Stability of defection, optimisation of strategies and the limits of memory in the Prisoner's Dilemma.

Nikoleta E. Glynatsi Vincent A. Knight

Abstract

Memory-one strategies are a set of iterated prisoners dilemma strategies that have been praised for their mathematical tractability and robustness. This manuscript investigates *best response* memory-one strategies and studies them as a multidimensional optimisation problem. Though extortionate memory-one strategies have gained much attention, we prove that best response memory-one strategies do not behave in an extortionate way. For memory one strategies to be evolutionary robust they need to be able to behave in a forgiving way. We also provide evidence that memory-one strategies suffer from the limitation of their memory and can be out performed in multi agent interactions by longer memory strategies.

1 Introduction

The Prisoner's Dilemma (PD) is a two player game used in understanding the evolution of co-operative behaviour, formally introduced in [9]. Each player has two options, to cooperate (C) or to defect (D). The decisions are made simultaneously and independently. The normal form representation of the game is given by:

$$S_p = \begin{pmatrix} R & S \\ T & P \end{pmatrix} \quad S_q = \begin{pmatrix} R & T \\ S & P \end{pmatrix} \tag{1}$$

where S_p represents the utilities of the row player and S_q the utilities of the column player. The payoffs, (R, P, S, T), are constrained by equations (2) and (3). Constraint (2) ensures that defection dominates cooperation and constraint (3) ensures that there is a dilemma; the sum of the utilities for both players is better when both choose to cooperate. The most common values used in the literature are $(3, 1, 0, 5)$ [3].

$$T > R > P > S \tag{2}$$

$$2R > T + S \tag{3}$$

The PD is a one shot game, however it is commonly studied in a manner where the history of the interactions matters. The repeated form of the game is called the Iterated Prisoner's Dilemma (IPD) and in the 1980s, following the work of [4, 5] it attracted the attention of the scientific community. In [4] and [5], the first well known computer tournaments of the IPD were performed. A total of 13 and 63 strategies were submitted respectively in the form of computer code. The contestants competed against each other, a copy of themselves and a random strategy, and the winner was then decided on the average score a strategy achieved (not the total number of wins). The contestants were given access to the entire history of a match, however, how many turns of history a strategy would incorporate, refereed to as the *memory size* of a strategy, was a result of the particular strategic decisions made by the author. The winning strategy of both tournaments was the strategy called Tit for Tat and it's success, in both tournaments, came as a surprise. Tit for Tat was a simple, forgiving strategy that opened each interaction by cooperation, but it had managed to defeat far more complicated opponents. Tit for Tat provided evidence that being nice can be advantageous and became the major paradigm for reciprocal altruism.

Another trait of the strategy is that it considers only the previous move of the opponent. These type of strategies are called *reactive* [24] and are a subset of so called *memory-one* strategies, which incorporate both players' recent moves. Reactive and memory-one strategies have been studied thoroughly in literature [25, 26]. They have gained most of their attention when a curtain set of memory-one strategies was introduced in [29]. [30] stated that "Press and Dyson have fundamentally changed the viewpoint on the Prisoner's Dilemma".

Zero-determinant strategies (ZD) are a special case of memory-one and extortionate strategies. They chose their actions so that a linear relationship is forced between their score and that of the opponent, reassuring that they will always receive at least as much as their opponents. ZD strategies are indeed mathematically unique and are proven to be robust in pairwise interactions. Their true effectiveness in tournament interactions [19] and evolutionary dynamics [2, 18] has been questioned by several works.

The purpose of this work is to consider a given memory-one strategy in a similar fashion to [29], however whilst [29] found a way for a player to manipulate a given opponent, this work will consider a multidimensional optimisation approach to identify the best response memory-one to a group of opponents. The main questions we raise are concerned with:

- A compact method of identifying the best response memory-one strategy against a given set of opponents.
- The behaviour of a best response memory-one strategy and whether it behaves extortionate, similar to [29].
- The factors that make a best response memory-one strategy evolutionary robust.
- A well designed framework that allows the comparison of an optimal memory one strategy, and a more complex strategy that has a larger memory and was obtained through contemporary reinforcement learning techniques.
- An identification of conditions for which defection is known to be a best response; thus identifying environments where cooperation can not occur.

2 The utility

One specific advantage of memory-one strategies is their mathematical tractability. They can be represented completely as a vector of $\mathbb{R}^4$. This originates from [24] where it is stated that if a strategy is concerned with only the outcome of a single turn then there are four possible 'states' the strategy could be in; CC, CD, DC, CC. Therefore, a memory-one strategy can be denoted by the probability vector of cooperating after each of these states; $p = (p_1, p_2, p_3, p_4) \in \mathbb{R}^4_{[0,1]}$. In an IPD match two memory-one strategies are moving from state to state, at each turn with a given probability. This exact behaviour can be modeled as a stochastic process, and more specifically as a Markov chain (Figure 1). The corresponding transition matrix M of Figure 1 is given below,

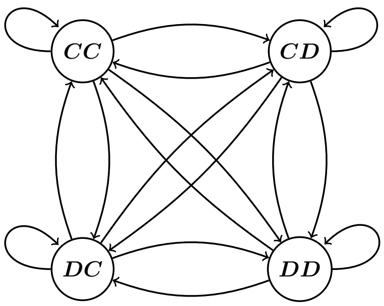


Figure 1: markov

$$M = \begin{bmatrix} p_1q_1 & p_1(-q_1+1) & q_1(-p_1+1) & (-p_1+1)(-q_1+1) \\ p_2q_3 & p_2(-q_3+1) & q_3(-p_2+1) & (-p_2+1)(-q_3+1) \\ p_3q_2 & p_3(-q_2+1) & q_2(-p_3+1) & (-p_3+1)(-q_2+1) \\ p_4q_4 & p_4(-q_4+1) & q_4(-p_4+1) & (-p_4+1)(-q_4+1) \end{bmatrix}$$

The long run steady state probability v is the solution to $vM = v$. The stationary vector v can be combined with the payoff matrices of equation (1) and the expected payoffs for each player can be estimated without simulating the actual interactions. More specifically, the utility for a memory-one strategy p against an opponent q, denoted as $u_q(p)$, is defined by,

$$u_q(p) = v \times (R, P, S, T). \tag{4}$$

In Theorem 1, the first theoretical results of the manuscript is presented, that is that $u_q(p)$ is given by a ratio of two quadratic forms [16]. To the authors knowledge this is the first work that has been done on the form of $u_q(p)$.

