Humans versus computer algorithms in repeated mixed strategy games

Spiliopoulos, Leonidas

University of Sydney

9 January 2008

Humans versus computer algorithms in repeated mixed strategy games

Leonidas Spiliopoulos

University of Sydney

8th January, 2008

Abstract This paper is concerned with the modeling of strategic change in humans' behavior when facing different types of opponents. In order to implement this efficiently a mixed experimental setup was used where subjects played a game with a unique mixed strategy Nash equilibrium for 100 rounds against 3 preprogrammed computer algorithms (CAs) designed to exploit different modes of play. In this context, substituting human opponents with computer algorithms designed to exploit commonly occurring human behavior increases the experimental control of the researcher allowing for more powerful statistical tests. The results indicate that subjects significantly change their behavior conditional on the type of CA opponent, exhibiting within-subjects heterogeneity, but that there exists comparatively little between-subjects heterogeneity since players seemed to follow very similar strategies against each algorithm. Simple heuristics, such as win-stay/lose-shift, were found to model subjects and make out of sample predictions as well as, if not better than, more complicated models such as individually estimated EWA learning models which suffered from overfitting. Subjects modified their strategies in the direction of better response as calculated from CA simulations of various learning models, albeit not perfectly. Examples include the observation that subjects randomized more effectively as the pattern recognition depth of the CAs increased, and the drastic reduction in the use of the win-stay/lose-shift heuristic when facing a CA designed to exploit this behavior.

JEL classification number: C9; C63; C70; C72; C73; C91

Keywords: Behavioral game theory; Learning; Experimental economics; Simulations; Experience weighted attraction learning; Simulations; Repeated games; Mixed Strategy Nash equilibria; Economics and psychology

Correspondence to: lspi4871@usyd.edu.au

1 Introduction

This paper addresses strategic change in humans' behavior conditional on opponents' play in a repeated game with a unique mixed strategy Nash equilibrium. In order to implement this efficiently subjects played the same game for 100 rounds against three preprogrammed computer algorithms designed to exploit different modes of play. The main research goals are to model subjects' behavior and ascertain how they altered their strategies against different opponents.

The majority of the experimental game theory studies have used human subjects playing against other human subjects, which complicates matters greatly for the following reason. Due to the bidirectional interaction of the human players it is difficult to focus solely on an individual and understand how his strategy may be changing and adapting in response to his opponent's play. This occurs because in such experiments there is no experimental control over a subject's opponent thereby severely limiting the questions researchers can investigate.

This study seeks to restore control to the experimenter by replacing one player with a computer algorithm (CA). This will allow the systematic observation of how people adapt and learn when playing against different types of opponents through the experimenter's manipulation of the CAs. The ability to pit humans against computer algorithms specially designed to target known or hypothesized human weaknesses in randomization is another important advantage over human versus human experiments. This will provide subjects with a strong monetary incentive to learn to avoid such damaging behavior since the CAs will be able to detect and exploit this behavior better than human opponents. Out of the three CAs used in this study, the *fp2* CA detected two-period patterns in subjects' behavior within a fictitious play framework, the *fp3* CA detected three-period patterns and finally the *spd* CA detected whether subjects were using the win-stay/lose-shift (*ws/ls*) heuristic or conditioning on the CA's first lagged action in general.

This paper is structured in the following manner. Section 2 presents the existing literature of human versus computer algorithm experiments examining game theoretic behavior. A discussion of the experimental setup is given in Section 3, with particular emphasis on the computer algorithms employed and the use of a between- and within-subjects experimental framework, alternatively referred to as a mixed design.

Section 4 provides general statistical analyses, such as whether the payoffs to subjects were significantly different depending on the CA opponent, whether the position of presentation of the CAs affected payoffs, as would be the case if subjects learned to play the game better over time, and whether the degree to which players' behavior was i.i.d. was affected by the type of CA opponent. The best performing CA was clearly the three-period pattern detecting algorithm *fp3,* which forced subjects to attain significantly lower payoffs compared to the other two CAs. Also, subjects were more likely to exhibit i.i.d. behavior when playing against this CA than others.

Section 5 includes two different approaches to modeling human behavior. The first approach, in Section 5.1, models subjects' action choices by probit regressions with lagged human and CA actions of up to five time lags as independent variables. A different approach of modeling human behavior is taken in Section 5.2 which estimates Experience Weighted Attraction (EWA) learning models (Camerer and Ho, 1999). The probit and EWA models are estimated both by pooling behavior according to the CA opponent, and also individually for each human/CA pair. The most important results of these analyses are that behavior is not i.i.d., as the lagged actions are statistically significant in explaining behavior, and that the pooled models perform better in cross-validation data

compared to the individually estimated models. Hence, this provides strong evidence that the heterogeneity exhibited by the subjects is within-subjects rather than between-subjects. Within-subjects heterogeneity occurs because subjects have a common set of different strategies available but choose the same strategy conditional on the type of opponent they face. In this case, observed heterogeneity can be driven either by changes in the game or by a change in the type of opponent. Between-subjects heterogeneity occurs because subjects do not have a common set of strategies so that the heterogeneity that is observed is purely attributed to each player and is not conditional upon their opponent. For example, if players are bounded rational to different degrees then this implies that their strategy sets are different and/or that their decision rules for selecting which strategy to use given the game and opponent are different[1]. Interestingly, parameter estimates of the individually estimated models were quite different from those of the pooled models due to overfitting. Hence, great care should be taken by researchers when drawing conclusions and making inferences regarding heterogeneity from parameter estimates of individually estimated models. Strategies implemented by subjects tended to be simple heuristics rather than complex models, which is similar to the viewpoint advanced by Gigerenzer (2000) that humans have a toolbox of simple yet effective heuristics at their disposal that they call upon depending on which is better adapted to the problem at hand, in this case the type of CA opponent.

Section 6 analyzes collected qualitative data, such as subjects' verbal expression of what their strategies were, and their estimates of the probability distribution over their own and opponent's actions during the previous rounds of play. Interestingly, the estimates over their own actions were significantly better calibrated than those over their opponents' actions.

Finally, the optimality of the behavior of subjects is examined through the use of simulations in Section 7. Since there is no technique available to exactly determine the optimal behavior against each of the CAs, a number of learning rules with varying parameter values and pattern detection capabilities were defined and a tournament was conducted where each of these learning rules was pitted against the CAs. The best performing learning rules were thus postulated as approximating the optimal strategy against each of the CAs. The main conclusion is that although the subjects did not precisely learn to play these strategies they did better respond by avoiding many of the learning rules or parameter values that were particularly inferior, and by modifying their behavior in the direction of best response. For example, subjects were quite effective in changing their behavior as measured by changes in the memory weighting parameter, the probability distribution over actions, the temporal correlation of actions and their use of the *ws/ls* heuristic.

Conclusions and possible future directions for research are summarized in Section 8 and an interesting comparison with a similar experiment (Barraclough et al., 2004) performed with monkey subjects that produced surprisingly similar results is given in Appendix A.

2 Literature review of human versus computer studies

The existing literature of humans versus computer algorithms is particularly limited and interestingly enough is characterized by experiments run in two different time periods separated by a relatively

[1] Theoretically, it could also be the case that the strategy sets are identical and that only the strategy decision rules differ between players. However, if evolutionary pressures and learning are powerful enough to direct players to have near identical strategy sets then in all likelihood they would also influence the decision rules employed by the players to the same degree.

large interlude. The first wave of research was in the late 1960s and early 1970s probably due to the widespread introduction of computers to academic institutions during this time period. There was a pronounced silence thereafter as game theory research was primarily directed to theoretical advancements with little interest in behavioral and experimental approaches. The second wave of research was inspired by the widespread surge of interest in experimental and behavioral economics, and the increase in the availability of computing power (and accompanying fall in cost), coupled with more user friendly programming languages.

The following section discusses the studies that are the most important and relevant to this paper's research agenda. Examples of research using computerized opponents in other strategic games such as bargaining and auction experiments include Walker et al. (1987), Smith and Walker (1993), Blount (1995) and Dursch et al. (2005).

2.1 Interdependent decision strategies in zero-sum games: A computer controlled study (Messick, 1967)

Messick (1967) conducted a pioneering experiment where humans were called upon to play against computer algorithms. The subjects faced three different algorithms, the first was a minimax strategy, the second was a *weighted fictitious play* (*wfp*) algorithm with perfect memory, and the third a fictitious play algorithm with a fixed memory window of five time period lags. The game was a 3×3 repeated zero-sum game consisting of 150 rounds. The main results are the following.

1. As expected, subjects' payoffs were not significantly different from the MSNE payoffs when playing against the minimax algorithm, but were significantly greater than MSNE payoffs when playing against the two fictitious play algorithms (payoffs were highest against the five period memory algorithm).
2. Subjects' play varied greatly across the three algorithms. In particular, the *ws/ls* heuristic accounted for roughly 70% of choices against the two fictitious play algorithms versus 49% against the minimax algorithm.
3. The fictitious play CAs were susceptible to exploitation because they led to behavior that manifested positive serial correlation which can easily be exploited by a *ws/ls* algorithm.
4. Also, the variability in choice behavior was found to be significantly lower against the two fictitious play algorithms which can be attributed to the result that any strategy against a minimax algorithm has the same expected payoff. However, against the non-minimax fictitious play algorithms, subjects as expected honed into a successful counterstrategy and tended to adhere to it leading to less variability i.e. experimented less with novel strategies.

2.2 The learning of strategies in a simple, two-person zero-sum game without a saddlepoint (Fox, 1972)

Fox (1972) had subjects play a 2×2 zero-sum game with a unique MSNE, where they confronted a rational opponent and a non-rational opponent that played a non-optimal mixed strategy independent of human play. Hence, a best response to this non-optimal strategy would be to play one of the two actions with probability one. The study concludes that:

1. Subjects changed their play in the direction of their best response when playing against the non-optimal algorithm, but did not play optimally i.e. play their best response with probability one, or at least not significantly different from one.
2. When subjects played against the optimal algorithm, and therefore had no financial incentive to change strategies, they still moved towards the MSNE. It was conjectured that subjects may be trying to reduce the riskiness of strategies by manipulating the variance of payoffs.

2.3 Strategic interaction in iterated zero-sum games (Coricelli, 2005)

Coricelli (2005) has human subjects playing a 4×4 game with a unique MSNE against two different CAs. This particular game was originally promoted by O'Neill (1987) based on its advantage that it is not dependent on the assumption of a linear utility function. The first strategy of the computer algorithm is to choose its action based on a *wfp* CA with infinite memory, and for the second strategy the CA again updates frequencies of actions in the same way with the only exception that it expects human subjects to overalternate as regards one of the action choices. Such overalternation has been documented in the psychology literature as occurring often when humans try to randomize, see Rapoport and Budescu (1997). Subjects were randomly assigned to two conditions to control for possible order effects in the presentation of the two algorithms which they played in succession. They were informed that they were playing against computer algorithms rather than humans. The most important results are:

1. Human play differed significantly across the two algorithms with subjects employing effective counter strategies to the second algorithm.
2. The data is inconsistent with the minimax hypothesis as human play is correlated with lagged own actions, lagged opponent actions and their interactions (up to and including two lags).

2.4 Learning about Learning in Games through Experimental Control of Strategic Interdependence (Shachat and Swarthout, 2002)

Shachat and Swarthout (2002) have human subjects play against two computer algorithms (reinforcement learning and EWA) in two games. The first is a zero-sum asymmetric matching pennies game called Pursue-Evade and the second is called Gamble-Safe and has a unique MSNE that does not coincide with the minimax solution to the game. In the first 23 rounds humans were playing against other humans and then play for each human subject was seamlessly switched over to a computer algorithm opponent, unbeknown to the subjects. They find the following interesting results:

1. Human play is not significantly affected by whether they are playing against humans or computer algorithms.
2. Each algorithm's frequency choices adjust linearly toward the best response to its human opponent's non-equilibrium action frequencies.
3. Although algorithms are better responding, their response is too weak to give them a competitive payoff advantage against humans. This is primarily due to the fact that the computer algorithms' decision rules are probabilistic so that they are by construct better responding to opponents' play rather than best responding.

4. Human subjects were not able to consistently exploit the predictability of the algorithms to gain a payoff advantage.

2.5 Do we detect and exploit mixed strategy play by opponents? (Shachat and Swarthout, 2004)

Shachat and Swarthout (2004) conduct an experiment where human subjects play against computer algorithms that are programmed to play various (not necessarily optimal) mixed strategies. The game they employ is a zero-sum asymmetric matching pennies game where the unique MSNE is for both players to play their Left action with probability two-thirds and their Right action with probability one-third. Subjects were assigned to be either row or column players and were then told that they would play 200 rounds against the same opponent. They were not informed that they were playing against a computerized opponent and steps were taken to mask the fact that the opponent was a computer. Each subject played against the same computer program which implemented a mixed strategy with probability of playing Left of either 19%, 27%, 35%, 51%, 59%, 67%, 75%, 83% or 91%. Such a wide range of probability distributions for the computer algorithms would then allow them to test the sensitivity of human subjects to a large range of deviations from the MSNE. They find the following important results:

1. They find that subjects' play is very likely to move in the direction of their best response if the computer algorithm's choice frequency was at least 15% away from the equilibrium MSNE proportion. As such they were able in these cases to gain a statistically significant payoff advantage against the computer algorithms.
2. There is considerable heterogeneity as regards the degree that subjects best responded and maximized their potential payoffs.

2.6 Proposed modifications and extensions to the existing literature

The use of a mixed-design experimental setup, in particular the within-subjects component, will allow for more powerful statistical testing than in other studies. This will be conducive for testing the performance of different types of computer algorithms, whether human subjects transfer learning from a previous opponent to subsequent opponents and identifying how subjects' strategic behavior changes with the computer algorithm they face.

None of the previous experiments have employed computer algorithms that are capable of explicit second- and higher-order pattern detection as the learning models used in these studies, reinforcement learning, EWA, weighted fictitious play are all capable of observing only the first-order play of human subjects. Consequently, subjects could manipulate the CAs in these studies by resorting to non-i.i.d. behavior that would be undetectable to the CAs. Hence, a logical approach for designing more effective CAs would be to incorporate pattern detection capabilities. Also, a second reason for incorporating pattern detection capabilities into CAs is that the weight of the psychological research finds that humans have problem randomizing or producing i.i.d. time series. For these reasons, pattern detecting CAs are an obvious extension that should be examined.

Another reason why CAs did not manage to significantly outperform their human opponents in these studies is because of the use of stochastic decision rules, which are not aggressive enough in exploiting the subjects' weaknesses. The best performance of the CAs in these studies was to

Table 1 Game payoffs

		Opponent's actions	
		Brown	White
Own actions	Blue	108, -80	-32, 60
	Yellow	-32, 60	28, 0

approximately achieve the MSNE payoffs, from which it can be inferred that these CAs were predominantly defensive algorithms. It is important to use more aggressive algorithms because this in turn leads to greater payoff incentives to human subjects to perform well, as any weaknesses they exhibit will result in larger payoff losses. Furthermore, the payoffs of the game employed in this study were carefully chosen to lead to a high degree of curvature in the payoff surface, thereby providing the human subjects with significant incentives and motivation to maximize payoffs.

This paper will also incorporate the use of open-ended questions requesting subjects to express their strategies in their own words. Such responses could be useful because they are direct elicitations of subjects' strategies instead of indirect elicitations such as fitting models to behavioral time series, which are plagued by various problems. Another innovation employed in this paper is that after playing each algorithm the subjects will be asked to state what they thought was the probability distribution over their own actions and over the CA's actions. By comparing these to the observed probability distributions it is possible to ascertain what information the subjects used in strategizing. In general, using both indirect and direct methods of elicitation in conjunction should lead to better inference than using either of these methods in solitude.

3 Methodology

3.1 General experimental setup

The experiment was run in a computer lab at Mediterranean College in Athens, Greece and all subjects participated in the experiment and interacted through computers running the Comlab Games Software[2]. Undergraduate students were randomly recruited through the use of fliers on campus and majored in business studies, psychology or computer science. Three sessions of eight students and one session of seven students were run for a total of 31 subjects.

The 2×2 constant-sum game presented in Table 1 was chosen with the intent to keep the game simple and easily understandable by subjects. Every subject played against each one of the three CAs 100 times, so that the game was played 300 times in total by each subject. The MSNE strategies of this game are playing blue and brown 30% of the time, and yellow and white 70%. It was desirable to place the MSNE probability of this game far from the trivial value of 0.5 in order to distinguish it from uniform play which subjects tend to find natural to play, at least initially when they have no experience[3]. On the other hand placing the MSNE too far away from 0.5 means that the probability of one action will be rather small and therefore the probability of this action being played twice in a

[2] This software is freely available at http://www.comlabgames.com and allows the design and implementation of a wide variety of game theory or decision making experiments.

[3] This tendency was verified in the data as the probability of playing the yellow action in the first ten rounds of play of each subject was 0.54.

row would be even smaller. Hence it would be exceedingly difficult for subjects to detect deviations from i.i.d. behavior, and in particular n-period strategies. It was with these considerations in mind that a MSNE of 70% in one action and 30% in the other action was chosen.

Given that this paper is concerned with changes in the strategic behavior of subjects it is imperative that the incentives for modifying and learning optimal strategies be appreciable. A game which has a "flat minimum" around the MSNE will not give subjects an incentive to converge to it and therefore is not an efficient game for testing this hypothesis. In this experiment the problem is analogous as subjects will not have an incentive to adapt their strategies much if the game has a very flat payoff function in the mixed strategy space[4]. The payoffs of the experimental game were specifically chosen so that the curvature of the payoff surface is high enough to provide significant payoff incentives. Table 2 gives the percentage change in payoffs compared to the baseline MSNE payoffs for various mixed strategies.

The use of negative payoffs in experimental games requires caution because if payoff incentive mechanisms are used and a subject's total payoffs are negative then an issue arises as this would imply that the subject would have to pay the experimenter. One solution is to pay a lump sum to the subject for participation that is large enough to cover the maximum possible loss. This is not practical however, as it would require either a very large lump sum payment or the use of a game with a flat payoff function so that the maximum possible loss is not large. Setting all payoffs in the game greater than or equal to zero also restricts the curvature of the payoffs function, thereby reducing the monetary incentives of the game. This can be seen intuitively with the following example. The maximum possible proportional loss compared to the MSNE payoffs is 100%, which would occur if a subject received the payoff of zero in each round. However, the use of negative payoffs would allow for a maximum proportional loss of greater than 100% which as a consequence would also increase the general curvature of the payoff surface. Hence, there is a tradeoff between increasing the curvature of the payoff function using negative payoffs and increasing the risk that a subject will finish the experiment with negative payoffs. This implies that the optimal solution will be interior and will therefore involve the use of negative payoffs. A pilot study of the implemented game supported the contention that total payoffs were unlikely to be negative. This is confirmed by the actual data from the experiment for which there did not exist any subjects who finished the experiment with negative payoffs, the lowest average payoffs were 4.93. Also, there were no subjects with negative average payoffs in the last 100 rounds of the experiment that might have affected subjects' behavior. These results justify the choice of the game's payoffs as they allow for the benefits of negative payoffs through increased payoff function curvature without any adverse effects as subjects were practically not in danger of earning negative payoffs.

The following observation should be kept in mind throughout the analysis of the data and discussion of subjects' strategies. A general result of games with a unique MSNE is that if player A chooses to behave according to the MSNE then by definition player B will be indifferent to his own strategies as all of them will earn the MSNE payoffs. However if the game is not constant-sum, player B's choice of strategy will affect the payoffs of player A and therefore the MSNE is not a minimax solution to this game. Since, the game employed in this study is constant-sum, if player B is indifferent then by definition player A must also be indifferent. In other words, playing the

[4] One of the earliest discussions of the problem of flat payoff functions is Harrison (1989), which critiques experimental first-price auction research.

Table 2 Percentage change in payoffs compared to the MSNE payoffs

		% of times brown action is played																
		10	12.5	15	17.5	20	22.5	25	27.5	30	32.5	35	37.5	40	42.5	45	47.5	50
% of times blue action is played	10	80	70	60	50	40	30	20	10	0	-10	-20	-30	-40	-50	-60	-70	-80
	12.5	70	61.25	52.5	43.75	35	26.25	17.5	8.75	0	-8.75	-17.5	-26.25	-35	-43.75	-52.5	-61.25	-70
	15	60	52.5	45	37.5	30	22.5	15	7.5	0	-7.5	-15	-22.5	-30	-37.5	-45	-52.5	-60
	17.5	50	43.75	37.5	31.25	25	18.75	12.5	6.25	0	-6.25	-12.5	-18.75	-25	-31.25	-37.5	-43.75	-50
	20	40	35	30	25	20	15	10	5	0	-5	-10	-15	-20	-25	-30	-35	-40
	22.5	30	26.25	22.5	18.75	15	11.25	7.5	3.75	0	-3.75	-7.5	-11.25	-15	-18.75	-22.5	-26.25	-30
	25	20	17.5	15	12.5	10	7.5	5	2.5	0	-2.5	-5	-7.5	-10	-12.5	-15	-17.5	-20
	27.5	10	8.75	7.5	6.25	5	3.75	2.5	1.25	0	-1.25	-2.5	-3.75	-5	-6.25	-7.5	-8.75	-10
	30	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	32.5	-10	-8.75	-7.5	-6.25	-5	-3.75	-2.5	-1.25	0	1.25	2.5	3.75	5	6.25	7.5	8.75	10
	35	-20	-17.5	-15	-12.5	-10	-7.5	-5	-2.5	0	2.5	5	7.5	10	12.5	15	17.5	20
	37.5	-30	-26.25	-22.5	-18.75	-15	-11.25	-7.5	-3.75	0	3.75	7.5	11.25	15	18.75	22.5	26.25	30
	40	-40	-35	-30	-25	-20	-15	-10	-5	0	5	10	15	20	25	30	35	40
	42.5	-50	-43.75	-37.5	-31.25	-25	-18.75	-12.5	-6.25	0	6.25	12.5	18.75	25	31.25	37.5	43.75	50
	45	-60	-52.5	-45	-37.5	-30	-22.5	-15	-7.5	0	7.5	15	22.5	30	37.5	45	52.5	60
	47.5	-70	-61.25	-52.5	-43.75	-35	-26.25	-17.5	-8.75	0	8.75	17.5	26.25	35	43.75	52.5	61.25	70
	50	-80	-70	-60	-50	-40	-30	-20	-10	0	10	20	30	40	50	60	70	80

MSNE is also a minimax solution to the game, as such it would be a serious violation of rationality if subjects in the long run were observed to be earning less than the MSNE payoffs. It is for this reason that throughout the analysis the MSNE payoffs will often be used as a reference point.

3.2 Computer algorithm selection

Apart from manipulating the curvature of the payoff surface of the game to increase incentives, it is also necessary to have CAs that are aggressive at exploiting opponents' flawed behavior to capitalize on the curvature of the payoff surface. Hence, the computer algorithms were specifically designed to exploit some limitations of human behavior such as the imperfect randomisation exhibited by subjects in Rapoport and Budescu (1997) and Budescu and Rapoport (1994). The three algorithms shall be referred to as *fp2*, *fp3* and *spd* (single period detector). Each algorithm can be broken down into two components, the belief learning rule and the decision rule which maps the beliefs generated by the learning rule into actions. A description of the belief learning rules of the CAs is provided below, followed by a discussion of the decision rules employed.

3.2.1 Belief learning models of the computer algorithms The *fp2* algorithm is a belief learning model that keeps track of the frequency of occurrence of two-period strategies and generates a belief for the probability of its opponent playing an action conditional on the previous action. The *fp3* algorithm is an analog of the *fp2* belief model that tracks three-period strategies (third-order play) and then generates beliefs for the probability distribution of its opponents' action conditional on the two previous actions.

Prior experimental evidence, such as Messick (1967), suggests that humans quite often use a very simple strategy in games, the *ws/ls* heuristic[5]. In order to guard against the possibility that humans would use this heuristic (or even the exact opposite heuristic i.e. changing strategy when winning and using the same strategy when losing) the *spd* algorithm was implemented which detects whether players adhere to this heuristic on average and then best responds to this belief. The computer algorithm keeps a count of the number of times a subject's action was consistent with the *ws/ls* heuristic minus the number of times it was inconsistent. Whenever this count is positive then the CA assumes the subject's next response will be the one prescribed by the *ws/ls* heuristic and therefore best responds to this. If the count is negative, the CA assumes the subject will play the action that is inconsistent with *ws/ls* and will best respond to that. Hence, the *spd* algorithm is capable of detecting the behavior of subjects conditioning on the CA's immediately prior action.

In Section 7, simulations were run of these three CAs competing against other learning models with different parameter values, to determine approximate optimal strategies against each of these algorithms. The results can be used to ascertain whether humans have learned to better respond to the computer algorithms they played.

