Elicitation of Strategies in Four Variants of a Round-robin Tournament: The case of Goofspiel

Moshe Dror and Graham Kendall, *Senior Member, IEEE* and Amnon Rapoport.

***Abstract*—Goofspiel is a simple two-person zero-sum game for which there exist no known equilibrium strategies. To gain insight into what constitute winning strategies, we conducted a round-robin tournament in which participants were asked to provide computerized programs for playing the game with or without carryover. Each of these two variants was to be played under two quite different objective functions, namely, maximization of the cumulative number of points won across all opponents (as in Axelrod's tournament), and maximization of the probability of winning any given round. Our results show that there are, indeed, inherent differences in the results with respect to the complexity of the game and its objective function, and that winning strategies exhibit a level of sophistication, depth, and balance that are not captured by present models of adaptive learning.**

***Index Terms*—Goofspiel, strategy.**

I. Introduction

Goofspiel is a two-person zero-sum card game [38], [39], where points won by one player are effectively lost by the other player. The rules are very simple, yet the game presents computational challenges that are yet to be resolved. To the best of our knowledge, there exist no known optimal winning strategies for Goofspiel against an opponent using an arbitrary strategy.

In an attempt to gain insight into what constitute winning strategies for Goofspiel, [11] solicited fourteen computer programs (play strategies) written by graduate students in Management Information Systems and Computer Science and pitted them one against the other in a round-robin tournament. None of the 14 programs in the competition reported in [11] dominated all other programs. Moreover, the most successful program had inferior results against another program that overall performed rather poorly. Evidently, we have not learned how to play Goofspiel. Using a similar methodology, the purpose of the present paper is to examine a sequence of round-robin tournaments with two major extensions of the classical Goofspiel game that vary both the degree of complexity of the game and its objective function. Our purpose is to determine the properties of successful strategies for playing Goofspiel.

Axelrod [2] presents the results of the earliest well-known round-robin tournament of computerized strategies. His ground-breaking work has been motivated by the desire to find an answer to the question of what is the most effective strategy for playing the iterated Prisoner's Dilemma (PD) game (see [24] for a discussion on recent work). For that purpose, Axelrod invited game theorists, all familiar with the topic, to enter a round-robin finitely iterated PD game tournament designed to determine which strategies perform best in practice. In a second version of this PD tournament played a few years later, the rules of the competition were the same with the only exception that the exact number of iterations was not disclosed in advance. Thoughtful summaries and discussions of this research have been provided by [18], [27], [29], and many others. Subsequent research has extended the same methodology to other games (e.g., [7]), including variants of the PD game with an exit option [28], or finite memory ([22]. Although the present Goofspiel study shares the same methodology as the PD earlier studies, the conclusions that it draws about winning strategies in this class of two-person zero-sum games are quite different.

The outline of this paper is as follows: In Section II, we state the rules of the classical Goofspiel game. In Section III, we describe and discuss two of its extensions. Section IV summarizes several observations about learning in zero-sum games. The experimental framework is presented in Section V. The results of the competition are summarized in Section VI. Section VII concludes. The outline of the format of the computer program and software framework are the same as in [11], and are omitted from this paper.

M. Dror is with The University of Arizona, USA. e-mail: (mdror@eller.arizona.edu).

G. Kendall with the ASAP Research Group, School of Computer Science, University of Nottingham, UK and University of Nottingham Malaysia Campus. e-mail: (graham.kendall@nottingham.ac.uk or graham.kendall@nottingham.edu.my)

A. Rapoport is with the Graduate School of Management, University of California, Riverside, USA. e-mail: (amnonr@ucr.edu).

Manuscript received mmm dd, yyyy; revised mmm dd, yyyy.

Readers interested in the more detailed description of the computer strategies provided by the participants and/or the original 32 tables of the results for all the competitions in the tournament should access the supplementary file associated with this paper.

II. Goofspiel

In the standard (classical) Goofspiel game there are two players. One of the four suits is given to player I, a different suit to player II, and a third suit is shuffled and placed face down between the two players. The fourth suit is discarded and plays no further role in the game.

The game starts by revealing the top card from the shuffled suit, that has been placed between the two players. Each of the two players selects one of their cards as a bid for the value of the upturned card. These bids are made simultaneously. The player who selects a card with a higher value than their opponent wins the face-up card. In case of a tie between the two player's cards, the three exposed cards are discarded and no player wins any point. If one player has selected a higher card then his opponent, he keeps the card for accounting purposes. The two cards played by the players are discarded. Another card is now revealed and the game is repeated until all 13 face down cards have been exposed. We refer to this 13-card sequence as a *round*. The number of rounds is determined by the two players before the game commences.

In any given round, the player with the higher total value of cards won is the winner of that round. The numerical difference between the two players' total values is called the *winning margin* (or *point differential*). The objective in classical Goofspiel is to maximize the cumulative points won across all the rounds against a given opponent. With the cards in each suit marked from 1 to 13 ($Ace = 1, 2 = 2, \ldots, 10 = 10, J = 11, Q = 12, K = 13$), there is a fixed number of points (at most 91-1 = 90) to be won in each round. The value is 90, as when a player, plays their King, they are guaranteed to win the up card. This is at least an Ace (one point). Of course, in case of a tie(s) some (or even all) of the points in each round may be wasted. Importantly, maximizing the cumulative sum of the points won and maximizing the number of rounds won against an opponent are two different objectives; player $i, (i = I, II)$ may accumulate a higher total number of points than her opponent $j, (j = I, II, i \neq j)$, while player j records a higher number of wins. We discuss this issue below.

In contrast to Games with Finite Resources [16], the payoffs in Goofspiel are revealed at each move, one at a time. Therefore, there is no static payoff matrix because the pair of the cards chosen in each move is stochastic with respect to the payoff value.

Ross [30] notes that the number of pure strategies for each player is, $n^n \times \Pi_{k=1}^{n-1} k^{k(k+1)}$. For example, the number of pure strategies for $n = 4$ for Goofspiel is 8.4 billion (an 8.4 billion by 8.4 billion game). For $n = 5$ the number of pure strategies goes up to about 10^{23} [30]. Moreover, a 'successful' Goofspiel strategy must include randomization since it can be shown that any deterministic strategy can be easily defeated. The above 4-card example illustrates the computational challenge of constructing effective strategies for Goofspiel.

