Properties of Winning Iterated Prisoner's Dilemma Strategies.

Nikoleta E. Glynatsi, Vincent A. Knight, Marc Harper

Abstract

Researchers have explored the performance of Iterated Prisoner's Dilemma strategies for decades: from the celebrated performance of Tit for Tat, to the introduction of the zero-determinant strategies, to the use of sophisticated learning structures such as neural networks, many new strategies have been introduced and tested in a variety of tournaments and population dynamics. Typical results in the literature, however, rely on performance against a small number of somewhat arbitrarily selected strategies in a small number of tournaments, casting doubt on the generalizability of conclusions. We analyze a large collection of 195 strategies in 45606 tournaments, present the top performing strategies across multiple tournament types, and distill their salient features. The results show that there is not yet a single strategy that performs well in diverse Iterated Prisoner's Dilemma scenarios, nevertheless there are several properties that heavily influence the best performing strategies. This refines the properties described by R. Axelrod in light of recent and more diverse opponent populations to: be nice, be provocable and generous, be a little envious, be clever, and adapt to the environment. More precisely, we find that strategies perform best when their probability of cooperation matches the total tournament population's aggregate cooperation probabilities, or a proportion thereof in the case of noisy and probabilistically ending tournaments, and that the manner in which a strategy achieves the ideal cooperation rate is crucial. The features of high performing strategies help cast some light on why strategies such as Tit For Tat performed historically well in tournaments and why zero-determinant strategies typically do not fare well in tournament settings.

1 Background

The Iterated Prisoner's Dilemma (IPD) is a repeated two player game that models behavioural interactions, specifically interactions where self-interest clashes with collective interest. At each turn of the game both players, simultaneously and independently, decide between cooperation (C) and defection (D), given memory of all prior interactions. The payoffs for each player, at each turn, is influenced by their own choice and the choice of the other player. The payoffs of the game are defined by:

	Cooperate (C)	Defect (D)
Cooperate (C)	R	S
Defect (D)	T	P

where typically $T > R > P > S$ and $2R > T + S$. The most common values used in the literature [9] are $R = 3, P = 1, T = 5, S = 0$. These values are also used in this work.

Conceptualising strategies and understanding the best way of playing the game has been of interest to the scientific community since the formulation of the game in 1950 [13]. Following the computer tournaments of

R. Axelrod in the 1980's [6, 7], a strategy's performance in a round robin computer tournament became a common evaluation technique for newly designed strategies. Many tournaments have followed Axelrod's [10, 11, 17, 19, 33, 34] and now the literature and various codebases contain hundreds of strategies.

The winner of both of R. Axelrod's tournaments [6, 7] was the simple strategy Tit For Tat (TFT) which cooperates on the first turn and thereafter copies the previous action of its opponent, retailiating against defections with a defection, and forgiving a defection if followed by a cooperation. R. Axelrod concluded that the strategy's robustness was due to four properties, which he adapted into four suggestions on doing well in an IPD:

- Do not be envious by striving for a payoff larger than the opponent's payoff
- Be "nice"; Do not be the first to defect
- Reciprocate both cooperation and defection; Be provocable to retaliation and forgiveness
- Do not be too clever by scheming to exploit the opponent

Forgiveness is a strategy's ability to go from a DC to C aiming to achieve mutual cooperation again, the only way TFT would end up in DC, in environments without noise, is if it had received a defection and then retaliated. Subsequently, TFT would forgive an opponent that apologises (in a DC round) by returning to cooperation, since mutual cooperation is better than mutual defection. As a result of the strategy's strong performance in both tournaments, and moreover in a series of evolutionary experiments [9], TFT was often claimed to be a highly robust (and sometimes the most robust) strategy for the IPD.

There are strategies which have built upon TFT and the reciprocity based approach. In [10] a strategy called Gradual was introduced, constructed to have the same qualities as those of TFT with one addition. Gradual has a memory of the previous rounds of play of the game, recording the number of defections by the opponent and punishing them with a growing number of defections. It then enters a calming state in which it cooperates for two rounds. A strategy with the same intuition as Gradual is Adaptive Tit for Tat [35]. Adaptive Tit for Tat maintains a continually updated estimate of the opponent's behaviour and uses this estimate to condition its future actions.

Other work has built upon the limitations of TFT, and in some cases have shown that suggestions made by R. Axelrod did not necessarily apply in alternative environmental settings. In [11, 12, 25, 32] it was shown that TFT suffered in environments with noise. This was mainly due to the strategy being too provocable and its lack of generosity and contrition. Since TFT immediately punishes a defection, in a noisy environment it can get stuck in a repeated cycle of defections and cooperations. Some new strategies, more robust in tournaments with noise, were soon introduced, including Nice and Forgiving [11], Generous Tit For Tat [27], and Pavlov (aka Win Stay Lose Shift) [26], as well as later variants such as OmegaTFT [20]. The introduction of a new strategy is often accompanied by a claim that the new strategy is the best known, despite only being tested against a small number of opponents or against specific classes of opponents not necessarily representative of all possible or all published strategies. The lack of testing against formally defined strategies and tournament winners is understandable given the effort required to implement the hundreds of published IPD strategies. Implementing prior strategies faithfully is often extremely difficult or impossible due to insufficient descriptions and lack of published implementations or code. This calls into question any claims of superiority or robustness of newly introduced strategies.

A set of envious IPD strategies were introduced called zero-determinant strategies (ZDs) in [30]. These strategies attempt to force a linear relationship between stationary payoffs against other memory-one opponents, potentially ensuring that they receive a higher average payout. While ZDs were introduced with a small tournament in which some were reportedly successful [34], this result has not generally held in future work. In [17] a series of strategies trained using reinforcement learning were introduced, and a tournament containing over 200 strategies featured no ZDs ranking in top spots. Instead, the top ranked strategies were a set of "clever" (in the sense of R. Axelrod's characteristics) trained strategies based on lookup tables [8], hidden Markov models [17], and finite state automata [24]. Similarly, in [23], a set of evolutionarily-trained strategies, and a pre-selected set of known strategies, outperformed a collection of 12 ZDs.

Though only select pieces of work have been discussed, there is a broad collection of strategies in the literature, and new strategies and competitions are published frequently [15]. The question, however, still remains the same: what is the best way to play the game?

Compared to other works, where typically a few selected or introduced strategies are evaluated on a small number of tournaments and/or small number of opponents, this manuscript evaluates the performance of 195 strategies in 45606 tournaments. Furthermore, a large portion of the strategies used in this manuscript are drawn from known and named strategies in IPD literature, including many previous tournament winners, in contrast to other work that may have randomly generated many essentially arbitrary strategies (typically restrained to a class such as memory-one strategies, or those of a certain structural form such as finite state machines or deterministic memory-two strategies). Additionally, our tournaments come in a number of variations including standard tournaments emulating R. Axelrod's original tournaments, tournaments with noise, probabilistic match length, and both noise and probabilistic match length. This diversity of strategies and tournament types yields new insights and tests earlier claims in alternative settings against known powerful strategies.

The later part of the paper evaluates the impact of various strategy features on the performance of the strategies using standard statistical and machine learning techniques. These features include measures regarding a strategy's behaviour and measures regarding the tournaments. This rigorous analysis reinforces the discussion started by R. Axelrod and concludes that the properties of a successful strategy in the IPD are:

- ~~Do not be envious~~ Be a little bit envious
- Be "nice" in non-noisy environments or when game lengths are longer
- Reciprocate both cooperation and defection appropriately; Be provocable in tournaments with short matches, and generous when matches are longer
- ~~Do not be too clever~~ It's ok to be clever
- Adapt to the environment; Adjust to the mean population cooperation

The different tournament types as well as the data collection, made possible due to an open source library called Axelrod-Python (APL), are covered in Section 2. The raw and processed data sets have been made publicly available [14, 16] and can be used for further analysis and insights. Section 3, focuses on the best performing strategies for each type of tournament and overall. Section 4, explores the traits which contribute to good performance, and finally the results are summarised in Section 5. This manuscripts uses several parameters, introduced in the following sections. The full set of parameters and their definitions are given in Appendix **??**.