Theorem 1. *The expected utility of a memory-one strategy $p \in \mathbb{R}^4_{[0,1]}$ against a memory-one opponent $q \in \mathbb{R}^4_{[0,1]}$, denoted as $u_q(p)$, can be written as a ratio of two quadratic forms:*

$$u_q(p) = \frac{\frac{1}{2}pQp^T + cp + a}{\frac{1}{2}p\bar{Q}p^T + \bar{c}p + \bar{a}}, \tag{5}$$

where $Q, \bar{Q} \in \mathbb{R}^{4\times 4}$ are hollow matrices defined by the transition probabilities of the opponent q_1, q_2, q_3, q_4 as follows:

$$\boldsymbol{Q} = \begin{bmatrix} 0 & -(q_1-q_3)(q_2-5q_4-1) & q_3(q_1-q_2) & -5q_3(q_1-q_4) \\ -(q_1-q_3)(q_2-5q_4-1) & 0 & (q_2-q_3)(q_1-3q_4-1) & (q_3-q_4)(5q_1-3q_2-2) \\ q_3(q_1-q_2) & (q_2-q_3)(q_1-3q_4-1) & 0 & 3q_3(q_2-q_4) \\ -5q_3(q_1-q_4) & (q_3-q_4)(5q_1-3q_2-2) & 3q_3(q_2-q_4) & 0 \end{bmatrix}, \tag{6}$$

$$\bar{\boldsymbol{Q}} = \begin{bmatrix} 0 & -(q_1-q_3)(q_2-q_4-1) & (q_1-q_2)(q_3-q_4) & (q_1-q_4)(q_2-q_3-1) \\ -(q_1-q_3)(q_2-q_4-1) & 0 & (q_2-q_3)(q_1-q_4-1) & (q_1-q_2)(q_3-q_4) \\ (q_1-q_2)(q_3-q_4) & (q_2-q_3)(q_1-q_4-1) & 0 & -(q_2-q_4)(q_1-q_3-1) \\ (q_1-q_4)(q_2-q_3-1) & (q_1-q_2)(q_3-q_4) & -(q_2-q_4)(q_1-q_3-1) & 0 \end{bmatrix}. \tag{7}$$

c and $\bar{c} \in \mathbb{R}^{4\times 1}$ are similarly defined by:

$$c = \begin{bmatrix} q_1 (q_2 - 5q_4 - 1) \\ -(q_3 - 1)(q_2 - 5q_4 - 1) \\ -q_1q_2 + q_2q_3 + 3q_2q_4 + q_2 - q_3 \\ 5q_1q_4 - 3q_2q_4 - 5q_3q_4 + 5q_3 - 2q_4 \end{bmatrix}, \tag{8}$$

$$\bar{c} = \begin{bmatrix} q_1 (q_2 - q_4 - 1) \\ -(q_3 - 1)(q_2 - q_4 - 1) \\ -q_1q_2 + q_2q_3 + q_2 - q_3 + q_4 \\ q_1q_4 - q_2 - q_3q_4 + q_3 - q_4 + 1 \end{bmatrix}. \tag{9}$$

and $a = -q_2 + 5q_4 + 1$ *and* $\bar{a} = -q_2 + q_4 + 1$.

The proof of Theorem 1 is given in Appendix.

Numerical simulations have been carried out to validate the formulation of $u_q(p)$ as a quadratic ratio, a data set is available at. Figure 2 shows that the formulation successfully captures the simulated behaviour. The simulated utility, which is denoted as $U_q(p)$, has been calculated using [1] an open source research framework for the study of the IPD. The project is described in [17]. All of the aforementioned simulated results have been estimated using [1]. For smoothing the simulated results the simulated utility has been estimated in a tournament of 500 turns and 200 repetitions. The source code used in this manuscript is public and has been archieved here.

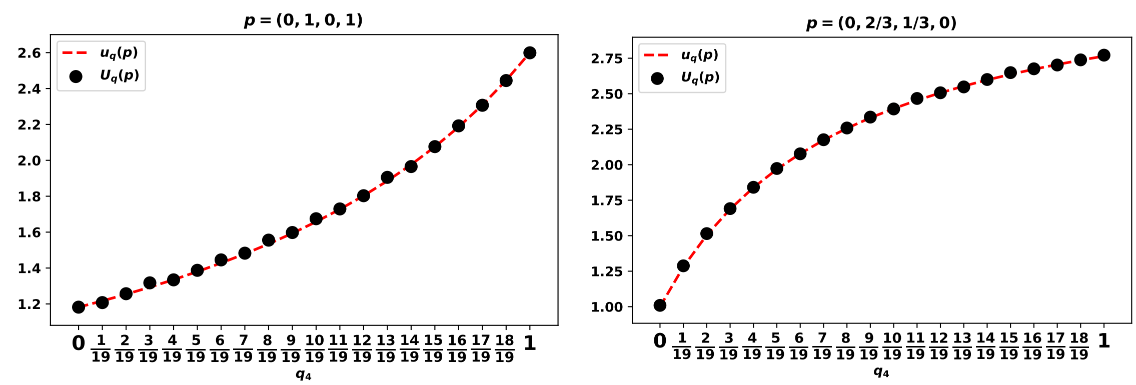


Figure 2: Differences between simulated and analytical results for $q = (\frac{1}{3}, \frac{1}{3}, \frac{1}{3}, q_4)$.

Theorem 1 can be extended to consider multiple opponents. The IPD is commonly studied in tournaments and/or Moran Processes where a strategy interacts with a number of opponents. The payoff of a player in such interactions is given by the average payoff the player received against each opponent. More specifically the expected utility of a memory-one strategy against a N number of opponents is given by Theorem 2.

Theorem 2. *The expected utility of a memory-one strategy* $p \in \mathbb{R}^4_{[0,1]}$ *against a group of opponents* $q^{(1)}, q^{(2)}, \dots, q^{(N)}$, *denoted as* $\frac{1}{N}\sum\limits_{i=1}^{N} {u_q}^{(i)}(p)$ *is given by:*

$$\frac{1}{N}\sum_{i=1}^{N} {u_q}^{(i)}(p) = \frac{1}{N} \frac{\sum\limits_{i=1}^{N} (\frac{1}{2}pQ^{(i)}p^T + c^{(i)}p + a^{(i)}) \prod\limits_{\substack{j=1 \\ j \neq i}}^{N} (\frac{1}{2}p\bar{Q}^{(i)}p^T + \bar{c}^{(i)}p + \bar{a}^{(i)})}{\prod\limits_{i=1}^{N} (\frac{1}{2}p\bar{Q}^{(i)}p^T + \bar{c}^{(i)}p + \bar{a}^{(i)})}. \tag{10}$$

Theorem 2 is validated against the strategies used in [30], Figure 3. The list of strategies from [30] alongside their original reference from the are given by Table 5 in the Appendix.

