who investigated a finitely-repeated-game model in which the pure strategies available to the agents were those that could be generated by machines utilizing no more than a certain number of states.

[13] Such automata are called *full automata*. On the other hand, *exact automata* do not have this restriction (as in Kalai and Stanford (1988)).

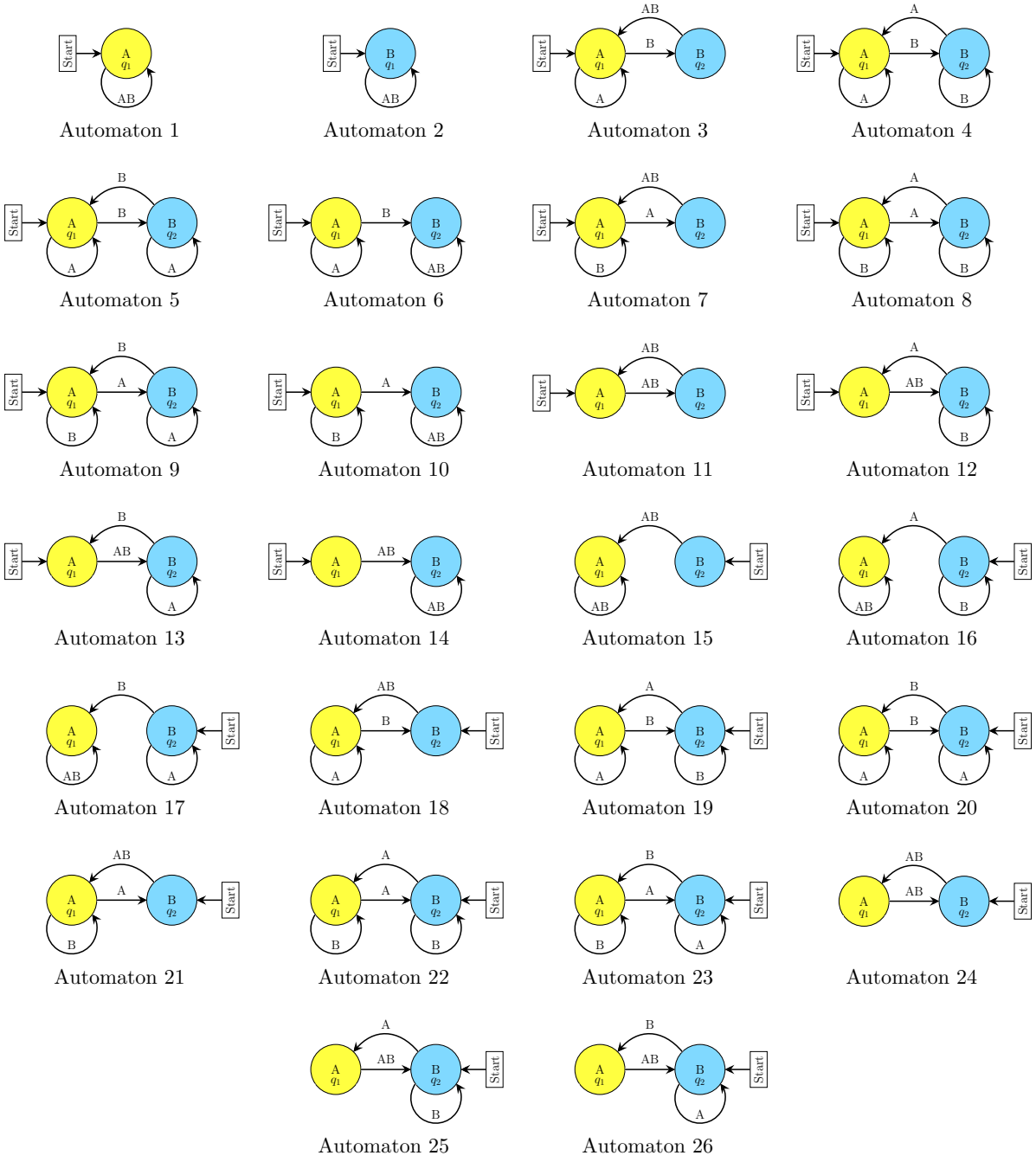


Figure 3: One-State and Two-State Automata

5.2 Mapping Between Histories and Repeated-Game Beliefs

Strategy learning models face a tough impediment in that beliefs are not directly observable. In the context of our proposed framework, we indicate next how beliefs are specified.[14] To determine the beliefs, let $h\left(t_1,t_2\right)=\left(a^{t_1},a^{t_1+1},\ldots,a^{t_2}\right)$ for $t_1\leq t_2$ be the truncated history between periods t_1 and t_2 (all inclusive) and $h\left(t,t-1\right)=\emptyset$ be the empty history. Also, let $\mathcal{T}_i(\chi)=\sum_{j=1}^{\chi}T_i(j)$ be the total number of periods at the end of player i's block χ, where $T_i(j)$ is the j^{th} block's length for player i. Then, strategy s_{-i} is consistent with $h^{\mathcal{T}_i(\chi)}$ for the last t' periods if

$$s_{-i}\left(h\left(\mathcal{T}_i(\chi)-t',\mathcal{T}_i(\chi)-t'-1+r\right)\right)=a_{-i}^{\mathcal{T}_i(\chi)-t'+r}\text{ for }r=0,\ldots,t'-1.$$

Define the fitness function $\mathcal{F}:S_{-i}\times\mathbb{N}\rightarrow[0,\mathcal{T}_i(\chi)]$ as[15]

$$\mathcal{F}\left(s_{-i},\chi\right)=\max\left\{t'|s_{-i}\text{ is consistent with }h^{\mathcal{T}_i(\chi)}\text{ for the last }t'\text{ periods}\right\}.^{16}$$

Define the belief function $\mathcal{B}:S_{-i}\times\mathbb{N}\rightarrow[0,1]$ as

$$\mathcal{B}(s_{-i},\chi)=\frac{\mathcal{F}(s_{-i},\chi)}{\sum\limits_{r\in S_{-i}}\mathcal{F}(r,\chi)},$$

which can be interpreted as player i's belief that the other player was using repeated-game strategy s_{-i} at the end of block χ. Once the beliefs over repeated-game strategies have been determined, player i can calculate his expected (foregone) payoff. The expected (forgone) payoff for player i of repeated-game strategy j over the χ^{th} block is given by

$$\mathcal{E}_i^j\left(\chi\right)=\sum_{s_{-i}\in S_{-i}}\pi_i(s_i^j,s_{-i}|_{h(s_{-i},\chi)})\cdot\mathcal{B}(s_{-i},\chi),$$

where $s_{-i}|_h$ is the continuation strategy induced by history h and

[14] Our specification of beliefs is different from the one in Chong, Camerer, and Ho (2006). In that study, a sophisticated lender holds a belief about the overall fraction of honest borrowers. Once the sophisticated lender chooses *loan*, his belief is updated in a Bayesian manner using borrower's choice probabilities. Our approach is more generalizable compared to the one advocated in that model, which is context-specific.

[15] We consider alternative fitness specifications in Appendix F.2.

[16] In the context of finite automata, let $h_i^t(\chi)$ be player i's action in the t^{th} period of block χ, and $s_{-i}=\left(Q_{-i},q_{-i}^0,f_{-i},\tau_{-i}\right)$ be a potential automaton for player $-i$. We say automaton s_{-i} is consistent with $h(\chi)$ for the last t' periods, if according to the history, it is possible that the other player played automaton s_{-i} in the last t' periods and, given player i's most recent action, the proposed automaton is in the starting state. Formally, automaton s_{-i} is consistent with $h(\chi)$ for the last t' periods if there exists some state $q^t\in Q_{-i}$ such that $h_{-i}^t(\chi)=f_{-i}\left(q^t\right)$ and $q^{t+1}=\tau_{-i}\left(q^t,h_i^t(\chi)\right)$ for all $\mathcal{T}_i(\chi)-t'+1\leq t\leq\mathcal{T}_i(\chi)$ and $q^{\mathcal{T}_i(\chi)+1}=q^0$. Note that our approach to fit action profiles onto repeated-game strategies implemented by finite automata is similar to the approach Engle-Warnick and Slonim (2006) used to infer subjects' strategies in the repeated Trust game.

$$h\left(s_{-i},\chi\right)=h\left(\mathcal{T}_i(\chi)-\mathcal{F}\left(s_{-i},\chi\right),\mathcal{T}_i(\chi)-1\right)$$

is the longest history such that s_{-i} is consistent with $h^{\mathcal{T}_i(\chi)}$.

5.3 Asynchronous Updating of Repeated-Game Strategies

Our formulation relaxes the synchronicity-of-updates constraint between the players, which is standard in the traditional action learning models; instead, players update their repeated-game strategies with the completion of a block of periods. The basic idea is that the probability of updating the strategy set depends on the expected block length, which is calculated *at the end of each period*. A long expected block length implies that a player is expecting to implement the specific repeated-game strategy for a longer time period than when the player has a short expected block length. In practice, when a player has a long expected block length, it means that the probability of updating his strategy set (or the probability of terminating the block) at the end of a period is lower than when the player has a short expected block length. The probability that player i updates his strategy set in period t, $\frac{1}{\mathcal{P}_i^t}$, is therefore determined endogenously via the expected length of the block term, $\mathcal{P}_i^t$, which is updated recursively; that is,

$$\mathcal{P}_i^t=\mathcal{P}_i^{t-1}-\frac{1}{\mathcal{P}_i^{t-1}}\frac{\left|\frac{1}{t-\underline{t}(\chi(t))}\sum_{s=\underline{t}(\chi(t))}^{t-1}g_i(a_i^s,a_{-i}^s)-\mathcal{E}_i^{s_i(\chi(t))}\left(\chi(t)\right)\right|}{\bar{g}-\underline{g}},$$

where $\underline{t}(\chi)$ is the first period of block χ, and $\chi(t)$ is the block corresponding to period t. In addition, $\bar{g}=\max_{a_1,a_2,j}g_j\left(a_1,a_2\right)$ is the highest stage-game payoff attainable by either player, and $\underline{g}=\min_{a_1,a_2,j}g_j\left(a_1,a_2\right)$ is the lowest stage-game payoff attainable to either player. The normalization by $\frac{1}{\bar{g}-\underline{g}}$ ensures that the expected block length is invariant to affine transformations of the stage-game payoffs. The variable $\mathcal{P}_i^t$ begins with an initial value $\mathcal{P}_i^0$. This prior value can be thought of as reflecting pre-game experience, either, due to learning transferred from other games, or due to (publicly) available information. The law of motion of the expected block length depends on the absolute difference between the *actual* average payoff thus far in the block and the *expected* payoff of strategy s_i. The expected payoff for player i, $\mathcal{E}_i^{s_i(\chi(t))}\left(\chi(t)\right)$, is the average payoff that player i expects (anticipates) to receive during block $\chi(t)$ and is calculated at the beginning of the block.[17] The difference between actual and expected payoff is thus a proxy for (outcome-based) surprise. If a player receives a payoff similar to his expectations, then surprise is low hence the probability of updating the strategy set increases by a relatively small amount. On

[17]Notice that the expected payoff is taken over the expected block length and no more. In line with Jéhiel (1995) and Jéhiel (1998), our approach assumes that players have a limited ability to forecast the future. Naturally, the average payoff is obtained over the length of the foresight.

the other hand, if a player receives a payoff that is drastically different from his expectations, then surprise is high hence the probability of updating the strategy set increases by a relative large amount. As Erev and Haruvy (2013) indicate, surprise triggers change; that is, inertia decreases in the presence of a surprising outcome.[18] In addition, we impose a qualitative control on the impact of surprise on the expected block length. Multiplying the absolute difference by $\frac{1}{\mathcal{P}_i^{t-1}}$ ensures that when the expected block length is long, surprise has a smaller impact on the expected block length than when the expected block length is short.

An alternative approach is to impose simultaneous strategy-updates for all players. We feel that such a venue, even though substantially simpler, would constitute a major shortcoming of the proposed model and its power as a behavioral model. First, such direction would imply that players would enter some sort of (possibly illegal!) contract before commencing the game. Second, such contract would need to be binding; otherwise, why would a player adhere to it? For instance, a player whose strategy has not been performing well, may have a change of heart and decide to update prematurely. Simultaneous strategy-updates would disallow such change of strategy. For these reasons, we prefer to forego simplicity in order to present a behavioral model, which is general and does not require unrealistic modeling assumptions. (Nevertheless we consider an approach with simultaneous strategy-updates in Appendix F.1 to highlight the value-added of the proposed approach with asynchronous updating of repeated-game strategies.)