2 Data collection

The data set generated for this manuscript was created using `Axelrod-Python` version 3.0.0. `Axelrod-Python` enables the simulation of Iterated Prisoner's Dilemma computer tournaments and contains an extensive list of strategies. Most of these strategies are described in the literature, with a few exceptions contributed specifically to the package. In this paper, we use a total of 195 strategies. You can find a list of these strategies in the Supplementary Material.

The package supports several tournament types, and this work considers standard, noisy, probabilistic ending, and noisy probabilistic ending tournaments. *Standard tournaments* are similar to Axelrod's well-known tournaments [6]. In these tournaments, there are N strategies, and each strategy plays an iterated game with n turns against all other strategies, not including self-interactions. *Noisy tournaments* also involve N strategies and n turns, but in each turn, there is a probability p_n that a player's action is flipped. *Probabilistic ending tournaments* consist of N strategies, and after each turn, a match between strategies ends with a given probability p_e. Finally, *noisy probabilistic ending tournaments* incorporate both a noise probability p_n and an ending probability p_e. For smoother results, each tournament is repeated k times, and this repetition factor was allowed to vary to assess the impact of smoothing. The winner of each tournament is determined based on the average score achieved by a strategy from the entire set of repetitions, not by the number of wins. The process of collecting tournament results is outlined in Algorithm 1. For each trial, we choose a random size N is selected, and a random list of N strategies from the 195 available. Subsequently, standard, noisy, probabilistic ending, and noisy probabilistic ending tournaments are conducted for the selected list of strategies. The parameters for the tournaments, as well as the number of repetitions, are chosen once for each trial.

Algorithm 1: Tournament Data Collection Algorithm

for *seed* $\in [0, 11420]$ **do**

- $N \leftarrow$ randomly select integer $\in [3, 195]$;
- players $\leftarrow$ randomly select N players;
- $k \leftarrow$ randomly select integer $\in [10, 100]$;
- $n \leftarrow$ randomly select integer $\in [1, 200]$;
- $p_n \leftarrow$ randomly select float $\in [0, 1]$;
- $p_e \leftarrow$ randomly select float $\in [0, 1]$;
- result standard $\leftarrow$ Axelrod.tournament(players, n, k);
- result noisy $\leftarrow$ Axelrod.tournament(players, n, p_n, k);
- result probabilistic ending $\leftarrow$ Axelrod.tournament(players, p_e, k);
- result noisy probabilistic ending $\leftarrow$ Axelrod.tournament(players, p_n, p_e, k);

return *result standard, result noisy, result probabilistic ending, result noisy probabilistic ending*;

We have run a total of 11400 trials of Algorithm 1. For each trial, we collect the results for four different tournaments, resulting in a total of 45606 (11400×4) tournament results. Each tournament outputs a result summary in the form of Table 1. Each strategy has participated, on average, in 5154 tournaments of each type. The strategy with the maximum participation in each tournament type is Inverse Punisher with 5639 entries. The strategy with the minimum entries is EvolvedLookerUp 1 1 1, which was selected in 4693 trials. During the data collection process, we allowed the probabilities of noise and tournament ending to vary between 0 and 1. However, commonly used values for these probabilities are $p_n < 0.5$ and $p_e < 0.1$ (for more details, see the Supplementary Material). Therefore, in this context, we will focus on these subsets for the three tournament types: noisy, probabilistic ending, and noisy probabilistic ending. The results presented here pertain to these subsets. Specifically, there are a total of 5650, 1134, and 565 runs for each of the three tournament types, respectively. We also provide an analysis of the paper considering the entire datasets, and these results are presented in the Supplementary Material.

						Rates							
Rank	Name	Median score	Cooperation rating (C_r)	Win	Initial C	CC	CD	DC	DD	CC to C	CD to C	DC to C	DD to C
0	EvolvedLookerUp2 2 2	2.97	0.705	28.0	1.0	0.639	0.066	0.189	0.106	0.836	0.481	0.568	0.8
1	Evolved FSM 16 Noise 05	2.875	0.697	21.0	1.0	0.676	0.020	0.135	0.168	0.985	0.571	0.392	0.07
2	PSO Gambler 1 1 1	2.874	0.684	23.0	1.0	0.651	0.034	0.152	0.164	1.000	0.283	0.000	0.136
3	PSO Gambler Mem1	2.861	0.706	23.0	1.0	0.663	0.042	0.145	0.150	1.000	0.510	0.000	0.122
4	Winner12	2.835	0.682	20.0	1.0	0.651	0.031	0.141	0.177	1.000	0.441	0.000	0.462
...	...	...	...	...	...	...	...	...	...	...	...	...	...

Table 1: **Result Summary Example of a Tournament.** A result summary consists of N rows, with each row containing information for each strategy that participated in the tournament. This information includes the strategy's rank (R), median score, the cooperation rate (C_r), the number of match wins, and the probability that the strategy cooperated in the opening move. Additionally, it provides the probabilities of a strategy being in any of the four states (CC, CD, DC, DD) and the cooperation rate after each state.

3 Top ranked strategies

A strategy has participated in multiple tournaments of each type, and to evaluate its overall performance, we introduce a measure called the *n*ormalized rank. In each tournament, the strategies receive a rank, where 0 denotes that the strategy was the winner, and $N-1$ indicates that the strategy came last in the tournament. The normalized rank, denoted as r, is calculated as $r = \frac{R}{N-1}$. Thus, the rank a strategy achieved over the number of players in the tournament. The performance of the strategies is assessed based on the *median of the normalized rank*, denoted as $\bar{r}$.

For example, let's consider the well-known strategies Tit For Tat and Gradual. Each strategy participated in several tournaments of each type (see Figure 1). We show the distribution of the ranks of these strategies in each of the four tournaments. We can observe that in tournaments with the presence of noise, Tit For Tat has a normally distributed normalized rank around 1/2. In tournaments without noise, the strategy performs better, achieving its best performance in probabilistic ending tournaments with a median normalized rank of 0.298. In comparison, Gradual's performance has longer tails, indicating that there were tournaments where the strategy performed very well or very poorly. Overall, Gradual achieves a lower median rank, signifying that it performs better than Tit For Tat except in the case of noisy and probabilistic ending tournaments.

The top 15 strategies for each tournament type, based on $\bar{r}$, are presented in Table 2, while the r distributions for the top-ranked strategies can be found in Figure 2. In standard tournaments, 10 out of the 15 top strategies were introduced in [17]. These strategies are based on finite state automata (FSM), hidden Markov models (HMM), artificial neural networks (ANN), lookup tables (LookerUp), and stochastic lookup tables (Gambler). They have been trained using reinforcement learning algorithms (evolutionary and particle swarm algorithms) to perform well against a subset of the strategies in `Axelrod-Python` in a standard tournament. Thus, their performance in the specific setting was anticipated, although still noteworthy given the random sampling of tournament participants. DoubleCrosser and BackStabber, both from the `Axelrod-Python` , use the number of turns and are set to defect in the last two rounds. These strategies can be characterized as "cheaters" because their source code allows them to know the number of turns (unless the match has a probabilistic ending). These strategies were expected to not perform as well in tournaments where the number of turns is not specified. Finally, Winner 12 [23] and DBS [5] are both from the literature. DBS is a strategy specifically designed for noisy environments; however, it ranks highly in standard tournaments as well. Similarly, the fourth-ranked player, Evolved FSM 16 Noise 05, was trained for noisy tournaments yet performs well in standard tournaments.