Only a few theoretical results for Goofspiel are known [30]. For instance, if Player I observes that Player II chooses her card at random in each turn, then to maximize her expected payoff value in a repeated Goofspiel, Player I should match the upturned value with their own card value [10]. This is known as the *the upcard-matching strategy* or in short the *Matching Strategy* – (MS).

III. Extensions

As mentioned earlier, the focus of our study is on identifying effective strategies for two major extensions of Goofspiel, one concerned with the complexity of the game and the other with its objective function. In what follows we describe these two extensions in some detail.

A. Goofspiel with Carryover

The first extension (or variant) of the classical Goofspiel, which we term *Goofspiel tournament with carryover*, was introduced in [30]. The difference between the classical and carryover variants of Goofspiel is quite simple. In the carryover variant, if both players I and II bid the same value (i.e., a tie) for a given exposed card in step t, provided that step t is not the last step in the sequence, then the value of the exposed card in step t is added to the value of the next exposed card in step $t + 1$ and the game continues as before. If a tie occurs on the last exposed card in the round, then no player wins the last value. Despite this minor change in the rules in case of a tie, its effect on the complexity of the game is profound. Ross claims that Goofspiel with carryover is a much more difficult game to analyze or even compute the number of pure strategies. To our knowledge, there are no published results on Goofspiel with carryover. Using a 2×2 within-subject design, the present study reports the results of both the classical and carryover variants of the game. Each of these two variants is played in a round-robin tournament under two different objective functions.

B. Objective Functions

To motivate these two objective functions we re-examined the round-robin tournament reported in [2]. In this first tournament, Axelrod invited professional game theorists to write computer programs that participated in a round-robin tournament for exactly 200 moves (a game) against each opponent.

Axelrod reported that TIT-FOR-TAT (TFT), the simplest of the 14 programs that had been submitted, won the tournament ([2], p. 31). To remind the reader, TFT chooses the cooperative strategy at step $t = 1$. Thereafter, at each t it mimics the opponent's move at step $t - 1$. TFT is not the first to defect, but it is not the first to cooperate either except by design on step $t = 1$. TFT only carries a memory of the outcome of the last step, plays each move as if it is the last in the game, and learns nothing about its opponent during the entire course of the game. TFT cannot win any particular round with another strategy. It can, at best, tie. It is able to win on total points by losing to many opponents, but still getting a relatively high score The opponents then get lower scores against other opponents. Given this, if we evaluate TFT on pairwise wins, it would perform poorly.

Analyzing the results from that first tournament shows that if we produce rankings (number of wins versus total number of points) a Spearman rank correlation between the two rankings is $r_s = -0.154$. On the basis of this result, we cannot reject the null hypothesis that the two rankings in Axelrod's first tournament are independent ($p > 0.05$).

One might possibly expect that the two objective functions of maximizing the probability of winning against any given program and maximizing the total number of points won across all the opponents in a round-robin tournament are not perfectly correlated. But the finding reported above that in the first tournament of Axelrod these two objectives are not significantly correlated may come as a surprise. To further assess the generality of this finding, we searched for another tournament between computer programs playing iterated PD games with possibly different rules and a larger number of participants. Such a round-robin tournament was presented in [21]. In contrast to the tournament organized by Axelrod in 1984, the tournament in 2004 allowed for the addition of random noise, which resulted in the misrepresentation of some of the moves. Additionally, teams of genuine players could submit multiple programs and many did so. Altogether, the 2004 tournament included 223 computer programs [21]. The results of the tournament are displayed in http://www.prisoners-dilemma.com/results/cec04. Fortunately, both the number of pairwise competitions won by each of the 223 programs and the total sum of the number of points won across all the program's opponents are listed in the website. The correlation between the two scores (the number of competitions won and the total sum of the number of points won for $n = 223$) is $r = -0.45$; it is negative and highly significant ($p < 0.001$).

Jointly evaluated, these results suggest that to achieve generality in answering the basic research question of the present paper, more than a single explication of successful strategies should be explored. In what follows, for both the classical and carryover variants of Goofspiel tournaments, we present the results of round-robin tournaments under two different objective functions. One version has the objective of maximizing the cumulative value of exposed cards won against each opponent (of great consequence in zero-sum games), and in the other, simpler, version the objective is to maximize the probability of winning in each round.

IV. Comments on Learning in Games

Automated game playing has a long history that can be traced back to at least the 1950's [34], [32]. Samuel pioneered learning in games via the co-evolution of a checkers player [32], [33], where a population of agents played against each other in order to evolve ever better players. It was Samuel's work, as well as the recent defeat of Kasparov by Deep Blue [9], that motivated Fogel [12] to continue this challenge. He utilized evolutionary artificial neural networks as checkers board evaluators. The final player evolved by Fogel was able to play at the level of a human expert [13].

Fogel's work motivated other researchers to continue investigating learning methodologies for checkers [1], as well as investigating evolutionary neural networks for other games, such Blackjack [20], Pac-Man [25] and Go [8]. The methodology has also been used in other domains, such as predicting share prices [35], and forecasting inflation [3].

A co-evolutionary approach has been compared to Temporal Difference Learning (TDL) in evaluating Othello positions [26]. It found that TDL learns faster than a co-evolutionary approach although a suitably tuned co-evolutionary algorithm is able to learn better strategies, demonstrating that more research is needed to establish the most suitable algorithm. Indeed, recently Monte Carlo Tree Search (MCTS) [5] has attracted significant attention in game playing, motivated by its success in Computer Go [23], with games such as poker [31] being highlighted as being able to benefit from MCTS.