6 Results

6.1 Strategy Learning Models in the Experimental Phase

The results at the end of the pre-experimental phase parallel the knowledge subjects bring to the laboratory before the *actual* game-play begins. On the other hand, the experimental phase parallels the actual game-play. In the first period of the experimental phase, players are randomly rematched. Afterwards, pairs stay matched for 100 periods. In Figure 4, the predictions of the three strategy learning models are validated with the experimental data. We display the plots for the parameters that led to the best goodness of fit across all games.[19] The results displayed

[18] Note that this gap-based abstraction can be justified from the observation that the activity of certain dopamine related neurons is correlated with the difference between the expected and actual outcomes (see Schultz, Dayan, and Montague (1997), and Caplin and Dean (2007)).

[19] The procedure for the simulations of the strategy learning models closely matches that of the action learning models presented earlier. The goodness of fit measures how close the last 10 periods of the experimental phase are to the experimental data using the same approach described in Section 3. Across all strategy learning models, the initial value of $\mathcal{P}_i^0$ is set to 100 at the beginning of both the pre-experimental and experimental phase. To determine the parameters of the model that fit the data best, we ran a grid search over different parameter values. For the one-parameter strategy STEWA model, the grid search covered $\lambda \in \{2, 2.75, 3, 3.25, 3.5, 4, 5, 6, 9, 12\}$.

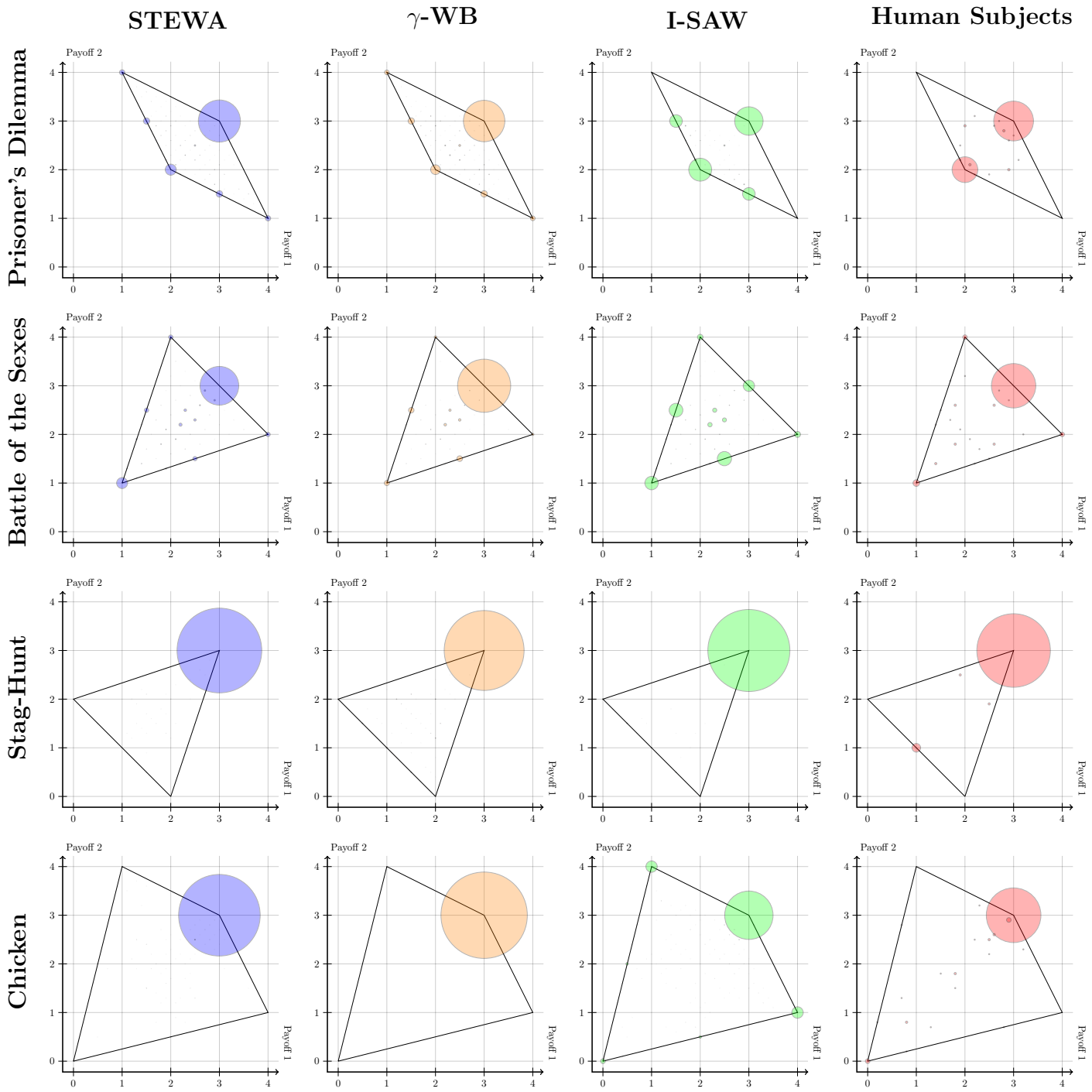


Figure 4: Strategy Learning Models in the Experimental Phase & Experimental Data

Notes: We validate the predictions of the three strategy learning models in the experimental phase with human data from the experiments of Mathevet and Romero (2012). The computational simulations and the experimental results with human subjects show the relative frequency of each payoff combination over the last 10 periods of game-play. The relative frequency of a payoff combination is denoted by a circle located on the coordinates that correspond to that combination; the larger the circle, the higher the (relative) frequency of that combination. We also display the set of feasible payoffs.

are averages taken over the *last 10 periods* of game-play. Subjects predominantly cooperate in the Prisoner's Dilemma game. However, there is also a large number of subjects who end up at the mutual defection outcome. The strategy learning models in Figure 4 are able to deliver both; that is, they predict convergence to the mutual cooperation outcome as well as convergence to the mutual defection outcome. Strategy learning models support sophisticated behaviors, such as punishments for non-conformity to the cooperative outcome that drive many pairs to select strategies that converge to the cooperative outcome. We also note that all three models predict that a small number of pairs will end up at the symmetric payoff $(3.0, 1.5)$. The latter payoff is not supported by the experimental data yet arises in the simulations when Automaton 2, which implements the "Always Defect" strategy is paired with Automaton 5, which implements the "Win-Stay, Lose-Shift" strategy. Such a pair alternates between (B, A) and (B, B). In the Battle of the Sexes game, subjects predominantly alternate between the two pure-strategy Nash equilibria. A small number of subjects ends up at one of the two symmetric pure-strategy Nash equilibria and a few subjects end up at the $(1.0, 1.0)$ payoff. These trends are supported by all three strategy learning models. The ability of the strategy learning models to predict convergence to alternations between the two pure-strategy Nash equilibria can be attributed to the richer specification of strategies. We also note that the models predict that a small number of pairs will end up at the symmetric payoff $(2.5, 1.5)$. The latter payoff occurs in the simulations when Automaton 2, which implements the "Always B" strategy is paired with Automaton 24, which switches actions every period. Such a pair alternates between (B, B) and (B, A). Even though the latter pair is not supported by the experimental data, there are points in the experimental dataset that end up quite close. In the Stag-Hunt game, the prevalent outcome is for subjects to converge to the payoff-dominant Nash equilibrium. All three strategy learning models are able to make crisp predictions on the convergence to the payoff-dominant Nash equilibrium as shown in Figure 4. In the Chicken game subjects converge to the mutual conciliation outcome, which is also predicted by the three strategy learning models.

The goodness-of-fit results for the experimental phase are shown in Table 3. The results displayed pertain the parameter values that minimize the total Euclidean distance across all four games. The table shows that the strategy learning models approximate subjects' behavior substantially better than their respective action learning models.[20] In particular, all three strategy

For the strategy γ-WB model with parameters γ and λ, the grid search covered $\gamma \in \{0.6, 0.7, 0.75, 0.8, 0.9\}$ and $\lambda \in \{12, 15, 18, 21, 22, 23, 23.5, 24, 24.5, 25, 26, 27, 30\}$. Finally, as described in Table 2, the strategy I-SAW model has only four parameters relative to the six parameters of the action I-SAW model. The grid search was therefore performed over $\varepsilon_i \in \{0, 0.0005, 0.001\}$, $\rho_i \in \{.7, .9, 1\}$, $\omega \in \{0.1, 0.3, .5, 0.7\}$ and $\mu_i = 3$. The parameters that maximized goodness of fit for the strategy learning models were $\lambda = 3$ for STEWA; $\gamma = 0.7$ and $\lambda = 23$ for γ-WB; and $\varepsilon = 0, \mu = 3, \omega = 0.1$ and $\rho = 1$ for I-SAW.

[20] This finding is robust to all values of ε. Figure 12 in Appendix G displays the minimized total Euclidean distance for the entire range of ε across the four games.

learning models do unequivocally better across all four games relative to their action learning counterparts.

	STEWA		γ-WB		I-SAW	
Game	**Action**	**Strategy**	**Action**	**Strategy**	**Action**	**Strategy**
Prisoner's Dilemma	0.532	0.228	0.535	0.237	0.486	0.257
Battle of the Sexes	0.642	0.126	0.580	0.153	0.531	0.455
Stag-Hunt	0.191	0.165	0.782	0.122	0.614	0.140
Chicken	0.698	0.326	0.679	0.381	0.773	0.221
Total	2.064	0.846	2.575	0.893	2.404	1.073

Table 3: Strategy Learning vs. Action Learning

Notes: We compare the three models with repeated-game strategies and the respective models with action learning to the experimental data for an $\varepsilon = 0.5$. The columns indicate the Euclidean distance between the experimental data and the respective predictions of the three models across each game. The red columns indicate the distance between the experimental data and the predictions of the action learning models, whereas the blue columns indicate the distance between the experimental data and the predictions of the strategy learning models. If the predictions matched the data perfectly, then the distance in a game would have been 0. Note that to calculate each model's total Euclidean distance, we located the parameters that led to the best goodness of fit (minimum total Euclidean distance) across all four games.

6.2 Importance of Pre-Experimental Phase

One may wonder whether developing initial attractions via the pre-experimental phase is important in driving the results highlighted in the experimental phase. We examine the necessity of the pre-experimental phase in this subsection. In particular, we run a baseline experimental phase in which players do not participate in the pre-experimental phase, but rather start the experimental phase without any experience. The results are displayed in Table 4. The first row is reproduced from Table 3 and indicates the total Euclidean distance between the experimental data and the predictions of the strategy learning models for the last 10 periods. The second row indicates the total Euclidean distance between the experimental data and the predictions of the strategy learning models when there is no pre-experimental phase. All results are based on a discretization parameter $\varepsilon = 0.5$. The total Euclidean distance without the pre-experimental phase is 1.669, 1.603, and 1.292 for STEWA, γ-WB, and I-SAW, respectively. On the other hand, the total Eu-

clidean distance with the pre-experimental phase is 0.846, 0.893 and 1.073 for STEWA, γ-WB, and I-SAW, respectively. Therefore, in the simulations without the pre-experimental phase, the models' fit is relatively poorer, which necessitates the importance of developing initial attractions via the pre-experimental phase.

	STEWA	γ-WB	I-SAW
With Pre-Experimental Phase	0.846	0.893	1.073
Without Pre-Experimental Phase	1.669	1.603	1.292

Table 4: Effect of Pre-Experimental Phase on Fit

Notes: The columns indicate the total Euclidean distance between the laboratory experimental data and the respective predictions of the models in the experimental phase across all four games. The first row is reproduced from Table 3 and indicates the total Euclidean distance between the experimental data and the predictions of the strategy learning models for the last 10 periods. The second row indicates the total Euclidean distance between the experimental data and the predictions of the strategy learning models when there is no pre-experimental phase. All results are based on a discretization parameter $\varepsilon = 0.5$. If the predictions matched the data perfectly, then the distance would have been 0.

6.3 Inferred Rules of Behavior

The extension from actions to a simple class of repeated-game strategies improves significantly the predictions of the models in the games studied. It is also informative to compare the repeated-game strategies predicted by the models with the inferred repeated-game strategies in the experimental data. However, inferring repeated-game strategies either from the simulated or experimental data is inhibited by two serious hurdles. First, the set of possible strategies in repeated games is infinite. Second, only one finite history is observed and therefore no information is derived with respect to what would have been played under other histories. Identifying ex ante a subset of repeated-game strategies overcomes the first problem. To overcome the second problem we use a simple approach. We focus on specific rules of behavior to see what percentage of the simulated (actual) histories could have been generated by the specific rule over the last 10 periods of the interaction in the experimental phase (experiments).[21] Given that our focus is on the last 10

[21]Previous studies overcame the second problem using either the method of strategy elicitation (Selten, Mitzkewitz, and Uhlich (1997) and Dal Bó and Fréchette (2013)) or direct inference (Engle-Warnick and Slonim (2006) and Aoyagi and Fréchette (2009)). These methods are not applicable here. For one, strategies were not elicited in the experiments of Mathevet and Romero (2012). Furthermore, the continuation probability in the experiments of Mathevet and Romero (2012) was too high to allow for direct inference in the context suggested, for instance, by

periods of the interaction, rather than focusing on the 26 one-state and two-state automata, we consider the 15 corresponding rules of behavior. A *rule of behavior* is essentially an automaton without information about the starting state; that is, a three-tuple $(Q_i,\ f_i,\ \tau_i)$ where Q_i, f_i and τ_i are defined as before.[22] The results are displayed in Figure 5. The names of the rules of behavior are either based on the conventional terminology or on a brief description of the behavior. Recall that (A, A) is the cooperative outcome in all games except the Battle of the Sexes game; in the latter game, alternations between the two pure-strategy Nash equilibria ((A, B) and (B, A)) is the cooperative outcome.