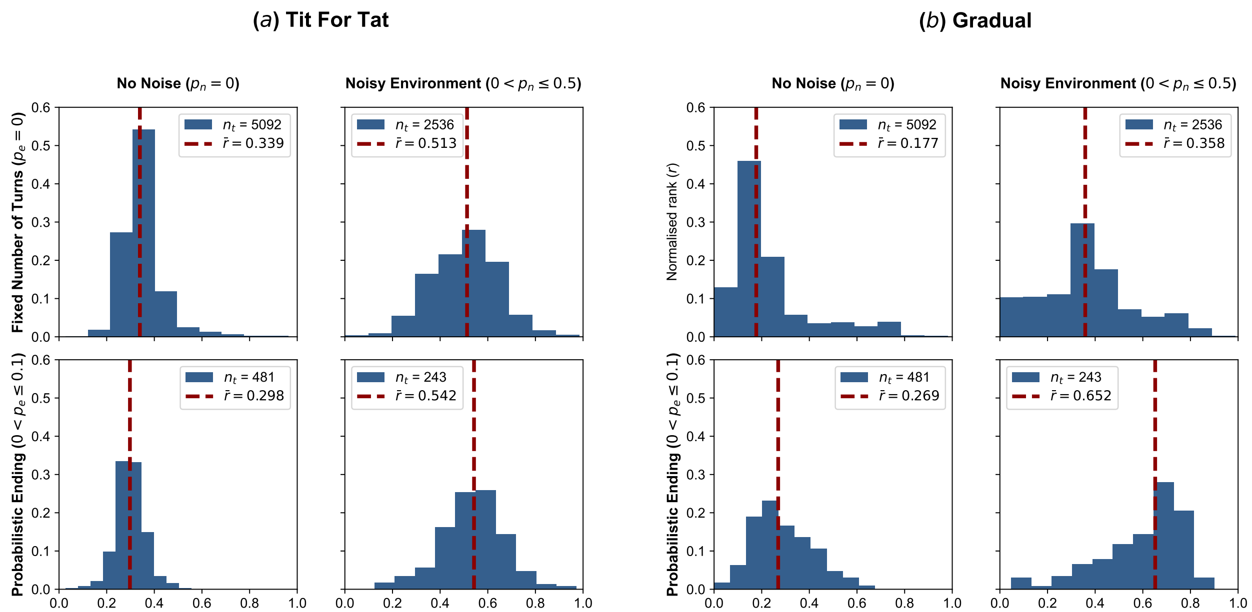


Figure 1: **Examples of normalized rank distributions for two strategies, Tit For Tat and Gradual.** We plot the distributions of r for the two strategies in the four tournament types. As a reminder, lower values of r correspond to better performances. In each plot, we also show the number of data points. Both strategies participated in a similar number of tournaments. Based on the median rank, which we use in this work to define overall performance, Tit For Tat performs best in probabilistic ending tournaments, whereas Gradual was in standard tournaments. The best performance of the Gradual strategy has been in standard tournaments where it achieved a $\bar{r}$ of 0.34.

In the case of noisy tournaments, the top-performing strategies include strategies specifically designed for noisy tournaments. These are DBS, Evolved FSM 16 Noise 05, Evolved ANN 5 Noise 05, PSO Gambler 2 2 2 Noise 05, and Omega Tit For Tat [20]. Omega TFT, a strategy designed to break the deadlocking cycles of CD and DC that TFT can fall into in noisy environments, places 10th. The rest of the top ranks are occupied by strategies that performed well in standard tournaments and deterministic strategies such as Spiteful Tit For Tat [1], Level Punisher [2], Eugine Nier [29].

The most effective strategies in probabilistic ending tournaments with $p_e < 0.1$ are a series of ensemble Meta strategies, trained strategies which performed well in standard tournaments, and Grudger [31] and Spiteful Tit for Tat [1]. The Meta strategies [31] utilize a team of strategies and aggregate the potential actions of the team members into a single action in various ways.

Overall, the analysis reveals that dominating strategies in standard tournaments were those trained using reinforcement learning techniques. In standard tournaments, these dominating strategies exhibited a clear trend of being trained through reinforcement learning techniques. Additionally, in environments with a noise probability strictly less than 0.1, successful strategies were purposefully designed or trained to adapt to noisy conditions. Furthermore, in tournaments with probabilistic endings, the highly ranked strategies leaned towards defecting strategies and trained finite state automata, as demonstrated by the works of Ashlock et al. [3, 4]. Their prominence in tournaments where the probability of the game ending after each turn exceeded 0.1 underscored their effectiveness. Similarly, in probabilistic tournaments with p_e less than 0.1, the highly ranked strategies were characterized by their reliance on the behavior of others. Notably, from the comprehensive set of strategies considered, no strategy demonstrated consistent success in noisy environments unless the noise value was limited to less than 0.1.

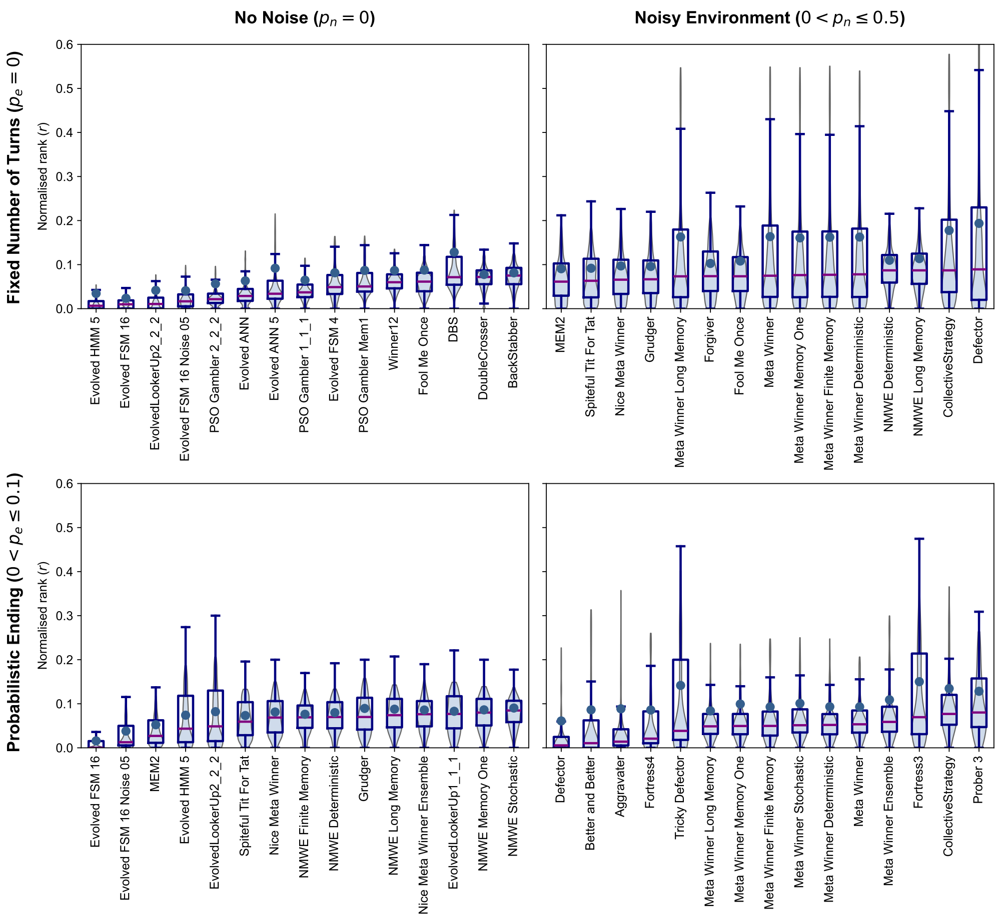


Figure 2: r distributions of the top 15 strategies in different environments. A lower value of $\bar{r}$ corresponds to a more successful performance. A strategy's r distribution skewed towards zero indicates that the strategy ranked highly in most tournaments it participated in. Most distributions are skewed towards zero except the distributions with unrestricted noise, supporting the conclusions from Table 2.

Though there is not a single strategy that consistently outranks all others in any of the distinct tournament types or even across the tournament types, there are specific types of strategies that have been repeatedly ranked in the top ranks. These include strategies that have been trained, strategies that retaliate, and strategies that adapt their behavior based on preassigned rules to achieve the highest outcome. These results contradict some of Axelrod's suggestions, and more specifically, the suggestions "Do not be clever" and "Do not be envious". We delve deeper into the crucial strategy features for success in the following section.

4 Evaluation of performance

For each strategy, for each tournament we have a variety of features, described in Table 3. These features are measures regarding a strategy's behaviour from the tournaments the strategies competed in as well as intrinsic properties such as whether a strategy is deterministic or stochastic.

