On Automated Agents' Rationality

Amos Azaria[1], Ariella Richardson[2], Avshalom Elmalech,
Avi Rosenfeld[2], Sarit Kraus[1,3], and David Sarne[1]

[1] Department of Computer Science, Bar-Ilan University, Ramat Gan 52900, Israel
[2] Dept. of Industrial Engineering Jerusalem College of Technology, Israel
[3] Institute for Advanced Computer Studies University of Maryland, MD 20742

Abstract. Agents that interact with humans are known to benefit from integrating behavioral science and exploiting the fact that humans are irrational. Therefore, when designing agents for interacting with automated agents, it is crucial to know whether the other agents are acting irrationally and if so to what extent. However, little is known about whether irrationality is found in automated agent design. Do automated agents suffer from irrationality? If so, is it similar in nature and extent to human irrationality? How do agents act in domains where human irrationality is motivated by emotion? This is the first time that extensive experimental evaluation was performed in order to resolve these questions. We evaluated agent rationality (for non-expert agents) in several environments and compared agent actions to human actions. We found that automated agents suffer from the same irrationality that humans display, although to a lesser degree.

1 Introduction

Automated Agents are integrated into countless environments, such as Electronic commerce, Web crawlers, Military agents, Space Exploration probes and Automated drivers. Due to the high importance of automated multi-agent environments, many competitions were established where automated agents compete with each other in order to achieve a goal [35,2,1,36].

Modeling agents is beneficial for agent-agent interaction [29]. However, building such a model is a complex issue, and furthermore, if the model built is too far from the actual opponent's behavior, using it may become detrimental [25,22]. How should designers plan their agents when opponent modeling is unavailable? Can any general assumptions be made on automated agents and used for agent design?

Research into peoples' behavior has found that people often do not make strictly rational decisions but instead use sub-optimal, bounded policies. This behavior has been attributed to a variety of reasons including: a lack of knowledge of one's own preferences, the effects of the task complexity, framing effects, the interplay between emotion and cognition, the problem of self-control, the value of anticipation, future discounting, anchoring and many other effects [37,23,5,11]. Since people do not usually use fully rational strategies themselves, agents based on game theory approach, which assume rational behavior in humans often perform poorly [28,20,8]. Many studies have shown that psychological factors and human decision-making theory are needed in order to develop a good model of true human behavior, which in turn is required for optimizing the performance of agents interacting with humans [16,19,26,30,28,7,9].

In domains where opponent modeling is infeasible, can we use rationality models or models of human behavior in order to design agents that interact with other agents (for non-expert-designed agents)? In order to do this we need to determine whether we can assign rationality traits to the agents: Are automated agents strictly rational, and if not how do they compare to humans? The answer to this question will provide useful guidelines for agent strategy planning when opponent agent modeling is infeasible, especially for domains where human rationality has been studied.

Throughout this paper, a rational player will refer to a player who tries to maximize his expected outcome and assumes that all other players are doing so as well. Irrational behavior will be indicated by a player who fails to compute a rational strategy, has his own subjective utility function which differs from the expected outcome or believes (possibly justifiably) that other players use irrational behavior. Note that, in cases in which opponents act irrationally, a player acting irrationally as well, may gain a greater expected outcome than a fully rational player.

We perform an extensive experiment evaluation of automated agents' behavior and compare it to fully rational and human behavior in three different games. Each game will allow the exploration of different aspects of agent rationality. In the first game we examine whether agents exhibit an irrational tendency to keep all options available (as humans are known to do). For the second game we use a costly exploration game. The second game expands our study as it enables the exhibition of behaviors that lie on either side of the equilibrium. In a case where agents are not purely rational, as we hypothesize, it remains to be seen whether they display the same type of irrational behavior as humans or sway in the opposite direction. The third game integrates activities which trigger emotions. We show that as one would expect humans become emotionally involved when playing this game. It remains to be seen how agents react. Do agents use strategies that mimic human emotions? Will they use these strategies if it involves a clear utility loss?

In this study we will determine whether automated agents are rational or not and to what extent, and whether or not they display behavior that is similar to humans. The findings of this research have practical implications for designing agents that interact with non-expert agents.

2 Related Work

Opponent modeling is of great importance when designing automated agents which interact with other agents [29]. Carmel and Markovitch [13] show that if an agent can build a model of its opposing agent, this model could be used to improve its own performance. McCracken and Bowling [25] propose a method for modeling an opponent in the Rock-Paper-Scissors domain. Lazaric et al. [22] propose a method for opponent modeling in Kuhn Poker (a degenerated version of poker). However, in more sophisticated games these methods become intractable.