It is important to note the remarkable ability of the rules of behavior to fit the experimental data. A single rule of behavior is able to explain the play of 80%, 62%, 95%, and 79% of the subjects in the Prisoner's Dilemma, Battle of the Sexes, Stag-Hunt and Chicken experiments, respectively. This provides strong evidence to support the claim that subjects are playing strategies leading to histories similar to the histories generated by the finite automata. In addition, we also observe a clear pattern between the simulated data and the experimental data. Rules of behavior that are able to explain a high percentage of the simulated data, are also able to explain a high percentage of the experimental data. Similarly, rules of behavior that explain a low percentage of the simulated data explain a low percentage of the experimental data. While this does not ensure that the inferred rules of behavior used by human subjects are the same as those inferred in the simulations, it does provide additional evidence that the rules that emerge in the simulations capture well the behavior of subjects in the laboratory.

Dal Bó and Fréchette (2013) resolved the issue of inferring the repeated-game strategies of the experimental subjects in the infinitely-repeated Prisoner's Dilemma game by asking them to design directly repeated-game strategies. The constructed repeated-game strategy of each subject would then be deployed to play the game in lieu of himself. Dal Bó and Fréchette found that subjects chose common cooperative repeated-game strategies, such as "Tit-For-Tat" and "Grim-Trigger." Looking at Figure 5, we see that in the Prisoner's Dilemma game, the two rules of behavior that attained the highest percentages in the experimental data were (4,19) and (6), each rule with 80%. The first rule of behavior implements a "Tit-For-Tat" strategy and the second rule implements the "Grim-Trigger" strategy. The same two rules are prevalent in the three strategy learning models. The rule of behavior corresponding to "Tit-For-Tat" is consistent with 63%, 62% and 59% of the STEWA, γ-WB and I-SAW, respectively. Even better, the rule of behavior

Engle-Warnick and Slonim (2006). In particular, in the latter study the continuation probability was 0.8, which implies an expected 5 periods of game-play. On the other hand, the continuation probability in the experiments of Mathevet and Romero (2012) was 0.99, which implies an expected 100 periods of game-play. Consequently, unless errors are allowed, it is impractical to fit such long sequences of action profiles onto repeated-game strategies. Thus, we truncate the sequence of action profiles to retain the last 10 action profiles in the sequence.

[22] An alternative interpretation is that a rule of behavior reflects recognition of a pattern (see Spiliopoulos (2012) and Spiliopoulos (2013)).

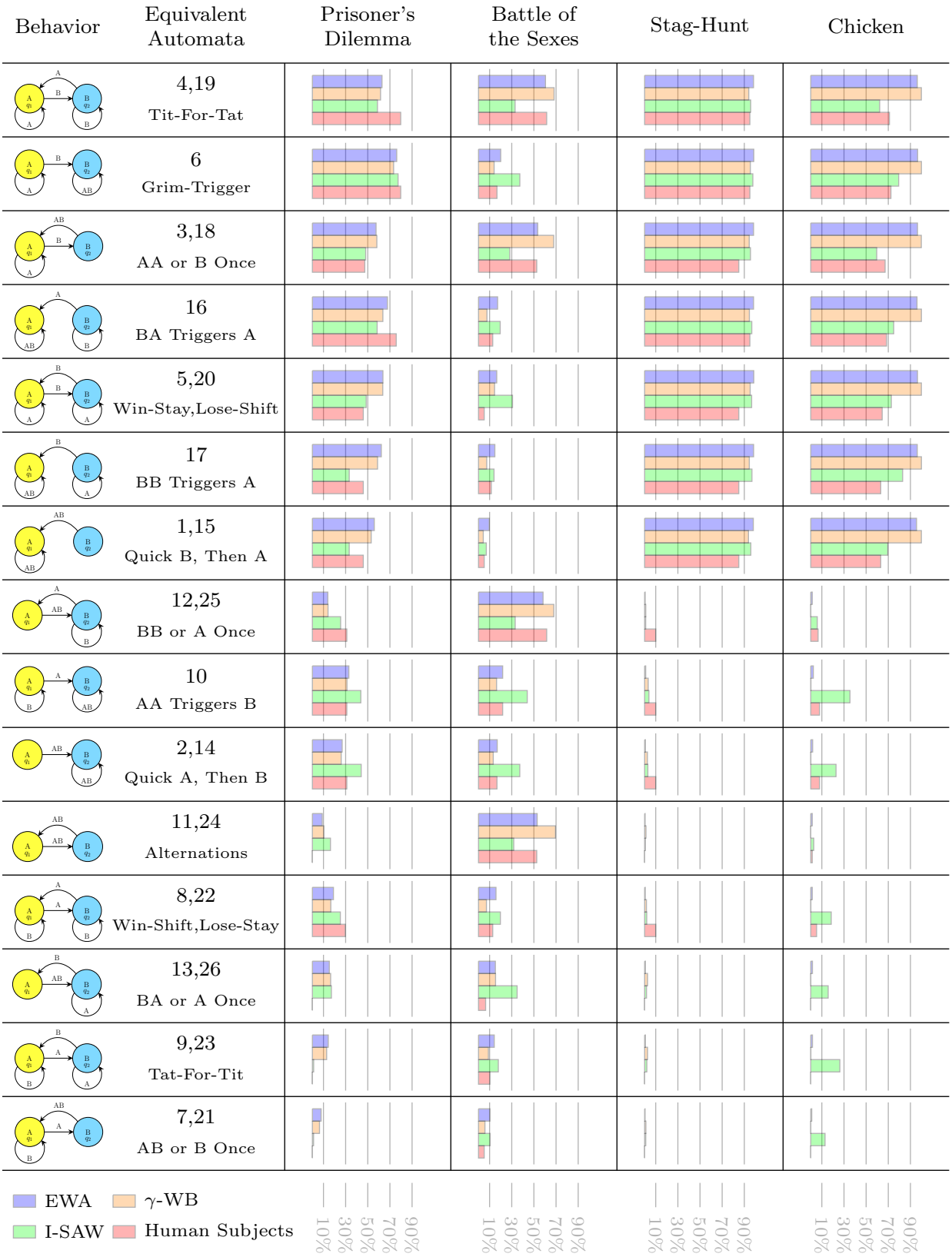


Figure 5: Inferred Rules of Behavior

Notes: The first column indicates the rule of behavior, whereas the second column indicates equivalent automata in the sense that they implement the same underlying behavior (as displayed in Figure 3) and either the rule's name or a brief description of the rule. Finally, the remaining four columns (one for each game) show with bar charts the percentage of the simulated (actual) histories that could have been generated by the specific rule over the last 10 periods of the interaction in the experimental phase (experiments). The rules of behavior are sorted in order of prevalence in the experimental data across all four games.

corresponding to "Grim-Trigger" is consistent with 76%, 73% and 77% of the STEWA, γ-WB and I-SAW, respectively. These results provide further evidence that the simulations are capturing rules of behavior similar to those applied in the laboratory. In the Battle of the Sexes game, the four most prevalent rules of behavior in the experimental data are: (3,18), (4,19), (11,14) and (12,25). The STEWA and γ-WB models capture well the precise same four rules of behavior, whereas the I-SAW model also displays high percentages in (6) and (10). Finally, similar to the Prisoner's Dilemma game, in the Stag-Hunt and Chicken games, rules of behavior (4,19) and (16) display the highest percentages. The strategy learning models in addition to the latter two rules of behavior also infer rules (1,15), (3,18), (5,20), (16) and (17). A common trait of all of these rules is that they are cooperative.

7 Conclusion

We propose a methodology in order to facilitate operability of belief learning models with repeated-game strategies. The methodology proposed is generalizable to a broad class of repeated games. We implement it by building on three proven action learning models: a self-tuning Experience Weighted Attraction model, a γ-Weighted Beliefs model, and an Inertia, Sampling and Weighting model. Additionally, their predictions with repeated-game strategies are validated with data from experiments with human subjects across four, symmetric 2×2 games: Prisoner's Dilemma, Battle of the Sexes, Stag-Hunt, and Chicken. The models with repeated-game strategies approximate subjects' behavior substantially better than their respective action learning models. Furthermore, we find that inferred rules of behavior in the experimental data coincide with those inferred in the strategy learning models. More specifically, in the Prisoner's Dilemma, Stag-Hunt, and Chicken, cooperative rules of behavior "Grim-Trigger" and "Tit-For-Tat" emerge as the most prevalent in the experimental dataset and the simulations, whereas in the Battle of the Sexes, cooperative rules implementing alternation between the two pure-strategy Nash equilibria emerge as the most prevalent in the experimental dataset and the simulations.

Ideally, the success of the proposed methodology will have to be evaluated across two important dimensions. First, it should be tested across a much broader array of games and models. Second, it should be tested under more complex environments with less tight strategy-complexity constraints. In this study, our focus has been capturing subjects' behavior in simple games, which required incorporating elements of bounded rationality via tight complexity constraints. However, there exists a plethora of situations where the agents' strategies might be more complicated. In such a case, our methodology would need to be revised to allow the use of automata carrying more than two states in order to capture more sophisticated strategies. It is important to highlight that as

the number of repeated-game strategies considered increases, the weight placed on a particular strategy will decrease. Consequently, for a large enough set of repeated-game strategies, the weight for any given strategy will approach zero. However, the fitness function proposed allows repeated-game strategies with similar characteristics to be *grouped together*, and though the weight on each strategy will indeed be small, the weight on the class of strategies will still be significant. For example, there may be a large number of repeated-game strategies that are reciprocal cooperators in the Prisoner's Dilemma game, and though the weight on each individual strategy could be small, the weight on the class of reciprocal cooperating strategies would still be large enough to make a player want to continue cooperating. Having said this, we do acknowledge that a fruitful direction for future research would be to reduce the centrality of finite automata as the carriers of agents' strategies.[23] Finally, another direction for future research would be to allow automata to commit errors. In this study, we assumed that agents' strategies were implemented by error-free automata. Agents, in real life, engage in actions that are constrained by the limitations of human nature and the surrounding environment. Thus, it would be interesting to test the susceptibility of the results to small amounts of perception and implementation errors.

[23]For instance, the cut-off strategies of Friedman and Oprea (2012) cannot be operationalized by finite automata.

References

Aoyagi, Masaki, and Guillaume R. Fréchette. "Collusion as Public Monitoring Becomes Noisy: Experimental Evidence." *Journal of Economic Theory* 144, 3: (2009) 1135–65.

Arifovic, Jasmina, Richard McKelvey, and Svetlana Pevnitskaya. "An Initial Implementation of the Turing Tournament to Learning in Repeated Two Person Games." *Games and Economic Behavior* 57: (2006) 93–122.

Aumann, Robert. "Survey of Repeated Games." In *Essays in Game Theory and Mathematical Economics in Honor of Oscar Morgestern*. Manheim: Bibliographisches Institut, 1981.

Brown, George W. "Iterated Solution of Games by Fictitious Play." In *Activity Analysis of Production and Allocation*, edited by Tjalling C. Koopmans. New York: Wiley, 1951.

Bush, Robert R., and Frederick Mosteller. "A Mathematical Model for Simple Learning." *Psychological Review* 58: (1951) 313–23.

Camerer, Colin F., and Teck-Hua Ho. "Experience Weighted Attraction Learning in Normal Form Games." *Econometrica* 67: (1999) 827–63.

Camerer, Colin F., Teck-Hua Ho, and Juin-Kuan Chong. "Sophisticated EWA Learning and Strategic Teaching in Repeated Games." *Journal of Economic Theory* 104: (2002) 137–88.

Caplin, Andrew, and Mark Dean. "The Neuroeconomic Theory of Learning." *American Economic Review Papers and Proceedings* 97, 2: (2007) 148–52.

Cheung, Yin-Wong, and Daniel Friedman. "Individual Learning in Normal Form Games: Some Laboratory Results." *Games and Economic Behavior* 19: (1997) 46–76.

Chmura, Thorste, Sebastian J. Goerg, and Reinhard Selten. "Learning in Repeated 2x2 Games." *Games and Economic Behavior* 76: (2012) 44–73.