The memory usage of strategies is the number of rounds of play used by the strategy divided by the number

	Standard		Noisy ($p_n < 0.5$)		Probabilistic ending ($p_e < 0.1$)		Noisy probabilistic ending	
	Name	$\bar{r}$	Name	$\bar{r}$	Name	$\bar{r}$	Name	$\bar{r}$
0	Evolved HMM 5	0.007	MEM2	0.061	Evolved FSM 16	0.0	Defector	0.006
1	Evolved FSM 16	0.01	Spiteful Tit For Tat	0.063	Evolved FSM 16 Noise 05	0.013	Better and Better	0.011
2	EvolvedLookerUp2_2_2	0.011	Nice Meta Winner	0.066	MEM2	0.027	Aggravater	0.014
3	Evolved FSM 16 Noise 05	0.017	Grudger	0.067	Evolved HMM 5	0.043	Fortress4	0.021
4	PSO Gambler 2_2_2	0.022	Meta Winner Long Memory	0.073	EvolvedLookerUp2_2_2	0.049	Tricky Defector	0.038
5	Evolved ANN	0.029	Forgiver	0.074	Spiteful Tit For Tat	0.059	Meta Winner Long Memory	0.049
6	Evolved ANN 5	0.034	Fool Me Once	0.074	Nice Meta Winner	0.069	Meta Winner Memory One	0.05
7	PSO Gambler 1_1_1	0.037	Meta Winner	0.075	NMWE Finite Memory	0.069	Meta Winner Finite Memory	0.05
8	Evolved FSM 4	0.049	Meta Winner Memory One	0.076	NMWE Deterministic	0.07	Meta Winner Stochastic	0.051
9	PSO Gambler Mem1	0.05	Meta Winner Finite Memory	0.077	Grudger	0.07	Meta Winner Deterministic	0.052
10	Winner12	0.06	Meta Winner Deterministic	0.078	NMWE Long Memory	0.074	Meta Winner	0.053
11	Fool Me Once	0.061	NMWE Deterministic	0.087	Nice Meta Winner Ensemble	0.076	Meta Winner Ensemble	0.059
12	DBS	0.071	NMWE Long Memory	0.087	EvolvedLookerUp1_1_1	0.077	Fortress3	0.07
13	DoubleCrosser	0.072	CollectiveStrategy	0.087	NMWE Memory One	0.08	CollectiveStrategy	0.077
14	BackStabber	0.075	Defector	0.089	NMWE Stochastic	0.085	Prober 3	0.08

Table 2: Top performances for each tournament type based on $\bar{r}$. The results of each type are based on 11420 unique tournaments. The results for noisy tournaments with $p_n < 0.1$ are based on 1151 tournaments, and for probabilistic ending tournaments with $p_e < 0.1$ on 1139. The top ranks indicate that trained strategies perform well in a variety of environments, but so do simple deterministic strategies. The normalised medians are close to 0 for most environments, except environments with noise not restricted to 0.1 regardless of the number of turns. Noisy and noisy probabilistic ending tournaments have the highest medians.

feature	feature explanation	source	value type	min value	max value
stochastic	If a strategy is stochastic	strategy classifier from APL	boolean	Na	Na
makes use of game	If a strategy makes used of the game information	strategy classifier from APL	boolean	Na	Na
makes use of length	If a strategy makes used of the number of turns	strategy classifier from APL	boolean	Na	Na
memory usage	The memory size of a strategy divided by the number of turns	memory size from APL	float	0	1
SSE	A measure of how far a strategy is from ZD behaviour	method described in [21]	float	0	1
max cooperating rate ($C_{\max}$)	The biggest cooperating rate in a given tournament	result summary	float	0	1
min cooperating rate ($C_{\min}$)	The smallest cooperating rate in a given tournament	result summary	float	0	1
median cooperating rate (C_{median})	The median cooperating rate in a given tournament	result summary	float	0	1
mean cooperating rate (C_{mean})	The mean cooperating rate in a given tournament	result summary	float	0	1
C_r / $C_{\max}$	A strategy's cooperating rate divided by the maximum	result summary	float	0	1
$C_{\min}$ / C_r	A strategy's cooperating rate divided by the minimum	result summary	float	0	1
C_r / C_{median}	A strategy's cooperating rate divided by the median	result summary	float	0	1
C_r / C_{mean}	A strategy's cooperating rate divided by the mean	result summary	float	0	1
C_r	The cooperating ratio of a strategy	result summary	float	0	1
CC to C rate	The probability a strategy will cooperate after a mutual cooperation	result summary	float	0	1
CD to C rate	The probability a strategy will cooperate after being betrayed by the opponent	result summary	float	0	1
DC to C rate	The probability a strategy will cooperate after betraying the opponent	result summary	float	0	1
DD to C rate	The probability a strategy will cooperate after a mutual defection	result summary	float	0	1
p_n	The probability of a player's action being flip at each interaction	trial summary	float	0	1
n	The number of turns	trial summary	integer	1	200
p_e	The probability of a match ending in the next turn	trial summary	float	0	1
N	The number of strategies in the tournament	trial summary	integer	3	195
k	The number of repetitions of a given tournament	trial summary	integer	10	100

Table 3: The features which are included in the performance evaluation analysis. Stochastic, makes use of length and makes use of game are APL classifiers that determine whether a strategy is stochastic or deterministic, whether it makes use of the number of turns or the game's payoffs. The memory usage is calculated as the number of turns the strategy considers to make an action (which is specified in the APL) divided by the number of turns. The SSE (introduced in [21]) shows how close a strategy is to behaving as a ZDs, and subsequently, in an extortionate way. The method identifies the ZDs closest to a given strategy and calculates the algebraic distance between them as the sum of squared error (SSE). A SSE value of 1 indicates no extortionate behaviour at all whereas a value of 0 indicates that a strategy is behaving as a ZDs. The rest of the features considered are the CC to C, CD to C, DC to C, and DD to C rates as well as cooperating ratio of a strategy, the minimum (C_{min}), maximum (C_{max}), mean (C_{mean}) and median (C_{median}) cooperating ratios of each tournament.

of turns in each match. For example, Winner12 uses the previous two rounds of play, and if participating in a match with 100 turns its memory usage would be 2/100. For strategies with an infinite memory size, for example Evolved FSM 16 Noise 05, memory usage is equal to 1. Note that for tournaments with a probabilistic ending the number of turns was not collected, so the memory usage feature is not used for probabilistic ending tournaments.

The correlation coefficients between the features of Table 3 the median score and the median normalised rank are given by Table 4. The correlation coefficients between all features of Table 3 have been calculated and a graphical representation can be found in the Appendix **??**.

	Standard		Noisy		Probabilistic ending		Noisy probabilistic ending		Overall	
	r	median score	r	median score	r	median score	r	median score	r	median score
CC to C rate	-0.501	0.501	0.414	-0.504	0.408	-0.323	0.260	0.022	0.108	0.081
CD to C rate	0.226	-0.199	0.456	-0.330	0.320	-0.017	0.205	-0.220	0.281	-0.177
C_r	-0.323	0.384	0.711	-0.678	0.714	-0.832	0.579	-0.135	0.360	-0.124
C_r / C_{max}	-0.323	0.381	0.616	-0.551	0.714	-0.833	0.536	-0.116	0.395	-0.265
C_r / C_{mean}	-0.331	0.358	0.731	-0.740	0.721	-0.861	0.649	-0.621	0.428	-0.439
C_r / C_{median}	-0.331	0.353	0.652	-0.669	0.712	-0.852	0.330	-0.466	0.294	-0.405
C_r / C_{min}	0.109	-0.080	-0.358	0.250	-0.134	0.150	-0.368	0.113	0.000	0.280
C_{max}	-0.000	0.049	0.000	0.023	-0.000	0.046	0.000	-0.004	-0.000	0.553
C_{mean}	-0.000	0.229	-0.000	0.271	0.000	0.200	0.000	0.690	-0.000	0.544
C_{median}	0.000	0.209	-0.000	0.240	-0.000	0.187	-0.000	0.673	0.000	-0.250
C_{min}	0.000	0.084	0.000	-0.017	-0.000	0.007	-0.000	0.041	-0.161	-0.190
DC to C rate	0.127	-0.100	0.509	-0.504	-0.018	0.033	0.341	-0.016	0.173	-0.088
DD to C rate	0.412	-0.396	0.533	-0.436	-0.103	0.176	0.378	-0.263	0.237	-0.239
N	0.000	-0.009	-0.000	0.002	-0.000	0.003	-0.000	0.001	-0.000	-0.001
k	0.000	-0.002	-0.000	0.003	-0.000	0.001	-0.000	-0.008	0.000	-0.001
n	0.000	-0.125	-0.000	-0.024	-	-	-	-	0.000	-0.074
p_e	-	-	-	-	0.000	0.165	0.000	-0.058	0.000	0.055
p_n	-	-	-0.000	0.207	-	-	-0.000	-0.650	-0.000	-0.256
Make use of game	-0.003	-0.022	0.025	-0.082	-0.053	-0.108	0.013	-0.016	-0.004	-0.053
Make use of length	-0.158	0.124	0.005	-0.123	-0.025	-0.090	0.014	-0.016	-0.041	-0.026
SSE	0.473	-0.452	0.463	-0.337	-0.156	0.223	0.305	-0.259	0.233	-0.167
memory usage	-0.082	0.095	-0.007	-0.017	-	-	-	-	-0.053	0.046
stochastic	0.006	-0.024	0.022	-0.026	0.002	-0.130	0.021	-0.013	0.013	-0.048

Table 4: Correlations between the features of Table 3 and the normalised rank and the median score.