3.2.2 Decision rules of the computer algorithms The choice of the decision rule is of paramount importance to experimental studies incorporating CAs as human subject opponents. There are two main classes of decision rules, however both have significant drawbacks. The first is a deterministic best-response rule which simply best responds to the beliefs generated by the CA. The second is to

[5] For a 2×2 game this is well defined as changing strategy necessarily implies choosing the other action.

employ a probabilistic rule, such as the logit decision rule. A best response rule has the advantage that in each period it fully exploits any patterns detected in human subjects, whereas a probabilistic rule would only best respond with a probability necessarily less than one. Therefore, the best response rule has the potential to exploit human behavior more than a probabilistic rule. For example, in Shachat and Swarthout (2004) although the CA's belief rule detected patterns in human behavior, because of the probabilistic nature of the decision rule the CAs could not achieve payoffs significantly higher than the MSNE payoffs. The problem with the best response rule is that it is much easier for subjects to reverse-engineer or figure out the belief model as there is no randomness built into it that would mask the underlying belief generation process. In fact, it may not even be necessary for human subjects to deduce the underlying belief generating mechanism, which may be quite complicated, because best response rules can fall into easily detectable patterns, or cycles, in their action choices. The CAs may then exhibit correlation in their action choices over time that can be exploited by simple heuristics such as *ws/ls*. On the other hand, a probabilistic rule with enough programmed randomness will be less likely to fall into such easily identifiable and exploitable patterns. It is clear that there is a non-monotonic relationship between the amount of noise injected into a decision rule and the algorithm's performance and therefore an optimal amount of noise will be an interior solution.

The usual probabilistic logit decision rules employed in the literature are functions of the expected payoffs of each action, as set forth in equation 1, where $E(\pi(a_i))$ is the expected payoff from action a_i and S_i is the discrete strategy set of player i. Hence, the amount of noise injected into each decision depends on the value of λ and the magnitude of the opponent's deviation from the MSNE prescription. If $\lambda \ll \infty$ then opponents with systematic but small deviations will not be exploited to a significant degree since the differences in the expected payoffs of the actions will not be large in magnitude. This study opts to decouple the relationship between the amount of noise injected into a decision from expected payoffs, thereby allowing for more aggressive behavior on the part of the CA. Also, standard probabilistic decision rules add noise to every single decision, which is excessive since patterns involve actions over many time periods and therefore they can probably be effectively concealed by injecting noise less often. For example, if an opponent is detecting n-period patterns then injecting noise on average once every n time periods will be sufficient to break this pattern[6].

$$P(a_i, t) = \frac{e^{\lambda_i \cdot E(\pi(a_i))}}{\sum_{a_i \in S_i} e^{\lambda_i \cdot E(\pi(a_i))}} \tag{1}$$

The final approach taken is to allow some CA responses to be pure best responses, with no added noise, whilst other responses are simply the MSNE of the game being played. In other words, instead of uniformly spreading noise out over all decisions, added noise is restricted to a subset of the actions. The decision rule employed for all CAs in this study is the following. In each round, with probability 0.8 the CA will best respond to its internally generated beliefs, and with probability 0.2 will simply choose an action using the MSNE prescription. This allows the computer algorithm to optimally respond with certainty to subjects' behavior 80% of the time whilst still randomizing according to the MSNE 20% of the time as a defensive move. There is also a fail-safe mechanism in case humans were able to significantly exploit the CAs, whereby if the average payoffs to the human

[6] No research was found in the literature regarding the theoretical optimal properties, or experimental comparisons, of decision rules - this is an obvious candidate for future research.

subject were larger than 20, the CA would revert to playing the MSNE until average payoffs fell back below 20. Finally, up until the fourth round the algorithms will be playing according to the MSNE prescription, in order to allow the algorithms to observe enough actions to generate beliefs.

3.2.3 Fp2 algorithm A schematic diagram of the operation of the *fp2* algorithm is provided in Figure 1, with the details of the equations and their updating provided below.

1. Starting from round 3, and for every round henceforth, after observing the action of the human subject update equations 2 and 3 for all j. Let the subscripts i and j denote two different players, then given actions a_j and a'_j, $I_t(a_j|a'_j)$ is an indicator function that takes a value of one if a_j was the action played at time t and a'_j was the action played at time $t-1$ and takes a value of zero otherwise[7]. Define for player i, the count of a_j at time t given action a'_j as[8]:

$$C_i(a_j|a'_j, t) = \frac{I_{t-1}(a_j|a'_j) + \sum_{u=1}^{t-2} I_{t-u-1}(a_j|a'_j)}{t-1} \tag{2}$$

   The *fp2* beliefs of player i of action a_j given action a'_j are then given as:

$$fp2_i(a_j|a'_j, t) = \frac{C_i(a_j|a'_j, t)}{\sum_{a_j \in S_j} C_i(a_j|a'_j)} \tag{3}$$

   where S_j is the discrete strategy set of player j[9].
2. If $t < 5$ or average payoffs to the human subject are higher than 20 proceed to step 3, otherwise proceed to step 4.
3. Choose the computer action probabilistically according to the game's MSNE and proceed to step 1.
4. Proceed to step 5 with probability 0.8 or to step 3 with probability 0.2.
5. Observe realized a'_j and if $fp2_i(a_j = blue|a'_j, t) < 0.7$ then choose brown as the computer action, if the inequality is reversed choose white and in the case of equality choose according to the MSNE. This step is essentially a best response to the value of $fp2_i(a_j = blue|a'_j, t)$.
6. Repeat starting from step 1.

The above algorithm is essentially a variant of *wfp* which instead of counting the past frequency of single-period actions and using that to predict future behavior, counts the past frequency of two temporally consecutive actions. Equation 2 implicitly assumes that there is no memory decay at all so that all past actions are remembered perfectly. This was consciously chosen because a pilot study showed that increasing memory decay made the computer algorithms much more predictable and open to exploitation, a result that is also corroborated by Messick (1967).

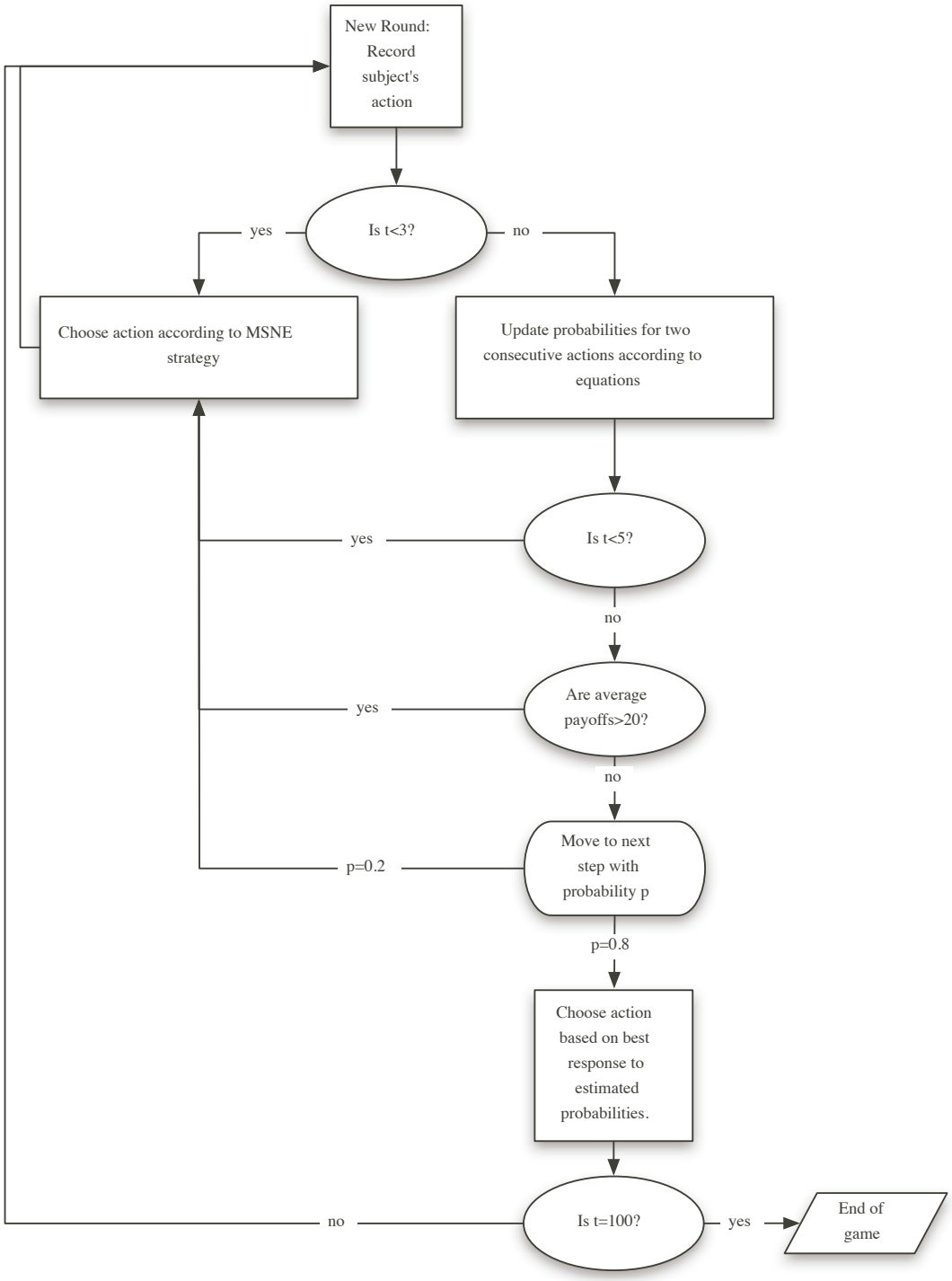
3.2.4 Fp3 algorithm The *fp3* algorithm is identical to the *fp2* algorithm in every way except that it keeps track of three temporally consecutive actions instead of just two. Equations 4 and 5 determine the probability updating procedure and Figure 2 gives a schematic representation of the algorithm.

[7] At $t = 1$ the indicator function takes the value of zero for all actions a_j, since a time period $t = 0$ does not exist and therefore actions are not observable.

[8] Note that this formulation implictly assumes perfect memory as past information is weighted equally in the algorithm.

[9] This definition assumes that the denominator is not zero i.e. that the action a'_j has been played at least once in the past. In cases where a'_j has not been observed beliefs are assumed to be given by a uniform distribution over $a_j \in S_j$.

Figure 1 *Fp2* algorithm



1. Starting from round 4, after observing the action of the human subject update the following equations. Given actions a_j, a'_j a''_j, $I_t(a_j|a'_j, a''_j)$ is an indicator function that takes a value of one if a_j was the action played at time t and a'_j and a''_j were the actions played at time $t-1$ and $t-2$ respectively, and takes a value of zero otherwise:

$$C_i(a_j|(a'_j, a''_j), t) = \frac{I_{t-1}(a_j|(a'_j, a''_j)) + \sum_{u=1}^{t-2} I_{t-u-1}(a_j|(a'_j, a''_j))}{t-1} \tag{4}$$

   The *fp3* beliefs of player i of action a_j given actions a'_j and a''_j are then given as:

$$fp3_i(a_j|(a'_j, a''_j), t) = \frac{C_i(a_j|(a'_j, a''_j), t)}{\sum_{a_j \in S_j} C_i(a_j|(a'_j, a''_j), t)} \tag{5}$$

2. If $t < 5$ or average payoffs to the human subject are higher than 20 proceed to step 3 otherwise proceed to step 4.
3. Choose the computer action probabilistically according to the game's MSNE and start again at step 1.
4. Proceed to step 5 with probability 0.8 or to step 3 with probability 0.2.
5. Observe realized actions a'_j and a''_j, and if $fp3_i(a_j = blue|(a'_j, a''_j), t) < 0.7$ then choose brown as the computer action, if the inequality is reversed choose white and in the case of equality choose according to the MSNE. This step is essentially a best response to the value of $fp3_i(a_j = blue|(a'_j, a''_j), t)$.
6. Repeat starting from step 1.

3.2.5 Spd algorithm The *spd* algorithm is inherently different from the previous two algorithms because instead of examining whether human subjects are conditioning on their own lagged actions, it examines whether subjects are using the win-stay/lose-shift heuristic (or its opposite, the win-shift/lose-stay heuristic).

1. Starting from round 4, after observing the human subject's action update equation 6 where $W_i(3)$ is initialized to be zero. The indicator function, $I_{t-1}(a'_j|a''_i, a''_j)$ takes the value one if the subject's action a'_j was consistent with the *ws/ls* heuristic and the value of -1 otherwise:

$$W_i(t) = W_i(t-1) + I_{t-1}(a'_j|a''_i, a''_j) \tag{6}$$

   The variable $W_i(t)$ is just the net sum of the number of times the subject has exhibited *ws/ls* behavior - a count of zero means that the *ws/ls* action was played 50% of the time, which is what would be expected by chance.
2. If $t < 5$ or average payoffs to the human subject are higher than 20 proceed to step 3 otherwise proceed to step 4.
3. Choose the computer action probabilistically according to the game's MSNE and start again at step 1.
4. Proceed to step 5 with probability 0.8 or to step 3 with probability 0.2.
5. If $W_i(t) > 0$, then assume the subject will play the *ws/ls* prescribed action and best respond to that, if $W_i(t) < 0$ assume the subject will not play the *ws/ls* prescribed action and best respond to that, and if $W_i(t) = 0$ then play the MSNE.
6. Repeat starting from step 1.

Figure 2 *Fp3* algorithm

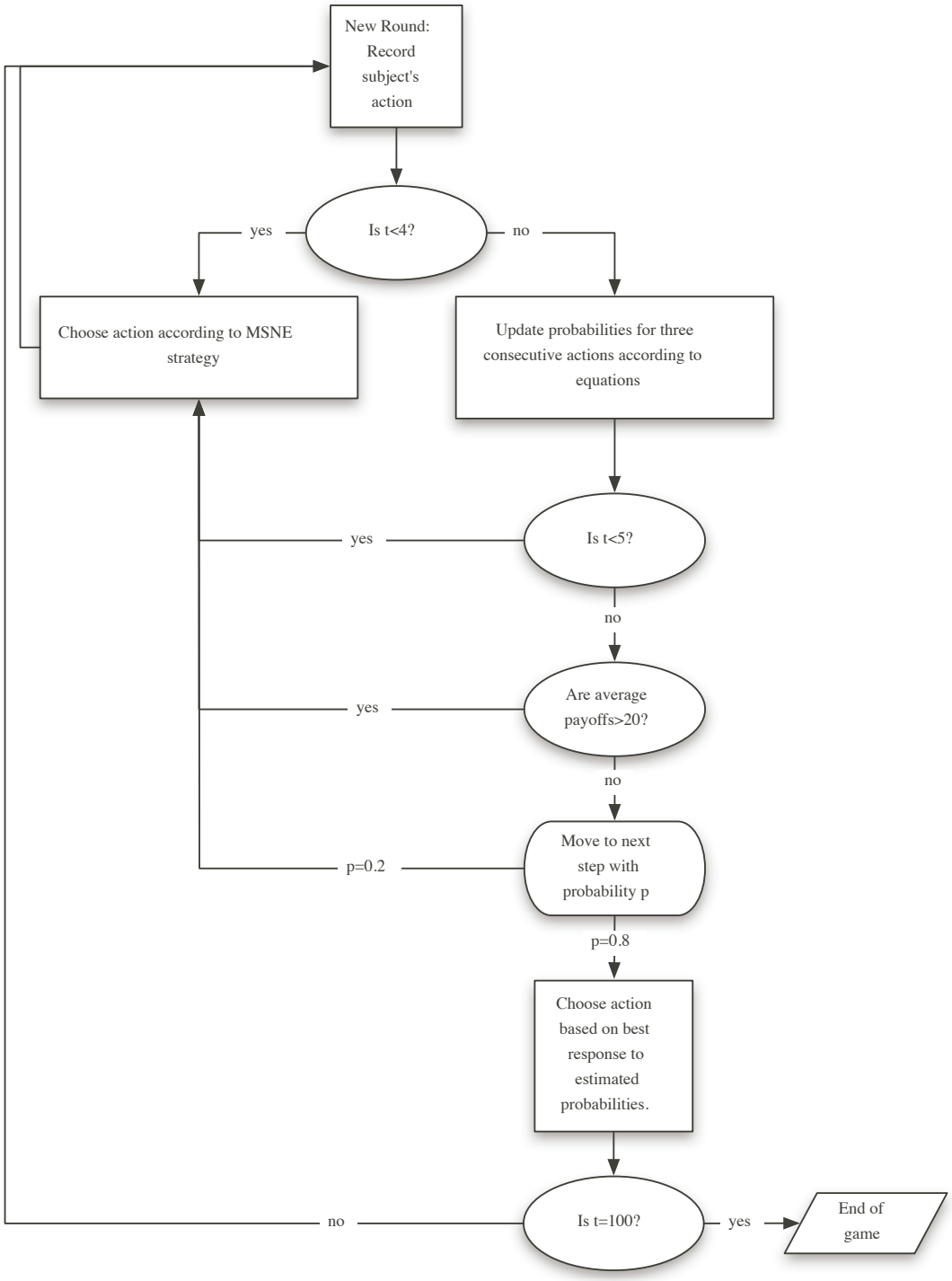
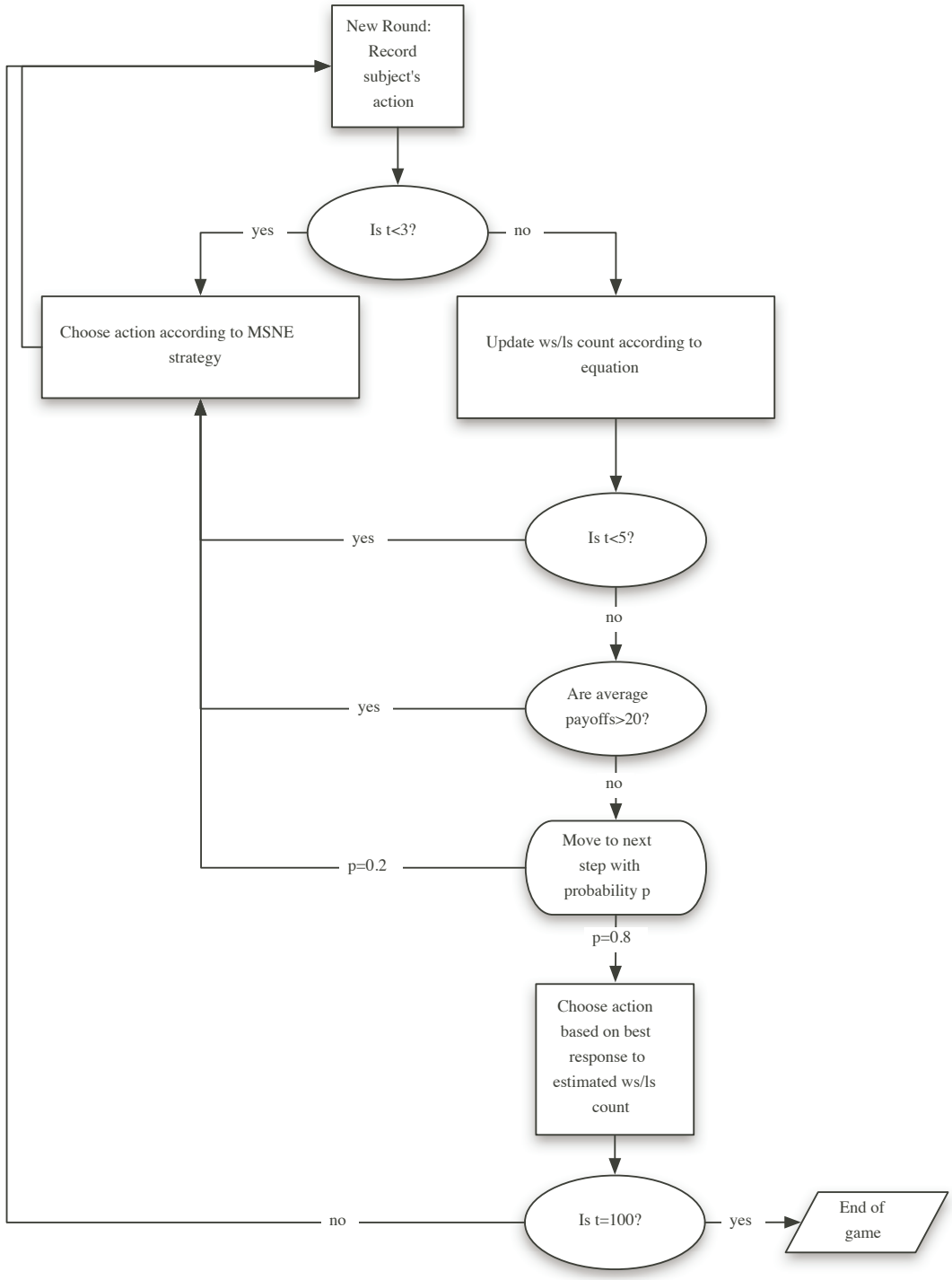


Figure 3 *Spd* Algorithm



3.3 Discussion of disclosure to participants

An important decision must be made regarding how much information to disclose to the human subjects in this experiment and in particular whether they are playing against a computer or another human subject. The issue of whether to use deception or to mislead experimental subjects is a hot topic, see Hertwig and Ortmann (2001) for a general discussion. Many economists are strongly against the use of any degree of deception (Davis and Holt, 1992), often on the basis that the use of deception is a public bad. Their reasoning is that in any single experiment a researcher may benefit from using deception, but on the whole this will give experimental economics a bad reputation and will cause future experiment participants to suspect that they are being deceived leading to a change in their behavior. Some economists, such as Bonetti (1998), contend that deception may be acceptable as long as it is of paramount importance to the experiment, and point to research on the use of deception in the psychology literature to argue that many of the negative effects feared by economists have not materialized.

This study provides full disclosure to subjects by clearly stating that they are playing against three different computer opponents, however absolutely no information or hint is given regarding how the computer algorithms work or how they differ from one another.

3.4 Monetary incentives to subjects

In order to provide incentives to subjects to actively participate in the game and seek out optimal strategies, subjects were given performance dependent monetary compensation for participating in the experiment. Subjects' monetary reward depended on the average number of points they had amassed against all three computer algorithms - the expected value of this according to the MSNE was 10. They would receive the average number of points in euro, hence the expected monetary reward from MSNE play is €10. Given that the experiment lasted roughly an hour this is an hourly rate of €10 per hour which compares favourably to alternative sources of income for students. According to Eurostat (2006), the minimum monthly wage for non-manual workers in Greece in 2006 is €668, roughly €4.18 per hour (assuming 40 hour weeks). Additionally, two prizes were offered to the subjects who achieved the two highest average payoffs after the completion of all experimental runs. The subject with the highest average payoff received an additional €30, whilst the second best performer received €20. These additional monetary incentives can be a particularly strong incentive to participants due to what the the psychology literature has documented as the overconfidence effect. Numerous studies, see Svenson (1981) and Gilovich et al. (2002) for examples, have shown that people tend to think that they are above average performers in general. Therefore, subjects will tend to believe that they have a larger probability of winning one of the two top prizes than they actually do, thereby providing an even stronger incentive to perform well in the experiment.

3.5 Information and data collection

Data on each subject's action choice was collected after each round. In game theory experiments it is often very difficult to recover the true parameters of the underlying behavioral models from

the data due to the bidirectional nature of subjects' learning which can lead to chaotic time series that on the surface appear to be random, even if the underlying rules are deterministic. One way of alleviating this problem is to use open-ended questions asking participants to state what their strategies were in their own words. Every 25 rounds subjects will have to respond to an open-ended question asking what their strategy was in the last 25 rounds. This will provide some insight as to what learning rule players are using against each CA and whether this changes during the repeated rounds of play.

Such a technique can also help pinpoint models that may be appropriate thereby narrowing down the field somewhat and guiding modeling. After the last (100th) round against each CA, subjects were asked to state how many times they thought their opponent played each action and how many times they thought they played each of their own actions. It will be interesting to see whether these probabilities will be close to the actual probabilities played. Their knowledge or lack thereof of how game play evolved may give some important information as to the underlying cognitive processes involved.

A comparison can be made by classifying human subject behavior according to their responses in the open-ended questions and the strategies that they appear to be playing as implied by learning models estimated on the data. These two may match if strategies are the result of conscious, structured strategizing on behalf of the subjects. However, if on the other hand subjects' learning is more of an automatic, subconscious learning process then these may diverge. In all likelihood subjects' play will be a fusion of both conscious and subconscious learning.

3.6 The information set of the subjects

After making a move each subject will be informed as to what move her opponent made, her current payoff in that period, her average payoff and her total payoff so far in the game. The reasoning behind this feedback is that it becomes easier for subjects to keep track of how well they are doing and therefore should induce greater responsiveness and payoff maximizing behavior.