Chong, Juin-Kuan, Colin F. Camerer, and Teck-Hua Ho. "A Learning-Based Model of Repeated Games with Incomplete Information." *Games and Economic Behavior* 55: (2006) 340–71.

Cooper, David, and John Kagel. "Lessons Learned: Generalizing Learning Across Games." *American Economic Review* 93: (2003) 202–07.

Costa-Gomes, Miguel, Vincent Crawford, and Bruno Broseta. "Cognition and Behavior in Normal-Form Games: An Experimental Study." *Econometrica* 69: (2001) 1193–1237.

Cournot, Augustine. *Recherches sur les Principles Mathematiques de la Theorie des Richesses*. London: Haffner, 1960.

Dal Bó, Pedro, and Guillaume R. Fréchette. "Strategy Choice in the Infinitely Repeated Prisoner's Dilemma.", 2013. Working Paper.

Ellison, Glenn. "Learning from Personal Experience: One Rational Guy and the Justification of Myopia." *Games and Economic Behavior* 19: (1997) 180–210.

Engle-Warnick, Jim, and Robert Slonim. "Inferring Repeated-Game Strategies from Actions: Evidence from Trust Game Experiments." *Economic Theory* 28, 3: (2006) 603–32.

Erev, Ido, Eyal Ert, and Alvin E. Roth. "A Choice Prediction Competition for Market Entry Games: An Introduction." *Games* 1: (2010) 117–36.

Erev, Ido, and Ernan Haruvy. "Generality and the Role of Descriptive Learning Models." *Journal of Mathematical Psychology* 49, 5: (2005) 357–71.

———. "Learning and the Economics of Small Decisions." In *The Handbook of Experimental Economics, Vol. 2*, edited by John H. Kagel, and Alvin E. Roth, Princeton University Press, 2013.

Erev, Ido, and Alvin E. Roth. "Predicting How People Play Games: Reinforcement Learning in Experimental Games with Unique, Mixed Strategy Equilibria." *American Economic Review* 88: (1998) 848–81.

Estes, William K., and Clete J. Burke. "A Theory of Stimulus Variability in Learning." *Psychological Review* 60: (1953) 276–86.

Fischbacher, Urs. "z-Tree: Zurich toolbox for ready-made economic experiments." *Experimental Economics* 10, 2: (2007) 171–8.

Friedman, Daniel, and Ryan Oprea. "A Continuous Dilemma." *American Economic Review* 102, 1: (2012) 337–63.

Fudenberg, Drew, and David K. Levine. "Reputation and Equilibrium Selection in Games with a Patient Player." *Econometrica* 57: (1989) 759–78.

Grosskopf, Brit, Ido Erev, and Eldad Yechiam. "Foregone with the Wind: Indirect Payoff Information and its Implications for Choice." *International Journal of Game Theory* 34, 2: (2006) 285–302.

Hanaki, Nobuyuki, Rajiv Sethi, Ido Erev, and Alexander Peterhansl. "Learning Strategies." *Journal of Economic Behavior and Organization* 56: (2005) 523–42.

Ho, Teck-Hua, Colin F. Camerer, and Juin-Kuan Chong. "Self-tuning Experience Weighted Attraction Learning in Games." *Journal of Economic Theory* 133: (2007) 177–98.

Hopkins, Ed. "Two Competing Models of How People Learn in Games." *Econometrica* 70: (2002) 2141–66.

Hyndman, Kyle, Erkut Y. Ozbay, Andrew Schotter, and Wolf Ze'ev Ehrblatt. "Convergence: An Experimental Study of Teaching and Learning in Repeated Games." *Journal of the European Economic Association* 10, 3: (2012) 573–604.

Ianni, Antonella. "Learning Strict Nash Equilibria Through Reinforcement." *Journal of Mathematical Economics* (forthcoming).

Jéhiel, Philippe. "Limited Horizon Forecast in epeated Alternate Games." *Journal of Economic Theory* 67: (1995) 497–519.

———. "Learning to Play Limited Forecast Equilibria." *Games and Economic Behavior* 22: (1998) 274–98.

———. "Limited Foresight May Force Cooperation." *Review of Economic Studies* 68: (2001) 369–91.

Kalai, Ehud, and William Stanford. "Finite Rationality and Interpersonal Complexity in Repeated Games." *Econometrica* 56: (1988) 397–410.

Laslier, Jean-Francois, Richard Topol, and Bernard Walliser. "A Behavioral Learning Process in Games." *Games and Economic Behavior* 37: (2001) 340–66.

Laslier, Jean-Francois, and Bernard Walliser. "A Reinforcement Learning Process in Extesive Form Games." *International Journal of Game Theory* 33: (2005) 219–27.

Mathevet, Laurent, and Julian Romero. "Predictive Repeated Game Theory: Measures and Experiments.", 2012. Mimeo.

McKelvey, Richard, and Thomas R. Palfrey. "Playing in the Dark: Information, Learning, and Coordination in Repeated Games.", 2001. Mimeo.

Mookherjee, Dilip, and Barry Sopher. "Learning Behavior in an Experimental Matching Pennies Game." *Games and Economic Behavior* 7, 1: (1994) 62–91.

Moore, Edward F. "Gedanken Experiments on Sequential Machines." *Annals of Mathematical Studies* 34: (1956) 129–53.

Neyman, Abraham. "Bounded Complexity Justifies Cooperation in the Finitely Repeated Prisoner's Dilemma." *Economics Letters* 19: (1985) 227–229.

Nyarko, Yaw, and Andrew Schotter. "An Experimental Study of Belief Learning Using Elicited Beliefs." *Econometrica* 70, 3: (2002) 971–1005.

Rabin, Matthew. "An Approach to Incorporating Psychology into Economics." *American Economic Review* 103, 3: (2013) 617–22.

Rapoport, Amnon, and Ido Erev. "Magic, Reinforcement Learning and Coordination in a Market Entry Game." *Games and Economic Behavior* 23: (1998) 146–75.

Rapoport, Anatol, and Albert M. Chammah. *Prisoner's Dilemma*. Ann Arbor: University of Michigan Press, 1965.

Rapoport, Anatol, Melvin J. Guyer, and David G. Gordon. *The 2X2 Game*. Ann Arbor: University of Michigan Press, 1976.

Schultz, Wolfram, Peter Dayan, and P. Read Montague. "A Neural Substrate of Prediction and Reward." *Science* 275, 5306: (1997) 1596–99.

Selten, Reinhard, Michael Mitzkewitz, and Gerald R. Uhlich. "Duopoly Strategies Programmed by Experienced Players." *Econometrica* 65, 3: (1997) 517–55.

Simon, Herbert. *Administrative Behavior: A Study of Decision-Making Processes in Administrative Organizations*. The Free Press, 1947.

Spiliopoulos, Leonidas. "Pattern Recognition and Subjectie Belief Learning in a Repeated Constant-Sum Game." *Games and Economic Behavior* 75: (2012) 921–35.

———. "Beyond Fictitious Play Beliefs: Incorporating Pattern Recognition and Similarity Matching." *Games and Economic Behavior* 81: (2013) 69–85.

Stahl, O. Dale. "Evidence Based Rules and Learning in Symmetric Normal-Form Games." *International Journal of Game Theory* 28: (1999) 111–30.

Stahl, O. Dale, and Ernan Haruvy. "Between-Game Rule Learning in Dissimilar Symmetric Normal-Form Games." *Games and Economic Behavior* 74: (2012) 208–21.

Thorndike, Edward Lee. "Animal Intelligence: An Experimental Study of the Associative Process in Animals." *Psychological Review, Monograph Supplements,* 8: (1898) Chapter II.

Van Huyck, John B., Ramond C. Battalio, and Frederick W. Rankin. "Selection Dynamics and Adaptive Behavior Without Much Information." *Economic Theory* 33, 1: (2007) 53–65.

A Experimental Data

Mathevet and Romero (2012) provide experimental data on the four games reported in Figure 1. The experimental sessions were run at the Vernon Smith Experimental Economics Laboratory at Purdue University. Subjects interacted on computers using an interface that was programmed with the z-Tree software (Fischbacher (2007)). Subjects' final payoffs consisted of the sum of their earnings from all periods of the experiment. With the completion of the experiment, subjects were paid in private their cash earnings. The average payoff was \$16.85. The game-play consisted of a fixed matching protocol. The continuation probability for an additional period was 0.99 and was common knowledge in all experimental sessions. The experimental data consist of 37, 39, 20 and 38 observations in the Prisoner's Dilemma game, Battle of the Sexes game, Stag-Hunt game and Chicken game, respectively. The experimental data points displayed below are the average payoffs per pair over the last 10 periods of game-play.

Points	(3.0,3.0)	(2.0,2.0)	(1.0,1.0)	(2.0,4.0)	(0.0,0.0)	(1.9,2.5)	(1.8,2.6)	(2.9,2.9)	(1.8,1.8)	(2.6,2.6)	(2.7,3.0)	(1.7,3.1)	(4.0,2.0)	(2.6,2.9)	(2.9,2.0)	(0.8,0.8)	(3.1,2.2)	(2.0,2.9)	(2.1,2.1)	(1.5,2.5)	(2.8,0.7)	(2.9,2.7)	(2.5,2.5)	(2.5,2.2)	(2.0,3.2)	(2.8,2.8)	(1.4,2.2)	(2.2,1.9)	(0.2,0.8)	(3.2,2.3)	(1.7,2.3)	(1.3,0.7)	(1.4,1.4)	(2.1,1.9)	(1.5,1.8)	Total
PD	17	11	0	0	0	1	0	0	0	0	1	0	0	1	1	0	1	1	1	0	0	0	0	0	0	1	0	1	0	0	0	0	0	0	0	37
BO	20	0	3	3	0	0	2	0	1	1	0	1	1	0	0	0	0	0	0	1	0	1	0	0	1	0	1	0	0	0	1	0	1	1	0	39
SH	17	0	2	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	20
CH	24	0	0	0	2	0	0	2	1	1	0	0	0	0	0	1	0	0	0	0	1	0	1	1	0	0	0	0	1	1	0	1	0	0	1	38

Table 5: Experimental Data

B Convergence Details

The convergence details of the strategy learning models in the pre-experimental phase are displayed in Table 6. The details are displayed for the parameters that minimized the Euclidean distance in the experimental phase of the simulations. Recall that each simulation is broken up into epochs of 100 periods. The simulation runs until the average epoch payoff of the pair has not changed by more than 0.01 from the previous epoch (in terms of Euclidean distance) in 20 consecutive epochs, or 50,000 periods, which ever occurs first. For the Prisoner's Dilemma game and the Stag-Hunt game, all of the simulations converged within 50,000 periods. In the Battle of the Sexes a significant number of the simulations did not converge within 50,000 periods. The pairs that did not converge according to the criteria had both players playing their preferred action B repeatedly in, essentially, a war of attrition, while waiting for the other player to give in. This led to an average payoff of $(1.0, 1.0)$ over the course of the epoch. However, even one deviation from this payoff over the 100-period epoch led to a payoff different enough that did not satisfy the convergence criterion. Some of the simulations of the strategy I-SAW model did not converge according to the criterion in the Chicken game. Pairs in these simulations were playing the mixed strategy Nash equilibrium, which led to relatively unstable epoch payoffs.

	STEWA		**γ-WB**		**I-SAW**	
Game	**Average**	**%NC**	**Average**	**%NC**	**Average**	**%NC**
Prisoner's Dilemma	20,750	0	14,582	0	14,332	0
Battle of the Sexes	28,459	41	21,416	15	32,979	58
Stag-Hunt	7,171	0	3,387	0	2,117	0
Chicken	8,704	3	4,044	0	23,252	42

Table 6: Details on Convergence in the Pre-experimental Phase

Notes: The two columns of each strategy learning model contain details about the average number of periods to convergence and the percentage of the simulations that did not converge within 50,000 periods. The convergence details are displayed for the parameters that minimized the Euclidean distance in the experimental phase.

C Models with Strategy Learning

We implement the proposed methodology to three proven action learning models: the self-tuning Experience Weighted Attraction model of Ho, Camerer, and Chong (2007), the γ-Weighted Beliefs model of Cheung and Friedman (1997) and the Inertia, Sampling and Weighting model of Erev, Ert, and Roth (2010). The details on the modeling assumptions are provided below.