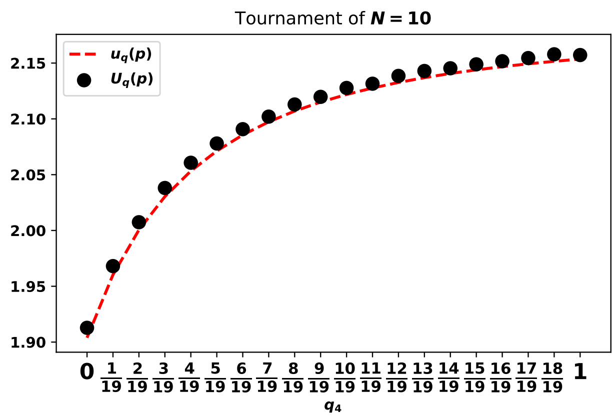


Figure 3: Results of memory-one strategies against the strategies in Table 5.

Furthermore, using the same list of strategies the hypothesis the utility against a group of strategies could be captured by the utility against the mean opponent, thus:

$$\frac{1}{N}\sum_{i=1}^{N} {u_q}^{(i)}(p) = u_{\frac{1}{N}\sum\limits_{i=1}^{N} q^{(i)}}(p), \tag{11}$$

has been checked. Condition (11) fails and numerical evidence is given by Figure 4.

Theorem 2 can be used to identify best responses in the case of memory-one strategies. In the following sections several theoretical results are presented and the advantages of analytical formulation of become evident.

3 Best responses to memory-one players

A *best response* is the strategy which corresponds to the most favorable outcome. A best response memory-one strategy corresponds to the p^* for which $\sum u_{q^{(i)}}(p^*)$ for $i \in \{1, \dots, N\}$ is maximized. This is considered as a multi dimensional optimisation problem where the decision variable is the vector p, the solitary constraint is that $p \in \mathbb{R}^4_{[0,1]}$ and the objective function is a sum of quadratic ratios. The optimisation problem is formally given by (12).

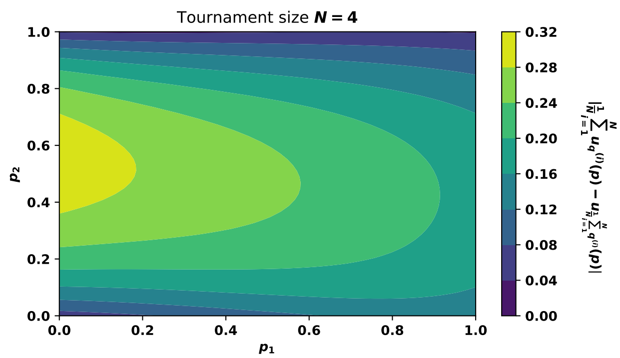


Figure 4: The difference between the average utility and against the utility against the average player of the strategies in [30]. A positive difference indicates that the condition (11) does not hold.

$$\begin{aligned} \max_p : & \sum_{i=1}^{N} {u_q}^{(i)}(p) \\ \text{such that} : & \ p \in \mathbb{R}_{[0,1]} \end{aligned} \tag{12}$$

Optimising this particular ratio of quadratic forms is not trivial. It can be verified empirically for the case of a single opponent that there exist at least one point for which the definition of concavity does not hold. Though [6, 8] are also concerned with a non concave ratios of quadratic forms, in both the numerator and the denominator of the fractional problem were concave or that the denominator was greater than zero; both assumptions fail here. These results are established in Theorem 3.

Theorem 3. *The utility of a player p against an opponent q, $u_q(p)$ given by (5), is not concave. Furthermore neither the numerator or the denominator of (5), are concave.*

The non concavity of $u(p)$ indicates multiple local optimal points. The approach taken here is to introduce a compact way of constructing the candidate set of all local optimal points. Once the set is defined the point that maximises (10) corresponds to the best response strategy, this approach transforms the continuous optimisation problem in to a discrete problem. The problem considered is a bounded because $p \in \mathbb{R}^4_{[0,1]}$. The candidate solutions will exist either at the boundaries of the feasible solution space, or within that space. The method of Lagrange Multipliers [7] and KarushKuhnTucker conditions [10] are based on this.

These lead to Lemma 4 which presents the best response memory-one strategy to a group of opponents.

Lemma 4. *The optimal behaviour of a memory-one strategy player $p^* \in \mathbb{R}^4_{[0,1]}$ against a set of N opponents $\{q^{(1)}, q^{(2)}, \dots, q^{(N)}\}$ for $q^{(i)} \in \mathbb{R}^4_{[0,1]}$ is established by:*

$$p^* = \text{argmax}(\sum_{i=1}^{N} u_q(p)), \ p \in S_q.$$

The set S_q is defined as all the possible combinations of:

$$S_q = \left\{ p \in \mathbb{R}^4 \middle\| \begin{array}{l} \bullet \quad p_j \in \{0,1\} \quad and \quad \dfrac{d}{dp_k} \sum\limits_{i=1}^{N} u_q^{(i)}(p) = 0 \quad \forall \quad j \in J \quad \& \quad k \in K \quad \forall \quad J, K \\ \qquad\qquad\qquad\qquad where \quad J \cap K = \emptyset \quad and \quad J \cup K = \{1,2,3,4\} \\ \bullet \quad p \in \{0,1\}^4 \end{array} \right\}.$$

Proof in the Appendix.

The derivate $\frac{d}{dp} \sum\limits_{i=1}^{N} u_q(p)$ is given by:

$$\frac{d}{dp} \sum_{i=1}^{N} {u_q}^{(i)}(p) = \\ = \frac{(\sum\limits_{i=1}^{N} {Q_N^{(i)}}' \prod_{\substack{j=1 \\ j \neq i}}^{N} Q_D^{(i)} + \sum\limits_{i=1}^{N} {Q_D^{(i)}}' \sum_{\substack{j=1 \\ j \neq i}}^{N} Q_N^{(i)} \prod^{N}_{\substack{j=1 \\ j \neq \{i,j\}}} Q_D^{(i)}) \times \prod\limits_{i=1}^{N} Q_D^{(i)} - (\sum\limits_{i=1}^{N} {Q_D^{(i)}}' y - vk \prod_{\substack{j=1 \\ j \neq i}}^{N} Q_D^{(i)}) \times (\sum\limits_{i=1}^{N} Q_N^{(i)} \prod_{\substack{j=1 \\ j \neq i}}^{N} Q_D^{(i)})}{(\prod\limits_{i=1}^{N} Q_D^{(i)})^2} \tag{13}$$