When playing repeated two-person games, it is common for the players to attempt learning how to play the game (improve their game performance) as the game progresses by forming beliefs about their opponents' strategies based on the history of play. Learning how to play games is an important topic with a vast literature. Major examples include reinforcement-based and belief-based models of adaptive learning, as well as models that postulate minimization of regret. However, these models are not suited to capture the sophistication and complexity of game strategies that experts employ in playing Goofspiel. They assume myopic behavior, stationary environments, and parameter values that remain fixed over iterations of the stage game; these attributes do not characterize many real-life conflicts, even one as simple as Goofspiel.

The existence of pure-strategy Nash equilibrium in two-person zero-sum games is usually claimed by reference to Zermelo's Theorem (see [36]). However, Zermelo's Theorem does not apply to Goofspiel because of the uncertainty introduced by the simultaneous card selection of both players. Since we know that no pure strategy may consistently outperform general strategies in Goofspiel (there is no pure-strategy Nash equilibrium for Goofspiel), we ought to consider learning mixed strategies see [15]. However, non-stationarity in the opponent's strategies in Goofspiel derails the known classical approaches in the literature on learning in games. Subsequently, we are forced to resort to experimental examinations and computational search for successful Goofspiel playing strategies by conducting computerized round- robin competitions as described in [11] and extended in this paper.

V. Experimental Framework

In this section, we discuss the 4×4×2 round-robin competitions that were conducted in the course of this study. Each player submitted 4 different versions of Goofspiel. Each version participated in 4 round-robin competitions of 10, 20, 30, and 50 rounds. For each version, we calculated the number of points and the numbers of rounds won for a total of 32 tables. Their tabulated results are presented in 32 tables which are available in the supplementary file. Each of the 32 tables reflects the results of a single, 11-player, round-robin Goofspiel competition specified by the winning criterion (objective function), the specific form of Goofspiel (with or without carryover), and the number of rounds played by each pair of programs. Each entry in these tables reports the point differential between a pair of players in a single execution of a specific competition with a fixed number of rounds. For illustration, Table I presents the results of a 10-round round-robin Goofspiel with carryover competition when the players' objective is to maximize the cumulative number of points won across the ten rounds against each opponent. Table I shows that program 0209 won in 10 rounds against RANDOM with a point differential of 307 points. Program 0203 won by total of 42 points more than RANDOM when playing ten rounds, but lost by a total of 126 points against program 0209 and by a total of 134 points against program 0205.

TABLE I
Score(P1-P2) - Carryover maximizing points - 10 rounds

P1 \ P2	Random	0209	0205	0203	0210	0207	0206	0202	0204	0208	0201
Random	-	-	-	-	-	-	-	-	-	-	-
0209	307	-	-	-	-	-	-	-	-	-	-
0205	102	188	-	-	-	-	-	-	-	-	-
0203	42	-126	-134	-	-	-	-	-	-	-	-
0210	68	-136	-27	10	-	-	-	-	-	-	-
0207	-20	154	24	-244	2	-	-	-	-	-	-
0206	186	-254	-2	66	-247	198	-	-	-	-	-
0202	136	83	-223	-17	-2	16	72	-	-	-	-
0204	212	-207	-207	-55	-142	15	-144	-86	-	-	-
0208	195	190	-19	-54	65	-22	243	204	302	-	-
0201	260	358	51	136	166	-49	256	132	213	93	-

The computer programs were submitted by ten Ph.D. students from the University of Arizona. We have added the random card selection strategy (RANDOM) to each round-robin competition as a benchmark strategy. The competitions were held with 10, 20, 30, and 50 repeated games (rounds) against each opponent in each competition. Competitions with increasing number of rounds aim to assess the rate of learning and subsequent strategy adjustment decisions encoded by the players in their programs. The players were told that their final grade in the course would partly be contingent on the performance of their entries.

A. *Short Summary of the Players' Strategies*

This subsection provides a brief summary (the main idea) of each participant's playing strategies in our round-robin competition. More detailed descriptions are available in the supplementary file associated with this paper.

The genuine players – programs in the competitions – are labelled from 0201 to 0210, and Random.

(1) Player 0201 designed her strategy around the principal idea that the program ought to perform well when considering five reasonable play strategies that her opponent is likely (in her judgment) to play.

(2) The logic of Player 0202's strategy revolves around the idea of how much risk she ought to play in the current move. To that end, she calculates four ratios before each move and uses these ratios in a convex combination to calculate her daring signal which, in turn, determines her play card.

(3) Player 0203 aims to employ on each move either an offensive strategy, such as playing the upcard value $+x, (x =$ some nonnegative integer less than 5), or a defensive strategy, such as randomizing, and alternates between the two if there is no guaranteed good play.

(4) The strategy submitted by Player 0204 attempts to imitate the probabilistic play as described by Rhoads and Bartholdi (2012).

(5) Player 0205 bases her play strategy on a number of key elements: (i) valuation of cards; (ii) a learning routine; (iii) bounded random play, and (iv) defense mechanism against play learning by the opponents. Since this is the best performing program in our Goofspiel competition, we examine the reasons for its success in a subsequent section.

(6) Player 0206 selects her card for each move based on the ratio of the total number of points earned so far in the round divided by the total number of points spent so far in the round. This ratio is referred to as Point Quality. The intent on every move is to increase the difference of the two players' Point Quality values in her favor. The difference between the Point Quality scores determines between choosing an aggressive or non-aggressive play.

(7) Player 0207 first determines four intuitive Goofspiel strategies. In actual play the four strategies are computed iteratively based on their immediate or consecutive success or failure.

(8) Player 0208 essentially runs a rolling time horizon Monte Carlo simulation for two consecutive plays in each turn to determine her play card. In addition, the strategy calls at times for random play to guard against a play-learning opponent.

(9) Player 0209 assumes that her opponent is likely to select one of five fairly well-known strategies and attempts to learn from the play history which strategy was chosen by her opponent in order to respond with myopic play. It turns out to be one of our most successful strategies.

(10) Player 0210 opts to sacrifice the first few rounds by playing randomly so that she may gather data on her opponents play. Then, she introduces a number of rules that trigger a degree of aggressive play that also allows for randomization.