C.1 Self-tuning Experience Weighted Attraction (STEWA)

A detailed exposition of the STEWA model with actions can be found in Ho, Camerer, and Chong (2007). Analogous to the latter model, the STEWA model with strategies also consists of two variables that are updated once an agent switches strategies. The first variable is $N_i(\chi)$, which is interpreted as the number of observation-equivalents of past experience in block χ of player i. The second variable, denoted as $A_i^j(\chi)$, indicates player i's attraction to strategy j *after* the χ^{th} block of periods. The variables $N_i(\chi)$ and $A_i^j(\chi)$ begin with some prior values, $N_i(0)$ and $A_i^j(0)$. These prior values can be thought of as reflecting pre-game experience, either due to learning transferred from different games or due to pre-play analysis. In addition, we use an indicator function $\mathbb{I}(x, y)$ that equals 1 if $x = y$ and 0 otherwise. The evolution of learning over the χ^{th} block with $\chi \geq 1$ is governed by the following rules:

$$N_i(\chi) = \phi_i(\chi) \cdot N_i(\chi - 1) + 1, \tag{1}$$

and

$$A_i^j(\chi) = \frac{\phi_i(\chi) \cdot N_i(\chi - 1) \cdot A_i^j(\chi - 1) + \mathbb{I}(s_i^j, s_i(\chi)) \cdot R_i(\chi) + \delta_i^j(\chi) \cdot \mathcal{E}_i^j(\chi)}{\phi_i(\chi) \cdot N_i(\chi - 1) + 1}. \tag{2}$$

Reinforcement Payoff

The reinforcement payoff in the proposed model, $R_i(\chi)$, is defined as the average payoff obtained by player i over the χ^{th} block,

$$R_i(\chi) = \frac{1}{T_i(\chi)} \sum_{a \in h(\chi)} g_i(a),$$

where $h(\chi)$ is the sequence of action profiles played in the χ^{th} block and $T_i(\chi)$ is the χ^{th} block's length for player i.

Expected Foregone Payoff

To calculate the forgone payoff $\mathcal{E}_i^j(\chi)$, players need to form beliefs about the current strategy of their opponent. To determine the beliefs, let $h(t_1, t_2) = \left(a^{t_1}, a^{t_1+1}, \ldots, a^{t_2}\right)$ for $t_1 \leq t_2$ be the truncated history between periods t_1 and t_2 (all inclusive). Also, let $h(t, t-1) = \emptyset$ be the empty history. Let $\mathcal{T}_i(\chi) = \sum_{j=1}^{\chi} T_i(j)$ be the total number of periods at the end of player i's block χ. Then, strategy s_{-i}

is consistent with $h^{\mathcal{T}_i(\chi)}$ for the last t' periods if

$$s_{-i}\left(h\left(\mathcal{T}_i(\chi)-t',\mathcal{T}_i(\chi)-t'-1+r\right)\right)=a_{-i}^{\mathcal{T}_i(\chi)-t'+r} \text{ for } r=0,\ldots,t'-1.$$

Define the fitness function $\mathcal{F}: S_{-i}\times\mathbb{N}\to[0,\mathcal{T}_i(\chi)]$ as

$$\mathcal{F}\left(s_{-i},\chi\right)=\max\left\{t'|s_{-i} \text{ is consistent with } h^{\mathcal{T}_i(\chi)} \text{ for the last } t' \text{ periods}\right\}.$$

Define the belief function $\mathcal{B}: S_{-i}\times\mathbb{N}\to[0,1]$ as

$$\mathcal{B}(s_{-i},\chi)=\frac{\mathcal{F}(s_{-i},\chi)}{\sum\limits_{r\in S_{-i}}\mathcal{F}(r,\chi)},$$

which can be interpreted as player i's belief that the other player was using strategy s_{-i} at the end of block χ. Therefore, the expected forgone payoff for player i of strategy j over the χ^{th} block is given by

$$\mathcal{E}_i^j\left(\chi\right)=\sum_{s_{-i}\in S_{-i}}\pi_i(s_i^j,s_{-i}|_{h(s_{-i},\chi)})\cdot\mathcal{B}(s_{-i},\chi),$$

where $s_{-i}|_h$ is the continuation strategy induced by history h and

$$h\left(s_{-i},\chi\right)=h\left(\mathcal{T}_i(\chi)-\mathcal{F}\left(s_{-i},\chi\right),\mathcal{T}_i(\chi)-1\right)$$

is the longest history such that s_{-i} is consistent with $h^{\mathcal{T}_i(\chi)}$.

The Attention Function

The attention function $\delta(\cdot)$ determines the weight placed on forgone payoffs and is represented by the following function:

$$\delta_i^j(\chi)=\begin{cases}1 & \text{if } \mathcal{E}_i^j(\chi)\geq R_i(\chi) \text{ and } s_i^j\neq s_i(\chi)\\ 0 & \text{otherwise.}\end{cases}$$

The Decay Function

The decay rate function $\phi(\cdot)$ weighs lagged attractions. The core of the $\phi_i(\cdot)$ is a "surprise index," which indicates the difference between the other player's most recent strategy and the strategies he chose in the previous blocks. The averaged belief function $\sigma: S_{-i}\times\mathbb{N}\to[0,1]$

$$\sigma(s_{-i},\chi)=\frac{1}{\chi}\sum_{j=1}^{\chi}\mathcal{B}(s_{-i},j)$$

averages the beliefs over the χ blocks that the other player chose strategy s_{-i}. The surprise index $\mathcal{S}_i(\chi)$ simply sums up the squared deviations between each averaged belief $\sigma(s_{-i},\chi)$ and the immediate belief

$\mathcal{B}(s_{-i}, \chi)$; that is,

$$\mathcal{S}_i(\chi) = \sum_{s_{-i} \in S_{-i}} (\sigma(s_{-i}, \chi) - \mathcal{B}(s_{-i}, \chi))^2.$$

Thus, the surprise index captures the degree of change of the most recent beliefs from the historical average of beliefs. Note that it varies from zero (when there is belief persistence) to two (when a player is certain that the opponent just switched to a new strategy after playing a specific strategy from the beginning). The change-detecting decay rate of the χ^{th} block is then

$$\phi_i(\chi) = 1 - \frac{1}{2}\mathcal{S}_i(\chi).$$

Therefore, when player i's beliefs are not changing, $\phi_i(\chi) = 1$; that is, the player weighs previous attractions fully. Alternatively, when player i's beliefs are changing, then $\phi_i(\chi) = 0$; that is, the player puts no weight on previous attractions.

Attractions

Attractions determine probabilities of choosing strategies. We use the logit specification to calculate the choice probability of strategy j. Thus, the probability of a player i choosing strategy j, when he updates his strategy at the beginning of block $\chi + 1$, is

$$\mathbb{P}_i^j(\chi + 1) = \frac{e^{\lambda \cdot A_i^j(\chi)}}{\sum_k^K e^{\lambda \cdot A_i^k(\chi)}}.$$

The parameter $\lambda \geq 0$ measures the sensitivity of players to attractions.

Finally, players update their strategies with the completion of a block of periods. The probability that player i updates his strategy in period t is $\frac{1}{\mathcal{P}_i^t}$ and is determined endogenously via the expected length of the block term.

C.2 γ-Weighted Beliefs Model (γ-WB)

We first review briefly the γ-WB model of Cheung and Friedman (1997) with actions. Player i's action set is $\mathcal{A}_i = \{A, B\}$. Initially, a player updates his beliefs on the opponent's actions with parameter γ. In particular, he believes that the other player (player $-i$) will play action a_{-i} *after* period t with probability,

$$b_i\left(a_{-i}, t\right) = \frac{\sum_{r=1}^t \gamma^{t-r} \mathbb{I}\left(a_{-i}^r = a_{-i}\right)}{\sum_{r=1}^t \gamma^{t-r}}.$$

He then calculates the expected payoff of action a_i:

$$E_i(a_i, t) = \sum_{a_{-i} \in \mathcal{A}_{-i}} b_i\left(a_{-i}, t\right) g_i\left(a_i, a_{-i}\right).$$

Assuming that the number of observation-equivalents of past experience is given by $N_i(0) = 1$ and $N_i(t) = \gamma \cdot N_i\left(t-1\right) + 1$, we can rewrite the above expression recursively as,

$$E_i(a_i, t) = \frac{\gamma \cdot N_i\left(t-1\right) E\left(a_i, t-1\right) + \mathbb{I}\left(a_{-i}^t = a_{-i}\right) g_i\left(a_i, a_{-i}\right)}{\gamma \cdot N_i\left(t-1\right) + 1}.$$

Finally, an action is selected via the logit specification with parameter λ; that is, the probability of choosing action a_i in period $t+1$ is

$$\mathbb{P}_i\left(a_i, t+1\right) = \frac{e^{\lambda \cdot E_i(a_i,t)}}{\sum_{a_i \in \mathcal{A}_i} e^{\lambda \cdot E_i(a_i,t)}}.$$

The γ-WB model with strategy learning can be written in the same notation as the STEWA model with strategy learning. The number of observation-equivalents of past experience starts with $N_i(0) = 1$. The initial attractions are all equal and set to the expected payoff in the game if both players mix with probability 0.5. The attractions in this model evolve according to the following two rules and parameter γ:

$$N_i\left(\chi\right) = \gamma \cdot N_i\left(\chi - 1\right) + 1, \tag{3}$$

and

$$A_i^j\left(\chi\right) = \frac{\gamma \cdot N_i\left(\chi - 1\right) \cdot A_i^j\left(\chi - 1\right) + \mathcal{E}_i^j\left(\chi\right)}{N_i\left(\chi\right)}. \tag{4}$$

This model is equivalent to the γ-WB model of Cheung and Friedman (1997) if the set of automata is restricted to the two one-state automata. Notice that if $\gamma = 0$, then the attractions simplify to the expected payoff from the last block. Also, the probability of a player i choosing strategy j, when he updates his strategy at the beginning of block $\chi + 1$, is

$$\mathbb{P}_i^j(\chi + 1) = \frac{e^{\lambda \cdot A_i^j(\chi)}}{\sum_k^K e^{\lambda \cdot A_i^k(\chi)}}.$$

Finally, similar to the STEWA model with strategy learning, players update their strategies asynchronously. In period t, a player updates his strategy with probability $\frac{1}{\mathcal{P}_i^t}$.

C.3 Inertia, Sampling and Weighting (I-SAW)

We first review the I-SAW model of Erev, Ert, and Roth (2010) with actions. I-SAW is an instance-based model, which allows for three response modes: exploration, inertia and exploitation. In each period,

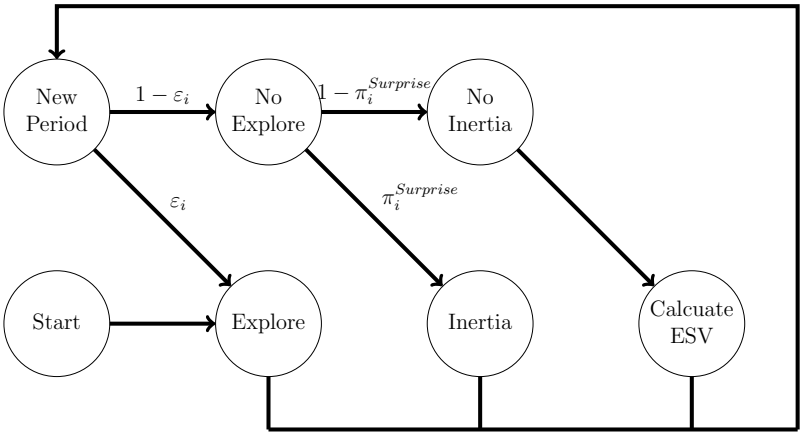


Figure 6: Schematic Description of I-SAW

Notes: I-SAW allows for three response modes: exploration, inertia and exploitation. In exploration trials, a player chooses amongst actions with some probability. Exploration occurs with probability ε_i. Inertia occurs with probability $(1-\varepsilon_i)\times\pi_i^{Surprise(t)}$. In this mode, a player repeats the last action. Exploitation occurs with probability $(1-\varepsilon_i)\times(1-\pi_i^{Surprise(t)})$. In exploitation trials, a player selects the action with the highest Estimated Subjective Value (ESV).

a player enters one of the modes with different probabilities. The I-SAW model with action learning can be summarized in the schematic in Figure 6. There are n players in the game. Each player has a set of parameters $(p_A, \varepsilon_i, \pi_i, \mu_i, \rho_i, \omega_i)$. The parameter $p_A \in [0,1]$ is the same for all agents. The other parameters are idiosyncratic with $\varepsilon_i \sim U[0,\varepsilon]$, $\pi_i \sim U[0,\pi]$, $\mu_i \sim U[0,\mu]$, $\rho_i \sim U[0,\rho]$ and $\omega_i \sim U[0,\omega]$. Dropping the subscripts for convenience, the parameters of the model are $(p_A, \varepsilon, \pi, \mu, \rho, \omega)$. Player i's action set is $\mathcal{A}_i = \{A, B\}$. Let a_i^t be the action of player i that was played in period t, where $h_i(t_1,t_2) = \left\{a_i^{t_1}, a_i^{t_1+1}, \ldots, a_i^{t_2}\right\}$ for $t_1 \leq t_2$. Similarly, let a_{-i}^t be the actions of players other than i in period t, where $h_{-i}(t_1,t_2) = \left\{a_{-i}^{t_1}, a_{-i}^{t_1+1}, \ldots, a_{-i}^{t_2}\right\}$ for $t_1 \leq t_2$. We explain next the three response modes.