3.7 Justification of within-subjects design and CA presentation order

There are two possible experimental designs that can be employed to observe how peoples' strategies change when faced with different opponents, between- or within-subjects structures. Between-subjects structures, where each subject would be assigned to play only one of the three CAs, have the disadvantage that they introduce idiosyncratic error when comparing across CA treatments and therefore reduce the power of subsequent statistical tests. On the other hand, in this case there is no issue of transfer of learning across the CA opponents. Since the games are the same regardless of the CA player, then in a within-subjects setup it is reasonable to assume that learning may carry over from the first 100 periods to the second 100 periods and so on, hence this must be taken into account. A between-subjects setup also has the following practical problem - for n different treatments (or CAs in this case) with x participants in each treatment an experimenter would need to find nx participants whereas the same can be achieved with x subjects in a within-subjects framework. This study opts for the within-subjects approach on the basis of the increase in power of statistical tests for a given number of subjects, whilst keeping in mind that learning may be occurring across

algorithms. In fact the hypothesis that learning is transferred from the history of play against the preceding CA opponents to later opponents will be formally tested.

The decision to use a within-subjects framework raises the issue of the presentation order of the CAs to subjects. Since this study is interested in testing differences in behavior and payoffs against the three different CAs it must resort to using a randomized order of presentation. This is because if learning does occur across the different algorithms then the algorithm to be presented last may appear to be a weaker opponent even though this may simply be due to the fact that subjects had more experience. In this case the order of the algorithms is perfectly correlated with the type of CA and therefore it would be impossible to separate the two effects. Ultimately for a completely randomized design all possible combinations of the orderings should be used, which in this case would amount to 3!. However, this would significantly reduce the number of subjects in each condition, so much so that the statistical power of tests would be significantly affected. Three conditions were used instead, which allowed each CA to occupy each position in the presentation order exactly once - the orders of presentation are *fp2-fp3-spd*, *fp3-spd-fp2* and *spd-fp2-fp3*. This implicitly assumes that only the position of the algorithm in the order of presentation affects behavior, but that behavior is independent of which particular algorithms preceded it.

4 General statistical analyses of game data

The use of a mixed-effects experimental design where each player faces all three algorithms, allows for more powerful statistical testing than conventional studies. The estimated models are linear mixed-effects models, which can capture both within- and between-subjects variance. Since subjects were randomly sampled from a population of possible subjects they should be modeled as random-effects rather than fixed effects. With random subject effects each individual's constant (or general ability) is a normally distributed random variable, whose mean and variance is estimated by the model. The general equation setup for the following analyses will be:

$$\textit{response=random subject effects + fixed treatment effects + error} \tag{7}$$

The treatments are fixed-effects and are modeled by the inclusion of appropriate dummy variables. There are two treatments each consisting of three levels. The algorithm treatment consists of three levels, namely the three CAs that the subjects faced, *fp2*, *fp3* and *spd*. Let A_{alg} denote a dummy variable equal to one whenever alg was the computer algorithm faced by the subject. Likewise, the position (or order) treatment consists of three levels i.e. whether the game was the first, second or third that a subject played. Let T_t be a dummy variable equal to one where t is equal to the position (or order of presentation) of the game. Adhering to this convention let the estimated coefficients of these treatment effects be denoted by the lower-case Greek equivalents, α_{alg} and τ_t. The dependent variable, payoffs, is denoted by π^t_{alg} with subscripts for each player dropped from the notation for simplicity.

The reference category for this model is the value of the dependent variable when subjects played against the *fp2* algorithm in the 1st position and is equal to the value of π^1_{fp2}. The estimated model is given in equation 8, where $\epsilon_{i,t} \sim N(0, \sigma^2_\epsilon)$ thereby assuming homoskedasticity of errors, and $\mu_i \sim N(0, \sigma^2_\mu)$ with variance-covariance matrix $\Omega_{3n\times 3n} = I_n \otimes \Sigma_{3\times 3}$, where $n = i \times t$ is the total

number of observations, I_n is the identity matrix and $\Sigma_{3\times3}$ is given in equation 9.

$$\pi_{i,t} = \pi^1_{fp2} + \alpha_{fp3}A_{fp3} + \alpha_{spd}A_{spd} + \tau_2 T_2 + \tau_3 T_3 + \mu_i + \epsilon_{i,t} \tag{8}$$

$$\Sigma_{3\times3} = \begin{bmatrix} \sigma^2_\epsilon + \sigma^2_\mu & \sigma^2_\mu & \sigma^2_\mu \\ \sigma^2_\mu & \sigma^2_\epsilon + \sigma^2_\mu & \sigma^2_\mu \\ \sigma^2_\mu & \sigma^2_\mu & \sigma^2_\epsilon + \sigma^2_\mu \end{bmatrix} \tag{9}$$

All analyses were performed in Stata (StataCorp, 2007) using the xtreg command, for maximum-likelihood estimation of random-effects models. Confidence intervals were boostrapped in order to circumvent the assumption of normality which was not justified by the data. Confidence intervals were created using the bias-corrected, accelerated percentile intervals (BC_a) method proposed by Efron (1987). Compared to standard bootstrapped confidence intervals, the BC_a method has the advantage of exhibiting second-order accuracy (compared to the first-order accuracy of alternative methods), and is transformation-respecting and range-preserving. The number of boostrap replications used is 2,500 which is higher than the recommendation of 2,000 replications made by Efron and Tibshirani (1994) for the BC_a method. The effects of multiple comparisons are accounted for by controlling the family-wise error rate (FWER) rather than the per-comparison error rate (PCER) as is usual, through the use of the Sidak (1967) correction. According to this correction, the family-wise error rate (FWER) for n pairwise comparisons is related to the PCER through the equation $1-(1-PCER)^{1/n}$. All confidence intervals will be calculated at a 5% FWER, and the respective PCER.

For ease of exposition when comparing contrasts, let π^t_{alg} denote the mean payoffs to players when they face the CA, alg, in position t. Note that $\pi^t_{fp3} = \pi^t_{fp2} + \alpha_{fp3}$, and similarly $\pi^t_{spd} = \pi^t_{fp2} + \alpha_{spd}$.

4.1 Are some algorithms better than others in exploiting human behavior?

Hypothesis Subjects' payoffs depend on the computer algorithm opponent they face, some algorithms will be more efficient than others at exploiting human weaknesses. Hence, payoffs averaged by computer opponent are significantly different from each other i.e. at least one of the coefficients α_{fp3}, α_{spd}, $\alpha_{spd} - \alpha_{fp3}$ will be significantly different from zero.

Testing this proposition or hypothesis entails three pairwise comparisons - the maximum possible number of combinations of two algorithms from a total of three algorithms. Assuming that there is no prior reason to believe that differences should be either positive or negative these three comparisons are two-tailed tests implying an appropriate PCER of 1.7%[10].

The results are displayed in Table 3 which shows that mean payoffs where highest against *spd*, followed by *fp2* and finally by *fp3*. The estimated difference in payoffs between *fp2* and *spd*, α_{spd}, is 1.099 which is not significantly different from zero at the appropriate PCER. Payoffs against *fp3* are significantly less than the payoffs from both the *fp2* and *spd* CAs making *fp3* the clear winner in terms of performance. Compared to these two CAs, payoffs against *fp3* are less by 3.265 and 4.363

[10] Although it is quite likely that the *fp3* algorithm will perform better than *fp2* against the human subjects because its pattern detecting capabilities are of a higher order, it is not possible to be sure of this. Therefore, preferring to err on the side of caution, a two-tailed test was employed.

Table 3 Payoff performance and its dependence on position and algorithm effects

	Coef.	Bias	Bootstrap s.e.	$lower_{98.3\%}^{2-tail}$	$upper_{98.3\%}^{2-tail}$	$lower_{98.3\%}^{1-tail}$
π_{fp2}^{1}	10.563	-0.005	1.078	7.964	13.147	
π_{fp3}^{1}	7.299	-0.018	1.074	4.503	9.726	
π_{spd}^{1}	11.662	-0.057	1.217	8.629	14.612	
π_{fp2}^{3}	13.109	-0.019	1.238	9.864	15.922	
π_{fp3}^{3}	9.844	-0.033	1.271	6.224	12.567	
π_{spd}^{3}	14.207	-0.071	1.205	11.352	17.232	
α_{fp3}	-3.265	-0.013	1.293	-6.459	-0.243	
α_{spd}	1.099	-0.052	1.356	-2.189	4.266	
$\alpha_{spd} - \alpha_{fp3}$	4.363	-0.038	1.352	1.256	8.059	
τ_2	2.609	0.020	1.257			0.037
$\tau_3 - \tau_2$	-0.063	0.010	1.403			-2.774
τ_3	2.545	0.030	1.313			-0.084
	Likelihood	$LR\,\chi^2(4)$	p-value			
	-285.30503	15.58	0.0036			

respectively revealing that the results are not only statistically significant but also economically significant in terms of magnitude.

In conclusion, the proposition that human subjects' payoff performance varies depending on the computer opponent is empirically verified as for two of the three pairwise comparisons the differences in payoffs were statistically significant, and the *fp3* CA is clearly the best at exploiting human play as it leads to significantly lower payoffs to subjects than the other two CAs.

4.2 Are payoffs against CAs significantly different from the MSNE payoffs, and is there a dependence upon the position?

Hypotheses Two hypotheses will be tested: the first that for some CAs in the first position subjects' payoffs are different from the MSNE payoffs, and secondly that the same is true of payoffs against CAs in the third position.

Human subjects may be able to exploit the CAs and earn higher than the MSNE payoffs, but at the same time the CAs are able to exploit human weaknesses, especially in randomizing. Testing whether subjects' payoffs are less than MSNE payoffs is important since subjects are always able to secure MSNE payoffs by simply playing according to the MSNE. Hence, such a finding would imply that subjects are not privy to what the MSNE solution is and were not able to learn it during the rounds of the game.

This section will firstly test for differences in games played during the first position and secondly for games played in the final position which would include learning throughout the experiment. These shall be treated as two different hypotheses for the sake of adjusting for pairwise comparisons.

Each proposition involves three pairwise comparisons, specifically comparing the mean payoffs grouped by the three CAs to the MSNE payoffs of 10. These tests will be two-tailed as it is not possible to rule out the case that subjects cannot utilize the MSNE solution. The appropriate PCER is 1.7% and the results are found in Table 3.

For games played in the first period, payoffs against *fp2* and *spd* are both higher than MSNE payoffs but they are not significantly different from 10 at the specified significance level. The mean payoffs against *fp3*, π^1_{fp3}, however are significantly less than 10, leading to the conclusion that the three-period pattern detecting capabilities of this algorithm are much more efficient at exploiting subjects than the simpler two-period pattern detecting capabilities of *fp*2.

Shifting attention to the payoffs from games played against the CAs in the last position, when the subjects have benefited from experience and learning, the results are quite different. Payoffs π^3_{fp2} and π^3_{spd} are again greater than MSNE payoffs, with the latter qualifying as a statistically significant result. Interestingly, payoffs π^3_{fp3} are estimated to be 9.844, much higher than π^1_{fp3}, and are now not statistically different from the MSNE payoffs.

In conclusion, there is evidence that by the final period humans are achieving payoffs significantly greater than MSNE payoffs against *spd*, whereas in the first period they are found to be playing significantly worse than MSNE payoffs against *fp3*. In the last position, none of the mean payoffs could be shown to be significantly less than the MSNE payoffs and therefore a hypothesis that the subjects have not learned to achieve at least MSNE payoffs by the end of the experiment can be rejected.

4.3 Do subjects exhibit game-specific and opponent-specific learning?

The answer to this question is approached from two different angles. Firstly, through the observation of the dependence of payoffs on their position in the presentation order, and secondly through the dependence of first-order behavior on the position.

4.3.1 Do payoffs depend on the order of presentation of each algorithm?

Hypothesis Human subjects exhibit transfer of learning across positions (and by implication across CAs), as measured by payoff performance, even though they are aware that the computer opponent has changed. This hypothesis implies that $\tau_2 > 0$ and $\tau_3 - \tau_2 > 0$.

A reasonable distinction between two types of learning can be made. The first type of learning is general knowledge about the properties of the game being played, which is independent of the CA opponent, and shall be referred to as game-specific learning. The second type of learning is opponent-specific learning which has to do with the learning of strategies that are effective against a particular CA.

These two types of learning can be expected to exhibit different properties throughout the experiment. Game-specific learning would be expected to be very strong initially, i.e. against the first CA, but would likely taper off quite early, in particular well before the end of the experiment. It is reasonable to expect that this type of learning will be transferred to the rounds played against the other two CAs as it is a general type of knowledge that will be useful against any opponent. Opponent-specific learning on the other hand, would be expected to be stronger during the initial rounds of play against each CA, and would likely not be perfectly transferred from one CA opponent to another. In practice, there may be some transfer of learning since subjects may continue to use strategies from the previous CA for lack of a better alternative in the initial rounds of playing against a new opponent. However, these strategies will likely be abandoned soon if they are not performing well against the new CA.

The statistical tests in this case are one-tailed as game-specific learning necessarily implies an increase in payoffs over time, and therefore only differences in payoffs from the immediately preceding time period should be examined. Learning from the first to second position necessarily implies that learning from the first to third period will be at least as much or more, depending on the strength of learning from the second to third position. A test for learning from the first to third position is redundant and therefore only two pairwise comparisons will be made, between the first and second time periods, and between the second and third time periods. All statistical tests will be one-tailed and adjusted for two pairwise comparisons, for a strictly controlled FWER of 5%.

The coefficients in Table 3 that test this proposition are those for τ_2 and $\tau_3 - \tau_2$, the first of which is significantly different from zero at the required significance level and economically significant as payoffs rise by 2.6. The value of $\tau_3 - \tau_2$ however is close to zero and not statistically significant implying that although there is significant transfer of learning from the first to the second and third periods in terms of an increase in mean payoffs, there is no additional transfer of new learning from the second to third time periods.

4.3.2 Do subjects strategically modify their first-order behavior against different CAs and is learned first-order behavior transferred across positions (and CAs)?

Hypothesis Subjects vary the first-order probability distribution (*fop*) over their actions depending on the CA they face. Hence, at least one of following coefficients in equation 10 must be significantly different from zero: α_{fp3}, α_{spd}, $\alpha_{spd} - \alpha_{fp3}$.

Let *fop* denote the first-order play (the percentage of yellow actions played by subjects is chosen as the reference class) and be the independent variable in Equation 10. As before, random subject effects are employed, captured by μ_i, and the treatment is the type of CA faced by subjects[11].

$$fop_{i,t} = fop_{fp2} + \alpha_{fp3} A_{fp3} + \alpha_{spd} A_{spd} + \mu_i + \epsilon_{i,t} \tag{10}$$

Table 4 reports that play against the *fp2* algorithm had the highest proportion of yellow actions followed by *fp3* and lastly by *spd*. First-order play against the *spd* and *fp2* CAs is significantly different from zero, and the difference $fop_{spd} - fop_{fp3}$, is close to being statistically significant as it ranges from -0.0019 to 0.1215, which is of considerable magnitude. In conclusion, there is statistical evidence that *fop* is used by subjects as a strategic instrument against different CAs.

Figure 4 graphs the evolution of first-order play across algorithms and positions. For example, the upper left subgraph plots the probability of playing yellow for the first 20 rounds (at a value of 20 on the x-axis), from the 21st to 40th rounds (value of 40 on the x-axis) et cetera. The graph immediately to the right of this subgraph plots the the probability when the *fp2* algorithm was faced in the second position. Within each subgraph, changes in probability indicate opponent-specific learning since they plot the same subjects playing against the same CA over the 100 rounds.

Transfer of learning is observed at the vertical lines indicating the crossover from one position to the next. A drop in the graph at these crossover points means that some opponent-specific knowledge acquired from playing the previous CA is discarded. However as long as the starting

[11] A linear model was used instead of a fractional response model on the basis of simplicity and intuitive interpretation of the magnitude of coefficients. Since all of the observations of the dependent variable were well within the bounds of zero and one results will not be affected significantly by this approach.

Figure 4 First-order play by CA and position

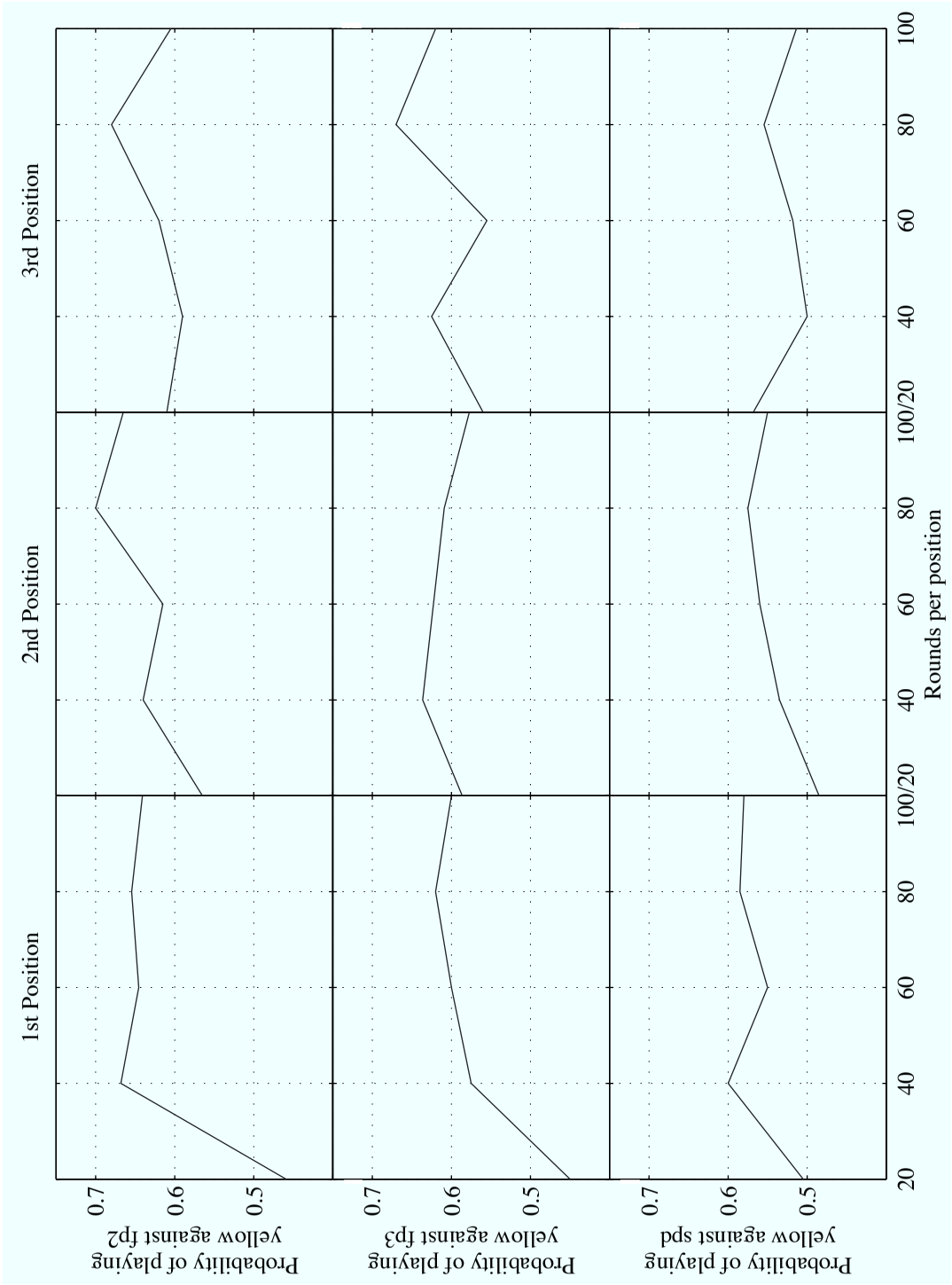


Table 4 Proportions of first-order play of yellow action across CA opponents

	Coef.	Bias	Bootstrap s.e.	$lower^{2-tail}_{98.3\%}$	$upper^{2-tail}_{98.3\%}$
fop_{fp2}	0.624	-0.0002	0.012	0.592	0.653
fop_{fp3}	0.594	-0.0007	0.011	0.564	0.619
fop_{spd}	0.545	0.0002	0.022	0.485	0.589
α_{fp3}	-0.029	-0.0004	0.016	-0.07	0.01
α_{spd}	-0.079	0.0005	0.025	-0.147	-0.024
$\alpha_{spd} - \alpha_{fp3}$	0.049	-0.0009	0.026	-0.002	0.122
	Likelihood	$LR\,\chi^2(2)$	p-value		
	91.138	11.65	0.003		

point of a subgraph is higher than the starting point of the subgraph of the previous position (i.e. the subgraph to the same row and to the left) this indicates some transfer of game-specific knowledge.

Against the *fp2* and *fp3* CAs there seems to be significant game-specific learning as the beginning of the second position graph is significantly higher than the beginning of the first position graph in both cases. However, comparing the second and third positions the starting point of the graphs are very close to each other indicating that no additional game-specific learning has occurred during the second position. At the crossover point from the first to the second position of the *spd* algorithm there is a drop back to the original level in the beginning, however comparing the end of the second position to the beginning of the third position yields a small increase. In general there does not seem to be much learning occurring as regards first-order play when the *spd* CA is the opponent, however it is very likely that other types of learning are taking place, such as second-order play or the frequency of use of heuristics such as *ws/ls*.

The largest changes in proportion of play occur in rounds 21-40 of the first position. By comparing the first 20 rounds of the first position to the last 20 rounds of the third position it is possible to see the final variation in first-order play after all learning has taken place. This comparison yields large increases in the proportion of yellow play for the *fp2* and *fp3* algorithm indicating significant learning, however play against the *spd* algorithm is approximately the same at the end of the experiment as it was as the beginning, with some variation however throughout.

4.4 Do subjects strategically modify their use of the win-stay/lose-shift heuristic against different CAs?

Hypothesis The *ws/ls* heuristic will be employed to different degrees by subjects depending on the CA which they are matched up with[12]. Hence, at least one of the following coefficients in equation 11 must be significantly different from zero: α_{fp3}, α_{spd}, $\alpha_{spd} - \alpha_{fp3}$.

The win-stay/lose-shift count, $W_i(t)$, is simply the difference between the number of times a subject's chosen action was consistent with this heuristic and the number of times it was inconsistent[13]. The modeled equation in this case, Equation 11, is similar to the one investigating first-order play with

[12] Intuitively, the *ws/ls* heuristic will be employed less against the *spd* algorithm because it is designed to exploit it implying a one-tail test. However, this hinges on the assumption of rationality of the subjects which is not indisputable and therefore the more cautious choice of a two-tailed test was preferred.

[13] This is equivalent to the output of the *ws/ls* heuristic employed in the *spd* algorithm.

Table 5 Win-stay/lose-shift behavior and its dependence on the CA opponent

	Coefficient	Bias	Bootstrap s.e.	$lower^{2-tail}_{98.3\%}$	$upper^{2-tail}_{98.3\%}$
W_{fp2}	35.774	0.067	3.346	27.657	43.608
W_{fp3}	30.742	0.013	2.557	24.94	37.234
W_{spd}	6.677	-0.03	2.759	-0.012	13.718
α_{fp3}	-5.032	-0.053	4.166	-15.199	5.33
α_{spd}	-29.097	-0.096	4.512	-39.264	-17.758
$\alpha_{spd} - \alpha_{fp3}$	-24.065	0.043	3.868	-14.626	-33.208
	Likelihood	$LR\,\chi^2(2)$	p-value		
	-396.556	43.74	0.000		

the exception that the independent variable is now the *ws/ls* count, and the results are given in Table 5.

$$W_i(t) = W_{fp2} + \alpha_{fp3}A_{fp3} + \alpha_{spd}A_{spd} + \mu_i + \epsilon_{i,t} \tag{11}$$

The value of W_{fp2} is roughly 36 translating into *ws/ls* play approximately 68% of the time, whilst W_{fp3} is less but the difference is not statistically significant. However when it comes to the *spd* algorithm there is a statistically and economically significant change in behavior compared to both the *fp2* and *fp3* CAs. As expected, since this algorithm is able to take advantage of players exhibiting *ws/ls* behaviour W_{spd} has fallen to 6.67 (translating into *ws/ls* behaviour only 53.5% of the time), which is not statistically different from zero.

4.5 Do subjects strategically modify the degree of randomness of their action choices against different CAs?

A common non-parametric test for randomness or i.i.d. behavior is the Wald-Wolfowitz runs test, which is based on the number of runs found in a time series. This statistic compares the observed number of runs of a dichotomous variable to the expected number of runs, given the observed frequencies. More than expected runs indicate the existence of overalternation in the time series, the opposite indicates underalternation or longer than random sequences. Table 6 presents the number of subjects for whom the hypothesis that they are playing randomly can be rejected by a Wald-Wolfowitz test conducted at the 5% significance level (two-tailed). The results are grouped by the two treatment groups i.e. CA opponents and position. The highest number of non-i.i.d. behaving subjects is found against the *spd* CA which is reasonable since it is not able to directly detect non-randomness. Also, more subjects were classified as playing randomly against *fp3* than against *fp2* which is a logical response as the increased pattern detecting capabilities of the former increases the incentives to adopt i.i.d. behavior.