Angluin [4] proposes an algorithm for modeling an automated agent as a Moore machine. Angluin assumes the existence of a teacher which replies 'Yes' for a correct conjecture or provides a counterexample on which the model and the machine disagree. This algorithm is polynomial in the number of states in the machine. However, the existence of such a teacher isn't likely when modeling an opponent. Carmel and Markovitch [13] relax the teacher assumption and provide a heuristic algorithm for learning Moore machines. The algorithm builds a model consistent with past examples, and when a new counterexample arrives it tries to extend the model in a minimal fashion. However, in practice an automated opponent is not likely to be limited to being a Moore machine but rather a more general Turing machine.

Various studies [10] have compared experimental results between the strategy method (in which a responder makes conditional decisions for each possible information set) to the more standard direct-response method (in which the responder observes the action of the first mover and then chooses a response). There is mixed evidence as to whether the two methods lead to similar results. While in the strategy method the subjects make decisions for all the possible options, in our work the agents must provide a general strategy since the number of possible options is extremely large. Our work should not be confused with research on the psychology of programming [31] that investigates how understanding psychological aspects of programming improves their ability to write error-less and readable code.

Human behavior is well studied and plays an important role in human-agent interaction. However, the evaluation of automated agent rationality is much less explored. Grosz et al. [18] introduced the Colored Trails game in order to investigate decision-making strategies in multi-agent situations. They showed that human players and agents do not play in the same way. Their results indicate that people design automated agents that don't play as well as human players, possibly because they cooperate less than people. It is interesting to note that the agents also do not adhere to the equilibrium strategy. Although the environment used in this game was a mixed human-agent environment, while we are studying environments composed only of agents, this research provides some insight into our problem.

Manistersky et al. [24] explored the evolution of automated agents in a negotiation environment. Designers were given a chance to improve the agents based on previous performance. They found that agents designed by humans seem to perform better than equilibrium agents but not as well as Pareto-optimal agents. The environment explored is a pure automated agent environment, such as the one we use. However, they did not compare the performance of agents to that of humans.

Unlike the above-mentioned prior work, the games we selected are purposely very simple compared to the Colored Trails and negotiation games. Neither do we aim to evaluate the performance of the agents described in this paper. Rather we are interested in studying the degree of rationality displayed by the agents relative to humans.

It seems that automated agents are not purely rational and are different from humans. We would like to know, however, how they relate to humans. Do the agents act in a unique fashion? Or are they displaying the same irrational behavior that humans use, but possibly to a lesser degree?

3 Door Game

The first experiment tests to what extent agents and people exhibit a tendency to keep all options viable, even when the cost of doing so is greater than the potential benefit.

This irrational tendency among humans has been demonstrated by Shin and Ariely [34] via the "Door Game".

In the basic version of the game, the player is faced with three doors (alternatives), each associated with a different distribution of payoffs. The distribution of each door is a priori unknown to the player. The player first chooses with which door to begin, and from that point on, any additional click on that door will yield a reward drawn from that distribution. At any time, the player can switch to any other door by clicking on it. Switching to another door enables the player to receive rewards from the distribution characterizing that door via additional clicks on it. The player can collect rewards only from the door to which she has most recently switched. Rewards are accumulated and the player's goal is to maximize gains given a limited "budget" of clicks. Once the player has used all of her clicks, the game terminates and she is paid the sum of her door-click pay-offs. The click that the player needs to "sacrifice" in order to switch doors is in fact a switching cost. This setting can be mapped to a Mutli-Armed Bandit problem [6], in which a rational strategy focuses most of the exploration in initial rounds and then sticks with the door with the highest average outcome.

Shin and Ariely also considered a variant in which each time a participant clicks on a door, the two other doors are reduced in size by $\frac{1}{15}$ of their original width. A single click on a shrinking door restores it to its original size and the process continues. Once a door shrinks to zero, it is eliminated for the remainder of the game. In the basic version Shin and Ariely show that human subjects followed the rational strategy, however, with the shrinking door variant, players tend to switch from door to door, in an effort to keep their options open. This resulted in a substantial performance degradation (in terms of the rewards accumulated) compared to choosing any single door and sticking with it.

Our experiment with the door game used the game variant with the diminishing doors. We followed a specific experimental design reported in [34] where the game is made up of two phases, each with a "budget" of 50 clicks. In the first phase (the exploration phase), the participants do not receive any payoff and were only notified of the payoff amount. The purpose of this phase is for the participants to identify the best door. This phase is long enough for a rational player to select a single door from which she does not need to divert for the entire second phase (while ignoring the vanishing of the other doors). In the second stage (the exploitation phase), the participants received the payoff obtained from the door on which they clicked.

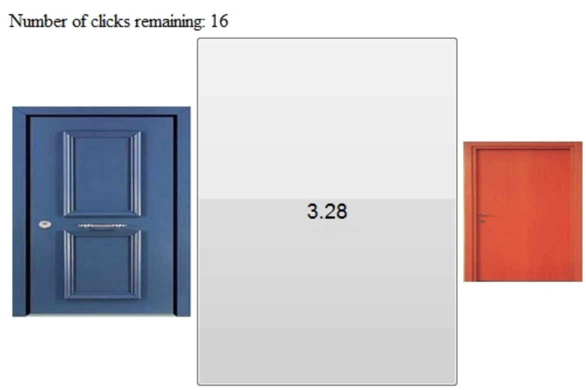
The experiment included 48 computer science students, each of whom programmed an agent capable of playing this game, as part of the students' regular course assignment. Their grades were in part calculated based on the performance of their specific agent. The students received a thorough explanation which included the game rules, the payoff accumulation method and the division into the two stages.