Exploration

In exploration, each player chooses action A with probability p_A and action B with probability $1-p_A$. The probabilities are the same for all players.

Inertia

The decision to enter the inertia mode depends on an endogenous parameter $Surprise(t) \in [0,1]$. A player might enter the inertia mode after period 2 with probability $\pi_i^{Surprise(t)}$, where $\pi_i \in [0,1]$. The probability of inertia is low when surprise is high and vice versa.

Exploitation

In exploitation trials, an individual selects the action with the highest Estimated Subjective Value (ESV). To determine the ESV, player i randomly selects μ_i elements from $h_{-i}(0,t-1)$ with replacement; let

us call this set $M_{-i}(0, t-1)$. This set is chosen according to the following: with probability ρ_i player chooses a_{-i}^{t-1} and with probability $1-\rho_{-i}$ player chooses uniformly over $h_{-i}(0, t-1)$. The same set M_{-i} is used for each $a_i \in \mathcal{A}_i$. The sample mean for action a_i' is then defined as

$$SampleM(a_i', t) = \frac{1}{|M_{-i}(0, t-1)|} \sum_{a_{-i} \in M_{-i}(0,t-1)} g_i\left(a_i', a_{-i}\right).$$

Then, player i's ESV of action a_i' is

$$ESV\left(a_i'\right) = (1-\omega_i) \cdot SampleM(a_i', t) + \omega_i \cdot GrandM(a_i', t),$$

where ω is the weight assigned on the payoff based on the entire history ($GrandM$) and $1-\omega$ is the weight assigned on the payoff based on the sample from the history ($SampleM$). Then, the player simply chooses the a_i' that maximizes ESV (and chooses randomly in ties).

The I-SAW model with strategy learning is different from the I-SAW with action learning in two important aspects. First, given that no exploration happens in a given period, the probability of inertia is now $(1-\frac{1}{\mathcal{P}_i^t})$ rather than $\pi_i^{Surprise}$. Therefore, there is one less parameter than before. Second, the calculation of the ESV now depends on the distribution of beliefs over strategies. The grand mean is thus

$$GrandM_i\left(s^j, \chi\right) = \frac{1}{\chi} \sum_{k=1}^{\chi} \mathcal{E}_i^j(k).$$

Next, let $M_i(\chi)$ be a set of μ_i numbers drawn with replacement from $\{1, 2, \ldots, \chi\}$. Then, the sample mean is

$$SampleM_i\left(s^j, \chi\right) = \frac{1}{|M_i(\chi)|} \sum_{k \in M_i(\chi)} \mathcal{E}_i^j(k),$$

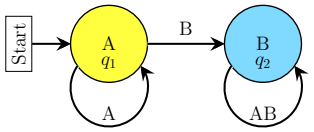
where the same set M_i is used for each $s_i \in S_i$. Finally, the ESV is calculated as

$$ESV_i\left(s^j, \chi\right) = (1-\omega_i) \cdot SampleM_i\left(s^j, \chi\right) + \omega_i \cdot GrandM\left(s^j, \chi\right).$$

The framework work as follows. In the first period, a strategy is randomly selected (i.e. exploration takes place). Concurrently, the expected payoff is calculated; this is the payoff attained if both players mix with probability 0.5. In the second period, a player explores with probability ε_i. With probability $(1-\varepsilon_i) \times (1-\frac{1}{\mathcal{P}_i^t})$ the player enters the inertia mode. If the player does not enter the inertia mode, then the block of periods is completed and a new block of periods starts off. In the beginning of the block, the player calculates the ESV of all strategies and chooses to play with the strategy that maximizes the ESV. The specific ESV becomes the new expected payoff, which is also used to calculate $\mathcal{P}_i^t$; the latter determines the probability of inertia.

D Finite Automata

A finite automaton is a mathematical model of a system with discrete inputs and outputs. The system can be in any one of a finite number of internal configurations or "states." The state of the system summarizes the information concerning past inputs that is needed to determine the behavior of the system on subsequent inputs. The specific type of finite automaton used here is a Moore machine. A *Moore machine* for player i, M_i, in a repeated game $G = (I, \{\mathcal{A}_i\}_{i \in I}, \{g_i\}_{i \in I})$ is a four-tuple $(Q_i, q_i^0, f_i, \tau_i)$ where Q_i is a finite set of internal states of which q_i^0 is specified to be the initial state, $f_i : Q_i \to \mathcal{A}_i$ is an output function that assigns an action to every state, and $\tau_i : Q_i \times \mathcal{A}_{-i} \to Q_i$ is the transition function that assigns a state to every two-tuple of state and other player's action. It is pertinent to note that the transition function depends only on the present state and the other player's action. This formalization fits the natural description of a strategy as i's plan of action in all possible circumstances that are consistent with i's plans. In contrast, the notion of a game-theoretic strategy for i requires the specification of an action for every possible history, including those that are inconsistent with i's plan of action. It is important to highlight that to formulate the game-theoretic notion of a strategy, one would *only* have to construct the transition function so that $\tau_i : Q_i \times \mathcal{A} \to Q_i$, instead of $\tau_i : Q_i \times \mathcal{A}_{-i} \to Q_i$.



$$Q_i = \{q_1, q_2\}$$
$$q_i^0 = q_1$$
$$f_i(q) = \begin{cases} A & \text{if } q = q_1 \\ B & \text{if } q = q_2 \end{cases}$$
$$\tau_i(q, a_{-i}) = \begin{cases} q_1 & \text{if } (q, a_{-i}) = (q_1, A) \\ q_2 & \text{otherwise} \end{cases}$$

Figure 7: Grim-Trigger Automaton

Notes: The vertices denote the states of the automaton, and the arcs labeled with the action of the other agent indicate the transition to the states.

In the first period, the state is q_i^0, and the automaton chooses the action $f_i(q_i^0)$. If a_{-i} is the action chosen by the other player in the first period, then the state of i's automaton changes to $\tau_i(q_i^0, a_{-i})$, and in the second period, i chooses the action dictated by f_i in that state. Then, the state changes again according to the transition function given the other agent's action. Thus, whenever the automaton is in some state q, it chooses the action $f_i(q)$, while the transition function τ_i specifies the automaton's transition from q (to a state) in response to the action taken by the other player. For example, the automaton $(Q_i, q_i^0, f_i, \tau_i)$ in Figure 7 carries out the "Grim-Trigger" strategy. In the transition diagram, a vertex denotes the internal state of the automaton with the prescribed agent's action indicated in the center, and the arcs labeled with the action of the other player indicate the transition to the states. Thus, the strategy chooses A, as long as both players have chosen A in every period in the past and chooses B otherwise.

E Evolution of Fit Within the Experimental Phase

The experimental phase consisted of a fixed-pair matching of 100 periods. In Figure 4, we displayed the plots of the models with strategy learning and the human experimental data, while averaging the *last* 10 periods of game-play. Furthermore, Table 3 indicated the Euclidean distance of the models with strategy learning for the same range of periods in the experimental phase. It is also informative to compare the models' fit to the experimental data in the later periods with the models' fit to the experimental data in the earlier periods. This way, we can determine the models' evolution of fit within the experimental phase.

In Figure 8, we plot the predictions of the three strategy learning models and the data from the experiments of Mathevet and Romero (2012). But this time, the computational simulations and the experimental results consist of averaging the *first* 10 periods of game-play. Furthermore, in Table 7, we indicate the total Euclidean distance between the laboratory experimental data and the respective predictions of the proposed models in the first 10 periods as well as the last 10 periods across all four games. Overall, the strategy learnings models in the first 10 periods of game-play do reasonably well. The total Euclidean distance across the four games is 1.622 in the STEWA model, 1.642 in the γ-WB model and 1.085 in the I-SAW model. Looking at Figure 8, we observe that the models do a fairly good job in fitting the experimental data in the Prisoner's Dilemma game where they predict mutual cooperation, and in the Stag-Hunt game where they predict the payoff-dominant Nash equilibrium. On the other hand, the STEWA and γ-WB models predict some alternations in the Battle of the Sexes game, and strong mutual conciliation in the Chicken game. Some mutual conciliation is indeed observed in the experimental data, but no alternations are observed in the Battle of the Sexes game. The latter games are clearly tougher to establish coordination from the beginning. Thus, we observe a lot of noisy behavior in the experimental data as a result of subjects' different backgrounds and abilities to internalize fully

	STEWA	γ-WB	I-SAW
Strategy Learning Models (Last 10 Periods)	0.846	0.893	1.073
Strategy Learning Models (First 10 Periods)	1.622	1.642	1.085

Table 7: Evolution of Fit within the Experimental Phase

Notes: The columns indicate the total Euclidean distance between the laboratory experimental data and the respective predictions of the models in the experimental phase across all four games. The first row is reproduced from Table 3 and indicates the Euclidean distance between the experimental data and the predictions of the strategy learning models for the last 10 periods. The second row indicates the Euclidean distance between the experimental data and the predictions of the strategy learning models for the first 10 periods. All results are based on a discretization parameter $\varepsilon = 0.5$. If the predictions matched the data perfectly, then the distance would have been 0.

the instructions and/or the game structure from the start. Such noisy behavior is hard to capture in the proposed models. However, the plots of Figure 4 confirm that as time goes by, learning takes place, which limits the amount of noise in subjects' behavior, and thus enables the proposed models to approximate subjects' behavior at the end remarkably well.

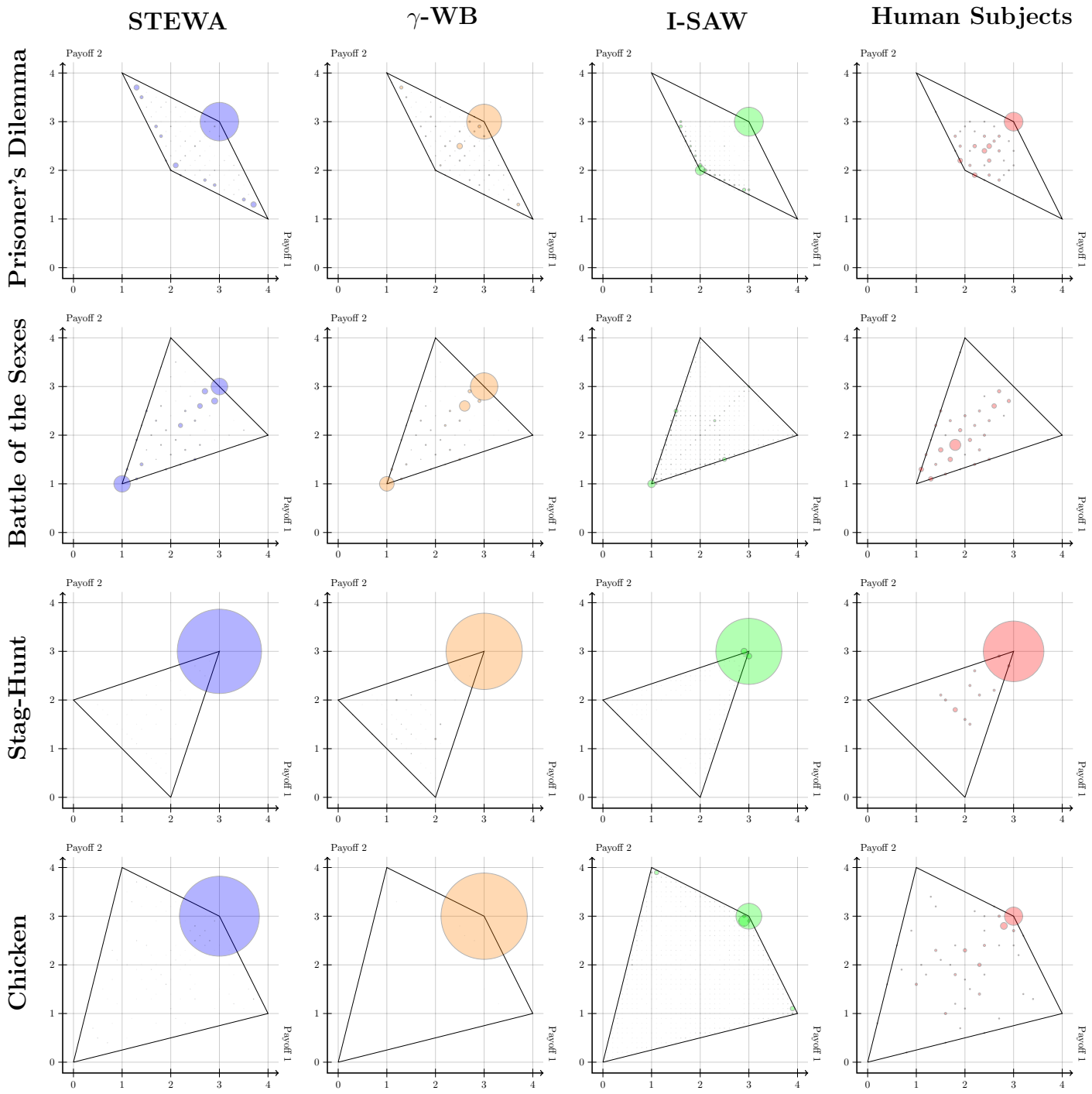


Figure 8: Strategy Learning Models & Experimental Data - First 10 Periods

Notes: We validate the predictions of the strategy learning models in the experimental phase with human data from the experiments of Mathevet and Romero (2012). The computational simulations and the experimental results with human subjects consist of averaging the first 10 periods of game-play.