VI. The Competition

The present section discusses the tabulated results (the original 32 tables - see the supplementary file) of the round-robin competition. Many different summaries can be tabulated based on the 32 basic tables. For instance, Tables II and III represent one such summary of the results. Table II depicts the results for Goofspiel with no carryover (and similarly Table III for Goofspiel with carryover) for 50-round competitions and the objective of maximizing the cumulative number of points won against each opponent in contrast to the objective of maximizing the number of rounds won against each opponent. The column labeled % in W. Point Diff. presents for every player the sum of the positive point differences against each of this player's opponents, dividing (normalizing) this sum by the maximum points a player can win in such Goofspiel competition $90 \times 10 \times 10$ and $90 \times 50 \times 10$, respectively. For instance, program 0205 had a winning point differential in a 50-round tournament of 21.0 % (Table II). It is the highest point differential indicating the relative strength of program 0205. A similar interpretation holds for Table III for Goofspiel with carryover. We note, however, that in Goofspiel with carryover program 0205 did not achieve the highest % in W.P. Diff. Rather, Program 0201 had a score of 13.2 versus 10.4 for program 0205.

Next we turn to Tables IV, V, VI, and VII, for the count of the number of defeated opponents in the 50 round-competitions. Table IV depicts for each program the number of opponents defeated by point differential and by the number of rounds differential for Goofspiel with no carryover and the objective of maximizing the total number of rounds won against each opponent in 50-round competitions. Table IV shows that program 0205 defeated all ten opponents by points won in the 50-round competitions, and nine opponents with respect to the number of rounds. However, with respect to rounds it lost to one opponent in the 50-round competition. From the round-robin results tables we note that in the 10-round competitions program 0205 did not lose and had one tie. On the other hand, program 0207 defeated five opponents by points in the 10-round competitions and six opponents in the 50-round tournament. In addition, program 0207 raised the number of defeated opponents from 6 to 7 when switching from the 10 to the 50-round competition. Regarding Table V with an objective of maximizing total points, there are three successful programs (0201. 0205, and 0208) that increased the number of defeated opponents when increasing the number of rounds from 10 to 50-round competitions. Tables VI and VII differ from Tables IV and V, respectively, by depicting the results from Goofspiel with carryover instead of the Goofspiel with no carryover.

For the purpose of identifying the most successful programs/players, we present in Table VIII for each player separately a 1×16 vector that lists the number of defeated opponents, ignoring ties, in 16 round-robin competitions starting with the 8 round-robin competitions of 10 rounds each in order of the tables, and followed in the same order by the 8 round-robin competitions of 50 rounds each. The entries in bold in each position in at least one of the 11 vectors strongly dominate all the non-bold entries in the same position in corresponding vectors. Program 0205 is by far the most successful program, with program 0208 second and 0201 third based on the number of the round robin competitions it dominated as reflected by the number of defeated programs. However, when assessing the success of a program by the smallest number of opponents defeated in any one of the 16 round robing competitions, program 0201 is number one with a minimum of seven defeated opponents, followed by 0205 with five, and 0208 with four, which is also true for 0210. At the other end of the spectrum, 0204 can be judged as the weakest program by virtue of maximum of four and minimum of one defeated opponents. We exclude Random from this ranking but note that Random defeated two opponents in one of the 16 competitions.

In an attempt to identify success patterns, we inspect the differences in program's results (the original 32 tables) as a function of the number of rounds in otherwise identical round-robin competitions. We note that programs 0205, 0203, and 0204 improve their success score, depicted by the winning point differential, when switching from 10 to 50 rounds: 14.5 to 21.0, 3.5 to 5.4, and 3.9 to 6.4 respectively. Similarly, programs 0205, 0203, and 0204 improve their success score regarding winning round differential from 47.0, 16.0, 12.0 to 61.6, 16.3, and 20.0, respectively. Similar improvement is registered for programs 0205, 0202, 0207, and 0206. This observation repeats itself for programs 0205, 0207, 0202, 0204. It is not entirely clear what we 'learn' from these observations. For instance, program 0203 has a built-in learning subroutine that aims to identify if the opponent employs a matching-like strategy and if so it triggers an optimizing response. For program 0202, one cannot identify a learning intent. Program 0205 has some learning and adjusting response built into the program. However, for Goofspiel with carryover this 'learning' phenomena disappears when the objective switches to maximization of total rounds won. That is, as depicted in the bottom half of Table III, program 0205's percentage of winning points differential and % of winning rounds differential both decrease when the number of round increases from 10 to 50. Therefore, we do not detect a trend of play improvements as a function of the number of rounds. If such a phenomenon exists, it is a function of other factors and not just the number of rounds.

A. *Why is program 0205 the most successful?*

Quoting from the description of program 0205: "...naive strategies that only consider the value of the upcard can easily become overwhelmed. For this reason, my strategy involves a number of more sophisticated elements here, including: (i) *Tiered valuation of cards*, (ii) *A learning algorithm based on monitoring of historical play*, (iii) *Bounded random play*, (iv) *Mitigations against learning algorithms*."

"Four versions of my Goofspiel algorithm were created: maximize points (carryover), maximize points (no carryover), maximize wins (carryover), and maximize points (no carryover). Each version uses the same basic set of strategies with a few unique features".

We note that program 0205 considers the strategic elements very carefully. For instance, "*In Goofspiel with no carryover and a goal of maximizing the number of rounds won, the aim is to win one more than half of the available points in a given round (to win at least 46 points). As the game progresses ties may occur causing points to be discarded and thus lowering the total number of points needed to win. Therefore at the start of each hand first calculate the number of points sufficient to win: Winning Score = (My Score + Opponent Score + Points remaining in deck)/2.*"

To find the 'playing path' to the winning score, the values of each card are added to find combinations that have a sum greater than the winning score. Using a tiered valuation system, these cards are stored in a winners' array because they are the cards that can win the game in the minimum number of plays. Once the winners have been determined, a second valuation tier called **helpers** is created. Any two helper cards may be used to replace any one winner card up to the highest winner. In the first hand the helpers are 9, 8, 7, and 6. The sum of any two helpers must be greater than or equal to the highest winner e.g., five is not a helper because despite $9+5 \geq 13$ being true, $6+5 \geq 13$ is not true. The final tier of the valuation system is **tossers**, which are cards that would require three or more to be useful. As the game progresses, however, the number of points needed to win changes for three reasons: (a) ties, (b) you may loose cards in your shortest path to victory, (c) you may win smaller cards, expanding your victory path options."...