In standard tournaments the features CC to C, C_r, C_r/C_{max} and the cooperating ratio compared to C_{median} and C_{mean} have a moderately negative effect on the normalised rank (smaller rank is better), and a moderate positive on the median score. The SSE error and the DD to C rate have the opposite effects. Thus, in standard tournaments behaving cooperatively corresponds to a more successful performance. Even though being nice generally pays off that does not hold against defective strategies. Being more cooperative after a mutual defection, that is not retaliating, is associated to lesser overall success in terms of normalised rank. Figure 3 confirms that the winners of standard tournaments always cooperate after a mutual cooperation and almost always defect after a mutual defection.

Compared to standard tournaments, in both noisy and in probabilistic ending tournaments the higher the rates of cooperation the lower a strategy's success and median score. A strategy would want to cooperate less than both the mean and median cooperator in such settings. In probabilistic ending tournaments the correlation coefficients have larger values, indicating a stronger effect. Thus a strategy will be punished more by its cooperative behaviour in probabilistic ending environments, supporting the results of Section 4 as well. The distributions of the C_r of the winners in both tournaments are given by Figure 4. It confirms that the winners in noisy tournaments cooperated less than 35% of the time and in probabilistic ending tournaments less than 10%. In noisy probabilistic ending tournaments and over all the tournaments' results, the only features that had a moderate effect are $C_r/C_{\text{mean}}, C_r/C_{\text{max}}$ and C_r. In such environments cooperative behaviour appears to be punished less than in noisy and probabilistic ending tournaments.

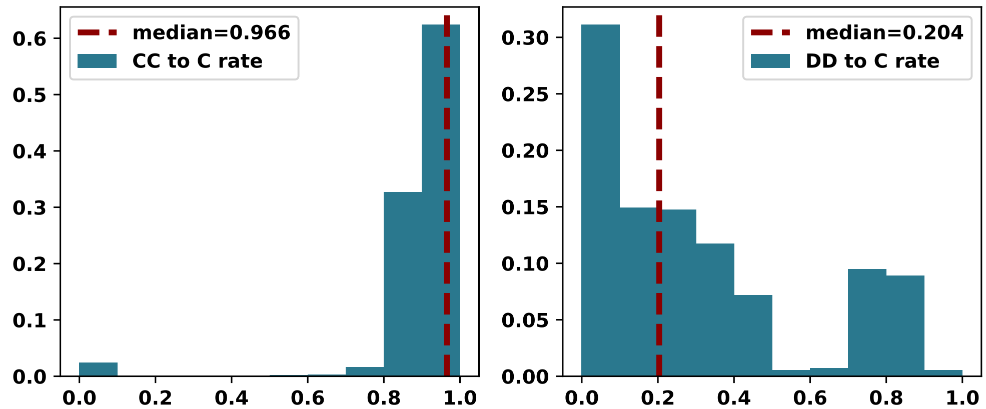


Figure 3: Distributions of CC to C and DD to C for the winners in standard tournaments.

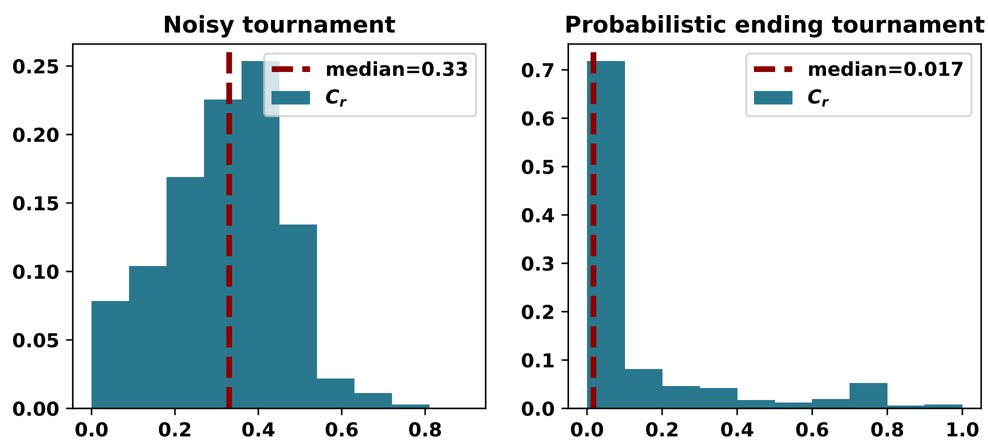


Figure 4: C_r distributions of the winners in noisy and in probabilistic ending tournaments.

A multivariate linear regression has been fitted to model the relationship between the features and the normalised rank. Based on the graphical representation of the correlation matrices given in Appendix **??** several of the features are highly correlated and have been removed before fitting the linear regression model. The features included are given by Table 5 alongside their corresponding p values in the distinct tournaments and their regression coefficients.

	Standard		Noisy		Probabilistic ending		Noisy probabilistic ending		Overall	
	R adjusted: 0.541		R adjusted: 0.639		R adjusted: 0.587		R adjusted: 0.577		R adjusted: 0.242	
	Coefficient	p-value	Coefficient	p-value	Coefficient	p-value	Coefficient	p-value	Coefficient	p-value
CC to C rate	-0.042	0.000	-0.007	0.000	0.017	0.000	0.111	0.0	-0.099	0.0
CD to C rate	0.297	0.000	-0.068	0.000	0.182	0.000	0.023	0.0	0.129	0.0
C_r / C_{max}	-	-	1.856	0.000	-	-	1.256	0.0	-	-
C_r / C_{mean}	-0.468	0.000	-0.577	0.000	0.525	0.000	-0.120	0.0	0.300	0.0
C_{max}	-0.071	0.000	-	-	-0.022	0.391	1.130	0.0	-	-
C_{mean}	0.118	0.000	-2.558	0.000	-0.023	0.001	-1.489	0.0	-	-
C_{min}	-0.161	0.000	-1.179	0.000	-0.170	0.000	-	-	-	-
C_{min} / C_r	0.057	0.000	-0.320	0.000	0.125	0.000	-	-	-0.103	0.0
DC to C rate	0.198	0.000	0.040	0.000	-0.030	0.000	0.022	0.0	0.064	0.0
k	0.000	0.319	0.000	0.020	0.000	0.002	0.000	0.0	-	-
n	0.000	0.000	-	-	-	-	-	-	-	-
p_e	-	-	-	-	0.000	0.847	-0.083	0.0	-	-
p_n	-	-	-0.048	0.000	-	-	-	-	-	-
SSE	0.258	0.000	0.153	0.000	-0.041	0.000	0.100	0.0	0.056	0.0
constant	0.697	0.000	1.522	0.000	-0.057	0.019	-0.472	0.0	0.178	0.0
memory usage	-0.010	0.000	-0.000	0.035	-	-	-	-	-	-

Table 5: Results of multivariate linear regressions with r as the dependent variable. R squared is reported for each model.

A multivariate linear regression has also be fitted on the median score. The coefficients and p values of the features can be found in Appendix **??**. This approach leads to similar conclusions.

The feature C_r/C_{mean} has a statistically significant effect across all models and a high regression coefficient. It has both a positive and negative impact on the normalised rank depending on the environment. For standard tournaments, Figure 5 gives the distributions of several features for the winners of standard tournaments. The C_r/C_{mean} distribution of the winner is also given in Figure 5. A value of $C_r/C_{\text{mean}} = 1$ implies that the cooperating ratio of the winner was the same as the mean cooperating ratio of the tournament, and in standard tournaments, the median is 1. Therefore, an effective strategy in standard tournaments was the mean cooperator of its respective tournament.