For $\frac{d}{dp} \sum\limits_{i=1}^{N} u_q(p)$ to equal zero then:

$$\sum_{i=1}^{N} {Q_N^{(i)}}' \prod_{\substack{j=1 \\ j \neq i}}^{N} Q_D^{(i)} + \sum_{i=1}^{N} {Q_D^{(i)}}' \sum_{\substack{j=1 \\ j \neq i}}^{N} Q_N^{(i)} \prod_{\substack{j=1 \\ j \neq \{i,j\}}}^{N} Q_D^{(i)}) \times \prod_{i=1}^{N} Q_D^{(i)} - (\sum_{i=1}^{N} {Q_D^{(i)}}' y - vk \prod_{\substack{j=1 \\ j \neq i}}^{N} Q_D^{(i)}) \times (\sum_{i=1}^{N} Q_N^{(i)} \prod_{\substack{j=1 \\ j \neq i}}^{N} Q_D^{(i)} = 0, \quad while$$

$$(\prod_{i=1}^{N} Q_D^{(i)})^2 \neq 0.$$

Constructing the subset S_q is analytical possible. The points for any or none of $p_i \in \{0,1\}$ for $i \in 1,2,3,4$ are trivial. Finding the roots of the partial derivatives $\frac{d}{dp} \sum\limits_{i=1}^{N} u_q(p)$ is feasible using resultant theory. Resultant theory [15] allow us to solve systems of polynomials by the calculation of a resultant. However, for large systems these quickly become intractable. Because of that no further analytical consideration is given to problems described here.

So far we have provided an analytical formulation that can estimate best response memory-one strategies against a number of opponents. This will be revisited and solved numerically in Section 4. In the following subsection we present a theoretical results which has been possible due to the formulation discussed here.

3.1 Stability of defection

A topic often written about is the emergence of cooperation and when it is possible. Whether it is in a tournament setting [3] or in spatial interactions [28], researchers have establish several conditions under which cooperation is possible. The formulation of best responses discussed here allow us to provide an identification of conditions for which defection is known to be a best response in a memory-one environment; thus identifying environments where cooperation can not occur.

The results are presented in Lemma 5.

Lemma 5. *In a tournament of N players where $q^{(i)} = (q_1^{(i)}, q_2^{(i)}, q_3^{(i)}, q_4^{(i)})$ defection is a best response if the transition probabilities of the opponents satisfy the condition:*

$$\sum_{i=1}^{N} (c^{(i)T} \bar{a}^{(i)} - \bar{c}^{(i)T} a^{(i)}) \leq 0 \tag{14}$$

Proof. For defection to be evolutionary stable the derivative of the utility at the point $p = (0, 0, 0, 0)$ must be negative. This would indicate that the utility function is only declining from that point onwards.

Substituting $p = (0, 0, 0, 0)$ in equation (13) which gives:

$$\sum_{i=1}^{N} (c^{(i)T} \bar{a}^{(i)} - \bar{c}^{(i)T} a^{(i)}) \prod_{\substack{j = 1 \\ j \neq i}}^{N} (\bar{a}^{(i)})^2 \tag{15}$$

The second term $\prod_{\substack{j = 1 \\ j \neq i}}^{N} (\bar{a}^{(i)})^2$ is always positive, however, the sign of the first term $\sum_{i=1}^{N} (c^{(i)T} \bar{a}^{(i)} - \bar{c}^{(i)T} a^{(i)})$ can vary based on the transition probabilities of the opponents. Thus the sign of the derivative is negative if and only if $\sum_{i=1}^{N} (c^{(i)T} \bar{a}^{(i)} - \bar{c}^{(i)T} a^{(i)}) \leq 0$. □

4 Numerical experiments

In this section best responses are explored numerically. Best responses are estimated using Bayesian optimisation algorithm, which is a global optimisation algorithm, introduced in [22], that has proven to outperform many other popular algorithms [14]. Differential evolution had also been considered, however it was not selected due to Bayesian being computationally faster.

Bayesian optimisation tries to find values for the decision variables for which the utility of a player is maximised over a given time of calls. Consider the problem of (12) where $N = 2$ and p^* is being estimated. Figure 5 illustrates the change of the utility function over number of calls. The default number of iterations that have been used in this work is 60. After 60 calls the convergence of the utility is checked. If the utility is not converged then calls are increased by 20. This is repeated until the utility reaches convergence.

Bayesian optimisation will be used not only to estimate best response memory-one strategies but also to explore a series of best response cases, such as evolutionary memory-one and longer memory best responses. This is done so we can gain a better understanding of memory-one strategies, their behaviour, robustness and limitations.

4.1 Best response memory-one strategies for $N = 2$

The first case considered is that of best response memory-one strategies in tournaments in order to understand whether best responses behave like ZD. A large data set of best response memory-one strategies when $N = 2$

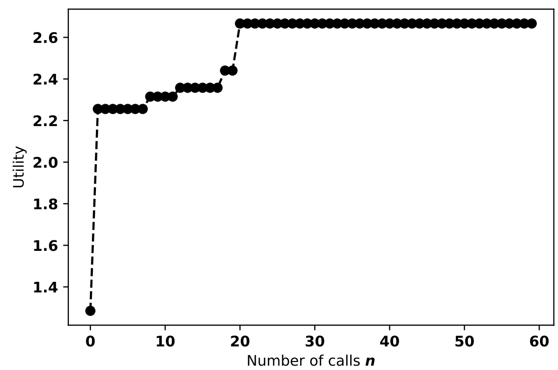
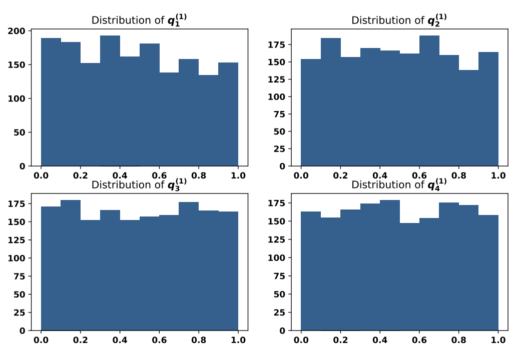


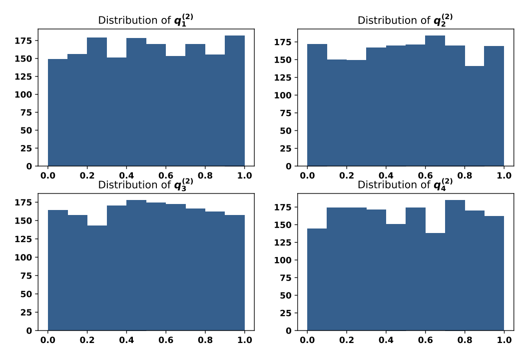
Figure 5: Utility over time of calls using Bayesian optimisation. The opponents are $q^{(1)} = (\frac{1}{3}, \frac{1}{3}, \frac{1}{3}, \frac{1}{3})$ and $q^{(2)} = (\frac{1}{3}, \frac{1}{3}, \frac{1}{3}, \frac{1}{3})$.

has been generated and is available here. $N = 2$ has been chosen as it's the smallest possible size of a tournament.