"Monitoring Historic Play: If we have seen a single card frequency $\geq$ 0.5 played for a given upcard, we assume it is the card they will play thus will play that card $+1$. If we do not have it, we try the next highest card until we find one that is available." In addition, 0205 has what she referrers to as “**Sanity Check:** ... If the card you are playing is more than three higher than the upcard and the upcard is not a winner then the **bounded strategy** takes over. **Bounded Random Play:**...play one of the highest available cards, randomly selected... However, if my opponent has cards higher than all of the cards in my bounded subset, I play my lowest available card.”

In summary, program 0205 presents a carefully considered and very balanced approach for selecting the play card in each turn. For extensions of strategy 0205 and a more complete evaluation of the success of a modified version of program 0205, see [17].

B. Results for program 0205: The two objectives with and without carryover – 50 round round-robin competitions

Since one of the aims of this paper is to study the differences in complexity and the objective functions of the different competitions, we return to the round-robin results to determine if the results suggest inherent differences. Our conclusion is that there are inherent differences in the results with respect to complexity and the two objectives. We singled out program 0205 as the representative program and restricted the examination to the 50-round competitions. We report the performance of program 0205 in the two dimensions against each of the other nine programs in four 50-round competitions.

We observe a clear and consistent difference between the results for Goofspiel with carryover (CO) and without carryover (NCO) for both objectives. For instance, playing 50 rounds against program 0201, program 0205 won by 138 and 194 more points than program 0201 when the objectives were, respectively, Max the number of rounds won and Max the total points won for the carryover games. Clearly, the objective function does matter in the formulation of program 0205. For the games with no carryover program 0205 won by 685 and 571 more points than program 0201 when the objectives were, respectively, Max the number of rounds won and Max the total points. With respect to number of win differentials, program 0205 won by 10 rounds more than program 0201 for both objectives in games with carryover and by 28 and 21 rounds in games with no carryover when the objectives were, respectively, Max the number of rounds won and Max the total points.

Differences exist between the results in games with and without carryover and even more drastic differences with respect to the two objective functions. The results depict drastic differences for the two game formats but less drastic with respect to objectives, and the opposite holds when program 0205 plays program 0206. When playing 0206, program 0205 wins by points differentials for both games and both objectives but looses by 7 both times with respect to number of wins differentials when maximizing the number of rounds won.

VII. Conclusions and Discussion

To place the results of our tournament in perspective, a comparison between the preset Goofspiel tournament and Axelrod’s previous tournaments is warranted. Essentially, viewing the finitely iterated PD game as the E. Coli of social psychology ([2], p. 30), and finding previous experimental literature on the PD to be of no help, Axelrod proposed a computer tournament for studying the effective choices in the iterated PD game that satisfy two major requirements: (i) that the effectiveness in playing the game depends not only upon the characteristics of a particular strategy, but also on the nature of other strategies with which it must interact; (ii) that at any round of the game the effective strategy should take into account the entire history of the repeated interaction up to that point. To satisfy these requirements, Axelrod made three important choices that every tournament must consider: (1) the format of the tournament, (2) the objective function that the players are assumed to optimize, and (3) the population of the contestants. Without giving an explanation as to the reason, Axelrod opted to structure the multi-player competition as a round-robin tournament. He solicited programs from professional game theorists, and instructed them to maximize the cumulative sum of points won against all the competitors. None of these choices is mandated by Axelrod’s original goal. For instance, rather than structuring the competition as a round-robin tournament, one might have structured it as a single elimination tournament. As we stated earlier, alternative objective functions might have been proposed including maximization of the number of wins or maximization of the cumulative point differential across all competitors. There is also a choice between alternative populations of players based on their expertise or risk-taking attitude; in fact, in his second tournament the request for a population of expert game theorists was relaxed and, instead, the tournament was open to the entire public. Our conjecture is that the outcomes of the multiple tournaments of the iterated PD reported in the literature in the last 30 years, and in particular the predominance of TFT, do not generalize beyond the particular combination of tournament format and objective function that they all share.

Our search for effective strategies for Goofspiel was predicated on the same three critical choices made by Axelrod. The participants were PhD students from management information systems and computer science. The project appeared to motivate them, as exhibited by the time and effort they put into the devising computer programs for the tournament. The multi-player competition was similarly structured as a round-robin tournament, thereby conditioning the effectiveness of any particular strategy on the nature of all the strategies with which it might interact. And one of the four variants of the programs required the participants to maximize the cumulative sum of points won against all the competitors as in Axelrod's tournament. However, the results of the Goofspiel tournament could not have been more different. The most effective strategy, that has dominated the tournament in all the four variants of Goofspiel, does not share the simple and highly transparent properties of TFT or its extensions studied in [28], [37], and others. In particular, it is a highly sophisticated program that monitors the history of play, forms hypotheses about the opponent, considers alternative strategies that are conditioned on the state of the competition at any move, and selectively randomizes the choice of cards in order to keep the opponent off track. It has depth and balance, not unlike the characteristics of a good chess computer program, which are completely absent in TFT or its extensions.

One possible reason for the difference between the fundamental characteristics of these two efficient programs may have to do with the differences between the two games. The only uncertainty in playing the iterated PD game is strategic: A player does not know which of the two options her opponent is about to choose on any particular move. In contrast, the player in Goofspiel faces two sources of uncertainty on any particular move, one strategic and the other environmental due to the random ordering of the cards in the suit that is being exposed. In addition, the PD is a nonzero-sum game in which the Pareto dominant outcome is achieved if both players cooperate, whereas Goofspiel is a zero-sum game in which preferences over the outcomes are diametrically opposed, thereby excluding any incentive to cooperate. Finally, whereas the PD presents each player with the same binary choice on each move, Goofspiel presents each player with multiple options that differ across moves. These features of Goofspiel, which may exist in other repeatedly interactive games in one combination or another, call for considerable sophistication and balance between different objectives in devising efficient strategies. Chess, Go, and backgammon, all zero-sum games that still defy complete analysis, serve as instructive examples.