52 human subjects were recruited using Amazon's Mechanical Turk service [3], which is a crowd sourcing web service that coordinates the supply and demand of tasks which require human intelligence in order to be completed. Amazon Mechanical Turk has become an important tool for running experiments with human subjects and has been established as a viable method for data collection [27].

The subjects first received appropriate instructions. In order to assure that they fully understood the game, the subjects had to first play the game when no score was recorded, and only then they could play the actual game. In order to encourage them to play seriously they received a bonus which was proportionate to their performance.

The GUI presented the doors to the human subjects, with each door size changing according to the clicks made. Figure 1(a) demonstrates a screen-shot of a game where the user has not yet clicked on a door. Figure 1(b) shows a screen-shot where the user has picked the middle door and the other two doors have started shrinking.

(a)



(b)

Fig. 1. A screen-shot of the door game user interface: (a) before user clicks on door (b) user clicked on middle door

As discussed above, rational behavior in this game would be to dedicate all clicks in the first phase to exploration and stick to a single door throughout the second phase. Therefore, any switching between doors during the second phase can be classified as irrational behavior. In our experiment, 77% of the human subjects switched doors during the second phase of the game, comparable to only 29% of the agents programmed for this task. While the human subjects switched doors 14.80 times on average, in the last 50 clicks, the automated agents did so only 3.02 times on average. All results reported throughout all experimental sections are statistically significant with $\alpha = 0.05$ (using either student t-test or Fisher's exact test). These results indicate that, just as humans, also automated agents act irrationally to preserver options, even at a great expense. However, the extent of this irrational behavior is much less for automated agents. This provides the first answer to the questions we raised.

In the door game, irrational behavior could only be displayed in a single direction, as rational behavior required not switching doors at all. We next investigate what happens when rational (or optimal) behavior has a value which can be exceeded in both directions. The irrational player may adhere to a value which is either greater or less than the rational one. Assuming that the agents will act irrationally again, will they display the same type of behavior as humans or will they sway in the opposite direction?

4 Search Game

The second irrational behavior we inspect is the property observed in human consumers who tend to perform too short a search prior to purchasing a product [15]. This domain is particularly interesting (assuming the agents are irrational) as we can observe whether they display the human tendency to perform a short search, or if they exhibit a different type of irrational behavior and perform too long a search.

The sequential exploration problem considers an individual facing a number of possible available opportunities (e.g., to buy a product) out of which she can choose only one. The value of each opportunity to the agent is a priori unknown. Instead, only its probability distribution function is known, and revealing the true value of an opportunity is costly. The problem, to which a broad class of real-world situations can be mapped, was formally introduced by Weitzman [38] along with its optimal (cost-minimizing or utility-maximizing) strategy. The optimal strategy derives from the trade-off between the benefit from the potential improvement in the quality of the results which the agent may further obtain with the additional exploration and the costs of carrying out such exploration. The optimal strategy is based on setting a reservation value (a threshold) for each opportunity and choosing to obtain the value of the opportunity associated with the minimum reservation value at any time (terminating the exploration once the minimum value obtained so far is less than the minimum reservation value of any of the remaining opportunities). Intuitively, the reservation value of an opportunity is the value where the agent is precisely indifferent: the expected marginal benefit from obtaining the value of the opportunity exactly equals the cost of obtaining that additional value.

Much evidence has been given in the literature to differences between the behaviors exhibited by people and the theoretic-optimal strategy in costly exploration settings.The major difference reported relates to the tendency of people to terminate their search before the theoretic-optimal strategy would have done so [32,14,15].

In order to evaluate the existence and extent of this phenomena in automated agents, we used 31 agents whose strategies were programmed by computer science students (each agent by a different student) as part of their regular course assignments. (The group of students who designed these agents is different from the group that designed the agents for the door game.) Each agent receives as input a list of opportunities, their distribution of values and the cost of obtaining these values. The agent had to decide at any time the value of which opportunity to obtain next (incurring the appropriate cost) and when to terminate the exploration. Each student's grade in the assignment was correlated with her agent's performance. As part of their assignment, students provided documentation that described the algorithm used for managing the exploration. To compare the results with people, we used a GUI-based experimental infrastructure, simulating a price-search environment. Each opportunity represented a store associated with a different distribution of prices and a cost for obtaining the true price (represented as a "parking cost" for parking next to that store). Figure 2 presents a screen-shot of this GUI. Querying a store is done by clicking the "Check" button below it, in which case the true price of the store becomes known and the parking cost of that store is added to the accumulated cost. The game terminates when clicking the "Buy" button (available only in stores whose prices are known).