F Alternative Specifications

F.1 Asynchronous Updating vs. Synchronous Updating

The third rule proposed requires that players update their repeated-game strategies with the completion of a block of periods. Furthermore, the length of a block is determined endogenously and is based on the surprise-triggers-change regularity identified by Erev and Haruvy (2013). Alternatively, our framework could dictate synchronous strategy-updates for all players. Although, such direction is behaviorally unrealistic (for the reasons outlined in Section 5.3), nevertheless, we feel compelled to rerun the simulations with simultaneous (synchronous) strategy-updates for all players in order to highlight the value-added of asynchronous updating of repeated-game strategies in approximating subjects' behavior.

The simulations of the strategy learning models with synchronous updating are run in an analogous fashion to those of the strategy learning models with asynchronous updating; the *only* difference is that we forego the asynchronous-updating-of-strategies equation and instead, introduce a parameter as the probability of updating the strategy set. Therefore, in addition to the existing parameters of each model, we introduce one more parameter: ρ. The three models have been calibrated based on a grid search. The plots of the experimental phase are displayed in Figure 9. Furthermore, we indicate in Table 8, the total Euclidean distance between the models with synchronicity in updating in the experimental phase and the experimental data across the four games.

The synchronous strategy learning models perform quite well in the Stag-Hunt game and the Chicken game. On the other had, they perform relatively well in the Prisoner's Dilemma game and the Battle of the Sexes game. Coordination on the cooperative outcome in the Prisoner's Dilemma game is the prevalent outcome in the experimental data albeit, as indicated in Figure 9, some pairs end up defecting. The synchronous strategy learning models capture the cooperative outcome well, but have hard time capturing the defecting outcome as shown in Figure 9. Applying asynchronous updating of strategies in this specific game, makes it more difficult for players to coordinate on the cooperative outcome. For example, a pair of players might be using cooperative strategies with triggers to defecting states in case of non-conformity to the cooperative outcome (for instance, "Grim-Trigger" strategies). The asynchronous updating is likely to lead to implementation of these strategies at different time periods. Consequently, one of the players might be in a defecting state when the other player decides to implement the specific strategy thus leading the pair to an endless string of retaliations i.e. mutual defection. Despite the fair performance of the synchronous strategy learning models in the Prisoner's Dilemma game and the Battle of the Sexes game, the models with synchronous updating do quite well in the other two games. Thus, for modelers who desire simplicity, the synchronous strategy learning models are a good alternative. However, for our purposes, the proposed modeling framework with asynchronous updating of repeated-game strategies is still a better choice given that it incorporates elements of psychological realism and economic relevance as envisioned by Rabin (2013). Additionally, the three strategy learning models with asynchronous updating do unequivocally better across all four games relative to the respective strategy learning models with synchronous updating as shown in Table 8.

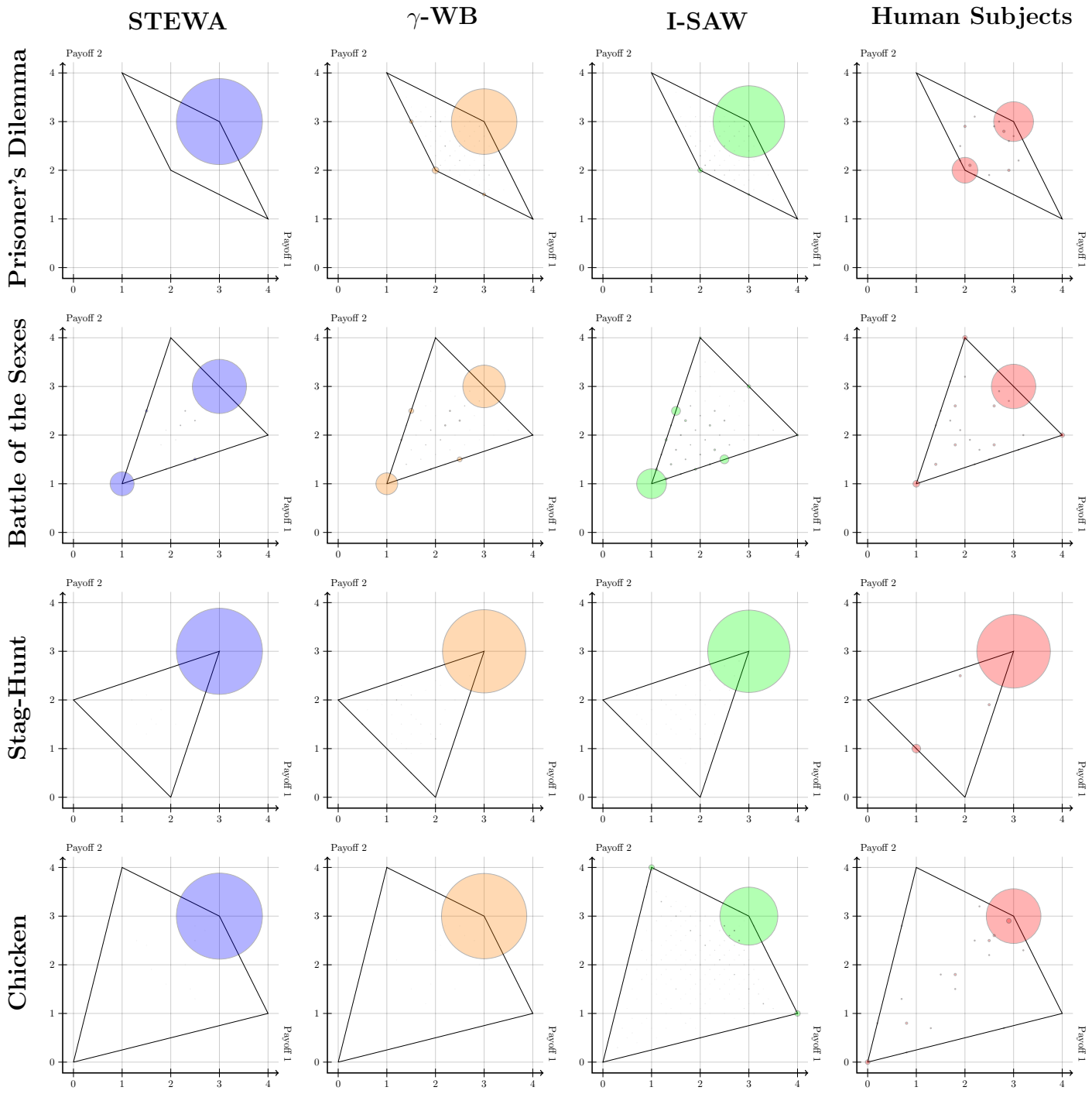


Figure 9: Strategy Learning Models with Synchronous Updating in the Experimental Phase & Experimental Data

Notes: We validate the predictions of the three strategy learning models with synchronous updating of strategies in the experimental phase with human data from the experiments of Mathevet and Romero (2012). The computational simulations and the experimental results with human subjects consist of averaging the last 10 periods of game-play.

F.2 Alternative Fitness Specifications

One of the difficulties in coping with repeated-game strategies is that it is not possible to observe the precise strategy of the opponent despite observing the history of play. One way to get around this difficulty is to use reinforcement learning, which doesn't require knowledge of the opponent's strategy.[24] Yet a big drawback of reinforcement learning models is that such approach can take a long time to converge. In order for a given repeated-game strategy's attraction to be updated in a reinforcement-learning model, the strategy must be played first. As the set of possible repeated-game strategies increases, the speed of convergence in reinforcement learning models deteriorates.[25] Another way to get around the difficulty of formulating beliefs about the repeated-game strategy of the opponent is to apply a fitness function. This approach is particularly attractive as complementing reinforcement learning with a belief-based component expedites convergence. When beliefs are added to the model, the attractions for every repeated-game strategy are updated at the end of every block. Therefore, the attraction on a strong strategy can start to increase with the first attraction-update; in contrast, implementing only the reinforcement component, keeps a strong strategy unaffected in terms of attraction-weights until it gets selected. For instance, the fitness function proposed here counts the number of consecutive fits between the candidate repeated-game strategy of the opponent and the observed sequence of actions profiles starting from the most recent and going backwards. We discuss next two alternative fitness specifications.

The first fitness function is a memory-one specification in which player i develops beliefs about player $-i$'s strategy in the χ^{th} block. Recall that the χ^{th} block's length for player i is denoted by $T_i(\chi)$. The block is divided into one-period observations of the following form: $\left(a_i^{t(\chi)-1}, a_{-i}^{t(\chi)-1}\right) \Rightarrow a_{-i}^{t(\chi)}$, where $t(\chi)$ is the t^{th} period corresponding to block χ. A strategy s_{-i} is said to support $\left(a_i^{t(\chi)-1}, a_{-i}^{t(\chi)-1}\right) \Rightarrow a_{-i}^{t(\chi)}$ if there exists some history $h^{t(\chi)} \in \mathcal{H}$ such that the last action profile of $h^{t(\chi)}$ is $\left(a_i^{t(\chi)-1}, a_{-i}^{t(\chi)-1}\right)$ and $s_{-i}\left(h^{t(\chi)}\right) = a_{-i}^{t(\chi)}$. The first alternative fitness function can then be written as,

$$\mathcal{F}^1\left(s_{-i}, \chi\right) = \sum_{r=0}^{T_i(\chi)-2} \mathbb{I}\left(s_{-i} \text{ supports } \left(a_i^r, a_{-i}^r\right) \Rightarrow a_{-i}^{r+1}\right).$$

[24] Reinforcement learning operates according to the "law of effect," which was formulated in the doctoral dissertation of Thorndike (1898). In principle, it assumes that a strategy is "reinforced" by the payoff it earned and that the propensity to choose a strategy depends, in some way, on its stock of reinforcement. Over the last decade, a growing body of research has studied analytically the properties of reinforcement learning both in normal-form and extensive-form games. On one hand, Laslier, Topol, and Walliser (2001) examined the convergence properties in repeated, finite, two-player, normal-form games in a learning process where each player uses the Cumulative Proportional Reinforcement (CPR) rule on strategies. The authors proved that the process converges with positive probability towards any strict pure Nash equilibrium. Related theoretical results were also given in Hopkins (2002) and Ianni (2013). On the other hand, Laslier and Walliser (2005) showed that when the CPR rule is applied on actions in a repeated, finite, extensive-form game with perfect information and generic (no ties for any player) payoffs, the process converges with probability one to the (unique) subgame perfect equilibrium. These contributions in economic theory were preceded by a large body of literature that originated in the work of mathematical psychologists in the 1950s (see for example, Bush and Mosteller (1951) and Estes and Burke (1953)).

[25] In addition, several studies show that providing foregone payoff information affects learning, which suggests that players do not simply reinforce chosen strategies (see, for instance the study of Mookherjee and Sopher (1994)).

The second fitness function is a memory-two specification similar to the one above. In this case, the block is divided into two-period observations of the following form: $\left(a_i^{t(\chi)-2}, a_{-i}^{t(\chi)-2}\right) \Rightarrow \left(a_i^{t(\chi)-1}, a_{-i}^{t(\chi)-1}\right) \Rightarrow a_{-i}^{t(\chi)}$. A strategy s_{-i} is said to support $\left(a_i^{t(\chi)-2}, a_{-i}^{t(\chi)-2}\right) \Rightarrow \left(a_i^{t(\chi)-1}, a_{-i}^{t(\chi)-1}\right) \Rightarrow a_{-i}^{t(\chi)}$ if there exists some history $h^{t(\chi)} \in \mathcal{H}$ such that $\left(a_i^{t(\chi)-2}, a_{-i}^{t(\chi)-2}\right)$ and $\left(a_i^{t(\chi)-1}, a_{-i}^{t(\chi)-1}\right)$ are the last two action profiles of $h^{t(\chi)}$ and $s_{-i}\left(a_i^{t(\chi)-2}, a_{-i}^{t(\chi)-2}\right) = a_{-i}^{t(\chi)-1}$ and $s_{-i}\left(a_i^{t(\chi)-1}, a_{-i}^{t(\chi)-1}\right) = a_{-i}^{t(\chi)}$. The second alternative fitness specification can be written as,

$$\mathcal{F}^2\left(s_{-i}, \chi\right) = \sum_{r=0}^{T_i(\chi)-3} \mathbb{I}\left(s_{-i} \text{ supports } \left(a_i^r, a_{-i}^r\right) \Rightarrow \left(a_i^{r+1}, a_{-i}^{r+1}\right) \Rightarrow a_{-i}^{r+2}\right).$$

We display the plots for each alternative fitness specification in the experimental phase in Figures 10-11 along with the experimental data. The plots use the parameters that led to the best goodness of fit across all four games. In addition, Table 8 shows the total Euclidean distance between the laboratory experimental data and the respective predictions of the models with the alternative fitness specifications in the experimental phase across all four games. At the aggregate level, the models with the alternative fitness functions do relatively well compared to their respective action learning models, but not as good as the strategy learning models with the specific fitness function proposed.