VIII. Future Research Directions

This paper has tackled Goofspiel in the same way Axelrod tackled the Prisoner?s Dilemma. That is, we invited competitors to participate, and conducted a round- robin competition. We restricted the number of entries to Ph.D. students. It would be interesting to open the competition to a larger number of competitors. Given the access that we now have to social media, email etc. (which Axelrod never had) it should be possible to hold a much larger competition, which should provide further insights.

As we discuss in Section IV, learning in games has a long and varied, history. It would be interesting to investigate some of the methodologies that others have utilized on Goofspiel. Monte Carlo Tree Search might prove useful in searching for the best card to play. It would also be interesting to consider evolutionary and co-evolutionary approaches in seeking good Goofspiel strategies.

Since the time that Axelrod popularized the Prisoner?s Dilemma a lot of research has been conducted, much of which has studied the game from a theoretical point of view. There has been limited theoretical work on Goofspiel, and this might prove to be a fruitful research direction.

References

[1] Al-Khateeb B. and Kendall G Introducing Individual and Social Learning Into Evolutionary Checkers. *IEEE Transactions on Computational Intelligence and AI in Games*, 4(4), 258-269, 2012.

[2] Axelrod, R. The Evolution of Cooperation. BASIC Books, New York, 1984.

[3] Binner J. M., Tino P, Tepper J, Anderson R, Jones B and Kendall G. Does money matter in inflation forecasting?. *Physica A: Statistical Mechanics and its Applications*, 389(21), 4793-4808, 2010.

[4] Brown, G.W. Iterative solution of games by fictitious play. in Activity Analysis of Production and Allocation. NewYork,NY,USA: Wiley, 1951, pp. 374-376

[5] Browne C . B., Powley E., Whitehouse D., Lucas S. M., Cowling P. I., Rohlfshagen P., Tavener S., Perez D., Samothrakis S. and Colton S. A Survey of Monte Carlo Tree Search Methods *IEEE Transactions on Computational Intelligence and AI in Games*, 4(1), 1-43, 2012

[6] Battigalli, P., M. Galli, and M.C. Molinari. Learning and Convergence to Equilibrium in Repeated Strategic Interactions: An Introductory Survey. *Ricerche Economiche* 46, 335-278, 1992.

[7] Billings, D. Thoughts on RoShamBo. *International Computer Games Association Journal* 23(1), 3-5, 2000.

[8] Cai X., Venayagamoorthy G. K. and Wunsch II D. C. Evolutionary swarm neural network game engine for Capture Go *Neural Networks*, 23(2), 295-305, 2010

[9] Campbell M., Hoane A. J. and Hsu F. H. Deep Blue *Artificial Intelligence*, 134(1-2), 57-83, 2002.

[10] Dror, M. Simple proof of Goofspiel: The game of pure strategy. *Advances of Applied Probability*, A Note, 21, 711-712, 1989.

[11] Dror, M., G. Kendall. Repeated Goofspiel: A Game of Pure Strategy. *IEEE Transactions on Computational Intelligence and AI in Games*, 5(4), 312–324, 2013.

[12] Fogel D. B. Blondie24: Playing at the Edge of AI. Morgan Kaufmann, 2001

[13] Fogel D. B. and Chellapilla K. Verifying Anaconda's expert rating by competing against Chinook: experiments in co-evolving a neural checkers player *Neurocomputing*, 42(1-4), 69-86, 200
[14] Foster, D.P. and H.P. Young. Learning, Hypothesis, and Nash Equilibrium. *Games and Economic Behavior* 45, 73-96, 2003.
[15] Fudenberg, D. and D.M. Kreps. Learning Mixed Equilibria. *Games and Economic Behavior* 5, 320-367, 1993.
[16] Gale, D. Information in games with finite resources. *Annals of Mathematical Studies* 39, 141-145, 1957.
[17] Grimes, M., M. Dror. Observations on a 'Good' Strategy for Playing Repeated Goofspiel. *MIS Working Paper*, University of Arizona, 2013.
[18] Hofstadter, D.R. The calculus of cooperation is tested through a lottery. *Scientific American*, 248, 14-28, 1983.
[19] Kalai E. and E. Lehrer. Rational Learning Leads to Nash Equilibrium. *Economerica* 67, 1019-1045, 1993.
[20] Kendall G and Smith C. The evolution of blackjack strategies. *In Proceedings of the The IEEE 2003 Congress on Evolutionary Computation (CEC2003)*, 2474-2481, Canberra, Australia, 2003
[21] Kendall, G., X. Yao, and S. Chong. The iterated prisoner's dilemma: 20 years on. World Scientific, 2007.
[22] Kretz, T. A round-robin tournament of the iterated prisoner's dilemma with complete memory-size-three strategies. *Complex Systems* 19, 363-389, 2011.
[23] Lee C. -S., Wang M. -H., Chaslot G. M. J. -B., Hoock J. -B.Rimmel A., Teytaud O., Tsai S. -R., Hsu S. -C. and Hong T. -P. The computational intelligence of MoGo revealed in Taiwan?s computer Go tournaments. *IEEE Trans. on Computational Intelligence and AI in Games*, 1(1), 73-89, 2009.
[24] Li, J., P. Hingston, and G. Kendall. Engineering design of strategies for winning iterated prisoner's dilemma competitions. *IEEE Trans. on Computational Intelligence and AI in Games*, 3(4), 348-360, 2011.
[25] Lucas S. M. Evolving a Neural Network Location Evaluator to Play Ms. Pac-Man *Proceedings of the 2005 IEEE Symposium on Computational Intelligence and Games (CIG 2005)*, 203-210, 2005
[26] Lucas S. M. and Runarsson T. P. Temporal Difference Learning Versus Co-Evolution for Acquiring Othello Position Evaluation *Proceedings of the 2006 IEEE Symposium on Computational Intelligence and Games (CIG 2006)*, 52-59, 2006
[27] Maynard Smith, J. Game theory and evolution of behaviour. *Behav. Brain Sci.* 7, 95-101, 1984.
[28] Phelan, S.E, R.J. Arend, and D.A. Seale. Using an iterated Prisoner's Dilemma with exit option to study alliance behavior: Results of a tournament and simulation. *Computational and Mathematical Organization Theory* 11, 339-356, 2005.
[29] Rapoport, A. Prisoner's Dilemma: Reflections and recollections. *Simulation Gaming* 26, 489-503, 1995.
[30] Ross, S.M. Goofspiel - The game of pure strategy. *Journal of Applied Probability* 8, 621-625, 1971.
[31] Rubin J. and Watson I. Computer poker: A review *Artificial Intelligence* 175(5-6), 958-987, 2011.
[32] Samuel A. L. Some studies in machine learning using the game of checkers, *IBM Journal of Research and Development* 3(3), 210-229, 1959.
[33] Samuel A. L. Some studies in machine learning using the game of checkers. II - Recent Progress. *IBM Journal of Research and Development*,11(6), 601–617, 1967.
[34] Shannon C. E. Programming a Computer for Playing Chess *Philosophical Magazine*, Ser.7, 41(314), 1950
[35] Kendall G. and Su Y. Imperfect Evolutionary Systems. *IEEE Transactions on Evolutionary Computation*, 11(3), 294-307, 2007.
[36] Schwalde, U. and P. Walker. Zermelo and the Early History of Game Theory. *Games and Economic Behavior* 34, 123-137, 2001.
[37] Slany, W. and W. Kienreich. On some winning strategies for the Iterated Prisoner's Dilemma or Mr. Nice Guy and the Cosa Nostra. CoRR abs/cs/0609017, 2006.
[38] Tucker, A., The Princeton Mathematical Community in the 30's, An Oral-History Project, Transcript Number 11 (PMC11), The Trustees of Princeton University, 1985.
[39] Tucker, A., Merrill Flood (Interview): The Princeton Mathematics Community in the 1930s: Transcript no. 30, 2000.