The data set contains a total of 1643 trials and 1643 different best responses. For each trial a set of 2 opponents is randomly generated, the memory-one best response against them is estimated and it's behaviour is being recorded. Though the probabilities q_i of the opponents are randomly generated, Figures 6a and 6b, show that they are uniformly distributed over the trial. Thus, the space full space of possible opponents has been covered.



(a) Distributions of first opponents' probabilities.



(b) Distributions of second opponents' probabilities.

It was briefly discussed in Section 1 that ZD strategies have received praised for their robustness against a single opponent. By forcing a linear relationship between the scores ZD strategies will always manage to receive a higher or the same payoff as their opponents. In tournaments the winner is defined by the average score a strategy received, thus winning against your opponent at each interaction does not guaranty a strategy's overall win. This manuscript argues that by trying to exploit their opponents ZD strategies suffer in multi opponent interaction where the payoffs matter. Compared to ZD best response memory-one strategies utilise their behaviour to gain the most from their interactions.

In [Knight 2019] the authors provided a method of measuring whether a strategy is ZD, based on it's estimated probabilities p. The method estimates the error of behaving as a ZD strategy defined as SSerror.

In [Knight2019], the method is applied to a large corpus of strategies. More specifically the SSerror is estimated for a total of 204 strategies at each repeated interaction with the rest 203 strategies. In this paper we similarly apply the SSerror to the best response memory one strategy, in repeated interactions against two random opponents each time. The distribution of the SSerror is shown in Figure 9 and a statistics summary by Table 4.

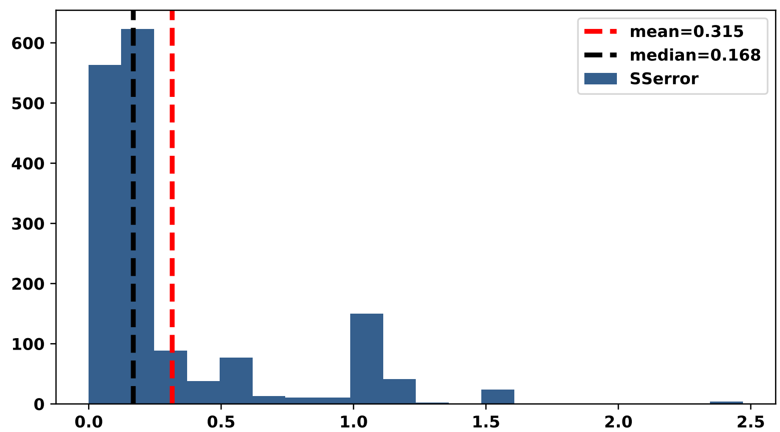


Figure 7: Distribution of SSerrors for memory-one best responses, when $N = 2$.

	SSerror
count	1643.000000
mean	0.315173
std	0.376325
min	0.000000
5%	0.011068
25%	0.075948
30%	0.104707
35%	0.126840
50%	0.168275
95%	1.058824
max	2.470588
median	0.168275
skew	1.939179
kurt	3.652869

Table 1: Summary statistics SSerror

Thought the distribution of SSerrors is skewed to left, indicating that the best response does exhibit a ZD behaviour, the best response is not a ZD. A positive measure of skewness and kurtosis indicate a heavy tail to the right. In several interaction best responses do not try to extortion their opponents. Best responses utilise the performance by behaving in a more variant way than ZD strategies. The following section the second experiment and the result of memory-one best responses in evolutionary dynamics are presented.

4.2 Memory-one best responses in evolutionary dynamics

As briefly discussed in Section 2, the IPD is commonly studied in Moran Processes, and generally in evolutionary processes. In evolutionary processes, a finite population is assumed where the strategies that compose the population can adapt and change their behaviour based on the outcomes of their interactions at each turn. A key in successfully being an evolution stable strategy (ESS) is self interactions. An ESS must be a best response not only to the opponents in the population, but also it has to be a best response to it's self.

Self interactions can easily be incorporated in the formulation that has been used in this paper. The utility of a memory-one strategy in an evolutionary setting is given by,

$$\frac{1}{N}\sum_{i=1}^{N} {u_q}^{(i)}(p) + u_p(p). \tag{16}$$

and respectively the optimisation problem of (12) is now re written as,

$$
\begin{aligned}
\max_p : \frac{1}{N} \sum_{i=1}^{N} {u_q}^{(i)}(p) + u_p(p) \\
\text{such that} : p \in \mathbb{R}_{[0,1]}
\end{aligned} \tag{17}
$$

We suggest an algorithmic approach for estimating the evolutionary best response memory one strategy (evo) called the algorithm *best response dynamics*, given by Algorithm 1.

Note that best response dynamics are commonly used in evolutionary game theory. Best response dynamics represent a class of strategy updating rules, where players strategies in the next round are determined by their best responses to some subset of the population, whether this might be in a large population model such as Moran Processes [18] or in a spatial model [27]. Moreover, in the theory of potential games, best response dynamics refers to a way of finding a pure Nash equilibrium by computing the best response for every player [23]. Here we have defined a combination of the two methods.

$p^{(t)} \leftarrow (1, 1, 1, 1)$;
while $p^{(t)}$ *not converged* **do**
    $p^{(t+1)} = \text{argmax} \frac{1}{N} \sum\limits_{i=1}^{N} {u_q}^{(i)}(p^{(t+1)}) + u_p^{(t)}(p^{(t+1)})$;
end

Algorithm 1: Best response dynamics Algorithm

The algorithm starts by setting an initial solution $p^{(1)} = (1, 1, 1, 1)$. Though a more optimal set of probabilities could be considered for an initial solution, it has been shown that the algorithm converges to same optimal solution even with more optimal starts. Once the initial solution is given then $p^{(2)}$ is estimated as the best response memory-one to the N opponents plus to $(1, 1, 1, 1)$. The current solution then changes to be the new best response memory-one, $p^{(2)}$. In the next step, $p^{(3)}$ is the best response memory-one to the N opponents plus to $p^{(2)}$. This is repeated until the same algorithms returns a solution that has already been evaluated. This is done in order to avoid cycles. Figure 8 illustrates a numerical example. The algorithm stops once it evaluates the same point again.

For each pair of opponents, from [ref], we have also recorder the evo strategy. Thus, a total of 1643 different evos have been estimated. Similarly, to previous results, the evo strategy does not behave as a ZD. A larger value of both the kurtosis and the skewness of the SSerror distribution indicate that evo is more variant, thus adaptable, than the equivalent best response, Figure 9 and Table 4.

To further understand the difference between the best response and evo memory-one strategy we consider the distributions of their respective transition probabilities (p_1, p_2, p_3, p_4), Figure 10. Though there is no significant difference between the medians of each distribution, Table 3, it is evident from Figure 10 that there is variation in the behaviors. Except from the case of p_3. If a strategy manages to get away with a defection, and receives a temptation payoff, the strategy will most certainly try another defection in the next round. That is true for both best response and evo memory-one.