The distributions of SSE and CD to C rate for the winners of standard tournaments are also given in Figure 5. The SSE distributions for the winners indicate that the strategy behaved in a ZD way in several tournaments, however, not constantly. The winners participated in matches where they did not try to extortionate their opponents. Furthermore, the CD to C distribution indicates that if a strategy were to defect against the winners the winners would reciprocate on average with a probability of 0.5.

Similarly for the rest of the different tournaments types, and the entire data set the distributions of C_r/C_{mean}, SSE and CD to C ratio are given by Figures 6, 8, 9 and 10.

Based on the C_r/C_{mean} distributions the successful strategies have adapted differently to the mean cooperator depending on the tournament type. In noisy tournaments where the median of the distribution is at 0.67, and thereupon the winners cooperated 67% of the time the mean cooperator did. In tournaments with noise and a probabilistic ending the winners cooperated 60%, whereas in settings that the type of the tournament can vary between all the types the winners cooperated 67% of the time the mean cooperator did. Lastly,

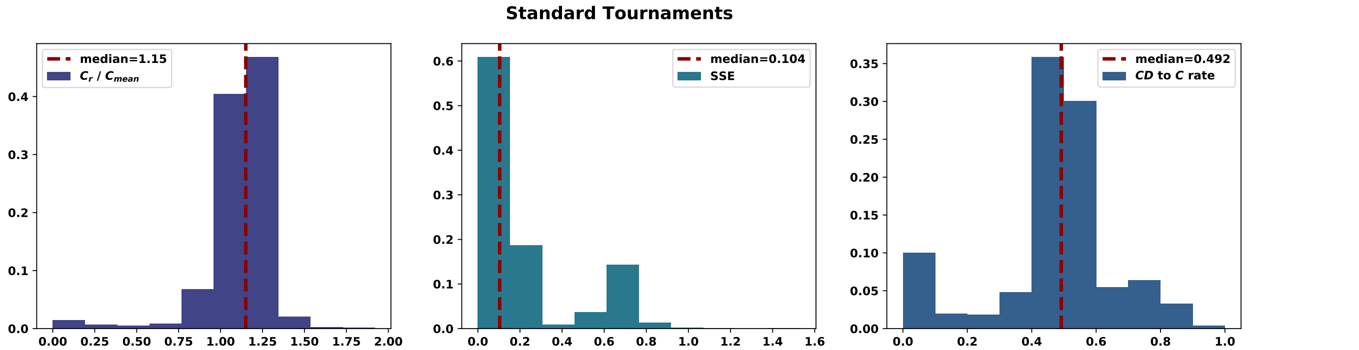


Figure 5: Distributions of C_r/C_{mean}, SSE and CD to C ratio for the winners of standard tournaments. A value of $C_r/C_{\text{mean}} = 1$ imply that the cooperating ratio of the winner was the same as the mean cooperating ratio of the tournament. An SSE distribution skewed towards 0 indicates a extortionate behaviour by the strategy.

in probabilistic ending tournaments above more defecting strategies prevail (Section 3), and this result is reflected here.

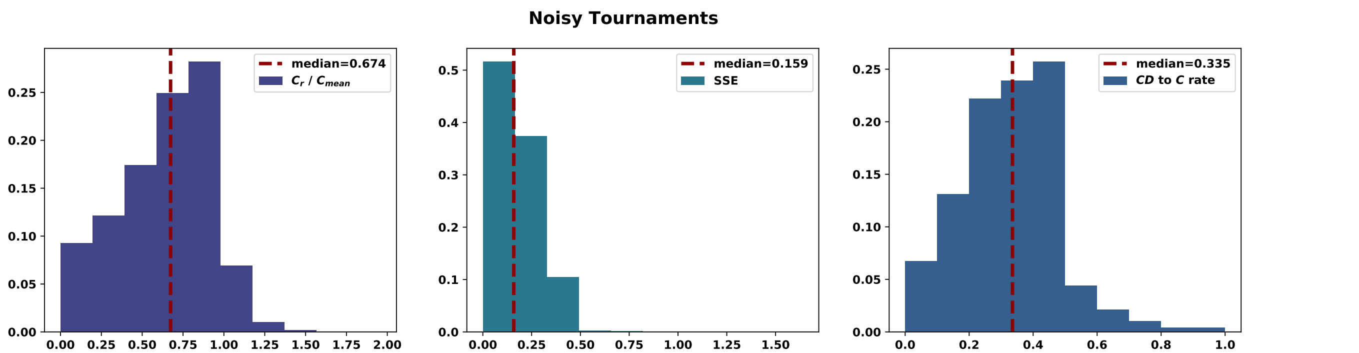
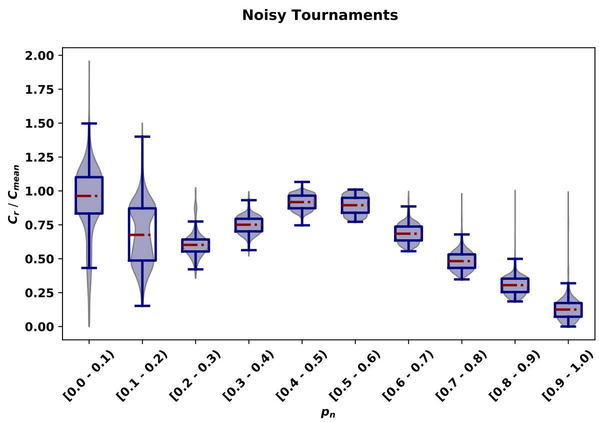


Figure 6: Distributions of C_r/C_{mean}, SSE and CD to C ratio for the winners of noisy tournaments.

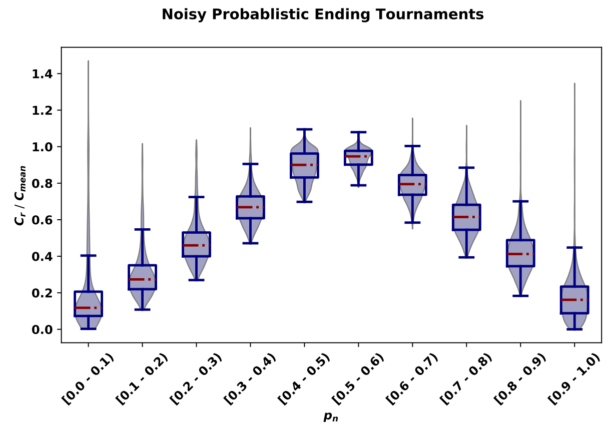
The probability of noise has been observed to substantially affect optimal behaviour. Figure 7 gives the ratio C_r/C_{mean} for the winners in tournaments with noise, over the probability of noise. From Figure 7a it is clear that the cooperating only 67% of the time the mean cooperator did is optimal only when $p_n \in [0.2, 0.4)$ and $p_n \in [0.6, 0.7]$. In environments with $p_n < 0.1$ the winners want to be close to the mean cooperator, similarly to standard tournaments, and as the probability of noise is exceeding 0.5 (where the game is effectively inverted) strategies should aim to be less and less cooperative.

Figure 7 gives C_r/C_{mean} for the winners over p_n in tournaments with noise and a probabilistic ending. The optimal proportions of cooperations are different now that the number of turns is not fixed, successful strategies want to be more defecting that the mean cooperator, that only changes when p_n approaches 0.5. Figure 7 demonstrates how the adjustments to C_r/C_{mean} change over the noise in the to the environment, and thus supports how important adapting to the environment is for a strategy to be successful.

The distributions of the SSE across the tournament types suggest that successful strategies exhibit some extortionate behaviour, but not constantly. ZDs are a set of strategies that are often envious as they try to exploit their opponents. The winners of the tournaments considered in this work are envious, but not as much as many ZDs. Though the exact interactions between the matches have not been recorded here, the work



(a) C_r/C_{mean} distribution for winners in noisy tournaments over p_n.



(b) C_r/C_{mean} distribution for winners in noisy probabilistic ending tournaments over p_n.