Acknowledgement: Special thanks to Shuo Zeng from University of Arizona for his help in running the computerized round robin competitions and to the 10 participants who submitted entries to the competition.

Moshe Dror Moshe Dror received an M.Sc. degree in Mathematical Methods and an I.E. degree – both from Columbia University. He also received a Ph.D. in Management Science from University of Maryland. He has conducted well-documented research on a variety of topics in logistics, scheduling, manufacturing, cooperative game theory/cost allocation, and on applications of agent theory in project management. He has published numerous book chapters and over 100 papers in the primary refereed journals which focus on management of operations – Management Science, Operations Research, Naval Research Logistics, IIE Transactions, Transportation Science, among others. He has also published in a number of mathematical and computer science journals such as SIAM J. on Discrete Mathematics, J. of Complexity, ORDER, IPL, Discrete Mathematics, Discrete Applied Mathematics, Discrete Optimization, International Journal of Computational Geometry & Applications, J. of Applied Probability, in addition to publications in a major economics journals I.J. of Game Theory and Games & Economic Behavior. He has edited a book on ARC ROUTING: Theory, Solutions, and Applications (2000), and co-edited a book MODELING UNCERTAINTY: An Examination of Stochastic Theory, Methods, and Applications (2002). He has also edited 3 special issues of journals (Operations Research, EJOR, and IIE Transactions).

Graham Kendall Graham Kendall (M03SM10) received the B.S. degree in computation (first class, honors) from the Institute of Science and Technology, University of Manchester, Manchester, U.K., in 1997 and the Ph.D. degree in computer science from the University of Nottingham, Nottingham, U.K., in 2001. His previous experience includes almost 20 years in the information technology industry where he held both technical and managerial positions. He is a Professor of Computer Science at the University of Nottingham and is currently based at their Malaysia Campus where he holds the position of Vice-Provost (Research and Knowledge Transfer). He is a Director of two companies (EventMAP Ltd., Nottingham, U.K.; Aptia Solutions Ltd., Nottingham, U.K.) and CEO of two companies (MyRIAD Solutions Sdn Bhd, Malaysia and MyResearch Sdn Bhd, Malaysia). He is a Fellow of the Operational Research Society. He is an Associate Editor of nine international journals, including two IEEE journals: the IEEE TRANSACTIONS ON EVOLUTIONARY COMPUTATION and the IEEE TRANSACTIONS ON COMPUTATIONAL INTELLIGENCE AND AI IN GAMES. He chaired the Multidisciplinary International Conference on Scheduling: Theory and Applications in 2003, 2005, 2007, 2009, and 2011, and has chaired several other international conferences, which has included establishing the IEEE Symposium on Computational Intelligence and Games. He has been awarded externally funded grants worth over 6 million from a variety of sources including Engineering and Physical Sciences Research Council (EPSRC) and commercial organizations.

Amnon Rapoport Amnon Rapoport Amnon Rapoport is a Distinguished Professor of Management at the A. Gary Anderson Graduate School of Management, University of California, Riverside, and an Eller Professor of Management at the University of Arizona (retired). The focus of his research has been interactive decision making. He received his Ph.D. degree in Quantitative Psychology from the University of North Carolina at Chapel Hill. He is the author or editor of nine books and more than two hundred refereed journal papers and book chapters. His published articles have appeared in such journals as Science, Management Science, Marketing Science, American Political Science Review, American Economic Review, Organizational Behavior and Human Decision Processes, Games and Economic Behavior, Journal of Mathematical Psychology, Experimental Economics, Manufacturing & Service Operation Management, Psychological Review, Transportation Research B, and the Journal of Economic Behavior and Organization. He has served as associate editor or member of the editorial board of the Journal of Mathematical Psychology, Experimental Economics, Organizational Behavior and Human Decision Processes, Decision, and Games and Economic Behavior. He has received numerous grants from such agencies as the National Science Foundation, the Hong Kong Grants Research Grants Council, United States - Israel Binational Science Foundation, and the United States Air Force. His current research focuses on dynamic pricing and on choice of routes in congestible networks.