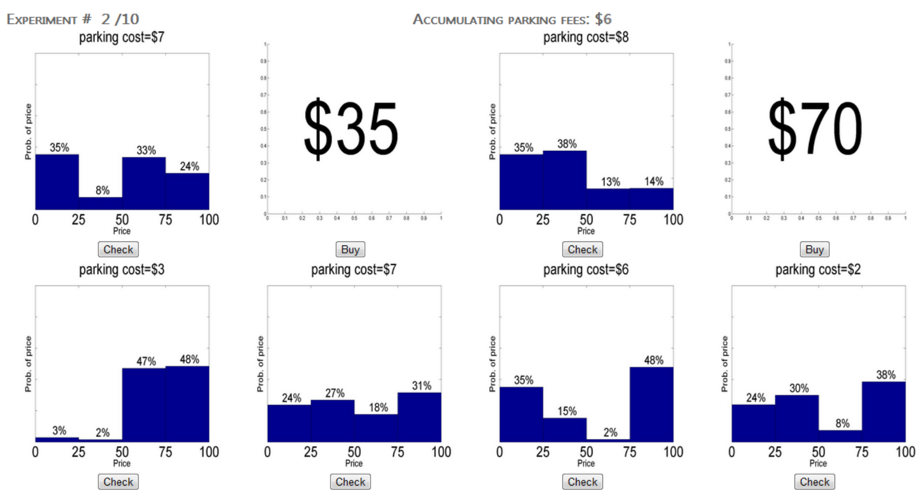


Fig. 2. A screen-shot of the search game user interface

Human subjects were recruited using Amazon's Mechanical Turk service. Overall 150 human subjects participated in this experiment. Subjects received a short textual description of the experiment, emphasizing the exploration aspects of the process and the way costs are accumulated, followed by a short video clip. Then, a series of practice games were played in order to make sure that the subject understood the experiment. Participants had to play at least three practice games; however, they could continue practicing until they felt ready to start the experiment.

We used 300 randomly generated exploration problems. Each problem contained 8 opportunities associated with distributions of values and costs. Distributions were formed as multi-rectangular distribution functions (i.e., based on rectangles of equal width, however each with a different distribution mass) defined over the continuous interval $0 - 100$. Costs were uniformly drawn from the interval $1 - 10$. Each agent was tested using all 300 problems, and each human subject was randomly assigned 10 problems out of this set.

Figure 3 depicts the average search extent in our experiments (measured as the number of opportunities explored within each game) for people, agents and the optimal exploration strategy (EQU). As observed from the figure, the tendency to terminate exploration earlier than is optimal, is reflected both in people's and agents' exploration patterns. This tendency, however, is observed to a greater extent with people, whereas with agents it is somehow less so. Thus we have shown that not only do agents display irrational behavior and on the other hand are more rational than humans, but we also show that the agents veer in the same irrational direction as the humans, as they stop their search before reaching the optimal point, just as humans do (rather than searching for longer than optimal).

In both of the games we have described so far one could maybe claim that the player did not fully understand what the rational behavior was, and therefore did not use it. The third game investigates a setting where we know that the human players understand what the rational behavior is, yet they still choose not to use it. In humans this would be explained as emotional involvement on the part of the player. Will agents still display irrational behavior when the designers are known to understand the rational strategy?

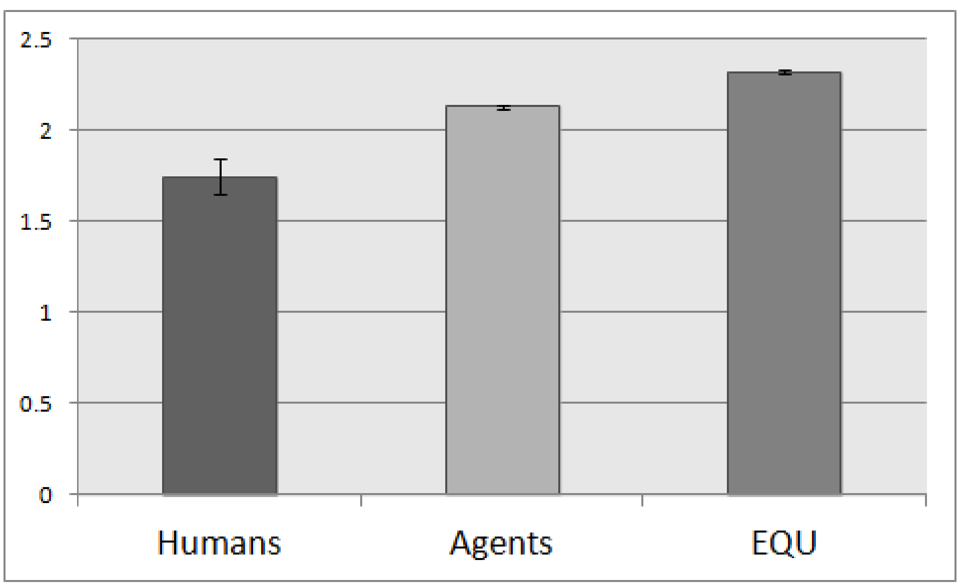


Fig. 3. People, agents and optimal agent search extent.