	STEWA	γ-WB	I-SAW
Action Learning Models	2.064	2.575	2.404
Strategy Learning Models	0.846	0.893	1.073
Synchronous Updating	1.254	1.112	1.291
Fitness Function #1	1.485	1.703	1.779
Fitness Function #2	1.398	1.542	1.695

Table 8: Alternative Specifications

Notes: The columns indicate the total Euclidean distance between the laboratory experimental data and the respective predictions of the models in the experimental phase across all four games. The first two rows are reproduced from Table 3, whereas the last three rows pertain the alternative specifications. All results are based on a discretization parameter $\varepsilon = 0.5$. If the predictions matched the data perfectly, then the distance would have been 0.

In addition, a value-added of the fitness function proposed in our model, beyond its overall superiority in approximating subjects' behavior well, is that it is behaviorally realistic, in sharp contrast to the alternative fitness specifications, which are somewhat naive. For instance, both alternative fitness functions assign equal weight to all actions in the block. Assume player i is trying to determine his opponent's

repeated-game strategy. Furthermore, assume his opponent played A in every period in the first half of the most recent block, but played B in every period in the second half of the most recent block. The two alternative fitness functions would assign equal weight to the "Always A" strategy and "Always B" strategy. However, the fitness function proposed would assign high weight to the "Always B" strategy, but 0 weight to the "Always A" strategy, because it starts from the most recent observation in the block and works backwards.

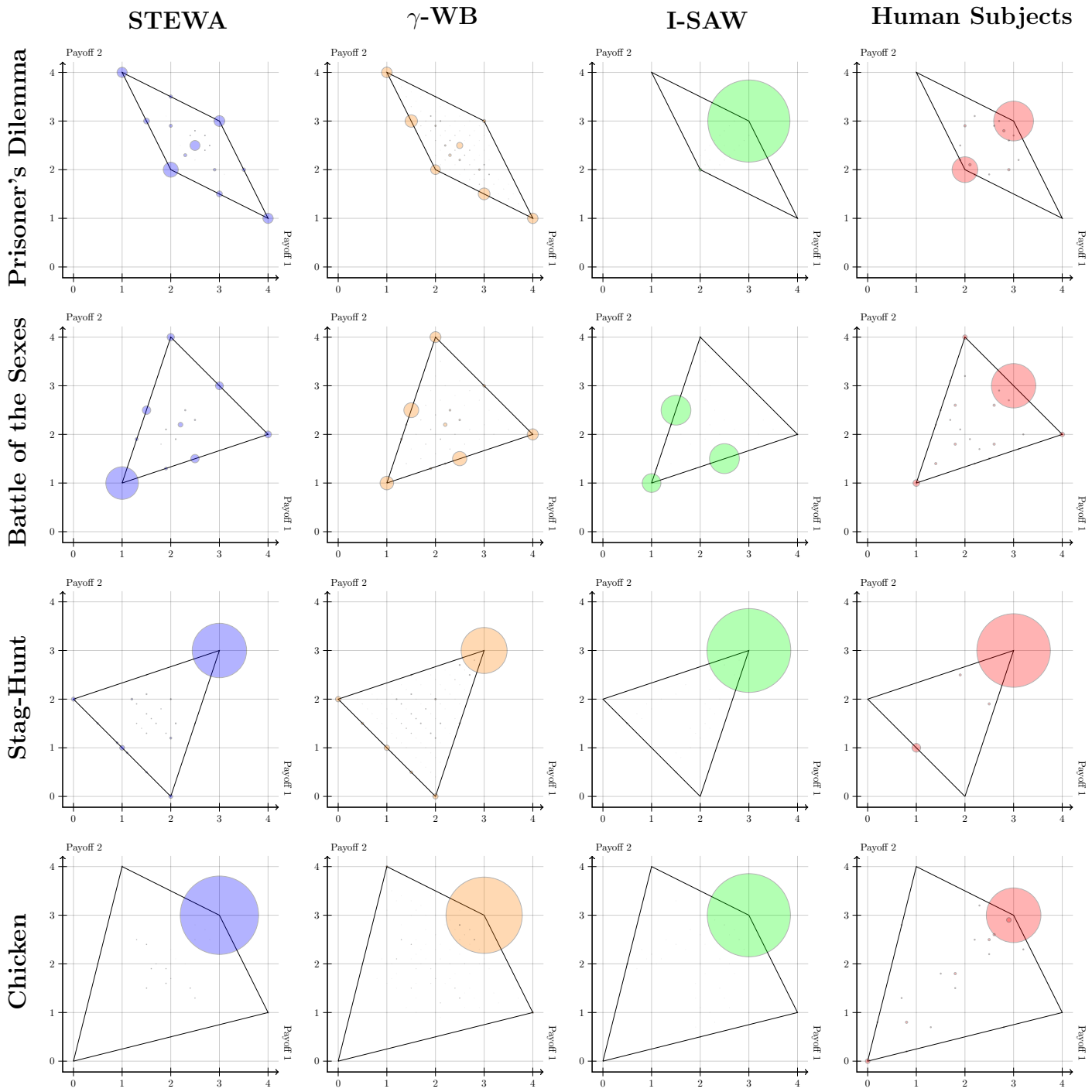


Figure 10: Strategy Learning Models with Fitness Function #1 in the Experimental Phase & Experimental Data

Notes: We validate the predictions of the three strategy learning models with fitness function #1 in the experimental phase with human data from the experiments of Mathevet and Romero (2012). The computational simulations and the experimental results with human subjects consist of averaging the last 10 periods of game-play.

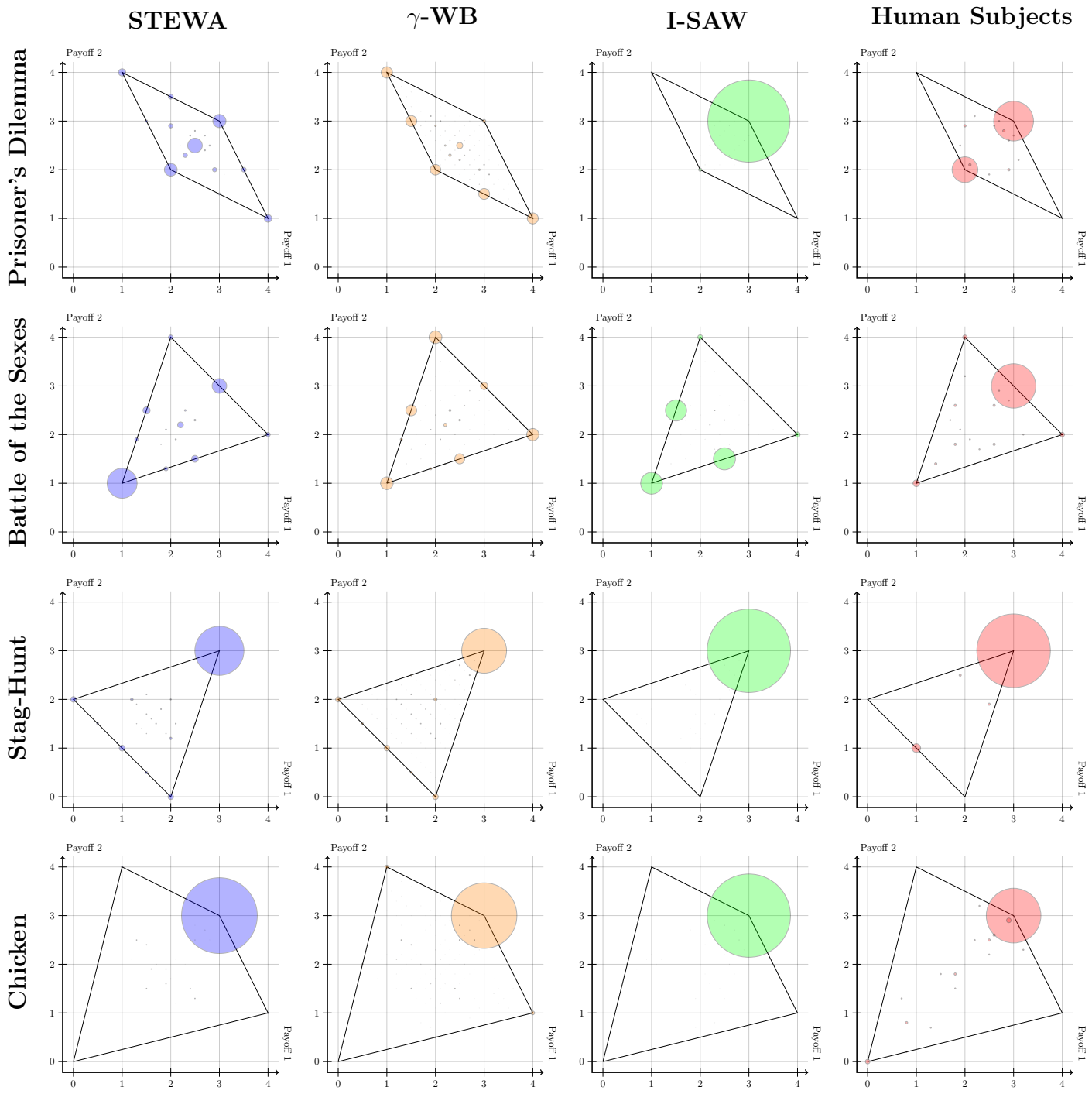


Figure 11: Strategy Learning Models with Fitness Function #2 in the Experimental Phase & Experimental Data

Notes: We validate the predictions of the three strategy learning models with fitness function #2 in the experimental phase with human data from the experiments of Mathevet and Romero (2012). The computational simulations and the experimental results with human subjects consist of averaging the last 10 periods of game-play.

G Sensitivity Analysis

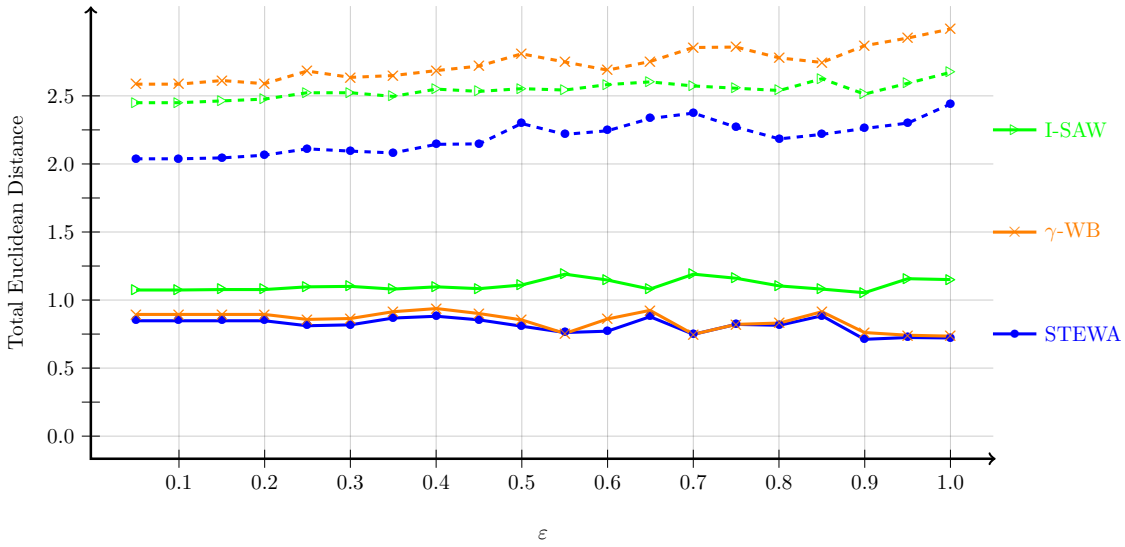


Figure 12: Sensitivity of Results to the Discretization Parameter ε

Notes: We compare the strategy learning models with the action learning models for the entire range of the discretization parameter ε. The action learning models are denoted with dashed lines, whereas the strategy learning models are denoted with solid lines.