TABLE II

PERCENTAGE IN WINNING POINT AND ROUND DIFFERENTIALS; NO CARRYOVER

	No Carryover, 50 Rounds					
	Obj: Maximize Total Points			Obj: Maximize Total Rounds		
Players	% in W. P. Diff.	% in W. R. Diff.	# of T.	% in W. P. Diff.	% in W. R. Diff.	# of T.
0209	4.0	18.6	1	9.2	32.0	0
0205	21.0	61.6	0	15.8	59.0	0
0203	5.4	16.2	0	5.5	20.8	0
0210	3.2	12.8	1	3.8	13.8	0
0207	6.0	37.6	0	5.6	31.4	0
0206	6.4	21.2	1	5.5	26.4	0
0202	5.2	17.6	1	5.0	19.2	0
0204	6.4	20.0	0	2.8	10.6	0
0208	7.0	29.2	0	7.8	28.0	0
0201	11.2	47.0	0	11.8	45.6	0
Random	0.01	0	0	0	0.01	0

TABLE III
PERCENTAGE IN WINNING POINT AND ROUND DIFFERENTIALS; CARRYOVER

With Carryover, 50 Rounds						
	Obj: Maximize Total Points			Obj: Maximize Total Rounds		
Players	% in W. P. Diff.	% in W. R. Diff.	# of T.	% in W. P. Diff.	% in W. R. Diff.	# of T.
0209	8.9	28.8	0	8.3	28.8	0
0205	10.4	32.8	0	9.0	33.6	0
0203	4.3	12.8	0	4.4	15.6	0
0210	5.1	15.6	1	5.3	12.8	0
0207	4.0	22.4	0	4.1	22.4	0
0206	4.4	9.6	0	4.1	16.0	0
0202	7.4	23.8	1	4.7	15.0	0
0204	4.0	12.4	0	3.6	12.8	0
0208	9.5	28.2	1	13.3	39.6.0	0
0201	13.2	46.4	0	13.3	42.8	0
Random	0	1.8	1	0.25	0.04	0

TABLE IV
NUMBER OF OPPONENTS DEFEATED BY POINTS AND ROUNDS WITH ROUND OBJ.: NO CARRYOVER

Max # of W. Rounds, 50 Rounds, No Carryover					
Players	# of O. Def. by Points		# of O. Def. by Rounds		
	Wins	Loses	Wins	Ties	Loses
0209	5	5	5	0	5
0205	10	0	9	0	1
0203	4	6	4	0	6
0210	4	6	4	0	6
0207	6	4	7	0	3
0206	5	5	8	0	2
0202	4	6	4	0	6
0204	1	9	2	0	8
0208	6	4	5	0	5
0201	8	2	8	0	2
Random	1	9	1	0	9

TABLE V
NUMBER OF OPPONENTS DEFEATED BY POINTS AND ROUNDS WITH POINTS OBJ.; NO CARRYOVER

Max Total Points, 50 Rounds, No Carryover					
Players	# of O. Def. by Points		# of O. Def. by Rounds		
	Wins	Loses	Wins	Ties	Loses
0209	2	8	2	1	7
0205	10	0	10	0	0
0203	6	4	5	0	5
0210	4	6	4	1	5
0207	5	5	6	0	4
0206	6	4	5	1	4
0202	4	6	3	1	6
0204	3	7	3	0	7
0208	6	4	5	0	5
0201	8	2	8	0	2
Random	1	9	1	0	9

TABLE VI
NUMBER OF OPPONENTS DEFEATED BY POINTS AND ROUNDS WITH POINTS OBJ.; CARRYOVER

Max Total Points, 50 Rounds, Carryover					
Players	# of O. Def. by Points		# of O. Def. by Rounds		
	Wins	Loses	Wins	Ties	Loses
0209	5	5	5	0	5
0205	10	0	10	0	0
0203	5	5	4	0	6
0210	5	5	5	1	4
0207	4	6	3	0	7
0206	3	7	4	0	6
0202	6	4	5	1	4
0204	2	8	2	0	8
0208	7	3	7	1	2
0201	8	2	7	1	2
Random	1	9	1	0	9

TABLE VII
NUMBER OF OPPONENTS DEFEATED BY POINTS AND ROUNDS WITH ROUND OBJ.; CARRYOVER

Max # of W. Rounds, 50 Rounds, Carryover					
Players	# of O. Def. by Points		# of O. Def. by Rounds		
	Wins	Loses	Wins	Ties	Loses
0209	5	5	5	0	5
0205	9	1	8	0	2
0203	4	6	4	0	6
0210	6	4	6	0	4
0207	3	7	3	0	7
0206	5	5	5	0	5
0202	5	5	5	0	5
0204	2	8	2	0	8
0208	9	1	9	0	1
0201	7	3	7	0	3
Random	1	9	1	0	9

TABLE VIII
FREQUENCY OF DEFEATED PROGRAMS

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
0201	9	7	7	7	**9**	**7**	8	7	8	8	8	8	8	7	7	7
0202	4	3	6	5	5	5	5	4	4	4	4	3	6	5	5	5
0203	5	5	5	5	4	4	3	3	4	4	6	5	5	4	4	4
0204	2	2	2	2	2	1	2	2	1	2	3	3	2	2	2	2
0205	**10**	**9**	**10**	**9**	8	5	**9**	7	**10**	**9**	**10**	**10**	**10**	**10**	**9**	8
0206	5	4	4	3	4	3	4	4	5	8	6	5	3	4	5	5
0207	5	6	6	6	5	4	3	3	6	7	5	6	4	3	3	3
0208	7	6	6	4	6	6	**9**	**8**	6	5	6	5	7	7	**9**	**9**
0209	7	5	3	3	5	5	5	5	5	5	2	2	5	5	5	5
0210	4	4	5	4	5	4	5	5	4	4	4	4	5	5	6	6
Rand.	0	0	1	1	1	1	2	1	1	1	1	1	1	1	1	1