Figure 7: C_r/C_{mean} distributions over intervals of p_n. These distributions model the optimal proportion of cooperation compared to C_{mean} as a function of (p_n).

of [17] which introduced the trained strategies that appeared in the top ranked strategies of Section 3 did. In [17] it was shown that clever strategies managed to achieve mutual cooperation with stronger strategies whilst exploiting the weaker strategies. This could explain the clever winners of our analysis, and would explain the SSE distributions. This could also be the reason why ZDs fail to appear in the tops ranks – they try to exploit all opponents and cannot actively adapt back to mutual cooperation against stronger strategies, which requires more depth of memory. Note that ZDs also tend to perform poorly in population games for a similar reason: they attempt to exploit other players using ZDs, failing to form a cooperative subpopulation [22]. This makes them good invaders but poor resisters of invasion.

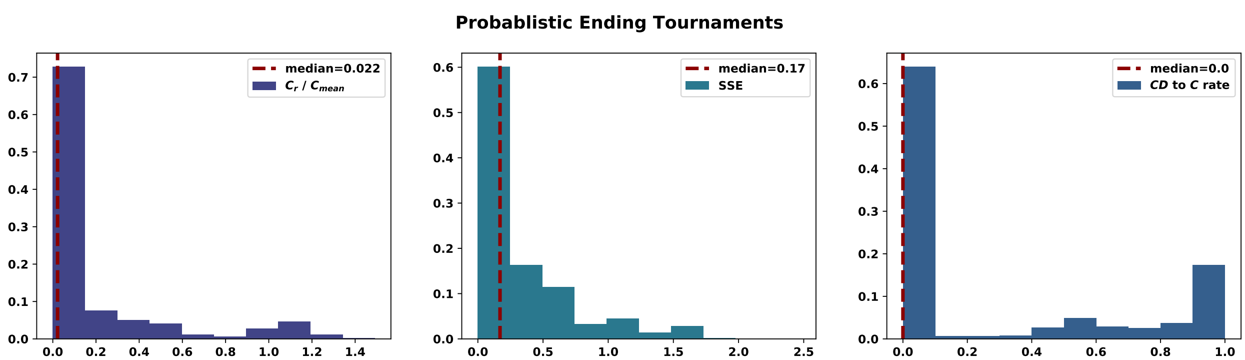


Figure 8: Distributions of C_r/C_{mean}, SSE and CD to C ratio for the winners of probabilistic ending tournaments.

The distributions of the CD to C rate evaluate the behaviour of a successful strategy after its opponent has defected against it. In standard tournaments it was observed that a successful strategy reciprocates with a probability of 0.5, and in a setting that the type of the tournament can vary between all the examined types a winning strategy would reciprocate on average with a probability of 0.58. In tournaments with noise a strategy is less likely to cooperate following a defection compared to standard tournaments, and in probabilistic ending tournaments a strategy will reciprocate a defection. This leads to adjusting the recommendation of being provocable to defections made by R. Axelrod. A strategy should be provocable in tournaments with short matches, but in the rest of the settings a strategy should be more generous.

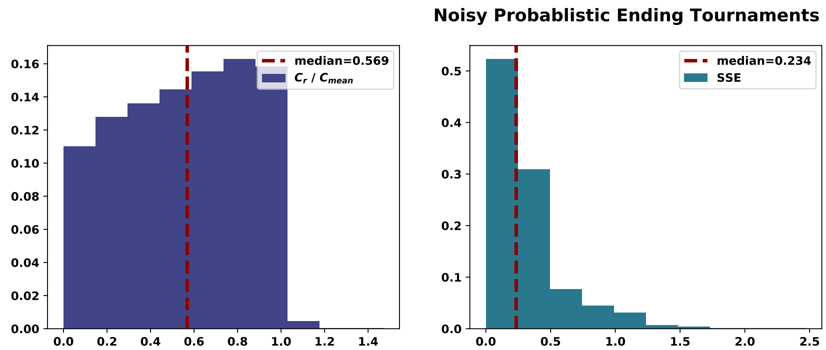


Figure 9: Distributions of C_r/C_{mean}, SSE and CD to C ratio for the winners of noisy probabilistic ending tournaments.

Further statistically significant features with strong effects include C_r/C_{min}, C_r/C_{max}, C_{min} and C_{max}. These add more emphasis on how important it is for a a strategy to adapt to its environment. Finally, the features number of turns, repetitions and the probabilities of noise and the game ending had no significant effects based on the multivariate regression models.

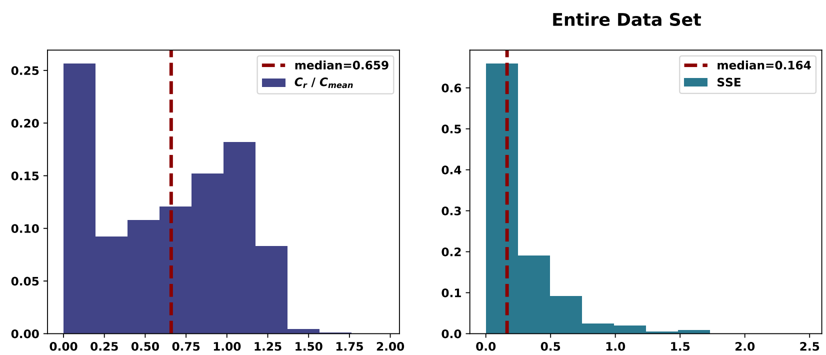


Figure 10: Distributions of C_r/C_{mean}, SSE and CD to C ratio for the winners over the tournaments of the entire data set.

A third method that evaluates the importance of the features in Table 3 using clustering and random forests can be found in the Appendix **??**. The results uphold the outcomes of the correlation and multivariate regression. It also evaluates the effects of the whether or not a strategy is stochastic, makes use of the knowledge of the utility values, or makes use of match length. These were not evaluated by the methods above because there are binary variables. The results showed that they have no significant effect on a strategy's performance.

5 Discussion

This manuscript explored the performance of 195 strategies of the IPD in 45606 computer tournaments. The collection of computer tournaments presented here is the largest and most diverse collection in the literature. The 195 strategies are drawn from the APL and include strategies from the IPD literature. The computer tournaments include tournaments of four different types.

So what is the best way of playing the IPD? And is there a single dominant strategy for the IPD?

There was not a single strategy within the collection of the 195 strategies that managed to perform well in all the tournaments variations it competed in. Even if on average a strategy ranked highly in a specific environment this did not guarantee its success over the different tournament types. Nevertheless, in Sections 3 and 4 we examined the best performing strategies across various tournament types and analysed their salient features. this demonstrated that there are properties associated with the success of strategies which in fact contradict the originally suggested properties of R. Axelrod [9].

We showed that complex or **clever** strategies can be effective, whether trained against a corpus of possible opponents or purposely designed to mitigate the impact of noise such as the DBS strategy. Moreover, we found some strategies designed or trained for noisy environments were also highly ranked in noise-free tournaments which reinforces the idea that strategies' complexity/cleverness is not necessarily a liability, rather it can confer adaptability to a more diverse set of environments. We also showed that while the type of exploitation attempted by ZDs is not typically effective in standard tournaments, **envious** strategies capable of both exploiting and not their opponents can be highly successful. Based on the results of [17] this could be because they are selectively exploiting weaker opponents while mutually cooperating with stronger opponents. Highly noisy or tournaments with short matches also favoured envious strategies. These environments mitigated the value of being nice. Uncertainty enables exploitation, reducing the ability of maintaining or enforcing mutual cooperation, while triggering grudging strategies to switch from typically cooperating to typically defecting.

The features analysis of the best performing strategies demonstrated that a strategy should reciprocate, as suggested by R. Axelrod, but it should relax its readiness to do so and be more **generous**. For noisy environments this is inline with the results of [11, 12, 25, 32], however, we also showed that generosity pays off even in standard settings, and that in fact the only setting a strategy would want to be too provocable is when the matches are not long. Forgiveness as defined by R. Axerlod was not explored here. This was mainly because the two round states were not recorded during the data collection. This could be a topic of future work that examines the impact of considering more rounds of history. The features analysis also concluded that there is a significant importance in **adapting to the environment**, and more specifically, to the mean cooperator. In standard tournaments a strategy would aim to be the mean cooperator while in noisy tournaments the best performing players cooperate at a lower rate than the tournament population on average. Moreover, the manner in which a strategy achieves a given cooperation rate relative to the tournament population average is important.