The difference between actual and observed runs, Table 7, can also be used to examine players' behavior and how it varies against CAs and positions. When competing against *fp2* and *fp3* subjects' play exhibits fewer runs than expected which means that they are underalternating. Against the *spd* algorithm however the average difference over all positions is positive indicating overalternating behavior.

Table 6 No. of subjects not exhibiting i.i.d. behavior

	Positions			
CA opponent	1st	2nd	3rd	All
fp2	3	3	5	11
fp3	4	1	3	8
spd	5	5	6	16
All	12	9	14	

Table 7 Difference between observed and expected number of runs

	Positions			
CA opponent	1st	2nd	3rd	All
fp2	-2.42	-0.47	-5.75	-2.87
fp3	-5.85	-0.31	-1.72	-2.55
spd	5.74	-3.81	5.36	2.52
All	-0.89	-1.49	-0.51	

Hypothesis The difference in observed and expected runs is varied by subjects according to the CA opponent[14]. Hence at least one of following coefficients in equation 12 is significantly different from zero: α_{fp3}, α_{spd}, $\alpha_{spd} - \alpha_{fp3}$.

Equation 12 regresses the difference in observed and expected runs, denoted by *runs*, on random subject effects and fixed-effects on the levels of the algorithm treatment. Confidence intervals are two-tailed at an adjusted PCER of 1.7% to account for three pairwise comparisons.

$$runs_{i,t} = runs_{fp2} + \alpha_{fp3} A_{fp3} + \alpha_{spd} A_{spd} + \mu_i + \epsilon_{i,t} \tag{12}$$

Table 8 Dependence of the difference between actual and expected runs

	Coef.	Bias	Bootstrap s.e.	$lower^{2-tail}_{98.3\%}$	$upper^{2-tail}_{98.3\%}$
$runs_{fp3} - runs_{fp2}$	0.315	0.052	1.577	-3.601	3.894
$runs_{spd} - runs_{fp2}$	5.390	-0.050	1.692	1.509	9.690
$runs_{spd} - runs_{fp3}$	5.074	-0.102	1.842	0.737	9.734
	Likelihood	$LR\,\chi^2(2)$	p-value		
	-323.474	14.62	0.0007		

The results of this panel regression are displayed in Table 8, which indicates that statistically significant differences exist in randomness of play between *fp2* and *spd*, and *fp3* and *spd*. In both cases the number of observed runs when playing against the *spd* CA increases by roughly 5.

[14] In accordance to previous similar arguments a two-tailed test was preferred even though a reasonable assumption is that $\alpha_{fp3} < 0$ since *fp3* is designed to detect higher order non-i.i.d. behavior and therefore a rational player should deviate less from the expected number of runs.

Table 9 Spearman correlation coefficients of first-order human and computer play by opponent

CA opponent	Spearman correlation coefficient	p-value
fp2	-0.259	0.1594
fp3	-0.378	0.036
spd	0.386	0.032

4.6 Joint first-order behavior of human subjects and CAs

Figure 5 displays for each subject/CA pair the joint observed proportion of action choices, allowing the assessment of the resulting dynamics of the two-way interaction between players. Data is broken down by CA opponent in order to clarify the difference in single-period characteristics of each game on the unit square of probabilities. Also, the figure is decomposed into four quadrants by the MSNE proportions of play. Assuming i.i.d. play by both players, Quadrants I and IV lead to lower than MSNE payoffs for human subjects and Quadrants II and III lead to higher than the MSNE payoffs.

It is clear from the spread of data points that play is not centered on the MSNE and that there is a pattern to the deviations. The first striking observation is that rarely do human subjects exhibit proportions greater than the MSNE of 0.7 whereas the computer algorithms were more prone to this, leading to a large concentration of datapoints in Quadrants III and IV. There are no datapoints in Quadrant II, only two on the border, so it clear than human subjects' superior payoffs are primarily due to play being concentrated in Quadrant III. This may be due to the fact that the payoff to action blue, 108, is much higher than other possible payoffs and may therefore be creating a bias to a smaller proportion of yellow play. Another interesting observation is that almost all of the *spd* algorithm datapoints are in Quadrant III implying that the *spd* algorithm almost always played white less than 70% of the time. On the other hand, the *fp2* and *fp3* algorithms ventured into Quadrant IV relatively often. Furthermore, there are few datapoints in Quadrant I, most of them belonging to play against the *spd* algorithm.

Table 9 demonstrates that for algorithms *fp2* and *fp3* there is a negative correlation between human and computer single-period play, whereas for the *spd* algorithm it is positive. However, no inference should be made about the direction of causality because of the inherent endogeneity of this system where one player's behavior affects the other's and vice versa. Rather these correlations should be viewed as representing the effects of the interaction between subjects and CAs and their influence on the joint distribution of first-order behavior on the unit square.

4.7 Payoffs and nth-order behavior

The determinant of subjects' payoffs is the joint play of both game players, but more interestingly this can be broken down into two distinct determinants. The first is the resulting first-order play of both players and the second is the resulting second- and higher orders of play. This segregation is interesting as it can help pinpoint why subjects or CAs achieved high payoffs. Table 10 presents the expected payoffs given the first-order play of both players, *fop*, where expected payoffs are calculated by using the realized joint first-order probabilities of play and multiplying by the relevant payoffs. If actual payoffs are higher than expected first-order payoffs this implies that subjects have taken advantage of the CAs through manipulation of second- and/or higher orders of play.

Figure 5 Scatterplot of first-order play of both players

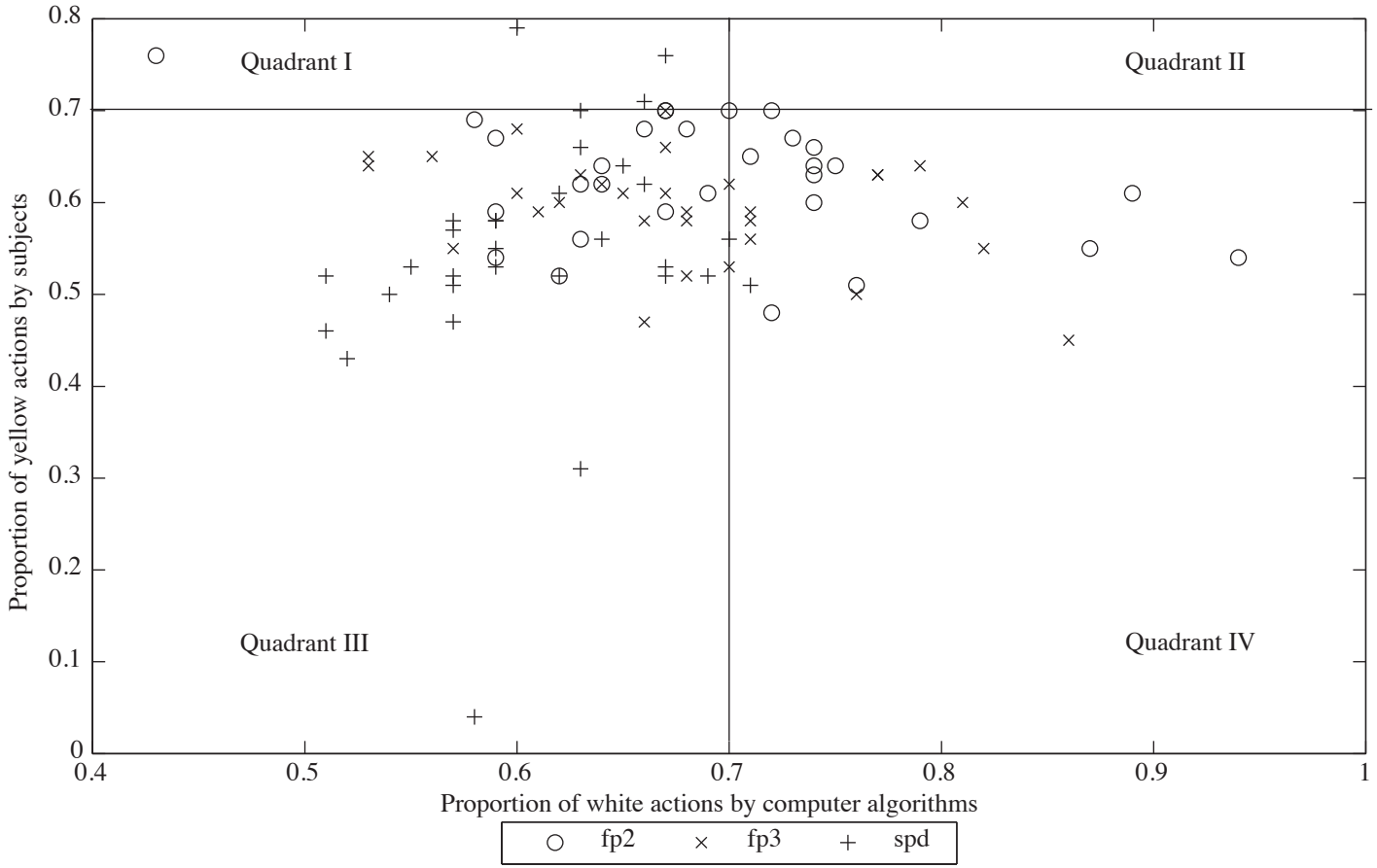


Table 10 Actual and expected payoffs given first-order play

CA	Expected payoffs given *fop* of both players	Actual payoffs	Coef.	Bias	Bootstrap s.e.	$lower_{95\%}^{2-tail}$	$upper_{95\%}^{2-tail}$
fp2	10.061	12.226	2.166	0.009	1.016	0.108	4.121
fp3	10.463	9.045	-1.418	-0.017	1.041	-3.676	0.498
spd	12.888	13.406	0.518	-0.009	0.891	-1.346	2.196

Expected payoffs given first-order play are higher than the MSNE payoffs against all computer algorithms, although marginally so against the *fp2* algorithm. Actual payoffs however are higher than MSNE payoffs only for the *fp2* and *spd* algorithm. Humans have exploited patterns when playing against the *fp2* and *spd* algorithms however only against the *fp2* algorithm is the difference between actual and first-order expected payoffs significantly greater than chance at the 5% significance level (applying a bootstrap BC_a method). Perhaps the most interesting result is that although first-order expected payoffs against the *fp3* algorithm are higher than the MSNE payoffs, actual payoffs where much less which implies that the *fp3* algorithm managed to exploit patterns in human behavior, which the *fp2* algorithm did not. This result is not entirely surprising as the *fp3* algorithm is capable of deeper pattern detection than the *fp2* algorithm. Note however, that although it is not significantly different from zero at the 5% significance level, the magnitude of the effect is quite large as actual payoffs are roughly 14% lower than the expected payoffs given first-order play.

5 Models of human behavior

This section proceeds by presenting two different models of behavior, probit regressions and EWA models, and explicitly examining the type of heterogeneity present in subjects' behavior. A comparison of the probit and EWA models will lead to some interesting conclusions regarding the relative value of simple heuristic models of behavior versus more complex models when opponent heterogeneity is controlled for by the use of a CA opponent.

5.1 Probit modeling

This section proceeds to examine how players in this game have conditioned on their own and opponent's past actions, and whether conditioning is dependent on the type of CA opponent. This analysis will be performed by maximum likelihood probit regressions, executed in Stata, with human actions at time t as dependent variables and dummy variables of human and computer actions for the previous 5 lags as independent variables $(H_{t-1}, ..., H_{t-5}, C_{t-1}, ..., C_{t-5})$.

Section 5.1.1 estimates 4 models, one pooled model for all the data collected, and one model for each CA opponent that the subjects played against, whilst Section 5.1.2 allows for greater heterogeneity in players by estimating a model for each human/CA pairing.

For ease of exposition and comparison all estimated coefficients are reported as the discrete change in the probability of playing an action when the dummy independent variable changes from zero to one (evaluated at the mean of the other independent variables), denoted by dP/dx. A robust

error approach was used, namely the Huber/White sandwich estimate of variance, Huber (1967) and White (1980).

The dataset is broken down into training and cross-validation (CV) subsets which will allow for the examination of possible overfitting of models to the data, and which will permit the deduction of the true out-of-sample predictive power of the models. The cross-validation set starts from the fourth time period and continues as every third datapoint thereafter i.e. the 4th, 7th, 10th...100th time periods - all other datapoints form the training set. This method of partitioning the data was chosen rather than assigning a contiguous block of data, such as the final rounds, so that it more accurately tracks the the fit of the model throughout all the rounds of the game.

5.1.1 Probit modeling pooled by CA opponent The estimated model will have current human play as the dependent variable, human and computer actions lagged up to 5 periods as the independent variables.

$$Pr(H_t = 1|H_{t-1}, ..., H_{t-5}, C_{t-1}, ..., C_{t-5}) =$$
$$\Phi_N(\beta_{h_{t-1}} H_{t-1} + ... + \beta_{h_{t-5}} H_{t-5} + \beta_{c_{t-1}} C_{t-1} + ... + \beta_{c_{t-5}} C_{t-5}) \quad (13)$$

In equation 13, H_t is a dummy variable equal to one if the human subject played the blue action at time t and zero otherwise, C_t is another dummy variable equal to one if the CA played the brown action at time t and zero otherwise. The transformation Φ_N is simply the cumulative distribution function of the normal distribution, which ensures that the fitted value of the dependent variable is bound between zero and one. This equation will be estimated on four different sets of data: data of play against all the algorithms pooled together and also against each of the algorithms separately. Table 11 provides the results of these probit regressions, where dP/dx is the estimated marginal effect of each independent variable, POI is the percentage inaccuracy on the training set, or equivalently the percentage of classification errors, POI-CV the percentage inaccuracy on the cross-validation set and Baseline is the error rate in classifying behavior if one simply predicted the most probable action from the data.

In fitting a single model of play against all the CAs, all the coefficients except $\beta_{h_{t-1}}$, $\beta_{h_{t-3}}$ and $\beta_{c_{t-3}}$ are statistically significant at the 5% level. The largest in magnitude coefficient is $\beta_{c_{t-1}}$, with $\beta_{c_{t-2}}$ also significantly large, indicating that players are conditioning primarily on the first and second lags of computer play. This model does better than a baseline model as the POI for the cross-validation set falls by roughly 5% points.

The models estimated separately for each of the computer algorithms are of greater interest as they allow comparisons and will provide information as to how subjects' play depends on their opponents' behavior. Figure 6 represents the estimated coefficients graphically for ease of comparison. Against the *fp2* algorithm the only statistically significant independent variables are $\beta_{h_{t-4}}$ and $\beta_{c_{t-1}}$, with the effect of $\beta_{c_{t-1}}$ on probability outweighing in magnitude that of $\beta_{h_{t-4}}$ by 2.7 times. Hence, human play against the *fp2* algorithm is driven primarily by conditioning on the last computer move[15].

[15] Given the chosen game, it is important to note that the *ws/ls* heuristic would manifest itself in a probit analysis as a large in magnitude coefficient on the first computer lag.

Table 11 Probit regressions of human play on 5 lags of human and computer play (pooled across all data and across each CA)

	All algorithms			*fp2*			*fp3*			*spd*		
	dP/dx	z-stat.	p-value	dP/dx	z-stat.	p-value	dP/dx	z-stat.	p-value	dP/dx	z-stat.	p-value
$\beta_{h_{t-1}}$	0.024	1.680	0.093	0.048	1.940	0.053	0.056	2.250	0.025	-0.021	-0.860	0.390
$\beta_{h_{t-2}}$	0.080	5.750	0.000	0.028	1.120	0.265	0.079	3.220	0.001	0.131	5.620	0.000
$\beta_{h_{t-3}}$	0.008	0.540	0.588	0.010	0.390	0.693	-0.004	-0.170	0.867	0.018	0.760	0.449
$\beta_{h_{t-4}}$	0.112	7.920	0.000	0.121	4.680	0.000	0.110	4.390	0.000	0.104	4.410	0.000
$\beta_{h_{t-5}}$	0.067	4.950	0.000	0.033	1.360	0.175	0.079	3.320	0.001	0.077	3.260	0.001
$\beta_{c_{t-1}}$	-0.235	-16.240	0.000	-0.326	-12.430	0.000	-0.287	-11.550	0.000	-0.092	-3.420	0.001
$\beta_{c_{t-2}}$	-0.111	-7.470	0.000	-0.048	-1.700	0.089	-0.056	-2.120	0.034	-0.125	-4.570	0.000
$\beta_{c_{t-3}}$	0.017	1.150	0.250	-0.048	-1.710	0.087	0.007	0.270	0.784	0.007	0.250	0.800
$\beta_{c_{t-4}}$	-0.053	-3.640	0.000	-0.007	-0.250	0.802	-0.043	-1.680	0.094	-0.061	-2.240	0.025
$\beta_{c_{t-5}}$	0.038	2.590	0.010	0.029	1.080	0.282	0.057	2.230	0.026	0.001	0.050	0.957
POI	36.17			32.16			34.36			39.07		
POI-CV	36.16			33.97			34.17			37.30		
Baseline	41.02			37.28			40.27			45.5		

Figure 6 Estimated probit coefficients of human play on human and computer lagged play

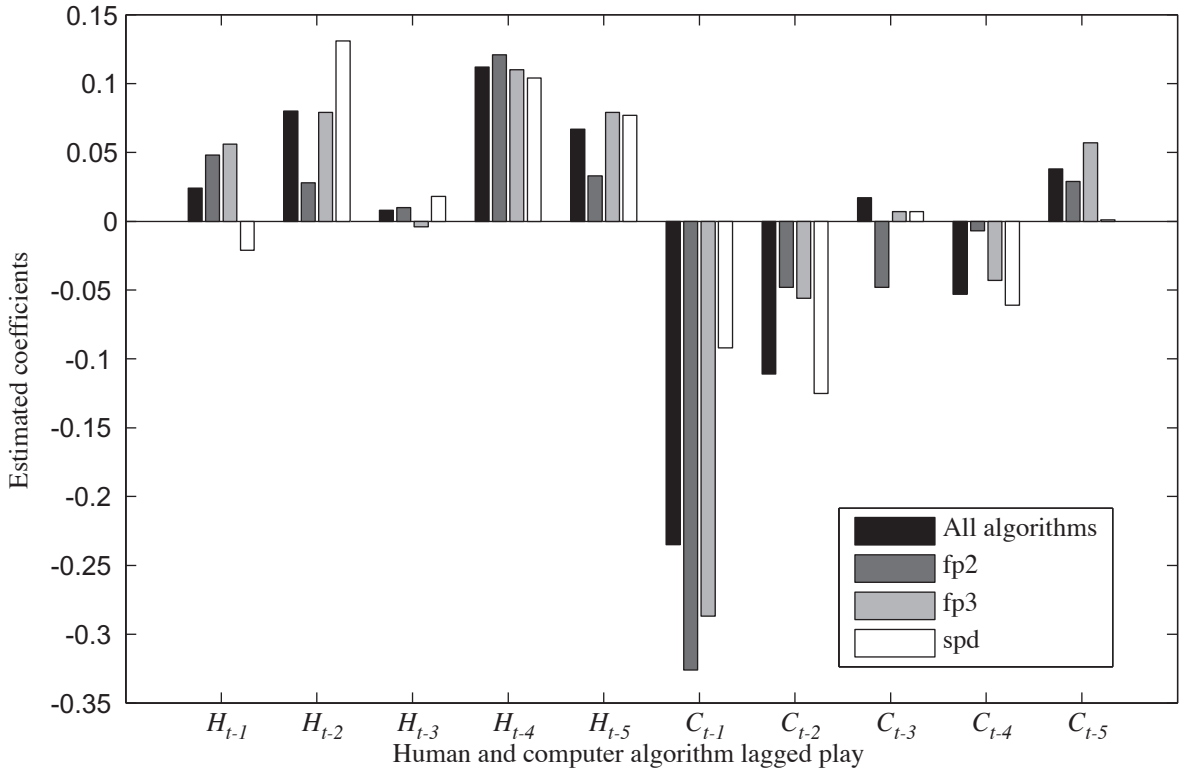


Table 12 Probit models of human play and dependence on C_{t-1}

	$\beta_{c_{t-1}}$			Error measures	
CA	dP/dx	z-stat.	p-value	POI	POI-CV
fp2	-0.32	-13.29	0.000	31.24	32.76
fp3	-0.264	-11.04	0.000	35.13	32.66

In comparison, against the *fp3* algorithm the statistically significant coefficients are $\beta_{h_{t-1}}$, $\beta_{h_{t-2}}$, $\beta_{h_{t-4}}$, $\beta_{h_{t-5}}$, $\beta_{c_{t-1}}$, $\beta_{c_{t-2}}$ and $\beta_{c_{t-5}}$. Due to the large number of observations however, statistically significant results do not necessarily imply that these variables have real economic significance. In particular, the two largest coefficients are again $\beta_{h_{t-4}}$ and $\beta_{c_{t-1}}$ which are many times larger than the coefficients of the other statistically significant variables. Again the main driving force seems to be the first computer lag indicating the use of the *ws/ls* heuristic just as was the case with the *fp2* algorithm. Shifting to the model estimated for games against the *spd* algorithm, the results now change significantly compared to those of *fp2* and *fp3*. Although the coefficient $\beta_{c_{t-1}}$ is still statistically significant it is roughly one third in magnitude of that estimated for the *fp2* and *fp3* models. This result indicates that subjects learned to condition less on C_{t-1} which is reasonable as the *spd* algorithm is designed to exploit the *ws/ls* heuristic (and therefore conditioning on C_{t-1}). The other important difference is that subjects are now conditioning much more on C_{t-2}, as the effect on the dependent variable is roughly two times higher compared to the *fp2* and *fp3* models. The coefficient $\beta_{h_{t-4}}$ is still large in magnitude and statistically significant but has been surpassed in importance by $\beta_{h_{t-2}}$.

In all cases the models have a lower POI for the cross-validation set than the baseline error rate indicating that significant information has been discovered in the lags of the human and computer play allowing for better prediction. Given the importance of C_{t-1} in predicting behavior against the *fp2* and *fp3* algorithms it is also of interest to regress human play only on C_{t-1} in order to determine whether this is adequate to explain behavior or whether more lags are needed. The model now simply becomes:

$$Pr(H_t = 1|C_{t-1}) = \Phi_N(\beta_{c_{t-1}} C_{t-1}) \tag{14}$$

Even though a reduced model will necessarily not fit as well as the full model as regards the likelihood function of the training data, it is possible that a reduced model may fare better when examining the POI of the cross-validation set.

This is verified by the results in Table 12 as in all cases there is an improvement in the cross validation POI indicating that the 5-lag models suffered from overfitting. This can occur because the greater the number of free parameters in a model the greater the ability of the model to fit the noise in the data rather than the true underlying signal. These models estimated by regressing only on C_{t-1} are essentially capturing the use of the win-stay/lose-shift algorithm. As regards behavior against the *spd* CA, it was not possible to significantly reduce the number of independent variables without negatively impacting the POI-CV. This indicates that a long memory process is being employed, a result that is supported by subsequent EWA modeling.

5.1.2 Individual probit modeling This section focuses attention on the effects of fitting probit models to each subject/CA pair. This allows for heterogeneity of individuals and therefore will provide a population distribution of parameter estimates. Table 13 gives the percentage of subject/CA pairs for which the estimated coefficient of each independent variable was significantly different from zero according to z-tests conducted at the 5% level. Human play was more often affected by the first lag of the computer algorithm's play followed by the second and fourth lag of humans' own play.

Turning now to the distribution of individual parameter estimates grouped by computer algorithm, the most interesting result is the change in the sign of the mean coefficient of the first human lag, from being positive when playing against *fp2* and *fp3* to becoming negative when subjects played the *spd* algorithm. In other words, subjects switch from positively correlated play to negatively correlated play. The other most striking result is the fall in the mean magnitude of the coefficient of C_{t-1} when subjects were playing against the *spd* algorithm. This shows that players have learned that conditioning on C_{t-1} leads to inferior payoffs against the *spd* algorithm. This is a rational reaction as the *spd* algorithm is designed to take advantage of exactly that type of behavior.

The significance statistics essentially mirror the above results as the percentage of player/algorithm combinations for which $\beta_{c_{t-1}}$ is statistically significant falls from 80.65% and 74.19%, for *fp2* and *fp3* respectively, to 12.90% for the *spd* algorithm. When playing against the *spd* algorithm the percentage of cases where $\beta_{c_{t-2}}$ is significant rises from 3.23% and 6.45%, for *fp2* and *fp3* respectively, to 22.58% - it seems that the conditioning shifts from C_{t-1} which is exploitable by the *spd* algorithm to the second lag, C_{t-2}, which is not exploitable by the *spd* algorithm. Also, against the *spd* algorithm the percentage of significant coefficients for lagged human play in general increases significantly.