5 Trust-Revenge Game

In this last game, we studied a more complex two-player domain. The Trust-Revenge Game, which will be described shortly, is designed to arouse three different emotions - trust, reciprocation (or fear) and revenge.

Research with human subjects on trust, reciprocation and revenge (or punishment) has been conducted in the past. The *investment game* was first introduced by Berg et al. [21]. In the *investment game* there are two types of players. Each player is given 10 chips at the beginning of the game. Players of type A are told that they can give some or all of their chips to a player of type B (this is the trust stage). The number of chips that Player A decides to give is multiplied by 3. Then Player B can give back some or all of what he was given (reciprocation stage). The subgame perfect equilibrium for this game is for both types of players to send nothing. Berg et al. conducted the experiment with students (human subjects). As expected, the human subjects did not act according to the subgame perfect equilibrium, and chips were transferred by both types of players.

Gneezy and Ariely [17] describe a variant of the investment game which includes an additional revenge phase. In their experiment, each of the two players starts off with $\$10$. The first player has to decide whether he wishes to end the game or to pass the full amount to the second player. If he decides to pass his money, the second player receives an additional $\$40$ for a total of $\$50$. Now the second player needs to decide whether to keep all of the money or to give half of the money back to the first player. If the second player decides to keep all of the money for herself, the first player may decide to take revenge on her and pay any amount from his own private money (up to $\$25$); this amount is multiplied by 2 and subtracted from the second player's revenue. Gneezy and Ariely's experiment showed that the first player often took revenge on the second player (when the second player kept all of the money for herself).

We use a variant of the game used by Gneezy and Ariely: the Trust-Revenge Game. This game is composed of three stages: *Trust*, *Reciprocate* and *Revenge*. This game is a "one-shot" game, i.e. after the three stages are completed, the game terminates (there are no repeated interactions). There are two types of players (A and B) in the game. At the beginning of the game Players A and B are both given a certain number of chips. The first stage is the *Trust* stage, where Player A is able to give any portion of his chips to Player B. There is a factor - the Trust Rate (**tr**) - by which the number of chips is multiplied when they are passed from Player A to Player B. The second stage is *Reciprocate*: after the chips have been transferred to Player B, Player B can decide how many chips to transfer back to Player A. Player B can transfer any number of chips (which she acquires) to player A. The third and final stage is *Revenge*: Player A plays another round in which he may pay any number of chips he has to the operator. Note that the chips are not transferred to anyone, merely subtracted from Player A's stack. However, in this round, Player B must pay a factor - Revenge Rate (**rr**) - on the number of chips Player A chose for revenge. Again, the chips are not transferred to anyone, merely subtracted from Player B's stack. Both the Trust Rate and the Revenge Rate are known to both players at the beginning of the game. Consider the following example: Assume that and $\mathbf{rr} = 6$. Assume that both players started with 10 chips. Trust stage: suppose Player A gives 5 chips to Player B. After applying the Trust Rate, Player B will receive 20 chips ($5 \cdot \mathbf{tr}$). Reciprocate stage: suppose Player B decides to give 7 chips to

Player A. At the end of this stage Player A has 12 chips $(5 + 7)$ and Player B has 23 chips $(30 - 7)$. Revenge stage: suppose Player A revenges 3 chips. At the end of this stage (which ends the game) Player A has 9 chips $(12 - 3)$, and Player B has 5 chips $(23 - 3 \cdot \mathbf{rr})$.

In this game there is a clear, unique subgame perfect equilibrium (SPE) strategy. In the revenge stage, there is no rational reason for Player A to revenge, therefore in the SPE there is no revenge. In the reciprocation stage there is no reason for Player B to reciprocate since she assumes that Player A is rational and that he will not revenge, therefore in the SPE there is no reciprocation. Therefore, in the trust stage there is no rational reason for Player A to trust since he knows that Player B will not reciprocate. Therefore the SPE says: ”don't revenge, don't reciprocate, don't trust”. Since the SPE expects a 0 chip action in each of the three stages, we refer to a positive transfer (or payment) in each of the stages as expressing "emotions". While the trust stage expresses trusting and the revenge stage expresses revenging, the reciprocation stage can either express reciprocation or fear of the other player's revenge. We do not claim, though, that actual emotions are the only explanation for a positive number of chips transferred. Obviously human subjects do not use the SPE and they show all of the "emotions" mentioned, We will examine what the agents do: Is human emotion embedded in the strategy of the agents?