Regarding the cases of p_1, p_2 and p_4:

- After the CC state where a mutual cooperation has occurred, the best responses memory-one strategy will make a binomial choice cooperation and defection. It is more likely however, that it will defect break the cycle of cooperation. In comparison the evo strategy is more stochastic. The two extreme peaks do not appear in the case of evo, and overall there is a slightly bigger insensitive for cooperation in the next round.

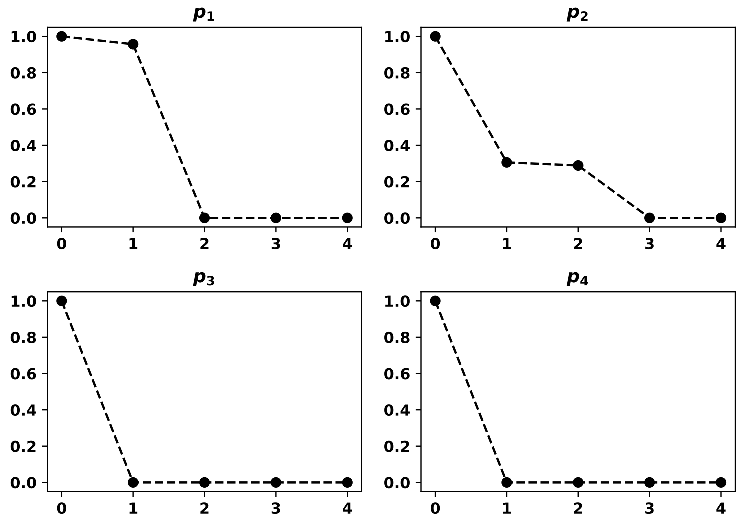


Figure 8: Best response dynamics with $N = 2$. More specifically, for $q^{(1)} = (0.2360, 0.1031, 0.3960, 0.1549)$ and $q^{(2)} = (0.0665, 0.4015, 0.9179, 0.8004)$.

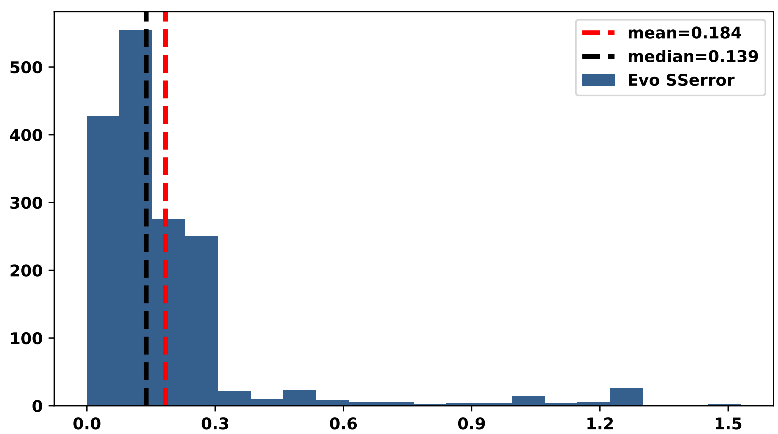


Figure 9: Distribution of sserrors for memory-one best responses, when $N = 2$

	Evo SSerror
count	1643.000000
mean	0.183524
std	0.216175
min	0.000000
5%	0.015543
25%	0.073650
30%	0.094743
35%	0.111698
50%	0.138574
95%	0.529412
max	1.529412
median	0.138574
skew	3.488164
kurt	13.206905

Table 2: Summary statistics SSerror

	Memory one Median	Evo Median	p-values
Distribution p_1	0.0	0.000000	0.0
Distribution p_2	0.0	0.173727	0.0
Distribution p_3	0.0	0.000000	0.0
Distribution p_4	0.0	0.000000	0.0

Table 3: A non parametric test, Wilcoxon Rank Sum, has been performed to tests the difference in the medians. A non parametric test is used because is evident that out data are skewed.

- Following the state of CD, a state that a strategy has been tricked, a best response memory-one strategy will very quickly punish it's opponent. On the other hand, the evo strategy is slightly more likely to cooperate again, to forgive it's opponent. This could be a result of self interaction and evo avoiding to punish a copy of themselves.
- Finally, in cases that a mutual defection has occur, the evo's probability of cooperating again is practically zero. Evo is not a forgiving strategy after a mutual defection, whereas a best response strategy is.

The difference of between the distributions of the transition probabilities has also been calculated to further confirm our results. This was done by subtracting the p of a best response memory one from the evo strategy. The differences are given in Figure 11. An interested point is that the difference in p_2 is following a normal distribution. *This is interesting. Why?*

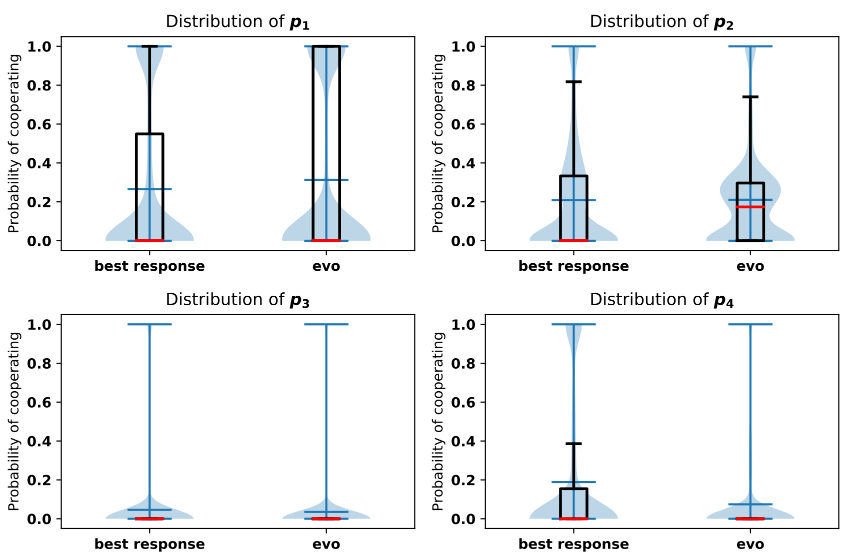


Figure 10: Distributions of p for both best response and evo memory-one strategies.