A comparison of the cross-validation performance of the CA pooled models and the individually estimated models in Table 14 will provide evidence whether between-subjects heterogeneity is important. Comparing the single probit model pooled across all CAs, the cross-validation performance is slightly better than that of the individually estimated probit models. This result is even more impressive if one accounts for the difference in the number of free parameters, which is enormous as the single model estimates ten parameters whereas the individual models estimate a total of 310 parameters. Comparing models pooled by the CA opponent with the cross-validation performance of individually estimated models averaged for each CA opponent, the pooled models performed better for the *fp2* and *fp3* CAs, and worse for the *spd* CA. The difference for the *fp3* CA is particularly large as the pooled model outperforms the mean performance of the individually estimated models by 4.23% points. On average, the pooled models perform better on cross-validation data than the individual models providing convincing evidence that there is relatively little between-subjects heterogeneity in the data.

5.2 EWA modeling

The learning literature is mainly devoted to two general types of learning algorithms and their variants: fictitious play algorithms and reinforcement learning. For a long time these two algorithms were regarded as distinct and not closely related, however Camerer and Ho (1999) changed this perception. They showed that fictitious play is an extension of reinforcement learning where all possible actions are reinforced in every time period rather than just the chosen action as is customary in reinforcement learning models. This discovery allows the nesting of both reinforcement and fictitious

Table 13 Percentage of player/game combinations where each independent variable is statistically significant from zero

		Coefficients									
	CA opponent	$\beta_{h_{t-1}}$	$\beta_{h_{t-2}}$	$\beta_{h_{t-3}}$	$\beta_{h_{t-4}}$	$\beta_{h_{t-5}}$	$\beta_{c_{t-1}}$	$\beta_{c_{t-2}}$	$\beta_{c_{t-3}}$	$\beta_{c_{t-4}}$	$\beta_{c_{t-5}}$
Mean	All	0.02	0.11	0.03	0.12	0.07	-0.27	-0.08	0.01	-0.05	0.03
value of	*fp2*	0.03	0.07	0.05	0.13	0.03	-0.37	-0.05	-0.02	-0.02	0.03
estimated	*fp3*	0.06	0.1	0	0.1	0.07	-0.32	-0.06	0.05	-0.02	0.03
coefficients	*spd*	-0.04	0.16	0.04	0.12	0.1	-0.11	-0.12	-0.01	-0.09	0.02
% of individual	All	7.53%	20.43%	8.60%	21.51%	11.83%	55.91%	10.75%	8.60%	10.75%	3.23%
coefficients	*fp2*	0.00%	6.45%	3.23%	22.58%	9.68%	80.65%	3.23%	12.90%	6.45%	3.23%
significantly different	*fp3*	6.45%	22.58%	6.45%	19.35%	6.45%	74.19%	6.45%	3.23%	6.45%	0.00%
from zero	*spd*	16.13%	32.26%	16.13%	22.58%	19.35%	12.90%	22.58%	9.68%	19.35%	6.45%

Table 14 Comparison of fit between pooled and individually estimated probit models grouped by algorithm

CA opponent	POI-CV of pooled probit models	POI-CV of individual probit models
All	36.16%	36.43%
fp2	33.97%	35.1%
fp3	34.17%	38.4%
spd	37.3%	35.78%

play learning models (and the spectrum between these two endpoints defined by how much weight is given to forgone payoffs) within a single model - the experience weighted attraction model or EWA for short. Hence, rather than model reinforcement learning models and fictitious play models separately we will instead fit the EWA model and compare the models by looking at the parameters that distinguish the two learning rules.

Equation 15 is the EWA updating formula for the attractions of each available action. These attractions are then normalized so that they sum up to one, thereby representing probabilities of playing each action, by implementing the logit decision rule in equation 16. Players are indexed by i (all other players by $-i$), individual strategies by j for a total of M_i strategies per player, $\pi_i(s_i^j, s_{-i}(t))$ are the payoffs to player i given strategies s_i^j at time t, and $I(s_i^j, s_i(t))$ is an indicator function which is equal to one if strategy j was played at time t by player i (i.e. if $s_i^j = s_i(t)$) and zero otherwise. The free parameters in the model are ϕ which represents a decay rate on the previous period attraction and can be thought of as strength of memory, δ controls how much, if at all, forgone payoffs affect attractions and κ controls how attractions grow over time - whether attractions are weighted averages of previous attractions or a cumulative sum of previous attractions. The parameter $N(t)$ is an experience weight and is modified according to equation 17 - note that if $N(0) = 0$, $\kappa = 0$ and $\phi = 1$ then the experience weight is simply the number of rounds played. The λ parameter in the logit decision rule determines the sensitivity of agents to differences in the attractions of the available actions. For example if $\lambda = 0$ then subjects are completely insensitive to attraction strengths and will play each action with the same probability. As λ increases, given differences in attractions of actions will lead to greater differences in the probability of playing actions. As $\lambda \to \infty$ this decision rule will tend to a strict best response rule with one action being played with probability one and the others with probabilities of zero i.e. it is what is often referred to as a hard decision rule rather than a soft decision rule. Before the game starts, at $t = 0$, the experience weight $N(t)$, is assigned an initial value, $N(0)$ to be estimated, as are all attractions $A_i^j(0)$ as well[16].

$$A_i^j(t) = \frac{\phi \cdot N(t-1) \cdot A_i^j(t-1) + \left[\delta + (1-\delta) \cdot I(s_i^j, s_i(t))\right] \cdot \pi_i(s_i^j, s_{-i}(t))}{N(t)} \tag{15}$$

$$P_i^j(t+1) = \frac{e^{\lambda \cdot A_i^j(t)}}{\sum_{k=1}^{M_i} e^{\lambda \cdot A_i^k(t)}} \tag{16}$$

$$N(t) = \phi \cdot (1-\kappa) \cdot N(t-1) + 1 \tag{17}$$

[16] In this case, given that there are two possible actions it is only necessary to estimate $A_i^j(0)$ for one action and normalize the other value to some constant, in this case zero. This is because of the parametric form of the logit decision rule which is such that adding a constant to all attractions does not change probabilities.

5.2.1 Special cases nested within the EWA model One of the alluring aspects of the EWA model is that it nests many of the learning models used in the literature, namely reinforcement learning, weighted fictitious play and cournot best response dynamics. As a result it is extremely easy to compare these standard models simply by observing the estimated values of certain parameters.

In standard reinforcement learning, only realized payoffs affect attractions and therefore only the attraction of the action played in the previous period will be adjusted. This can be accomplished in the EWA model by setting $\delta = 0$. If $\kappa = 1$ and $N(0) = 1$ then these three restrictions will reduce equation 15 to equation 18 which is the standard cumulative reinforcement learning model. Previous period attractions are decayed by a multiplicative factor of ϕ and are strengthened by the value of the realized payoff. If instead $\kappa = 0$ this will reduce to the weighted average reinforcement learning model.

$$A_i^j(t) = \phi \cdot A_i^j(t-1) + I(s_i^j, s_i(t)) \cdot \pi_i(s_i^j, s_{-i}(t)) \tag{18}$$

Camerer and Ho (1999) have shown that the difference between reinforcement learning and fictitious play learning is the updating of attractions not only based on realized payoffs but also on forgone payoffs i.e. payoffs that would have been achieved if the player had chosen the other action. Recall that δ is the parameter that regulates how much of an effect forgone payoffs have on attractions so that if both forgone payoffs and realized payoffs are given the same weight then $\delta = 1$. For the model to reduce perfectly to weighted fictitious play it is required that $\kappa = 0$ so that attractions at any time period are averages of previous attractions.

Cournot best response dynamics can be obtained for generalized payoffs from the model by setting $\phi = 0$ in the weighted fictitious play case so that there is no memory beyond the immediately previous time period and $\lambda = \infty$ so that the decision rule best responds. The third necessary condition for generalized payoffs is that $\delta = 1$ so that the model weights both foregone and realized payoffs equally. It is important however to note that given the payoffs of the game used in this experiment $\delta = 1$ is not a necessary condition. This is most easily seen by considering that cournot best response dynamics simply dictate that one should play the best response to the previous action of one's opponent. If $\phi = 0$ the EWA model reduces to:

$$A_i^j(t) = \Big[\delta + (1-\delta) \cdot I(s_i^j, s_i(t))\Big] \cdot \pi_i(s_i^j, s_{-i}(t)) \tag{19}$$

The payoffs used in this game allow the above equation to emulate cournot best response for all values of δ for the following reason. Given an opponent's action, the subject has only two possible actions, one of which has positive payoffs and the other negative payoffs. Hence, no matter what the value of δ the ordering of the attractions of actions remains invariant (since $\lambda = \infty$ the magnitude of the attractions is irrelevant). This would not be the case if given the computer's action the payoffs to the subjects' two possible actions had the same sign.

5.2.2 Estimation techniques of the EWA model The EWA model entails the concurrent estimation of six parameters and complicating matters further is the problem that as λ increases the EWA model slowly morphs from a probabilistic model to a deterministic model of best response. A deterministic best response learning model will not be sensitive to parameter changes and therefore the objective function (or error) surface will be less smooth and even discontinuous. Most typical optimization algorithms such as the various Newton methods involve the use of approximate derivatives to guide

Table 15 EWA models for pooled subjects against each algorithm

	Parameter estimates						Measures of fit	
CA opponent	κ	δ	ϕ	λ	$A_i^j(0)$	$N(0)$	POI	POI-CV
fp2	N/A	N/A	0	10	N/A	N/A	0.317	0.329
fp3	N/A	1	0.301	9.994	35.767	55.614	0.352	0.326
spd	0.27	0	1	7.134	11.015	54.726	0.423	0.41

N/A refers to parameters that are not identified given other estimated parameters

the search in the right direction, which is problematic for a non-smooth model. Hence, for high values of λ gradient descent techniques such as Newton-Raphson would be useless as the algorithm would remain stuck at the initial values.

Another problem concerns the choice of initial values for the optimization procedure because of the large number of estimated parameters which can cause the response surface of the objective function to have a large number of local minima. As a consequence this increases the probability of optimization algorithms becoming stuck at local minima rather than a global minimum.

These two problems are solved using the technique presented in Appendix B, namely by using genetic algorithms to propose a starting or initial point for the Nelder and Mead (1965) Simplex Method, which can deal with discontinuous functions[17].

The choice of the objective function of the EWA model to be minimized is important as it can influence the properties of parameter estimates. The MAD is preferred to the MSD as the error measure of choice for this application on the basis of the arguments detailed in Section C.

5.2.3 EWA modeling pooled by CA opponent EWA models were estimated by minimizing the MAD for all the games played against the same algorithm, leading to three different models of subjects' pooled behavior against each algorithm. The benefits of pooling subjects in this way and estimating single pooled parameters for all individuals rather than different parameters for each individual is that there are more degrees of freedom available for estimation and less chance of overfitting parameters to noise. The final purpose is to allow comparison of the three models, namely to observe the differences in parameter estimates which can be attributable to adaptation and learning conditional on opponent types.

The parameter estimates are given in Table 15 as well as the percentage of inaccuracies (POI), which is simply the percentage of classification errors[18].

An estimated value of ϕ equal to zero means that the model only uses the information from the immediately preceding time period and ignores all other information. Coupled with a high value of λ this means that a player is using cournot best response or the *ws/ls* heuristic. These two heuristics although apparently stemming from two different classes of learning models i.e. cournot best response is a special case of fictitious play and *ws/ls* is a special case of reinforcement learning, are indistinguishable in the 2×2 mixed strategy NE game employed in this study.

Interestingly, against the *fp2* algorithm the value of ϕ is zero, against the *fp3* algorithm 0.3 and against the *spd* algorithm it is virtually equal to one. The latter is consistent with the standard

[17] All computations were performed in Matlab (2007).

[18] Note that the estimated coefficient, λ, is high enough to lead to deterministic choice, so that predictions of the model are equal to zero or one. In this case, the MAD error measure is equivalent to the POI measure.

form of fictitious play where all past information is weighted equally and the player keeps track of the frequencies of past actions of his opponent. Values of λ are high enough to create deterministic choice rather than stochastic choice decisions. Another notable difference is that δ is equal to zero for the *spd* opponent but equal to one against the *fp3* algorithm.

Figures 7, 8 and 9 graph the results of sensitivity analyses on individual parameter estimates, when the other parameters are fixed at their estimated values. This will permit the examination of whether parameters are well identified and how important it is to include them in the model by looking at the effects of their variation on the error measures.

Since the optimal ϕ parameter against the *fp2* algorithm is zero and the estimate of λ is high enough for the decision to rule to produce deterministic behavior, it is apparent from equation 15 that the parameters κ, δ, $N(0)$ and $A_i^j(0)$ are not identifiable. Varying ϕ leads to large variations in the POI and POI-CV, and therefore we can be confident of the importance of this parameter in modeling subjects' behavior. There is no variation between values of zero and 0.2 as the POI measure can be insensitive to small changes in parameter values. The estimated value of λ causes significant changes in classification error rates over values between zero and 0.2 and stabilizes thereafter, implying that subjects' behavior was best modeled by deterministic choice rather than probabilistic choice.

Diverting attention to the EWA model for games played against the *fp3* CA, the optimal ϕ parameter is roughly 0.3, but there is no appreciable variation in the classification error rates for values between zero and 0.3 and therefore for all intents and purposes ϕ can be thought of as being equal to zero[19]. This is further supported by the behavior of the κ, $N(0)$ and $A_i^j(0)$ parameters which do not show appreciable variation in the POI and POI-CV. The only exception is the δ parameter which when varied from a value of zero to one leads to falls in the POI and POI-CV of approximately 2 percentage points. The behavior of λ is identical to the case of the *fp2* algorithm so that optimal classification occurs when choice is deterministic rather than probabilistic.

The results for behavior against the *spd* algorithm are interesting in the sense that they contrast sharply to the two previous algorithms. Values of ϕ between zero and 0.2 now provide the worst fit to the data in contrast to the EWA models of the other two CAs. The POI and POI-CV then exhibit a sharp drop at ϕ values roughly between 0.25 and 0.28 and remain relatively stable until just before a value of one where there is a sharp drop leading to the global minimum. At an optimal ϕ value of one the other parameters of the EWA model are now identifiable and therefore greater sensitivity of classification error rates to changes in these parameters should be observed. This is indeed the case with the exception of κ which varies minimally over the whole range. The value of δ on the other hand leads to an approximately monotone increase in roughly 6-7 percentage points in the POI and POI-CV as it changes from 0 to 1. The classification errors are mostly insensitive to changes in $A_i^j(0)$ except for the region between 0 and 10 which displays a sharp drop of roughly 10 percentage points. The $N(0)$ parameter and the classification errors exhibit a roughly monotone inverse relationship until an $N(0)$ value of approximately 50 with no significant correlation thereafter.

5.2.4 Individual EWA modeling Many studies in the behavioral game theory literature conclude that allowing for heterogeneity by modeling subjects individually is important as assuming homogeneity can lead to seriously biased estimates of learning parameters. This section proceeds with

[19] The classification error rate for the test set is at a minimum for all values between zero and roughly 0.29 so that the small dip in the POI for the training set is likely just a statistical aberration.

Figure 7 EWA (*fp2*) parameter sensitivity analysis

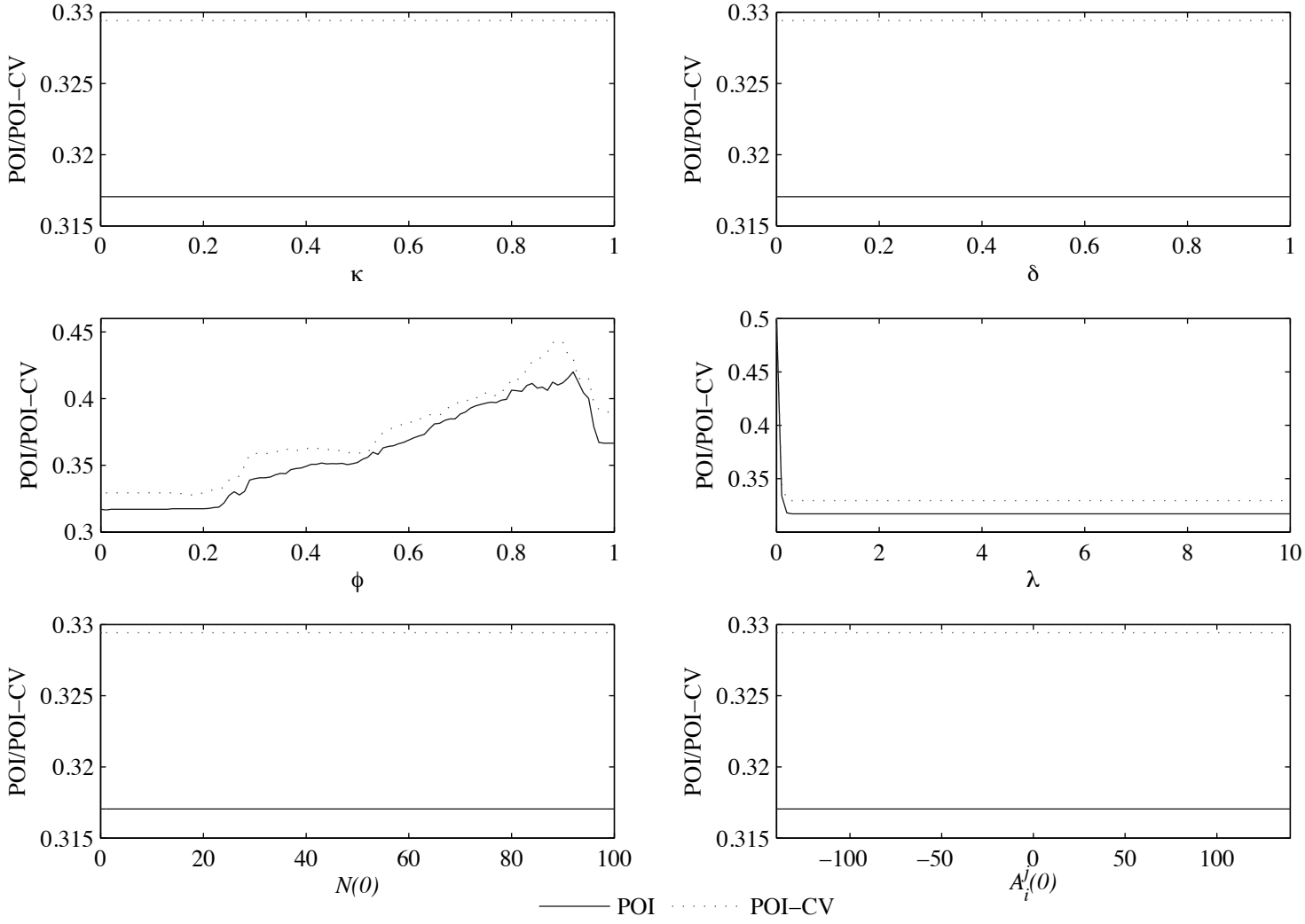


Figure 8 EWA (*fp3*) parameter sensitivity analysis

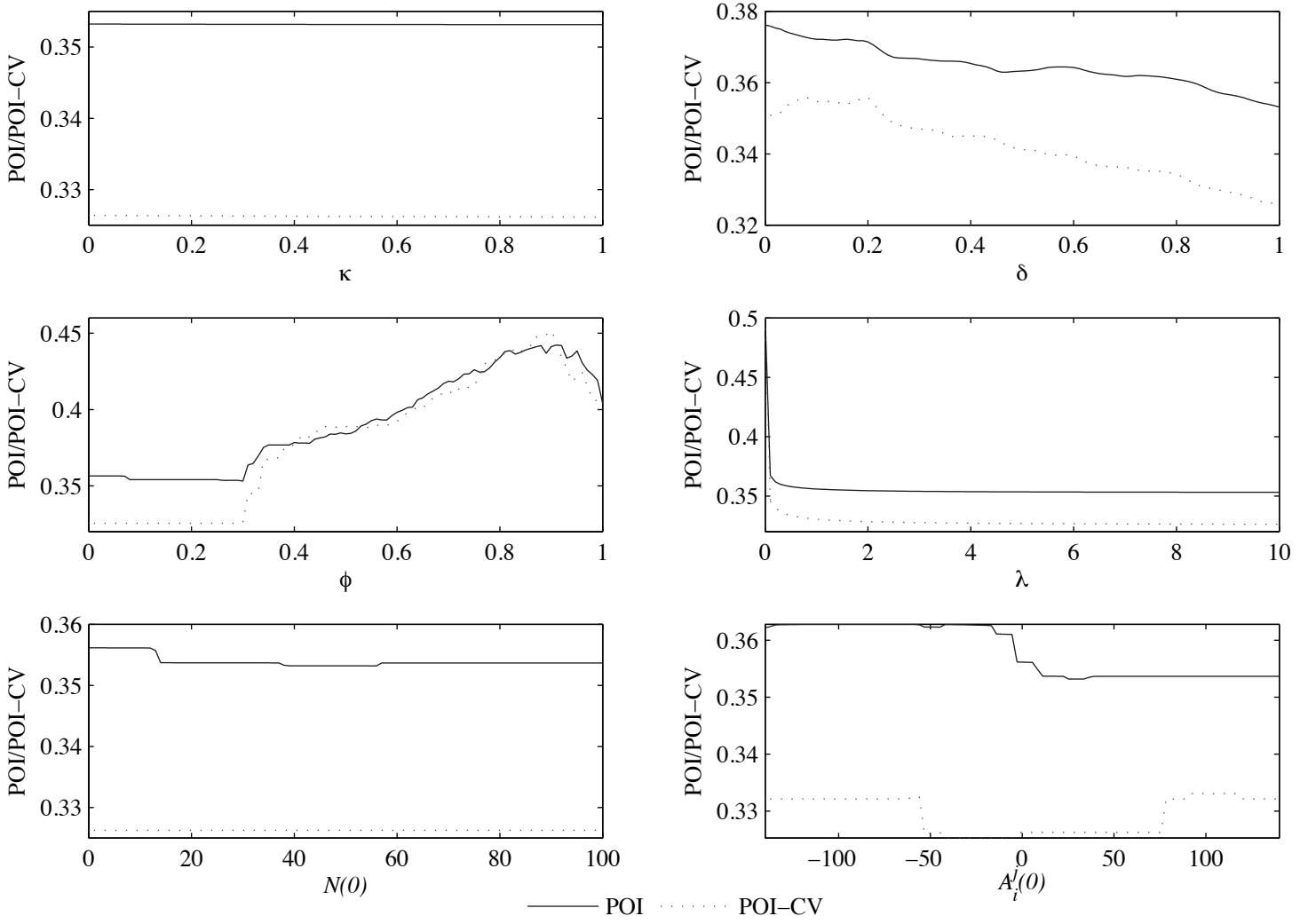


Figure 9 EWA (*spd*) parameter sensitivity analysis

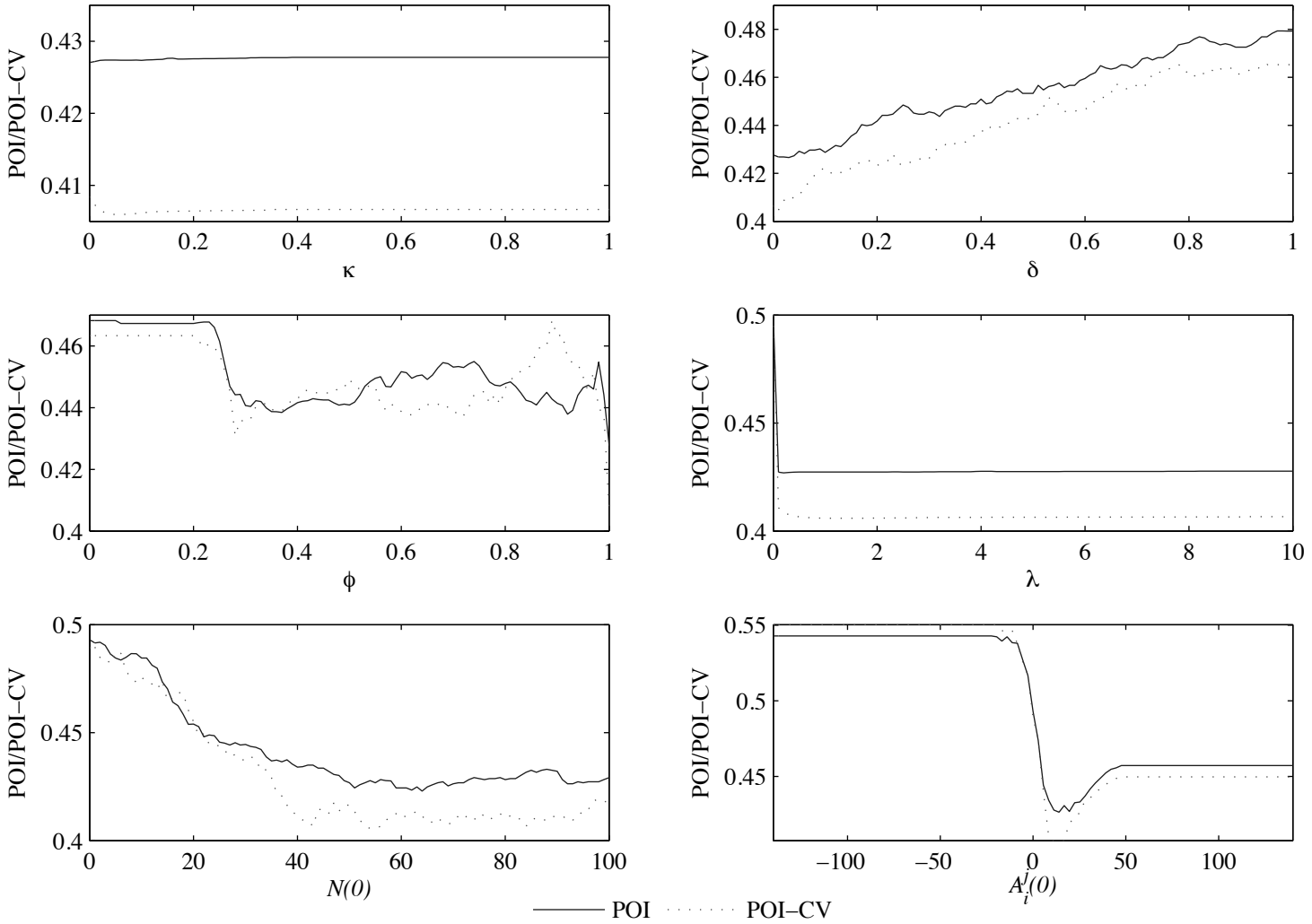


Table 16 Comparison of fit between pooled and individually estimated EWA models grouped by algorithm

	Pooled EWA model		Individual EWA models	
CA opponent	POI	POI-CV	POI	POI-CV
All	36.4%	35.5%	30.9%	36.36%
fp2	31.7%	32.9%	25.97%	33.28%
fp3	35.2%	32.6%	32.33%	35.48%
spd	42.3%	41.0%	34.4%	40.32%

a comparison of EWA models estimated individually and the previously discussed EWA models pooled by computer algorithm opponent. Table 16 communicates the POI of the estimated models for both the training and cross-validation set. The training set measures of fit are not as important as individually estimated models will provide better fits to the training data simply by virtue of the increase in the number of free parameters. This causes a potential problem of overfitting to the training data set without an improvement in the fit to previously unseen data in the cross-validation set. As expected, the POI for the training sets are lower for individually estimated models. However, the individually estimated models on average classify 0.86% points worse on the cross-validation set than the single model pooled against all CAs. Comparing POI-CV for play against the *fp2* CA the pooled model does slightly better whereas against the *fp3* CA the pooled model exhibits a significantly smaller POI-CV, 32.6% versus 35.48%. As regards the *spd* CA, the individually estimated models do slightly better, although the difference is small in magnitude and in economic significance. These results strongly suggest that there does not exist any significant between-subjects heterogeneity and that models pooled by computer opponent, are appropriate in modeling behavior.