We had a different set of 37 undergraduate computer science students compose automated agents for the Trust-Revenge Game. Each agent played twice against all of the other agents, once as Player A and once as Player B. The same students also played the game themselves via the web against other human subjects (not against agents). The students' grades depended on their agents' performance and its documentation. The agent's performance was measured according to its final result (relative to the number of chips) and did not depend on the opponent's performance or on the average performance. The students were explicitly informed of this grading policy. Their performance as humans in the web-based game was added as a bonus to their agents' performance, assuring that the incentive to play well was identical in both cases. The students knew that the automated agents would play with other automated agents, and, when playing on the web, they knew that they were playing against a human, however they did not know against whom they were playing and we assured them that this information would remain confidential. Table 1 shows the 3 settings that were used.

Settings Index	Player A Initial	Player B Initial	Trust Rate	Revenge Rate
1	10	10	3	3
2	10	10	6	6
3	20	0	6	6

Table 1. Settings Used in the Trust-Revenge Game

In this experiment we chose to have the same group of students who programmed the agents also play the game themselves as human subjects. The students first programmed their agent and then played the game themselves. We did so for the following reasons: 1. The Trust-Revenge game requires some familiarity with the population which each player is facing. Without this familiarity it is very hard to anticipate what the other player would do and therefore hard to know whether to trust her or not. 2. By having the students first build their agents and then play we guaranteed that the human subjects understood the game at least as much as the automated agents' programmers did. 3. Having the exact same group playing both as humans and programmers eliminates any culture or education bias.

We tested both the percentage of players that gave away chips in each stage of the game (Fig.4) and the number of chips transferred (or paid) on average at each stage (Figure 5). The transfer of chips in the first stage expresses trust, in the second it expresses either reciprocation or fear of revenge, and in the last stage it expresses revenge. As can be seen in Figures 4 and 5, all "emotions" are expressed both by the human subjects and by the automated agents. Although it is clear that there is a substantial degree of "emotion" expressed by the agents, they still clearly express less "emotion" than the humans.

Another important insight we would like to mention is that using the SPE when playing versus humans or automated agents doesn't yield the highest outcome. As can be seen in Figure 5, on average Player B reciprocated more than Player A trusted, with both humans and automated agents. No Player B ever reciprocated if Player A did not trust; therefore on average Player A gained from trusting (unlike the SPE which requires that there be no trust at all).

However, the Revenge stage is different. We know from the documentation we collected from the students who designed the agents that they understood that revenge would lower their final profit and that it was not beneficial in this single-shot game. Yet we were surprised to find that they still designed agents that revenge.

6 Discussion and Conclusions

Scientifically designed automated agents or automated agents designed by experts often implement a fully rational strategy. However, most pieces of software are written by amateurs (programmers with no more than a Bachelors' degree in computer science) and therefore are likely to "suffer" from the same irrationalities as humans (but to a reduced extent). Indeed, just as agents designed by experts behave more rationally, humans who acquire vast experience in a game become more rational players [33,12].

In all settings and environments that we tested, the automated agents exhibited irrational behavior. Agents paid chips or spent time and resources even though this is not considered rational behavior. The type of irrational behavior displayed by the agents is the same as the human irrationality, but to a lesser extent. Most surprisingly, even when the rational behavior was clear to the agents and irrational behavior consisted of a clear loss of utility, agents still exhibited irrational behavior (as was the case for the revenge stage of the third game). We would like to note, that while in the first two games the agents were much closer to the rational behavior than to human behavior, in the third game the opposite occurred. We suggest that this is because that the third game involved emotions, and therefore the players' subjective utility was different than the actual expected final stake. E.g. a human may gain subjective utility from the feeling that justice