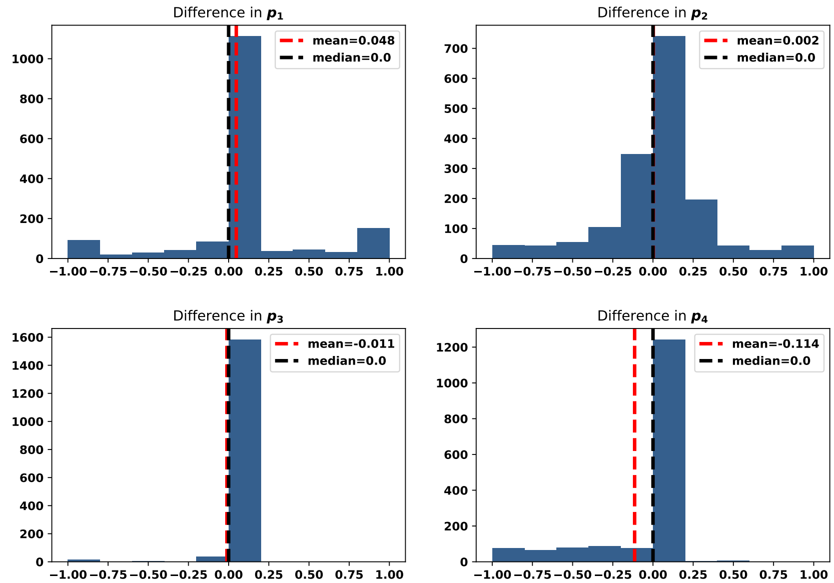


Figure 11: Differences of $p_{(i)}$ for $i \in \{1, 2, 3, 4\}$ between evos and best responses at each trial.

	Skewness	Kurtosis
Difference in p_1	0.155659	1.845893
Difference in p_2	-0.124756	2.717163
Difference in p_3	-6.766827	79.624372
Difference in p_4	-1.906567	3.759093

4.3 Longer memory best response

The third and final case considered in this paper focuses on proving that short memory strategies have limitations. In this section we present several empirical results that show that longer memories strategies can indeed perform better in cases of $N = 2$. This is achieved by comparing the performance of an optimised memory-one strategy to that of a trained long memory-one. The longer memory strategy selected is a strategy called *Gambler*, introduced and discussed in [11]. A Gambler strategy makes probabilistic decisions based on the opponent's first moves n_1, the opponent's last moves m_1 and the player's last moves m_2. This manuscript considers Gambler($n_1 = 2, m_1 = 1, m_2 = 1$). By considering the opponent's first two moves, the opponents last move and the player's last move, there are only 16 possible outcomes that can occur. Gambler also makes a probabilistic decision of cooperating in the first move. Combining these Gambler can then be written rigorously as:

$$F : \{\text{C, D}\}^{4 \cup 1} \rightarrow (0, 1)_{\mathbb{R}} \tag{18}$$

So this can be hard coded as an element of $(0,1)^{16+1}_{\mathbb{R}}$ one probability for each outcome plus the opening move. This is a 17 dimensional problem, which is solved numerically using Bayesian optimisation. The optimisation problem is given by:

$$\begin{aligned} \max_{p} : & \sum_{i=1}^{N} {U_q}^{(i)}(F) \\ \text{such that} : & \ F \in \mathbb{R}^{17}_{[0,1]} \end{aligned} \tag{19}$$

A graphical representation of Gambler and more specifically Gambler($n_1 = 2, m_1 = 1, m_2 = 1$) is given by Figure 12.

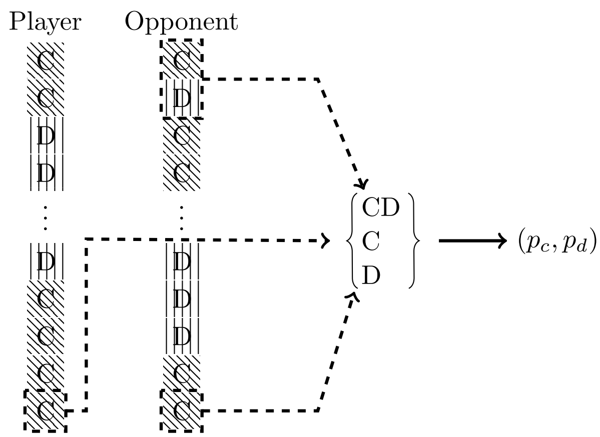


Figure 12: Graphical representation of Gambler.

Bayesian optimisation is used to numerically solve (19). Similarly to the other experiments, two random opponents are generated and the trained Gambler as well as the best response memory-one are recorded for each trial. A total of 89 trials have been recorded. The utility of both strategies for each trial is estimated, Figure 13.

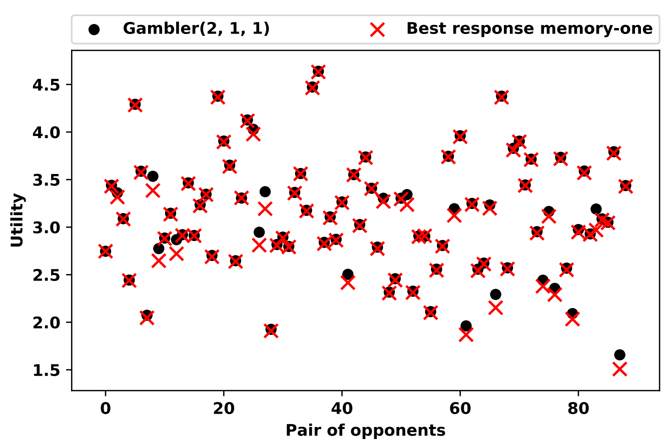


Figure 13: Utilities of Gambler and best response memory-one strategies for 89 different pair of opponents.

Though Gambler has an infinite memory (in order to remember the opponening moves of the opponent) the information the strategy considers is not significally larger than memory-one strategies. Even so, it is evident from Figure 13 that Gambler will always performs the same or betten than a best response memory one strategy, thus having a longer memory is beneficial.

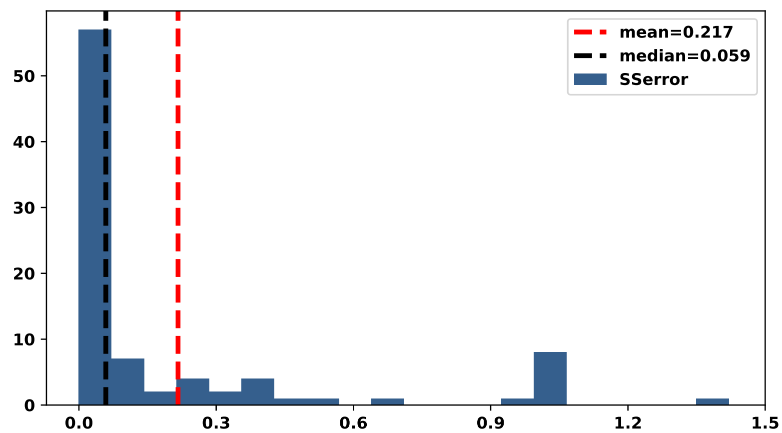


Figure 14: Distribution of sserrors for Gambler($n_1 = 2, m_1 = 1, m_2 = 1$).