The antithesis between the above results and the rest of the literature where modeling subject heterogeneity is found to be important deserves further attention and explanation. The most important distinction to keep in mind is that although this study does not find individual heterogeneity to be important it does find that allowing for heterogeneity conditional on one's opponent is very important. Hence, the results support within-subject heterogeneity which results from adaptation to one's opponent. The within-subjects heterogeneity seems to account for almost all of the heterogeneity in behavior without needing to postulate between-subject heterogeneity. In fact, subjects' behavior is surprisingly similar when facing the same opponents. Since in most studies in the literature subjects' opponents are not experimentally controlled, the observed behavioral heterogeneity was tacitly assumed to arise from between-subjects heterogeneity as there was no way to distinguish between within- and between-subjects heterogeneity.

Table 17 compares the estimated parameters of the pooled models to the average parameters of the individually estimated models. It is interesting to note that the differences in most cases are quite large with averages of individually estimated parameters usually being less extreme than pooled estimated parameters. This is reasonable since many parameters are bounded between zero and one and therefore the variations in estimated parameters occurring due to overfitting must always push the averages to less extreme values or the interior of the bounded interval. Pronounced differences occur for estimates of ϕ and δ with average values for individually estimated models near the middle of the probability space, whilst the pooled estimates lie at the extremes. Figure 10 plots the distributions of the individually estimated values of $\hat{\phi}$, which do not show any significant difference according to the CA opponent implying stability of this parameter, which however is

Table 17 Parameter estimates from pooled and individually estimated models

	Pooled estimated coefficients						Individually estimated coefficients					
CA	κ	δ	ϕ	λ	$A_i^j(0)$	$N(0)$	κ	δ	ϕ	λ	$A_i^j(0)$	$N(0)$
fp2	N/A	N/A	0.00	10.00	N/A	N/A	0.61	0.57	0.41	7.92	20.77	88.74
fp3	N/A	1.00	0.30	9.99	35.77	55.61	0.79	0.46	0.39	6.72	35.35	85.93
spd	0.27	0.00	1.00	7.13	11.02	54.73	0.80	0.36	0.51	8.07	-1.43	52.18

Table 18 Comparison of cross-validation performance of EWA and probit models

	Pooled EWA model		Pooled probit models (5 lags)		Pooled probit models (1 lag)	
CA	POI	POI-CV	POI	POI-CV	POI	POI-CV
fp2	31.7%	32.9%	32.16%	33.97%	31.24%	32.76%
fp3	35.2%	32.6%	34.36%	34.17%	35.13%	32.66%
spd	42.3%	41.0%	39.07%	37.30%		

clearly not the case according to the pooled parameter estimates. Also, against the *spd* CA κ is equal to 0.27 for the pooled model but equal to 0.8 for the individually estimated models. It is clear that the parameter estimates of these individually estimated models lead to wrong conclusions as to the real values of the learning parameters. This is an important qualification since at the extremes the EWA model is equivalent with simplified learning rules, such as the *ws/ls* heuristic, weighted fictitious play or standard reinforcement learning, and therefore it would lead us to reject simple heuristics in favour of unnecessarily complicated models. Tables 23, 24 and 25 in Appendix D report the individually estimated parameter values for all subjects against all computer algorithm opponents.

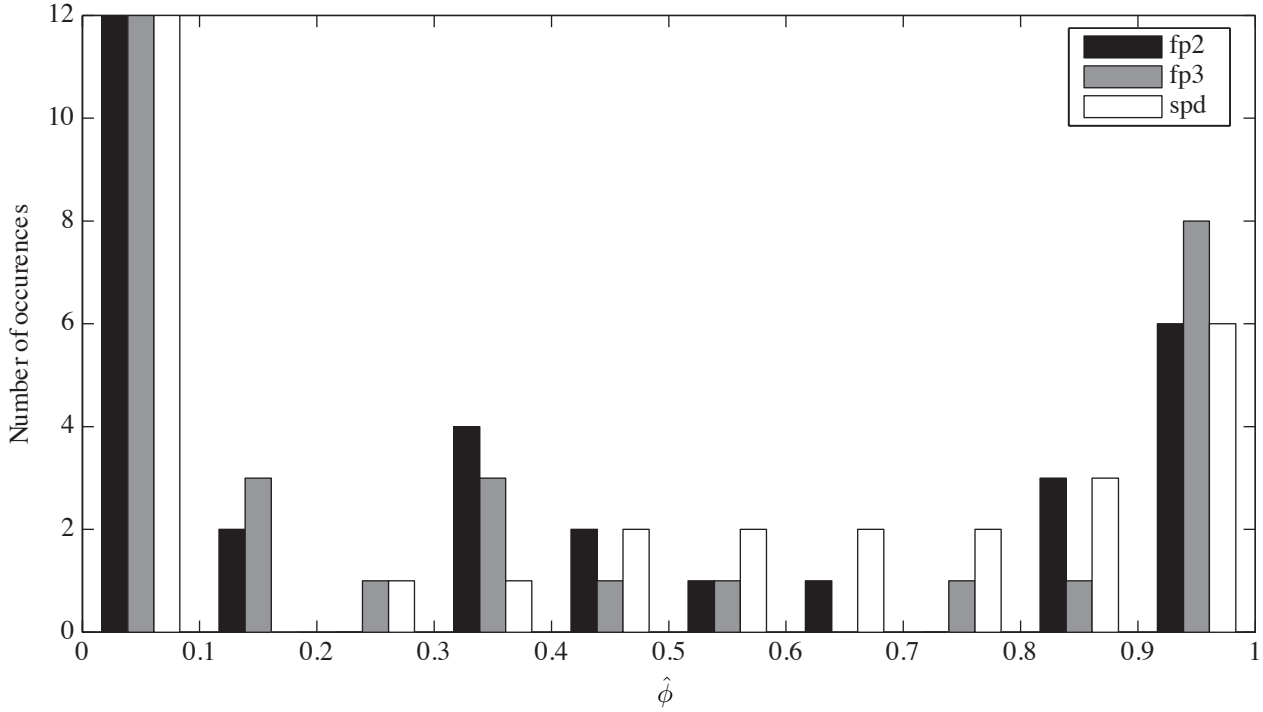
5.3 Comparison of EWA and probit models

The EWA and probit models presented above can be thought of as competing models of behavior. The most important differences between these two models is that the probit models have a discrete window of lagged observations whereas the EWA model uses all past information with exponentially decreasing weights (expect in the degenerate case where the memory parameter is equal to zero). Also, the probit model presented here did not directly incorporate payoff information[20], whereas the EWA model does so explicitly. A comparison of the performance of the probit and EWA models on training and cross-validation data yields some interesting conclusions. The accuracy of the various models on training and cross-validation data is recompiled in Table 18 for ease of comparison.

The last two columns document the models from Section 5.1.1 that included only one independent variable, the first lagged computer action, which were found to perform better out of sample than the 5 lag models of both own and CA's actions. In terms of the POI-CV of models of play against *fp2* and *fp3*, the EWA models and the 1-lag probit models are virtually identical. Taking into account that the probit model has significantly fewer free parameters than the EWA model, it is clear that it is the preferable model. On the other hand, comparing the EWA and 5-lag probit model for the *spd* CA the probit model performs significantly better than the EWA model, as is exemplified by a

[20] Although it does so indirectly because it includes the two determinants of payoffs, own and opponent's actions.

Figure 10 Histogram of $\hat{\phi}$



POI-CV of 37.3% versus 41% respectively. This difference is likely due to the fact that the probit model can better capture a non-monotonic dependence on lagged behavior, whereas the EWA model must have a decreasing dependence on the history of play because of the built-in exponential memory function[21].

On the basis of this information, the probit model appears in general to offer better predictive power on cross-validation data than the EWA, in particular whenever it is important to have a more flexible memory structure. However, this is conditional upon knowing what kind of opponent a player is facing. Therefore, in an uncontrolled environment where one's opponent is not known it is still possible for EWA models to have better predictive power than a probit model, particularly the 1-lag model, as the larger number of parameters of the EWA model will be better at representing the effects of opponents' heterogeneity in play. A clear distinction regarding how the value of models is measured is in order. If models are valued on the basis of how well they fit the true underlying behavior of subjects then comparisons should be made between models estimated against the same CA opponent, as is primarily done in this study, through the use of a human/computer experimental setup. If on the other hand, one is interested in the predictive power of a model in an uncontrolled environment, then the fit of the competing models should be compared when subjects are facing a wide variety of opponent types, thereby indirectly incorporating information about the relative prevalence of these different types in a population of players in the estimated parameters. In this case one can compare the single EWA and probit models pooled across all subjects, and indeed the POI-CV of the EWA is now smaller than that of the 5-lag probit models, 35.5% compared to 36.16%.

6 Analysis of questionnaire data

The style of analysis in this section shifts toward a more qualitative nature in the hope that it will provide insights which the quantitative models presented above cannot. After every 25 rounds subjects were asked to state what their decision strategy was in their own words, to provide clear examples of human strategizing. This question was open-ended to avoid leading subjects to specific strategies and to allow subjects to express any strategy without restriction to prespecified answers. The benefit of this is that it allows researchers to inspect individual thought processes, on the other hand however it would be desirable to be able to group answers into more generalized categories in order to infer trends across all subjects. Hence, each answer was coded as falling into either of the following categories: fictitious play, pattern detecting fictitious play, reinforcement learning, pattern detecting reinforcement learning, other types of reasoning, *ws/ls* heuristic, and random. The results of this categorization are displayed in Table 19. The difference between fictitious play (and reinforcement learning) with respect to pattern detecting fictitious play (and pattern detecting reinforcement learning) is that in the latter categories subjects displayed some type of multi-period or sequential thought instead of simply looking at single-period actions. Examples are given below of actual answers of subjects and how they were classified.

Fictitious play "The computer played white more often so I played yellow more often", "I think it started off playing white more often but then started playing brown more often"

[21] Referring back to Table 11, against the *spd* algorithm the coefficient of the first lag is smaller in magnitude than that of the second lag i.e. dependence is indeed non-decreasing, a property that will hinder the effectiveness of the EWA model.

Table 19 Classification of players into types based on the questionnaire answers

	Fictitious play		Reinforcement learning				
CA	single period	pattern	single period	pattern	*ws/ls*	Other reasoning	Random
All	12.1	8.9	6.7	12.1	2.2	4.6	53.5
fp2	11.3	10.5	6.5	10.5	2.4	5.6	53.2
fp3	11.3	7.3	5.6	15.3	1.6	3.2	55.6
spd	13.7	8.9	8.1	10.5	2.4	4.8	51.6

Pattern fictitious play "I thought the computer chose to play the same colour very often", "I am trying to figure out how many times in a row it uses each colour"

Reinforcement learning "I tried to count how often I won playing each colour", "I made my choice according to the changes in my score"

Pattern reinforcement learning "Playing three times yellow and one time blue seems to be a profitable combination", "When I lost two consecutive times playing blue I would play yellow and vice versa"

Other reasoning "I played yellow so as not to lose 80 points", "Normally you should play only blue because if you win you have made up for three losses"

Win/stay, lose/shift heuristic "Usually whenever I lost I would change colour", "If I lost playing yellow I would then choose blue and vice versa"

Random "I was playing according to chance", "I had no strategy"

From Table 19 it is evident that playing against different algorithms does not in general significantly affect the proportions of subjects classified under each type. The only striking difference occurs when comparing *fp2* and *fp3* computer opponents as there is a large increase in pattern reinforcement classification rates for the *fp3* algorithm opponent. The class of fictitious play learning algorithms account for 21% of responses with the class of reinforcement learning algorithms also accounting for 21% of responses[22]. A large percentage of responses stated that decisions were made randomly, 53.5%, and a much smaller percentage, 4.6%, gave some other kind of reasoning for how they were playing.

The most striking disparity between the data collected from the questionnaire and actual play is the fact that although *ws/ls* behavior is rampant, only a small percentage of responses acknowledged using it. This disparity begs the question of whether the use of the *ws/ls* heuristic is a controlled or automatic process as distinguished by Camerer et al. (2005). Controlled processes of neural functioning involve effortful serial processing and are invoked deliberately, implying that humans should have good introspective access to these processes. Automatic processes involve parallel processing of information, are effortless and people will have little introspective access as to how or why decisions were made. Since the subjects were not in general able to explain the use of the *ws/ls* heuristic, displaying little introspective access, this implies that in many cases the *ws/ls* heuristic may be an automatic process. Supporting this conclusion is the observation that such automatic processes are usually the result of the sculpting of the human brain by evolutionary pressures (Cosmides and Tooby, 1987). This would be very reasonable in the case of the *ws/ls* heuristic because it works admirably in any

[22] Although in this game the *ws/ls* heuristic makes equivalent decisions with a fictitious play learning rule with $\gamma = 0$, it has been included under the class of reinforcement learning rules because in every single case subjects described it in reinforcement learning terms rather than fictitious play terms.

Table 20 Predictions of own and opponents' first-order play

	Opponents' actions (white action)			Own actions (blue action)		
CA	% predicted	% actual	MAD	% predicted	% actual	MAD
All	51.24%	66.15%	20.98	41.13%	41.16%	11.60
fp2	53.41%	69.59%	23.83	39.07%	37.48%	11.38
fp3	51.90%	67.81%	21.26	38.65%	40.58%	12.71
spd	48.25%	60.75%	17.71	46.00%	45.61%	10.61

general situation where the environment exhibits relatively stable conditions, structured order and correlated behavior[23].

Given the distinction between automatic and controlled processes regarding introspective access, answers to the questionnaire can by definition only provide controlled processes as answers. When subjects answered that they played randomly or had no strategy they are referring to controlled processes. Hence, the abberation between the large percentage of answers categorized as random, the very small percentage of answers that explicitly stated the *ws/ls* heuristic but the widespread use of the *ws/ls* heuristic can be explained by the fact that the subjects' use of the *ws/ls* heuristic is predominantly an automatic process[24].

6.1 Accuracy of predictions of own and opponents' first-order play

At the end of 100 rounds of play against each CA each subject was asked to state the number of times she thought she had played blue in the last 100 rounds and also the number of times that her opponent had played white. Table 20 tabulates the predicted proportions and the actual proportions of own and opponents' play pooled against all algorithms and also by each computer opponent.

Predictions of opponents' play are not calibrated well with an average prediction of white play (against all algorithms) of 51.24% versus an actual value of 66.15%, whereas predictions of own play are surprisingly well calibrated with a predicted value of 41.13% versus 41.16%. Testimony that this result is important is the fact that even when predictions are broken down into each algorithm subjects were still well calibrated, with a maximum deviation of only 1.93% points. A measure of individual accuracy of predictions is the average individual absolute deviations of predictions from actual play or MAD. The difference between own and opponents' play is again striking as the MAD is much smaller for own play, 11.6%, compared to opponents' play, 20.98%. A Wilcoxon signed ranks test concludes that this difference is significantly different from zero at a p-value of less than 0.1% (z-value=-4.308).

The results for both the *fp2* and *fp3* CAs are very similar but against the *spd* algorithm the results are strikingly different. Calibration, especially for opponents' play, improves significantly as the difference between predicted and actual white play is 12.5% points for the *spd* algorithm,

[23] The notions of controlled and automatic processes are very similar to the notions of implicit and explicit learning, especially with respect to whether subjects consciously acknowledge them or whether they occur at a subconscious level.

[24] Different regions in the brain tend to be active depending on whether the subject is using a controlled or automatic process. Hence, the conjecture that the *ws/ls* heuristic is to a large extent an automatic process could be tested more rigorously by a neuroeconomic experiment that implements some technique of brain scanning whilst subjects are playing the game.

Table 21 Spearman correlations of predicted frequencies of own and opponents' play against actual play

	Prediction of own action frequencies		Prediction of opponent's action frequencies	
CA	ρ	p-value	ρ	p-value
All	0.3887	0.0002	0.077	0.4757
fp2	0.5671	0.0013	0.0755	0.6970
fp3	-0.0125	0.9467	0.2195	0.2354
spd	0.4646	0.0127	0.0271	0.8911

in comparison to 16.18% points and 15.91% points for the *fp2* and *fp3* algorithms respectively. Also, the MAD is lower when subjects played against the *spd* computer algorithm, especially for predictions of opponents' play. The MAD for predictions of opponents' play falls from 23.83 and 21.26 for *fp2* and *fp3* respectively to 17.71 for the *spd* algorithm. This improvement in predictions can be explained by referring back to the results from the EWA modeling - namely that against the *fp2* and *fp3* algorithms subjects predominantly used the *ws/ls* heuristic which does not require knowledge of own and opponents' play over the whole game whereas against the *spd* algorithm the estimated memory parameter was equal to one which implies the utilization of all past play of one's opponent. Hence, more attention must have been paid to keeping track of these variables leading to the improvement in calibration. However, conducting Friedman tests with the null hypothesis that the average MAD for predictions of own and opponents' play is the same across all CAs it is not possible to reject the null hypotheses at the 5% significance level. The χ^2 (and associated p-values) of MAD predictions of own and opponents' play are 1.52 (0.468) and 2.742 (0.254) respectively. Despite this failure to reject the null hypothesis, other evidence such as the EWA regressions which in the case of the *spd* algorithm concluded that players did use past information about opponents' actions, strengthens the argument that this result is not a statistical artifact. However, future studies should attempt to verify this by increasing the number of subjects and the power of the test to reject the null hypothesis.

The Spearman correlations of predicted frequencies of play on actual frequencies of play are given in Table 21. The Spearman correlation for own actions against all of the CAs, is much higher than that of opponent's actions, 0.39 (p-value=0.0002) versus 0.077 (p-value=0.4757) respectively, with only the former exhibiting statistical significance. Such a large difference is important as it is further strong evidence that subjects were not monitoring their opponents' first-order behavior closely. Breaking down the analysis by CA opponent reveals an interesting result. The Spearman correlation for own actions falls to zero in games against the *fp3* algorithm in stark contrast to the large in magnitude and statistically significant estimates against the *fp2* and *spd* algorithms. At the same time, against the *fp3* CA, there is an increase in the ρ value for opponent's action frequencies, which although not statistically significant at the 5% level, is appreciably higher in magnitude than the correlation against the other CAs. These simultaneous shifts in opposite directions point to a significant change in subjects attention to the history of play, and have occurred against *fp3*, the CA that was the best at detecting patterns and by implication the best at detecting conditioning on subjects' own history. Hence, the direction of the shift in attention could be considered to be a better response on behalf of the subjects.

7 Optimality of subjects' strategies

Agent-based computational economics (ACE) are increasingly being used to model the interactions of individual economic agents and to observe the resulting complex, non-linear dynamics of populations of agents, thereby allowing researchers to overcome some of the limitations of experimental economics[25]. Typically, heterogeneous agents are modeled as computer algorithms, and simulations of the interactions of a population of these different agents are performed. Simulations have been utilized in game theory to examine the evolutionary properties of agents and strategies such as in Axelrod (1985), where a tournament of different Prisoner's Dilemma strategies was conducted.

In order to assess how well subjects in the experiment are responding to the computer algorithms it is crucial to have some notion of what the best response to these algorithms is. This is not trivial as the CAs employed in this study are complex, and it is only through simulations that one can attempt to tackle this problem. However, there is no guarantee that an optimal strategy has been found as it is impossible to try all possible strategies against these algorithms. Instead, an approximate approach is condoned where a relatively broad sample of learning rules will be implemented. In particular, a wide class of reinforcement learning algorithms will be examined, with different depths of pattern recognition and values of the memory parameter. In one case it will be useful to also include a simple mixed strategy with varying first-order play. From this mini-tournament of learning rules we hope to get an approximate best response strategy against the CAs and to gain important insights into the evolutionary value and fitness of these learning rules.