This could potentially explain the early success of TFT. TFT naturally achieves a cooperation rate near C_{mean} by virtue of copying its opponent's last move while also minimizing instances where it is exploited by an opponent (cooperating while the opponent defects), at least in non-noisy tournaments. It could also explain why Tit For N Tats does not fare well for $N > 1$ – it fails to achieve the proper cooperation ratio by tolerating too many defections.

Similarly, our results could suggest an explanation regarding the intuitively unexpected effectiveness of memory-one strategies historically. Given that among the important features associated with success are the relative cooperation rate to the population average and the four memory-one probabilities of cooperating conditional on the previous round of play, these features can be optimized by a memory-one strategy such as TFT. Usage of more history becomes valuable when there are exploitable opponent patterns. This is indicated by the importance of SSE as a feature, showing that the first-approximation provided by a memory-one strategy is no longer sufficient.

These results highlight a central idea in evolutionary game theory in this context: the fitness landscape is a function of the population (where fitness in this case is tournament performance). While that may seem obvious now, it shows why historical tournament results on small or arbitrary populations of strategies have so often failed to produce generalizable results.

Overall, the five properties successful strategies need to have in a IPD competition based on the analysis that has been presented in this manuscript are:

- Be "nice" in non-noisy environments or when game lengths are longer
- Be provocable in tournaments with short matches, and generous when matches are longer
- Be a little bit envious
- Be clever
- Adapt to the environment (including the population of strategies).

The data set described in this work contains the largest number of IPD tournaments, to the authors knowledge. The raw data set is available at [16] and the processed data at [14]. Further data mining could be applied and provide new insights in the field.

References

[1] Lifl (1998) prison. `http://www.lifl.fr/IPD/ipd.frame.html`. Accessed: 2017-10-23.

[2] Eckhart A. Coopsim v0.9.9 beta 6. `https://github.com/jecki/CoopSim/`, 2015.

[3] W. Ashlock and D. Ashlock. Changes in prisoner's dilemma strategies over evolutionary time with different population sizes. In *2006 IEEE International Conference on Evolutionary Computation*, pages 297–304. IEEE, 2006.

[4] W. Ashlock, J. Tsang, and D. Ashlock. The evolution of exploitation. In *2014 IEEE Symposium on Foundations of Computational Intelligence (FOCI)*, pages 135–142. IEEE, 2014.

[5] T. C. Au and D. Nau. Accident or intention: that is the question (in the noisy iterated prisoner's dilemma). In *Proceedings of the fifth international joint conference on Autonomous agents and multiagent systems*, pages 561–568. ACM, 2006.

[6] R. Axelrod. Effective choice in the prisoner's dilemma. *Journal of Conflict Resolution*, 24(1):3–25, 1980.

[7] R. Axelrod. More effective choice in the prisoner's dilemma. *Journal of Conflict Resolution*, 24(3):379–403, 1980.

[8] R. Axelrod. The evolution of strategies in the iterated prisoner's dilemma. *Genetic Algorithms and Simulated Annealing*, pages 32–41, 1987.

[9] R. Axelrod and W. D. Hamilton. The evolution of cooperation. *science*, 211(4489):1390–1396, 1981.

[10] B. Beaufils, J. P. Delahaye, and P. Mathieu. Our meeting with gradual, a good strategy for the iterated prisoner's dilemma. In *Proceedings of the Fifth International Workshop on the Synthesis and Simulation of Living Systems*, pages 202–209, 1997.

[11] J. Bendor, R. M. Kramer, and S. Stout. When in doubt... cooperation in a noisy prisoner's dilemma. *The Journal of Conflict Resolution*, 35(4):691–719, 1991.

[12] C. Donninger. *Is it Always Efficient to be Nice? A Computer Simulation of Axelrod's Computer Tournament*. Physica-Verlag HD, Heidelberg, 1986.

[13] M. M. Flood. Some experimental games. *Management Science*, 5(1):5–26, 1958.

[14] N. E. Glynatsi. A data set of 45686 Iterated Prisoner's Dilemma tournaments' results. `https://doi.org/10.5281/zenodo.3516652`, October 2019.

[15] N. E. Glynatsi and V. A. Knight. A bibliometric study of research topics, collaboration and influence in the field of the iterated prisoner's dilemma, 2019.

[16] Nikoleta E. Glynatsi. A data set of 45686 Iterated Prisoner's Dilemma tournaments' results [RAW DATA], April 2020.

[17] M. Harper, V. Knight, M. Jones, G. Koutsovoulos, N. E. Glynatsi, and O. Campbell. Reinforcement learning produces dominant strategies for the iterated prisoner's dilemma. *PloS one*, 12(12):e0188046, 2017.

[18] J. D. Hunter. Matplotlib: A 2D graphics environment. *Computing In Science & Engineering*, 9(3):90–95, 2007.

[19] G. Kendall, X. Yao, and S. Y. Chong. *The iterated prisoners' dilemma: 20 years on*, volume 4. World Scientific, 2007.

[20] G. Kendall, X. Yao, and S. Y. Chong. *The iterated prisoners' dilemma: 20 years on*, volume 4. World Scientific, 2007.

[21] V. A. Knight, M. Harper, N. E. Glynatsi, and J. Gillard. Recognising and evaluating the effectiveness of extortion in the iterated prisoner's dilemma. *CoRR*, abs/1904.00973, 2019.

[22] Vincent Knight, Marc Harper, Nikoleta E Glynatsi, and Owen Campbell. Evolution reinforces cooperation with the emergence of self-recognition mechanisms: an empirical study of the moran process for the iterated prisoner's dilemma. *arXiv preprint arXiv:1707.06920*, 2017.

[23] P. Mathieu and J. P. Delahaye. New winning strategies for the iterated prisoner's dilemma. *Journal of Artificial Societies and Social Simulation*, 20(4):12, 2017.

[24] J. H. Miller. The coevolution of automata in the repeated prisoner's dilemma. *Journal of Economic Behavior and Organization*, 29(1):87 – 112, 1996.

[25] P. Molander. The optimal level of generosity in a selfish, uncertain environment. *The Journal of Conflict Resolution*, 29(4):611–618, 1985.

[26] M. Nowak and K. Sigmund. A strategy of win-stay, lose-shift that outperforms tit-for-tat in the prisoner's dilemma game. *Nature*, 364(6432):56, 1993.

[27] M. A. Nowak and K. Sigmund. Tit for tat in heterogeneous populations. *Nature*, 355(6357):250, 1992.

[28] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830, 2011.

[29] prase. Prisoner's dilemma tournament results. `https://www.lesswrong.com/posts/hamma4XgeNrsvAJv5/prisoner-s-dilemma-tournament-results`, 2011.

[30] W. H. Press and F. J. Dyson. Iterated prisoner's dilemma contains strategies that dominate any evolutionary opponent. *Proceedings of the National Academy of Sciences*, 109(26):10409–10413, 2012.

[31] The Axelrod project developers. Axelrod: 3.0.0. `http://dx.doi.org/10.5281/zenodo.807699`, April 2016.

[32] R. Selten and P. Hammerstein. Gaps in harley's argument on evolutionarily stable learning rules and in the logic of "tit for tat". *Behavioral and Brain Sciences*, 7(1):115–116, 1984.

[33] D. W. Stephens, C. M. McLinn, and J. R. Stevens. Discounting and reciprocity in an iterated prisoner's dilemma. *Science*, 298(5601):2216–2218, 2002.

[34] A. J. Stewart and J. B. Plotkin. Extortion and cooperation in the prisoner's dilemma. *Proceedings of the National Academy of Sciences*, 109(26):10134–10135, 2012.

[35] E. Tzafestas. Toward adaptive cooperative behavior. 2:334–340, Sep 2000.

[36] S. Walt, S. C. Colbert, and G. Varoquaux. The NumPy array: a structure for efficient numerical computation. *Computing in Science & Engineering*, 13(2):22–30, 2011.

6 Acknowledgements

A variety of software have been used in this work:

- The Axelrod-Python library for IPD simulations [31].
- The Matplotlib library for visualisation [18].
- The Numpy library for data manipulation [36].
- The scikit-learn library for data analysis [28].