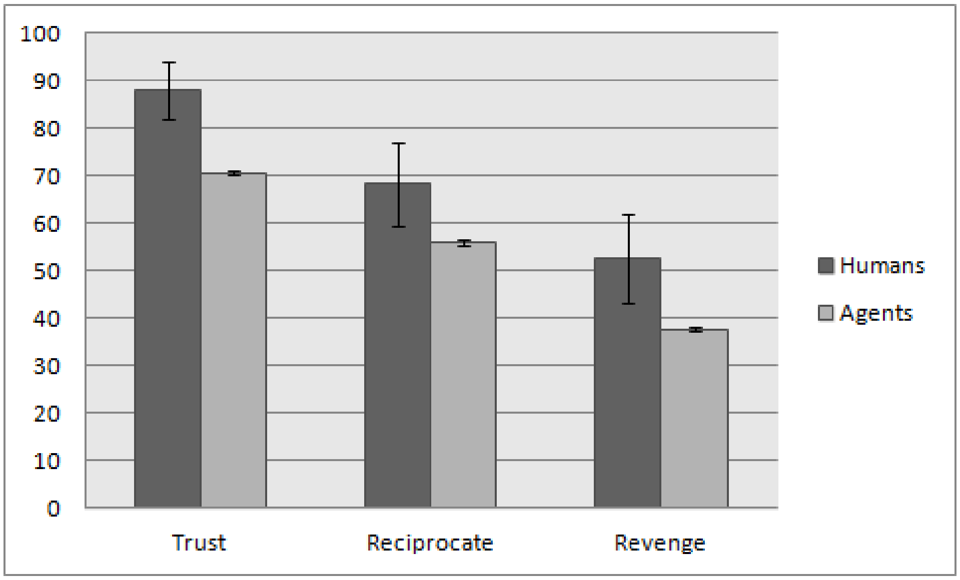


Fig. 4. Percent of Players Using Each Stage

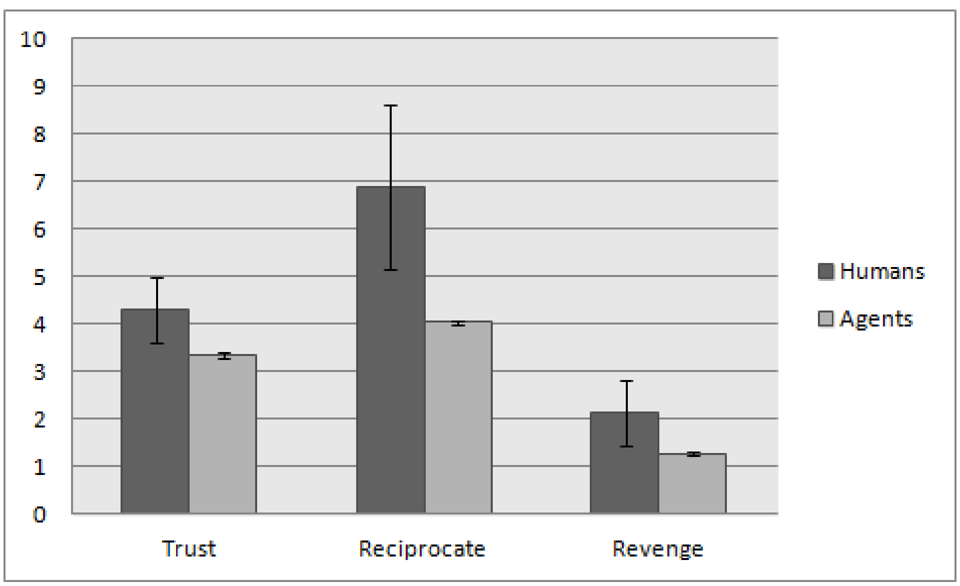


Fig. 5. Average Action in Each Stage (in chips)

is being made when he revenges, and therefore embeds this behavior in his agent. Quantifying and predicting the level of automated agents irrationality in comparison to that of humans is a topic for future work.

When building an automated agent that needs to interact with other automated agents, one needs to take into account that the other automated agents are very likely to show irrationalities that can be observed in humans. Assuming that other agents behave rationally when they actually do not, inhibits performance. In poker, for example, it is very likely that automated agents will exhibit attributes that humans use when playing, which many times aren't rational, such as over-bluffing. In electronic commerce, automated agents may be influenced by anchoring or the "sunk cost" effect. However, our results indicate that the automated agents are expected to show a smaller degree of these irrational behaviors. Therefore, considering human behavior when constructing an automated agent for playing poker, for electronic commerce or for other domains seems to be a promising approach.

7 Acknowledgment

This work was supported in part by ERC grant $\#267523$, the Google Inter-university center for Electronic Markets and Auctions, MURI grant number $W911NF-08-1-0144$ and ARO grants $W911NF0910206$ and $W911NF1110344$.