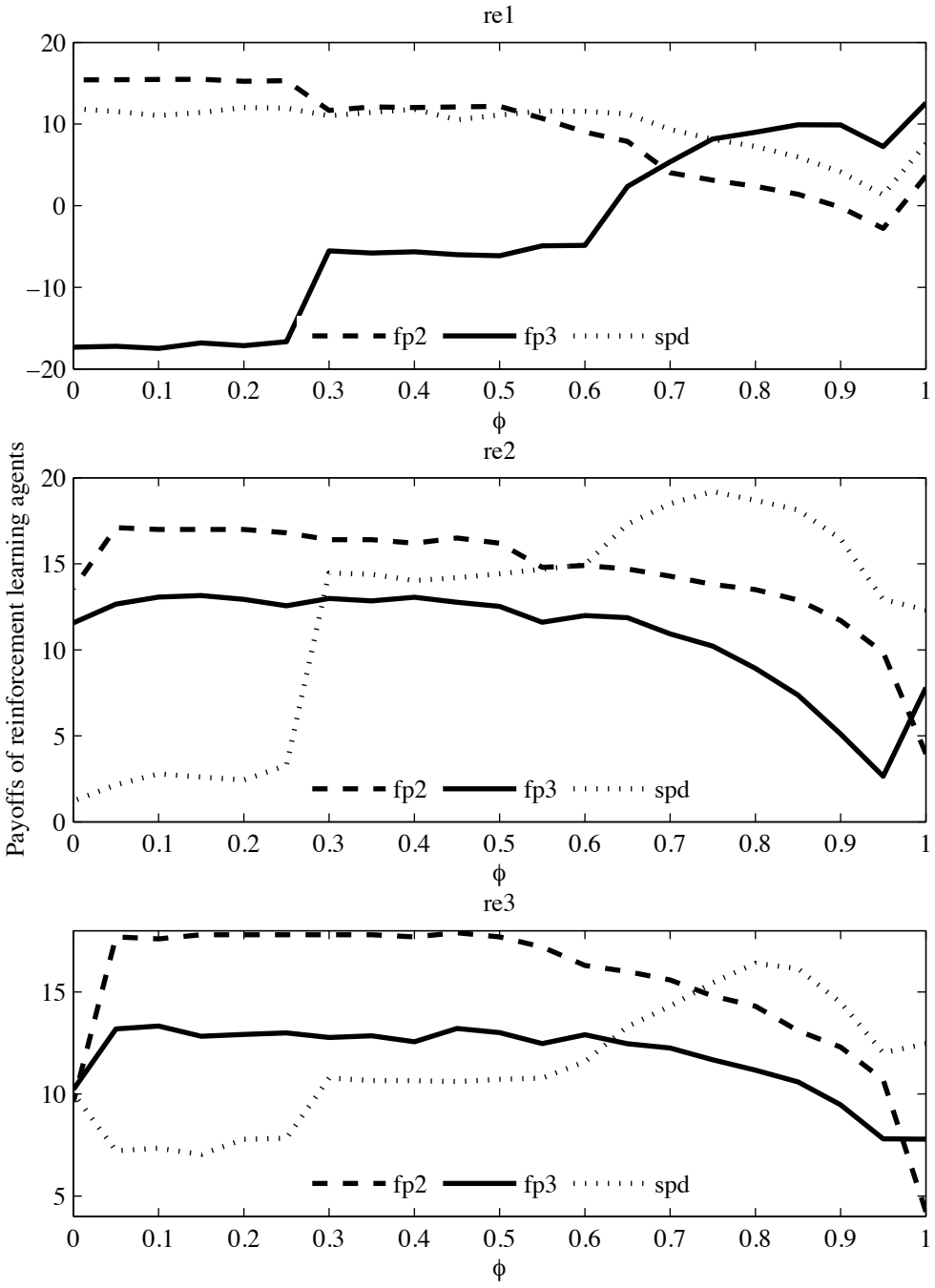
More specifically the three algorithms used in the experiment, *fp2*, *fp3* and *spd* were in the pool of algorithms and were pitted against three variants of reinforcement learning. *Re1* is a standard reinforcement learning algorithm that reinforces single period actions according to realized payoffs, *re2* and *re3* are variants which reinforce two and three consecutive period actions allowing for the detection of which patterns of responses are more successful. Each one of the reinforcement learning algorithms was matched against the *fp2*, *fp3* and *spd* algorithms. All the reinforcement learning algorithms were simulated with varying memory parameters, namely from zero to one in steps of 0.05 and their decision rules were deterministic best responses. Figures 11 and 12 graph the results of these simulations, both display the same information but in the former figure the results are grouped by the reinforcement learning algorithm and in the latter they are grouped by the CA opponent.

7.1 Simulations of reinforcement learning algorithms versus implemented CAs

Against the *fp2* algorithm, all the reinforcement learning algorithms achieve higher than MSNE payoffs for low values of γ but fall below MSNE payoffs for higher values. The highest payoff was achieved by *re3* which strictly dominated the other two models for all values of γ greater than roughly 0.05. *Re2* and *re3* are more robust to changes in γ as they earn greater than MSNE payoffs for γ up to roughly 0.95. It is important to note that *re1* with γ equal to zero, which is essentially the *ws/ls* heuristic, fares very well especially considering its computational simplicity. Even more impressive is the fact that its average payoff of 15.41 is considerably higher than the MSNE payoffs of 10 and relatively close to the highest possible average payoff of 17.9 accomplished by *re3* with γ equal to 0.45. As would be expected, an *re1* algorithm with high γ can be exploited by the *fp2*

[25] For an overview of agent-based computational economics applications the reader is referred to Tesfatsion and Judd (2006).

Figure 11 Payoffs to simulated agents playing against the computer algorithms (grouped by simulated agent)



algorithm which is capable of detecting two-period patterns whereas the *re1* algorithm reinforces only single-period actions. The maximum payoff achieved by the *re2* agent is 17.1 which is extremely close to that of the the *re3* agent and therefore the *re2* model is probably the most effective algorithm once the increased computational costs of implementing *re3* rather than *re2* are considered.

Against the *fp3* algorithm the picture is similar as all three reinforcement learning algorithms earn more than MSNE payoffs for small memory parameter values whilst performance declines signif-

icantly for longer memory structures. *Re3* achieves the highest payoff, 13.33, of all three algorithms at γ equal to 0.1. However, this is only slightly higher than the maximum of the *re2* algorithm, 13.15, and therefore after taking into account computational costs *re2* would again probably be the best option. Once more, given its simplicity the *re1* model with zero memory does extremely well although again the payoffs it achieves are lower than those associated with *re2* and *re3*. The *re1* agent's maximum payoff is 12.02 with γ equal to 0.2 with the average payoff at γ equal to zero only marginally less at 11.89, these results are roughly 20% higher than the MSNE payoffs. When faced with the *fp3* CA, since it detects three-period patterns it would be reasonable to expect that the *re2* algorithm would be more susceptible than when it is playing against an *fp2* algorithm. This is indeed the case as it dips below MSNE payoffs at a much lower value of γ, namely at 0.8.

Turning now to the *spd* algorithm, due to the very different nature of this algorithm compared to *fp2* and *fp3* it is expected that the results of the simulations will be very different. This is indeed the case, with the most striking feature being the reversal of the robust inverse relationship between memory depth and payoffs found for *fp2* and *fp3*. For all three *re* algorithms, payoffs were lowest for low values of γ, but these algorithms could achieve greater than MSNE payoffs by increasing their memory depth. *Re1* is strictly dominated by *re2* and *re3* over all parameter values whilst *re2* achieved the highest payoffs this time (but did not strictly dominate *re3* which did significantly better for low γ). The highest average payoff of 19.2 was achieved by *re2* when γ equaled 0.75, making the *spd* algorithm the worst performer when pitted against the chosen agents and optimal γ values. *Re1* average payoffs are below MSNE payoffs for all values of γ except one. This particularly bad result breaks the previous excellent results of *re1* and in particular of the *ws/ls* heuristic. However, this was to be expected as the *spd* algorithm was designed specifically to exploit *ws/ls* behavior in opponents.

Simulating various i.i.d. mixed strategies against the *fp2* and *fp3* algorithms would not be particularly interesting because these algorithms would obviously be able to take advantage of such deviations from the MSNE. However, the *spd* algorithm is not designed to directly exploit such behavior, and therefore mixed strategies of varying probability distributions over actions will be simulated against the *spd* algorithm in order to ascertain whether there may be a relatively easy way of exploiting this algorithm. The results are given in Figure 13 which plots average payoffs for varying values of the average proportion of blue actions played by the agent. The highest possible payoff of 12.52 is achieved by playing blue roughly 55% of the time, so that an improvement of roughly 25% over MSNE payoffs was possible against this algorithm only by modifying first-order play. However, this improvement is much less than the improvements achieved by the *re2* and *re3* algorithms and therefore there is a significant incentive for agents to switch to these algorithms despite the increased computational cost.

7.2 Have subjects learned the approximately optimal strategies derived from the simulations?

The EWA and probit models found subjects to be predominantly using the *ws/ls* heuristic against the *fp2* and *fp3* algorithms. Although the *ws/ls* heuristic (executed perfectly with no error or noise), or *re1(0)*, is a very good general strategy against the *fp2* and *fp3* algorithm as it earns more than the MSNE payoffs, subjects often did not perform significantly better than MSNE payoffs probably because of the noise in their decision processes. As a result of this, human subjects playing

Figure 12 Payoffs to simulated agents playing against the computer algorithms (grouped by computer algorithm)

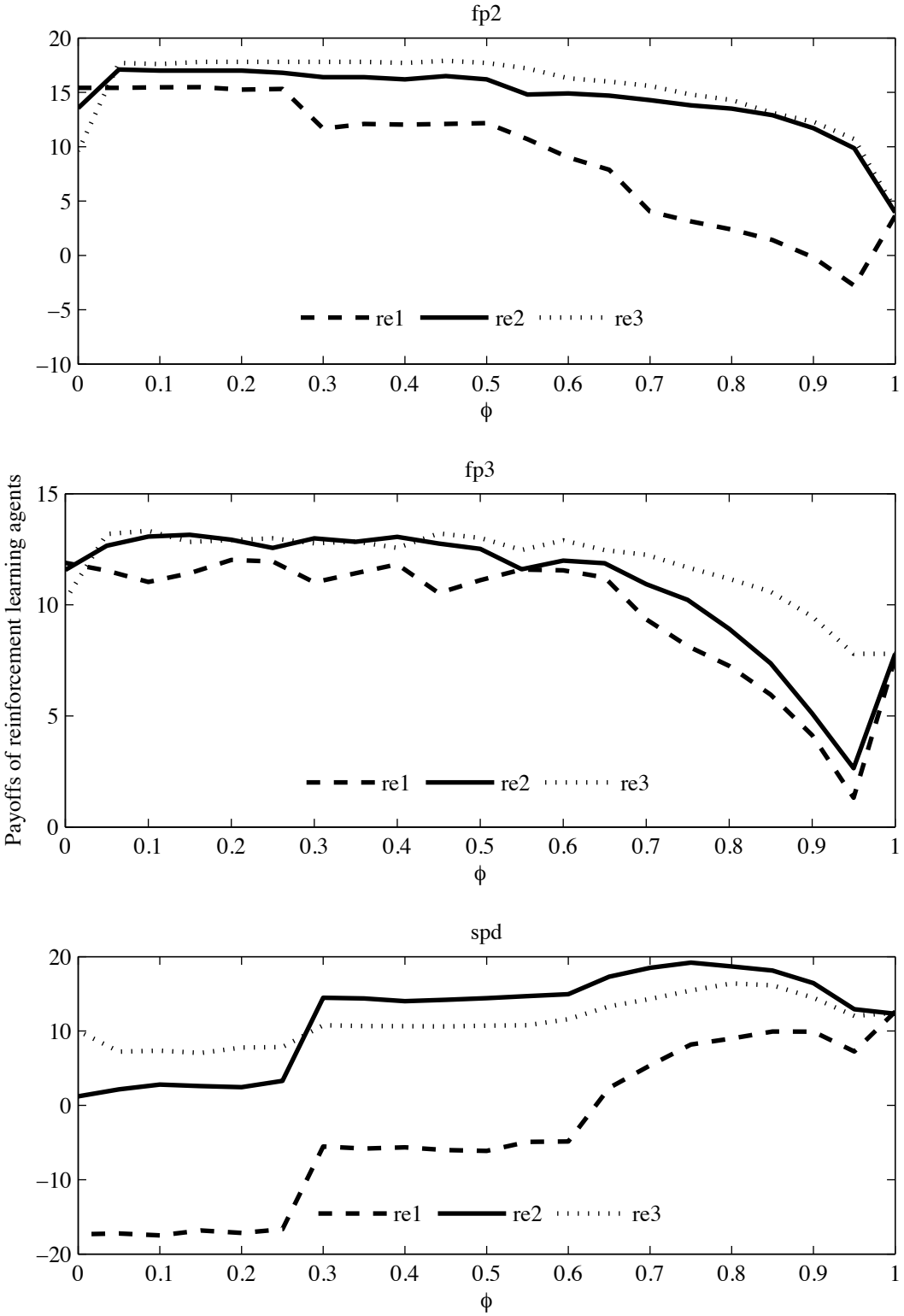
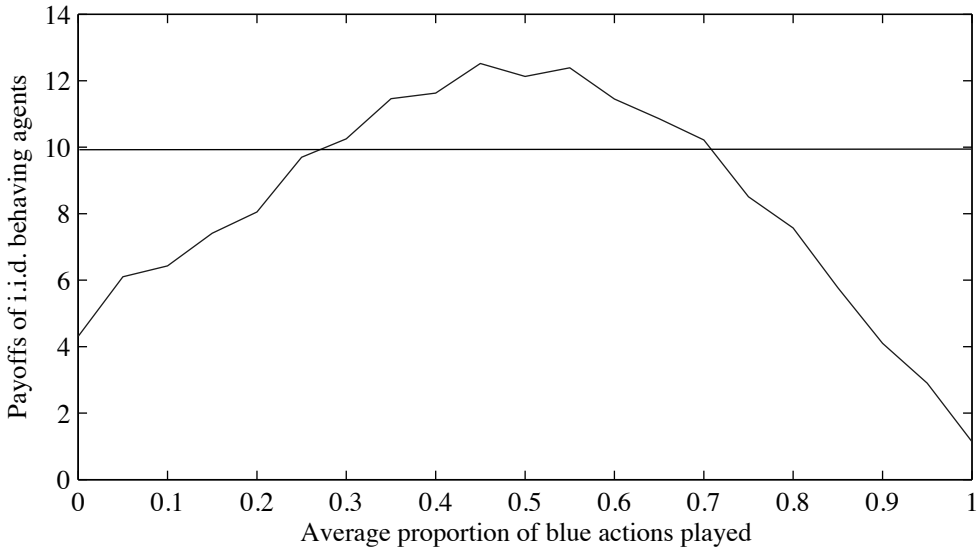


Figure 13 Payoffs to simulated agents with fixed i.i.d. proportions of actions against the *spd* CA



against the *fp2* algorithm did not earn payoffs significantly higher than the MSNE payoffs and against the *fp3* algorithm in the first position ended up making significantly less than this. In all fairness however, adopting *re2* and *re3* would have significant additional computational cost without necessarily large increases in payoffs. Against the *spd* algorithm humans did much better although again not as well as the simulated reinforcement learning algorithms suggest they could. However, average payoffs are quite close to what could be expected if subjects simply altered their first-order behavior (assuming i.i.d. behavior). In this case compared to the proportion of actions played against the other two algorithms, human subjects moved towards and ended up very close to the optimal first-order frequency of play against the *spd* algorithm of roughly 0.5, as shown in Figure 13.

Subjects displayed the ability to detect significant changes to optimal strategies such as the superiority of a small ϕ value against *fp2* and *fp3*, versus the superiority of a large ϕ parameter against *spd*. From the results in Section 4, they clearly modulated the properties of their nth-order behavior according to the ability of each algorithm to exploit deviations from i.i.d. behavior. Subjects shifted from underalternation against *fp2* and *fp3* to overalternation against the *spd* CA. Also, the number of players that were not deemed to be playing independently was much higher for the *spd* algorithm, which is not as adept at detecting such deviations from MSNE play as the *fp2* and *fp3* algorithms. From the probit regressions in Section 5.1.1, against the *spd* CA they quickly learned to avoid strongly conditioning on C_{t-1} which they had done against *fp2* and *fp3*, and instead conditioned strongly on C_{t-2} which the *spd* algorithm was not designed to detect. This matches well with the simulations against the *spd* algorithm where *re2* and *re3* clearly dominate *re1*, from which it can be inferred that higher-order information is indeed useful against the *spd* CA.

In conclusion, subjects avoided strategies that were clearly inferior and altered their strategies in the direction of best response, but did not perfectly adopt the optimal strategies derived from the simulations.

8 Conclusion

The purpose of this paper is to offer novel analyses to questions of strategy and behavioral modeling of humans engaged in games with a unique mixed strategy Nash equilibrium through the use of human versus computer algorithm experiments. The increase in experimental control through the use of computer algorithms as subjects' opponents allows for more powerful tests of hypotheses and also allows new hypotheses to be tested. As a consequence, it is possible to disentangle within-subjects heterogeneity resulting from adapting to opponents' behavior, and between-subjects heterogeneity resulting from differences in individuals' strategies against the same opponents.

The most important results to be derived from this research are the following. Subjects were found to adopt extremely similar strategies against the same opponents, so much so that models pooled by computer opponent (CA), did on average better at predicting out-of-sample behavior than individually estimated models. Hence, it is clear that between-subjects heterogeneity is not particularly important in modeling behavior. In contrast, it was found that subjects' behavior was strongly conditioned on their computer opponent, thereby exhibiting within-subjects heterogeneity. Subjects' play changed radically as the estimated memory parameter in the EWA models, which lies on the interval between zero and one, was estimated as being zero against two computer opponents and one against the other. In the probit models, conditioning against the *fp2* and *fp3* CAs was primarily on the first lag of the CA's action but against the *spd* CA this changed significantly as conditioning switched to the second lag of the CA's action. These results confirm that within-subjects heterogeneity is extremely important and is the major driver of the general heterogeneity observed in subjects' play.

Another important result is the effectiveness of simple heuristics in predicting out-of-sample behavior of subjects compared to more complex alternatives. Once experimental control is restored by using computer algorithms as subjects' opponents, the EWA models exhibit estimated parameters at the extremes of the EWA cube, where this rule collapses into simple heuristics like the win-stay/lose-shift heuristic if the memory parameter is zero or standard weighted fictitious play if it is equal to one i.e. perfect memory. These results are at the heart of what Gigerenzer (2000) and Gigerenzer and Selten (2001) refer to as an adaptive toolbox. They contend that humans are equipped with a toolbox of simple yet effective heuristics that have been sculpted by evolution to fit the properties of the environment, instead of the complex models usually postulated by economists.

Subjects were initially challenged to find effective counterstrategies to a pattern-detecting fictitious play algorithm which utilized the history of three consecutive time period actions of the subjects. They earned significantly less than MSNE payoffs when this was the first algorithm they faced, but with more experience they managed to raise payoffs to the MSNE level when they played this algorithm last. They were much more effective at outperforming a two-period pattern detecting fictitious play algorithm and an algorithm that detects whether subjects are conditioning play on the computer algorithm's previous action (this includes the popular win-stay/lose-shift heuristic).

Subjects exhibited surprisingly poor performance when asked to state what they thought was the frequency of the CA's actions in past rounds, whilst displaying significantly higher accuracy for their own actions. This may be evidence that reinforcement learning is a better representation of learning than fictitious play since the latter requires information about the probability distribution over opponents' actions but not over own actions. This striking result warrants further investigation and in particular should be expanded to examine n-period combinations of actions.

Transfer of learning was exhibited by subjects as they learned not only within rounds of play against the same algorithm, dubbed opponent-specific learning, but also across CAs, specifically between the first two of the three CAs presented to them, referred to as game-specific learning.

Simulations of pattern-detecting reinforcement and fictitious play algorithms competing against each other yielded some interesting conclusions. The first was the relatively good performance of the win-stay/lose-shift heuristic against much more complex algorithms such as two- and three-period pattern detecting fictitious play algorithms. This lends credibility to the evidence that subjects often bring the win-stay/lose-shift heuristic into experimental studies, as it is a strategy that seems to be effective in a wide variety of environments, especially ones exhibiting positive serial correlation. These simulations provided approximately optimal strategies against each CA, which can be used as a benchmark to determine whether subjects learned to adapt their behavior towards these strategies. In particular, first-order play, the use of the win-stay/lose-shift heuristic, the i.i.d. properties of behavior and the memory depth they employed were all found to differ significantly against the CAs indicating that subjects learned to tailor these strategies to each CA opponent. More importantly, the changes in behavior of these three strategies were in the direction of better responding to each CA, although subjects did not employ them optimally. For example, subjects' actions were closer to i.i.d. behavior when they faced the three-period pattern detecting algorithm, they stopped using the win-stay/lose-shift heuristic when playing against the *spd* CA which could detect this and they also increased their memory depth in the EWA model drastically against the *spd* CA which is exactly what the simulations prescribed.

The results from this study were extremely similar to the results of a similar study conducted with monkey subjects playing against computerized opponents. In particular, both studies used similar games and computer algorithms as opponents, and the estimated memory parameters of behavioral models were virtually identical, including the changes in these parameters with different computer opponents. From this it can be inferred that the learning process was likely an automatic or subconscious process, as otherwise it would be reasonable to assume that human and monkey behavior would diverge significantly.

Concluding, the increase in experimental control, through the use of computer algorithms as humans' opponents in this study, has allowed the disambiguation of within-subjects heterogeneity from between-subjects heterogeneity which was blurred in prior human versus human experiments. Furthermore, it has led to the observation that simple heuristics can explain out-of-sample data against the same opponent better than the complicated models usually postulated in the literature, such as EWA, which had a tendency to overfit the data. The simulations of various agents allowed the discovery of near optimal strategies against the computer algorithms used and provided insights as to the evolutionary fitness and robustness of various strategies when matched against each other. These conclusions were possible only due to the use of computerized agents or algorithms, testifying to the usefulness of such techniques in the research methods of experimental game theory.

Direct extensions of this work for future research could include a larger variety of computer algorithms, including pattern detection capabilities of higher order pattens. As computational power increases the simulations of different pattern detecting algorithms could be furthered to include a larger variety of algorithms and parameter values.

Further directions for this line of research include a detailed examination of the inherent tradeoffs regarding stochastic or noisy decision rules. In particular, the determination of the optimal amount

and parametric form of noise that would render the underlying belief formation model immune to detection by human opponents, but at the same time satisfy the competing goal of retaining enough aggressiveness to exploit human behavior. The use of human/CA experimental designs could be applied to other strategic situations, such as auction markets, and to exploit other behavioral weaknesses of humans, such as endowment effects, preference reversals, expected utility anomalies, the Winner's curse et cetera.

Changes in strategic behavior in this paper were determined by the inference of learning models and their parameters from action data. Another interesting methodology would be to combine this with a neuroeconomic approach whereby the players' neural activity would be monitored whilst participating in the experiments. Changes in strategizing would then be directly measured through neural activity in various areas of the brain rather than indirectly through estimation of learning rules, leading to invaluable new insights.

References

Axelrod, R. (1985). *The Evolution of Cooperation*. Basic Books.

Barraclough, D. J., M. L. Conroy, and D. Lee (2004, April). Prefrontal cortex and decision making in a mixed strategy game. *Nature Neuroscience 7*(4), 404–10.

Blount, S. (1995, August). When social outcomes aren't fair: The effect of casual attributions on preferences. *Organizational Behavior and Human Decision Processes 63*(2), 131–44.

Bonetti, S. (1998). Experimental economics and deception. *Journal of Economic Psychology 19*, 377–395.

Budescu, D. V. and A. Rapoport (1994). Subjective randomization in one- and two-person games. *Journal of Behavioral Decision Making 7*, 261–78.

Camerer, C., G. Loewenstein, and D. Prelec (2005, March). Neuroeconomics: How neuroscience can inform economics. *Journal of Economic Literature 43*(1), 9–64.

Camerer, C. F. and T. Ho (1999). Experience-weighted attraction learning in normal-form games. *Econometrica 67*, 827–74.

Coricelli, G. (2005). Strategic interaction in iterated zero-sum games. *Homo Oeconomicus, forthcoming*.

Cosmides, L. and J. Tooby (1987). From evolution to behavior: Evolutionary psychology as the missing link. In J. Dupre (Ed.), *The latest on the best: Essays on evolution and optimality*. Cambridge, MA: MIT Press.

Davis, D. and C. A. Holt (1992). *Experimental Economics*. Princeton University Press, Princeton, NJ.

Dursch, P., A. Kolb, J. Oechssler, and B. C. Schipper (2005). Rage against the machines: How subjects learn to play against computers. Technical report, University of California, Davis.

Efron, B. (1987). Better bootstrap confidence intervals. *Journal of the American Statistical Association 82*, 171–200.

Efron, B. and R. Tibshirani (1994). *An Introduction to the Bootstrap*. Chapman & Hall/CRC.

Eurostat (2006, 13 July). Minimum wages in the eu25.

Fisher, R. (1920). A mathematical examination of the methods of determining the accuracy of observation by the mean error and the mean square error. *Monthly Notes of the Royal Astronomical*

Society 80, 758–770.

Fox, J. (1972). The learning of strategies in a simple, two-person zero-sum game without saddlepoint. *Behavioral Science 17*, 300–308.

Gigerenzer, G. (2000). *Adaptive thinking: Rationality in the real world*. New York: Oxford University Press.

Gigerenzer, G. and R. Selten (Eds.) (2001). *Bounded rationality: The adaptive toolbox*. Cambridge, MA: MIT Press.

Gilovich, T., D. Griffin, and D. E. Kahneman (2002). *Heuristics and biases: The psychology of intuitive judgment*. Cambridge, UK: Cambridge University Press.

Gorard, S. (2005). The advantages of the mean deviation. *British Journal of Educational Studies 53*(4), 417–30.

Harrison, G. W. (1989). Theory and misbehavior of first-price auctions. *American Economic Review 79*(4), 749–62.

Hertwig, R. and A. Ortmann (2001). Experimental practices in economics: A methodological challenge for psychologists? *Behavioral and Brain Sciences 24*, 383–451.

Huber, P. (1981). *Robust Statistics*. New York, John Wiley and Sons.

Huber, P. J. (1967). The behavior of maximum likelihood estimates under nonstandard conditions. In *Proceedings of the Fifth Berkeley Symposium on Mathematical Statistics and Probability*, Volume 1, Berkeley, CA, pp. 221–223. University of California Press.

Matheson, I. (1990). A critical comparison of least absolute deviation fitting(robust) and least squares fittings: the importance of error distributions. *Computers & chemistry 14*(1), 49–57.

Matlab (2007). *Mathworks, Inc., Natick, MA*.

Messick, D. M. (1967). Interdependent decision strategies in zero-sum games: A computer-controlled study. *Behavioral Science 12*, 33–48.

Mitchell, M. (1999). *An Introduction to Genetic Algorithms* (Fifth ed.). The MIT Press.