References

1. AAAI. Annual computer poker competition. http://www.computerpokercompetition.org/, 2012.
2. AAMAS. Automated Negotiating Agents Competition (ANAC). http://mmi.tudelft.nl/anac/, 2012.
3. Amazon. Mechanical Turk services. http://www.mturk.com/, 2012.
4. Dana Angluin. Learning regular sets from queries and counterexamples. *Information and Computation*, 75(2):87–106, nov 1987.
5. D. Ariely, G. Loewenstein, and D. Prelec. Coherent arbitrariness: Stable demand curves without stable preferences. *Journal of Business Research*, 59(10-11):1053–1062, 2006.
6. Peter Auer, Nicolò Cesa-Bianchi, Yoav Freund, and Robert E. Schapire. Gambling in a rigged casino: the adversarial multi-armed bandit problem. In *Proc. of FOCS*, pages 322–331, 1995.
7. A. Azaria, Y. Aumann, and S. Kraus. Automated strategies for determining rewards for humanwork. In *AAAI*, 2012.
8. A. Azaria, Z. Rabinovich, S. Kraus, and C. V. Goldman. Strategic information disclosure to people with multiple alternatives. In *AAAI*, 2011.
9. A. Azaria, Z. Rabinovich, S. Kraus, C. V. Goldman, and O. Tsimhoni. Giving advice to people in path selection problems. In *AAMAS*, 2012.
10. Jordi Brandts and Gary Charness. The strategy versus the direct-response method: a first survey of experimental comparisons. *Experimental Economics*, 14(3):375–398, September 2011.
11. C. F. Camerer. *Behavioral Game Theory. Experiments in Strategic Interaction*, chapter 2. Princeton University Press, 2003.
12. Colin Camerer and Keith Weigelt. Experimental tests of a sequential equilibrium reputation model. *Econometrica*, 56(1):1–36, 1988.
13. David Carmel and Shaul Markovitch. Opponent modeling in multi-agent systems. In Gerhard Weiss and Sandip Sen, editors, *Adaption And Learning In Multi-Agent Systems*, volume 1042. 1996.
14. John D Claxton, Joseph N Fry, and Bernard Portis. A taxonomy of prepurchase information gathering patterns. *Journal of Consumer Research*, 1974.
15. David H. Furse, Girish N. Punj, and David W. Stewart. A typology of individual search strategies among purchasers of new automobiles. *Journal of Consumer Research*, 1984.
16. Y. Gal and A. Pfeffer. Modeling reciprocity in human bilateral negotiation. In *AAAI*, 2007.
17. A. Gneezy and D. Ariely. Don't get mad get even: On consumers' revenge. *manuscript*, 2010.
18. Barbara J. Grosz, Sarit Kraus, Shavit Talman, Boaz Stossel, and Moti Havlin. The influence of social dependencies on decision-making: Initial investigations with a new game. In *AAMAS*, pages 782–789, 2004.
19. K. Hindriks and D. Tykhonov. Opponent modelling in automated multi-issue negotiation using bayesian learning. In *AAMAS - '08*, pages 331–338, 2008.
20. P. Hoz-Weiss, S. Kraus, J. Wilkenfeld, D. R. Andersend, and A. Pate. Resolving crises through automated bilateral negotiations. *AIJl*, 172(1):1–18, 2008.
21. Berg Joyce, Dickhaut John, and McCabe Kevin. Trust, Reciprocity, and Social History. *Games and Economic Behavior*, 10(1):122–142, Jul 1995.
22. Alessandro Lazaric, Mario Quaresimale, and Marcello Restelli. On the usefulness of opponent modeling: the kuhn poker case study. In *AAMAS '08*, volume 3, pages 1345–1348, Richland, SC, 2008. International Foundation for Autonomous Agents and Multiagent Systems.

23. George Loewenstein. Anticipation and the valuation of delayed consumption. *Economic Journal*, 97(387):666–84, 1987.
24. Efrat Manistersky, Raz Lin, and Sarit Kraus. *The Development of the Strategic Behavior of Peer Designed Agents*. Lecture Notes in AI, to appear.
25. Peter McCracken and Michael Bowling. Safe strategies for agent modelling in games. In *AAAI Fall Symposium on Artificial Multi-agent Learning*, October 2004.
26. Y. Oshrat, R. Lin, and S. Kraus. Facing the challenge of human-agent negotiations via effective general opponent modeling. In *AAMAS*, 2009.
27. G. Paolacci, J. Chandler, and P. G. Ipeirotis. Running experiments on Amazon Mechanical Turk. *Judgment and Decision Making*, 5(5), 2010.
28. N. Peled, Y. Gal, and S. Kraus. A study of computational and human strategies in revelation games. In *Proc. of AAMAS*, 2011.
29. Patrick Riley and Manuela Veloso. Planning for distributed execution through use of probabilistic opponent models, 2001.
30. A. Rosenfeld and S. Kraus. Using aspiration adaptation theory to improve learning. In *AAMAS*, Taipei, 2011.
31. Jorma Sajaniemi. Guest editor's introduction: Psychology of programming : Looking into programmers heads. *Human Technology: An Interdisciplinary Journal on Humans in ICT Environments*, 2008.
32. Daniel Schunk and Joachim Winter. The relationship between risk attitudes and heuristics in search tasks: A laboratory experiment. *Economic Behavior Organization*, pages 347 – 360, 09.
33. Reinhard Selten and Rolf Stoecker. End behavior in sequences of finite prisoners dilemma supergames, 1986.
34. J. Shin and D. Ariely. Keeping doors open: The effect of unavailability on incentives to keep options viable. *Management Science*, pages 575–586, 2004.
35. TAC Team, Michael P. Wellman, Peter R. Wurman, Kevin O'Malley, Roshan Bangera, Shoude Lin, Daniel Reeves, and William E. Walsh. Designing the market game for a trading agent competition. *IEEE Internet Computing*, 5(2):43–51, mar 2001.
36. The iterated prisoner's dilemma competition. http://www.prisoners-dilemma.com/, 2012.
37. A. Tversky and D. Kahneman. The framing of decisions and the psychology of choice. *Science*, 211(4481):453–458, 1981.
38. Martin Weitzman. Optimal search for the best alternative. *Econometrica*, 47(3):641–54, May 1979.