Mitropoulos, A. (2001). On the measurement of the predictive success of learning theories in repeated games. *Economics Working Paper Archive EconWPA*.

Nelder, J. A. and R. Mead (1965). A simplex method for function minimization. *Computer Journal 7*, 308–313.

Nowak, M. and K. Sigmund (1993, July). A strategy of win-stay, lose-shift that outperforms tit-for-tat in the prisoner's dilemma game. *Nature 364*, 56–58.

Nyarko, Y. and A. Schotter (2002). An experimental study of belief learning using elicited beliefs. *Econometrica 70*(3), 971.

O'Neill, B. (1987). Nonmetric test of the minimax theory of two-person zerosum games. In *Proceedings of the National Academy of Sciences*, Volume 84, pp. 2106–9.

Rapoport, A. and D. Budescu (1997). Randomization in individual choice behavior. *Psychological Review 104*(603-617).

Shachat, J. and T. J. Swarthout (2002). Learning about learning in games through experimental control of strategic interdependence strategic interdependence. *Experimental 0310003, EconWPA*.

Shachat, J. and T. J. Swarthout (2004). Do we detect and exploit mixed strategy play by opponents? *Mathematical Methods of Operations Research 59*(3), 359–373.

Sidak, Z. (1967). Rectangular confidence regions for the means of multivariate normal distributions. *Journal of the American Statistical Association 62*, 626–633.

Siebrasse, N. (2000). Generalized win-stay, lose-shift is robust in the repeated prisoners' dilemma with noise played by multi-state automata.

Smith, V. L. and J. M. Walker (1993, April). Rewards, experience and decision costs in first price auctions. *Economic Inquiry 31*(2), 237–245.

Spiliopoulos, L. (2008). Do repeated players detect patterns in opponents? Revisiting the Nyarko and Schotter belief elicitation experiment.

StataCorp (2007). *Stata Statistical Software: Release 10*. College Station, TX: StataCorp LP.

Svenson, O. (1981). Are we all less risky and more skillful than our fellow drivers? *Acta Psychologica 47*, 143–48.

Tesfatsion, L. and K. L. Judd (2006). *Handbook of Computational Economics Volume 2*. Elsevier/North-Holland (Handbooks in Economics Series).

Walker, J. M., V. L. Smith, and J. C. Cox (1987). Bidding behavior in first-price sealed-bid auctions: Use of computerized nash competitors. *Economics Letters 23*(3), 239–244.

White, H. (1980). A heteroskedasticity-consistent covariance matrix estimator and a direct test for heteroskedasticity. *Econometrica 48*, 817–830.

Wilson, H. (1978). Least squares versus minimum absolute deviations estimation in linear models. *Decision Sciences 9*, 322–335.

A Comparison of results with Barraclough et al. (2004)

Barraclough et al. (2004) conducted a very similar study with monkey subjects playing against computer algorithms comparable to the ones investigated in this paper. What is even more interesting is that both papers exhibit extremely similar results for both human and monkey subjects, warranting special attention.

In their paper, two monkeys are engaged in playing a simple zero-sum game, matching pennies, against three different computer algorithms. All monkeys played the computer algorithms in the same order: algorithm 0 first, then algorithm 1 and finally algorithm 2.

Algorithm 0 chose both available actions with equal probability i.e. played the mixed strategy Nash equilibrium of this game. Against this algorithm the monkeys had no incentive to converge to the MSNE or any other particular strategy as their expected payoffs were independent of their actions.

Algorithm 1 had access to all the past moves of the monkeys in each trial and calculated the conditional probabilities of play given the last four actions of the monkeys. For each of these conditional probabilities binomial tests were conducted of the null hypothesis that these were not significantly different from zero at the 5% level. If the null hypothesis was not rejected for any of the tests then the computer algorithm assumed that the monkey was playing according to the MSNE and therefore also reverted to doing the same i.e. playing as algorithm 0. If only one of the tests was significant then this conditional probability was used to calculate the algorithm's best response. The algorithm responded probabilistically, in particular if the conditional probability that the monkey would play a particular action was p, then the computer algorithm would play its best response to that action also with probability p. In such a setup the optimal behavior of the subjects is to play according to the MSNE (including the i.i.d. condition). This algorithm is very similar to the *fp2* and *fp3* algorithms used in this paper with the difference that these algorithms did not test over many different time periods and did not perform significance tests as to whether deviations from i.i.d. behavior were significantly different from random deviations.

Algorithm 2 is the same as algorithm 1 with the addition of lagged payoffs (one to four lags) in the computer algorithm analysis. This means that the computer algorithm could now detect whether monkeys were conditioning on the algorithm's past play (through the interaction of payoffs and computer algorithm actions) and therefore the payoff maximizing strategy of the monkeys was to play not only independently of own past play but also independently of the computer algorithm's past play. Hence, algorithm 2 could detect and exploit possible *ws/ls* behavior on the part of the monkeys. This algorithm is analogous to the *spd* algorithm in this study since both are able to detect and exploit *ws/ls* behavior in opponents. The difference is that algorithm 2 could also detect conditioning on subjects' own actions whereas the *spd* algorithm could not, thereby opening an avenue for subjects to exploit this algorithm using strategies different from the *ws/ls* heuristic.

A comparison of the four main results of Barraclough et al. (2004) and analogous results from this paper follows.

1. The monkey subjects used the *ws/ls* heuristic with increasing frequency against algorithm 2, algorithm 1 and algorithm 0. Hence, they used the *ws/ls* heuristic least against algorithm 2 which was able to exploit this, and more when playing against algorithm 1 which could not detect it. What is interesting is that even against algorithm 0, where the *ws/ls* heuristic does

Table 22 Estimated memory decay parameters

Monkey	Algorithm 0	Algorithm 1	Algorithm 2
C	0.99	0.18	0.99
E	0.68	0.17	0.83
Average	0.84	0.17	0.91

not confer a competitive advantage, monkeys still had a bias towards using the *ws/ls* heuristic. This result together with the widespread findings that the *ws/ls* heuristic can be found in a wide range of games and human behavior lead us to conclude that it is the first strategic choice of human and animal subjects. Spiliopoulos (2008) finds that use of the *ws/ls* heuristic is quite widespread in the experimental data of Nyarko and Schotter (2002) and that simulations of the *ws/ls* heuristic against a fictitious play algorithm with full memory and a two-period pattern detecting variant of fictitious play conclude that the *ws/ls* outperforms these computationally more complex strategies. The *ws/ls* heuristic is a smart evolutionary strategy if environments are relatively stable and invariant as in this case, an action that provided a positive reinforcement in the past will also on average do so in the future i.e. the environment exhibits positively correlated events. Nowak and Sigmund (1993) find that the *ws/ls* heuristic outperforms the tit-for-tat strategy and is more robust and resistant to invasion from other strategies in iterated Prisoners' Dilemma games. Siebrasse (2000) discovers that a generalized version of the *ws/ls* heuristic dominates populations of strategies that evolve through time via the application of genetic algorithms.

2. Against algorithms 1 and 2 which are able to exploit non-i.i.d. behavior, such as conditioning on own past actions, the monkeys' play was much closer to i.i.d. behavior than against algorithm 0 which could not detect them. The use of algorithm 0 is interesting because it can be used to infer what strategies or dependencies are brought into the experiment rather than being learned in the experiment, since MSNE behavior by the CA leaves the subject indifferent to its own strategies.
3. Simple reinforcement learning models were modeled in their study requiring the estimation of a memory parameter, γ. Table 22 gives the estimated memory parameters from the reinforcement learning models they estimated. The most interesting results are for those of algorithms 1 and 2 which as argued above correspond to algorithms *fp2*, *fp3* and *spd*. Algorithms *fp2* and *fp3* are special cases of algorithm 1 and the *spd* algorithm is a special case of algorithm 2. In this paper's results we find that against *fp2* and *fp3* computer opponents the estimated γ was close to zero but against the *spd* algorithm this was close to one and exactly the same pattern emerges from their study. Algorithm 1, which is the analogue of *fp2* and *fp3*, exhibits a low average estimated γ, 0.17, whereas against algorithm 2, which is the analogue of the *spd* algorithm, the estimated γ is 0.91. Even though this study allowed for much more complex behavior by modeling subjects using the EWA learning rule, of which reinforcement learning is a special case, it is found that the same simple strategies predict behavior against similar algorithms in both our human subjects and their monkey subjects equally well.
4. Evidence was found of individual neurons encoding information as to the previous choice and reward of monkeys' actions and of the conjunction of these, which are necessary for the computation of a reinforcement learning algorithm.

B Computational details of model estimation and optimization

The problem of choosing initial parameter values for optimization algorithms is usually solved with one of two techniques.

1. Performing the optimization many times with randomly chosen parameter values from a restricted parameter space. The problem with this method is that for a large number of parameters it is necessary to use a large number of random initial points which means that an already computationally expensive optimization routine must be run a large number of times.
2. Performing a grid search over plausible parameter values and then choosing the best combination of parameter values (or a set of the best performing combinations) as initial starting points for the optimization algorithm. For a large number of parameters this technique is also computationally expensive as the number of possible initial grid values increases exponentially in the number of parameters (keeping the grid distance constant).

These two techniques are computationally inefficient because their search is not directed - in the first case points are chosen randomly, in the second case deterministically but arbitrarily. This occurs because the points are determined before the algorithm starts and are not updated with new information obtained during the execution of the algorithm. Incorporating this new information would lead to an increase in efficiency as the algorithm could increase sampling from areas of the parameter space that show promise and are more likely to lead to better solutions. We propose instead the use of genetic algorithms where the search is influenced and directed by information collected throughout the procedure, see Mitchell (1999) for an introduction.

Genetic algorithms use three main evolutionary principles or operators to guide the search for an optimum.

1. Start with an initial (usually randomly selected) population of combinations of parameters and estimate the objective function for each one.
2. Apply a selection rule by selecting a subset of the best performing parameter combinations which will serve as the parents of the next generation.
3. Apply a crossover rule to combine two parents and create children for the next generation. This entails randomly selecting features from the two parents and combining them to form a new parameter combination.
4. Apply a mutation rule to chosen parents so as to randomly change features of the parent and create a child for the next generation.

Repeating the above three rules for each successive generation creates a process similar to natural or Darwinian selection. The selection rule guides the algorithm so that it spends more time in regions of parameter space which have been more successful in the past and therefore reduces the computational time devoted to scanning clearly suboptimal regions in the parameter space. The drawback is that it may wrongly become stuck in a suboptimal area. This is where the mutation rule is useful since it forces the algorithm to keep experimenting in other areas of the parameter space regardless of their past performance. The crossover rule allows for the process to quickly and efficiently hone in to a good solution by combining the characteristics of good performers.

Genetic algorithms are very well suited for problems that are discontinuous, nondifferentiable, stochastic, or highly nonlinear. Although quite often the genetic algorithm procedure is used by itself

to perform the whole optimization routine, we propose to use it only to obtain initial parameter values. These will then be used to start another algorithm that is more suited to locally searching the parameter space and refining these initial parameter values found by the genetic algorithm. As argued in the main text, the family of gradient descent techniques may not be well suited to this specific problem and therefore the Nelder and Mead (1965) Simplex Method was implemented instead.

C Mean absolute deviations (MAD) versus mean square deviations (MSD) as an error measure of performance

Minimizing the MAD of a model instead of the MSD has a number of important advantages that are particularly important for this study. Mitropoulos (2001) finds that it is generally the case that MSD minimization tends to select learning rules that make predictions closer to a uniform distribution rather than a distribution of predictions near to the bounds. In response to this problem the MAD is proposed as the error measure, since it does not excessively penalize larger absolute errors. Gorard (2005) provides evidence for preferring MAD over MSD particularly on the grounds of efficiency. Although Fisher (1920) defended the use of MSD over MAD by arguing that the former was more efficient than the latter, the assumptions he made were extremely strong - he assumed normality and no observation or measurement error. Huber (1981) did away with these strict assumptions and found that MAD is in fact more efficient than MSD whenever the percentage of error points in the total observations is greater than 0.2%[26]. Wilson (1978) surmises that MAD is dramatically more efficient than MSD in the presence of outliers contaminating the dataset. The superiority of MAD over MSD for non-normal distributions was pointed out by Fisher himself and has been verified through the use of Monte Carlo techniques. This is extremely important for this study as error distributions of fitted stated beliefs will clearly be non-normal as argued earlier. In addition to this, the error distributions of the estimated models will not be evenly distributed around the function, i.e. will be asymmetric, in which case Matheson (1990) concludes again that MAD should be preferred over MSD.

D Tables of individually estimated EWA model parameters

[26] If the percentage of error points is 5% then MAD is twice as efficient as MSD.

Table 23 Parameter estimates and measures of fit of individually estimated EWA models of subjects playing against the *fp2* algorithm

	Parameter estimates						Measures of fit					
Player	κ	δ	ϕ	λ	$A_i^j(0)$	$N(0)$	POI	POI-CV	MSD	MSD-CV	MAD	MAD-CV
1	0.00	0.26	0.00	10.00	140.00	0.00	26.12	27.27	0.26	0.27	0.26	0.27
2	0.10	0.81	0.99	0.37	-131.05	5.66	20.15	42.42	0.19	0.40	0.20	0.41
3	1.00	0.36	0.66	9.39	-137.20	50.47	18.66	27.27	0.18	0.27	0.19	0.27
4	1.00	0.00	0.60	10.00	-0.21	673.13	29.10	30.30	0.29	0.30	0.29	0.30
5	1.00	1.00	0.33	9.99	-13.08	13.36	18.66	18.18	0.18	0.18	0.19	0.18
6	0.00	0.00	0.00	10.00	-134.68	0.00	33.58	39.39	0.33	0.39	0.34	0.39
7	0.00	1.00	0.00	10.00	94.87	94.99	14.18	24.24	0.14	0.24	0.14	0.24
8	0.99	0.66	0.86	1.37	-89.11	656.48	27.61	48.49	0.27	0.48	0.28	0.48
9	1.00	0.30	0.85	9.52	-11.48	0.01	35.08	45.46	0.35	0.45	0.35	0.45
10	1.00	0.37	0.00	10.00	140.00	0.00	20.15	24.24	0.20	0.24	0.20	0.24
11	0.90	0.99	0.84	8.41	77.09	32.35	39.55	45.45	0.39	0.45	0.40	0.45
12	0.00	0.00	0.00	10.00	140.00	78.10	30.60	45.46	0.30	0.45	0.31	0.45
13	0.00	0.00	0.00	10.00	-50.98	50.48	29.10	27.27	0.29	0.27	0.29	0.27
14	1.00	1.00	0.31	10.00	124.44	0.00	5.22	9.09	0.05	0.09	0.05	0.09
15	0.94	1.00	0.34	10.00	16.31	64.15	15.67	30.30	0.16	0.29	0.17	0.30
16	0.00	1.00	0.00	10.00	140.00	95.75	24.63	27.27	0.24	0.27	0.25	0.27
17	0.46	1.00	0.49	10.00	140.00	0.00	18.66	27.27	0.18	0.27	0.19	0.27
18	0.46	0.00	0.00	10.00	140.00	140.72	33.58	39.39	0.33	0.39	0.34	0.39
19	0.98	0.79	0.98	5.26	46.60	72.91	29.10	45.46	0.29	0.45	0.29	0.45
20	0.03	0.98	0.19	6.99	-50.73	33.21	18.66	24.24	0.18	0.24	0.19	0.24
21	1.00	0.00	0.42	0.00	-52.92	9.18	50.00	50.00	0.25	0.25	0.50	0.50
22	0.97	0.49	0.94	8.67	27.90	23.15	21.64	36.36	0.21	0.36	0.22	0.36
23	0.32	0.95	0.00	10.00	-41.23	0.00	33.58	36.36	0.33	0.36	0.34	0.36
24	1.00	0.11	0.00	10.00	-140.00	4.15	18.66	39.39	0.18	0.39	0.19	0.39
25	1.00	1.00	0.00	10.00	82.97	113.54	33.58	24.24	0.33	0.24	0.34	0.24
26	1.00	0.37	0.86	10.00	134.68	318.66	33.58	45.46	0.33	0.45	0.34	0.45
27	0.46	0.10	1.00	3.64	121.86	25.75	26.12	21.21	0.26	0.21	0.26	0.21
28	0.37	0.56	1.00	1.92	111.70	21.80	27.61	36.36	0.27	0.36	0.28	0.36
29	0.10	0.66	0.14	9.33	-22.06	55.86	24.63	24.24	0.24	0.24	0.25	0.24
30	0.82	1.00	0.00	10.00	-19.67	96.22	12.69	24.24	0.12	0.24	0.13	0.24
31	0.98	0.89	1.00	0.57	-140.00	20.71	35.08	45.46	0.34	0.45	0.34	0.46

Table 24 Parameter estimates and measures of fit of individually estimated EWA models of subjects playing against the *fp3* algorithm

	Parameter estimates						Measures of fit					
Player	κ	δ	ϕ	λ	$A_i^j(0)$	$N(0)$	POI	POI-CV	MSD	MSD-CV	MAD	MAD-CV
1	0.90	0.67	0.00	10.00	140.00	14.93	29.10	39.39	0.29	0.39	0.29	0.39
2	1.00	0.75	1.00	4.00	17.39	40.34	38.06	30.30	0.38	0.30	0.38	0.30
3	0.97	0.00	0.00	1.45	111.77	26.61	39.55	30.30	0.39	0.30	0.40	0.30
4	0.93	0.44	0.35	10.00	139.99	122.40	24.63	33.33	0.24	0.33	0.25	0.33
5	1.00	0.48	0.30	10.00	-66.04	2.11	17.16	18.18	0.17	0.18	0.17	0.18
6	1.00	0.00	0.99	4.90	-138.88	0.00	50.00	39.39	0.50	0.39	0.50	0.39
7	0.82	0.00	0.17	10.00	-140.00	156.13	18.66	30.30	0.18	0.30	0.19	0.30
8	0.99	1.00	0.93	0.01	140.00	1000.00	36.57	36.36	0.35	0.35	0.36	0.36
9	1.00	0.00	0.18	0.00	35.92	34.65	50.00	50.00	0.25	0.25	0.50	0.50
10	0.72	0.62	0.98	2.82	98.27	68.56	26.12	33.33	0.26	0.33	0.26	0.33
11	1.00	0.23	0.55	10.00	140.00	78.73	32.09	60.61	0.32	0.61	0.32	0.61
12	1.00	1.00	0.90	3.76	139.98	7.80	24.63	39.39	0.24	0.39	0.25	0.39
13	0.77	0.65	0.95	5.60	-82.58	38.19	21.64	33.33	0.21	0.31	0.22	0.32
14	0.91	0.00	0.13	10.00	-139.96	224.69	32.09	39.39	0.32	0.39	0.32	0.39
15	1.00	1.00	0.34	10.00	139.87	0.01	29.10	33.33	0.29	0.33	0.30	0.33
16	0.11	0.00	0.00	10.00	6.81	25.10	35.08	45.46	0.35	0.45	0.35	0.45
17	0.75	0.85	0.92	10.00	131.06	1.54	36.57	36.36	0.36	0.36	0.37	0.36
18	1.00	0.87	0.94	7.52	5.47	13.32	26.12	51.52	0.26	0.52	0.26	0.52
19	0.69	0.75	0.98	3.58	59.26	33.34	33.58	45.46	0.33	0.45	0.34	0.45
20	0.00	0.15	0.20	10.00	22.46	0.28	30.60	15.15	0.30	0.15	0.31	0.15
21	1.00	0.00	0.40	0.00	140.00	0.00	50.00	50.00	0.25	0.25	0.50	0.50
22	0.00	1.00	0.00	2.16	37.11	147.74	32.09	33.33	0.32	0.33	0.32	0.33
23	0.73	0.01	0.00	10.00	-140.00	8.28	20.15	18.18	0.20	0.18	0.20	0.18
24	1.00	0.80	0.00	10.00	41.76	0.00	29.10	15.15	0.29	0.15	0.29	0.15
25	1.00	0.97	0.00	10.00	-140.00	299.68	35.08	36.36	0.35	0.36	0.35	0.36
26	0.71	1.00	0.01	0.80	8.48	2.06	41.05	42.42	0.41	0.42	0.41	0.42
27	1.00	0.00	0.05	1.61	-118.82	0.16	33.58	27.27	0.33	0.27	0.34	0.27
28	0.94	0.17	0.74	10.00	86.50	73.68	18.66	42.42	0.18	0.42	0.19	0.42
29	0.58	0.00	0.00	10.00	140.00	48.05	39.55	24.24	0.39	0.24	0.40	0.24
30	0.57	0.66	0.00	10.00	140.00	0.00	20.15	30.30	0.20	0.30	0.20	0.30
31	0.35	0.11	0.00	10.00	140.00	195.56	51.49	39.39	0.51	0.39	0.51	0.39

Table 25 Parameter estimates and measures of fit of individually estimated EWA models of subjects playing against the *spd* algorithm

	Parameter estimates						Measures of fit					
Player	κ	δ	ϕ	λ	$A_i^j(0)$	$N(0)$	POI	POI-CV	MSD	MSD-CV	MAD	MAD-CV
1	0.89	0.00	0.43	7.96	15.35	0.00	36.57	30.30	0.36	0.30	0.37	0.30
2	0.97	0.07	0.99	0.28	43.48	30.57	27.61	33.33	0.27	0.33	0.28	0.33
3	1.00	1.00	0.73	10.00	-17.74	0.00	33.58	60.61	0.33	0.61	0.34	0.61
4	1.00	0.06	1.00	0.01	-76.17	77.36	5.22	0.00	0.05	0.00	0.05	0.00
5	0.61	0.81	0.00	10.00	79.28	51.06	36.57	30.30	0.36	0.30	0.37	0.30
6	1.00	0.00	0.92	10.00	70.20	204.93	24.63	45.46	0.24	0.46	0.25	0.46
7	0.14	0.32	0.98	4.68	35.62	97.40	24.63	15.15	0.24	0.15	0.25	0.15
8	1.00	0.56	0.50	10.00	-139.99	0.25	39.55	48.49	0.39	0.48	0.40	0.48
9	0.94	1.00	0.76	9.89	-29.30	7.77	33.58	42.42	0.33	0.42	0.34	0.42
10	0.45	0.56	0.00	10.00	135.98	0.00	51.49	48.49	0.51	0.48	0.51	0.48
11	1.00	0.00	0.26	9.89	-33.88	3.55	41.04	33.33	0.41	0.33	0.41	0.33
12	1.00	0.00	0.61	10.00	0.67	84.80	30.60	48.49	0.30	0.48	0.31	0.48
13	1.00	0.00	0.89	10.00	-7.27	0.19	21.64	15.15	0.21	0.15	0.22	0.15
14	0.95	0.11	0.85	10.00	-10.77	9.59	39.55	51.52	0.39	0.52	0.40	0.52
15	0.62	0.00	0.07	10.00	140.00	0.00	36.57	27.27	0.36	0.27	0.37	0.27
16	0.76	0.61	0.98	7.32	15.51	9.24	21.64	36.36	0.21	0.36	0.22	0.36
17	0.84	0.00	0.39	10.00	40.90	24.05	18.66	39.39	0.18	0.39	0.19	0.39
18	1.00	0.00	0.00	0.00	-57.82	41.20	50.00	50.00	0.25	0.25	0.50	0.50
19	1.00	0.04	0.63	9.89	129.62	95.07	35.08	45.46	0.35	0.45	0.35	0.45
20	0.03	1.00	0.00	1.16	-139.99	7.72	44.03	60.61	0.44	0.61	0.44	0.61
21	1.00	0.00	0.86	10.00	33.53	155.24	45.52	39.39	0.45	0.39	0.46	0.39
22	1.00	0.82	0.00	10.00	140.00	0.00	45.52	54.55	0.45	0.55	0.46	0.55
23	0.99	0.97	0.49	9.79	-107.79	36.88	30.60	39.39	0.30	0.39	0.31	0.39
24	0.98	0.01	0.91	3.43	-43.47	26.51	38.06	42.42	0.38	0.42	0.38	0.42
25	1.00	0.68	0.46	10.00	139.93	1.64	38.06	42.42	0.38	0.43	0.38	0.43
26	0.64	0.65	0.99	8.51	45.70	24.13	18.66	27.27	0.18	0.27	0.19	0.27
27	0.97	0.12	0.00	7.42	-64.15	184.60	48.51	60.61	0.48	0.61	0.49	0.61
28	0.82	1.00	0.00	10.00	-140.00	114.61	48.51	45.46	0.48	0.45	0.49	0.45
29	0.17	0.00	0.00	10.00	-6.24	47.91	36.57	48.49	0.36	0.48	0.37	0.48
30	0.16	0.75	0.90	9.98	-139.52	2.67	36.57	48.49	0.36	0.48	0.37	0.48
31	1.00	0.00	0.27	10.00	-95.98	278.62	27.61	39.39	0.28	0.39	0.29	0.40