	SSerror
count	89.000000
mean	0.216669
std	0.330357
min	0.000000
5%	0.010497
25%	0.058824
30%	0.058824
35%	0.058824
50%	0.058824
95%	1.058824
max	1.420346
median	0.058824
skew	2.091698
kurt	3.225540

Table 4: Summary statistics SSerror

5 Conclusion

6 Acknowledgements

A variety of software libraries have been used in this work:

- The Axelrod library for IPD simulations [1].
- The Scikit-optimize library for an implementation of Bayesian optimisation [12].
- The Matplotlib library for visualisation [13].
- The SymPy library for symbolic mathematics [21].
- The Numpy library for data manipulation [31].

References

[1] The Axelrod project developers . Axelrod: 4.4.0, April 2016.

[2] Christoph Adami and Arend Hintze. Evolutionary instability of zero-determinant strategies demonstrates that winning is not everything. *Nature communications*, 4:2193, 2013.

[3] R Axelrod and WD Hamilton. The evolution of cooperation. *Science*, 211(4489):1390–1396, 1981.

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

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

[6] Amir Beck and Marc Teboulle. A convex optimization approach for minimizing the ratio of indefinite quadratic functions over an ellipsoid. *Mathematical Programming*, 118(1):13–35, 2009.

[7] Dimitri P Bertsekas. *Constrained optimization and Lagrange multiplier methods*. Academic press, 2014.

[8] Hongyan Cai, Yanfei Wang, and Tao Yi. An approach for minimizing a quadratically constrained fractional quadratic problem with application to the communications over wireless channels. *Optimization Methods and Software*, 29(2):310–320, 2014.

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

[10] Giorgio Giorgi, Bienvenido Jiménez, and Vicente Novo. Approximate karush—kuhn—tucker condition in multiobjective optimization. *J. Optim. Theory Appl.*, 171(1):70–89, October 2016.

[11] Marc Harper, Vincent Knight, Martin Jones, Georgios Koutsovoulos, Nikoleta E. Glynatsi, and Owen Campbell. Reinforcement learning produces dominant strategies for the iterated prisoners dilemma. *PLOS ONE*, 12(12):1–33, 12 2017.

[12] Tim Head, MechCoder, Gilles Louppe, Iaroslav Shcherbatyi, fcharras, Z Vincius, cmmalone, Christopher Schrder, nel215, Nuno Campos, Todd Young, Stefano Cereda, Thomas Fan, rene rex, Kejia (KJ) Shi, Justus Schwabedal, carlosdanielcsantos, Hvass-Labs, Mikhail Pak, SoManyUsernamesTaken, Fred Callaway, Loc Estve, Lilian Besson, Mehdi Cherti, Karlson Pfannschmidt, Fabian Linzberger, Christophe Cauet, Anna Gut, Andreas Mueller, and Alexander Fabisch. scikit-optimize/scikit-optimize: v0.5.2, March 2018.

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

[14] Donald R Jones. A taxonomy of global optimization methods based on response surfaces. *Journal of global optimization*, 21(4):345–383, 2001.

[15] Gubjorn Jonsson and Stephen Vavasis. Accurate solution of polynomial equations using macaulay resultant matrices. *Mathematics of computation*, 74(249):221–262, 2005.

[16] Jeremy Kepner and John Gilbert. *Graph algorithms in the language of linear algebra*. SIAM, 2011.

[17] Vincent Knight, Owen Campbell, Marc Harper, Karol Langner, James Campbell, Thomas Campbell, Alex Carney, Martin Chorley, Cameron Davidson-Pilon, Kristian Glass, Tomáš Ehrlich, Martin Jones, Georgios Koutsovoulos, Holly Tibble, Müller Jochen, Geraint Palmer, Paul Slavin, Timothy Standen, Luis Visintini, and Karl Molden. An open reproducible framework for the study of the iterated prisoner's dilemma. 1(1), 2016.

[18] Vincent Knight, Marc Harper, Nikoleta E. Glynatsi, and Owen Campbell. Evolution reinforces cooperation with the emergence of self-recognition mechanisms: An empirical study of strategies in the moran process for the iterated prisoners dilemma. *PLOS ONE*, 13(10):1–33, 10 2018.

[19] Christopher Lee, Marc Harper, and Dashiell Fryer. The art of war: Beyond memory-one strategies in population games. *PLOS ONE*, 10(3):1–16, 03 2015.

[20] Jiawei Li, Philip Hingston, and Graham Kendall. Engineering design of strategies for winning iterated prisoner's dilemma competitions. *IEEE Transactions on Computational Intelligence and AI in Games*, 3(4):348–360, 2011.

[21] A. Meurer, C. P. Smith, M. Paprocki, O. Čertík, S. B. Kirpichev, M. Rocklin, A. Kumar, S. Ivanov, J. K. Moore, S. Singh, T. Rathnayake, S. Vig, B. E. Granger, R. P. Muller, F. Bonazzi, H. Gupta, S. Vats, F. Johansson, F. Pedregosa, M. J. Curry, A. R. Terrel, Š. Roučka, A. Saboo, I. Fernando, S. Kulal, R. Cimrman, and A. Scopatz. Sympy: symbolic computing in python. *PeerJ Computer Science*, 3, 2017.

[22] J. Močkus. On bayesian methods for seeking the extremum. In G. I. Marchuk, editor, *Optimization Techniques IFIP Technical Conference Novosibirsk, July 1–7, 1974*, pages 400–404, Berlin, Heidelberg, 1975. Springer Berlin Heidelberg.

[23] Noam Nisan, Tim Roughgarden, Eva Tardos, and Vijay V Vazirani. *Algorithmic game theory*. Cambridge University Press, 2007.

[24] Martin Nowak and Karl Sigmund. Game-dynamical aspects of the prisoner's dilemma. *Applied Mathematics and Computation*, 30(3):191–213, 1989.

[25] Martin Nowak and Karl Sigmund. The evolution of stochastic strategies in the prisoner's dilemma. *Acta Applicandae Mathematicae*, 20(3):247–265, 1990.

[26] Martin Nowak and Karl 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] Martin A Nowak and Robert M May. Evolutionary games and spatial chaos. *Nature*, 359(6398):826, 1992.

[28] Hisashi Ohtsuki, Christoph Hauert, Erez Lieberman, and Martin A Nowak. A simple rule for the evolution of cooperation on graphs and social networks. *Nature*, 441(7092):502, 2006.

[29] William H. Press and Freeman J. Dyson. Iterated prisoners dilemma contains strategies that dominate any evolutionary opponent. *Proceedings of the National Academy of Sciences*, 109(26):10409–10413, 2012.

[30] Alexander J. Stewart and Joshua B. Plotkin. Extortion and cooperation in the prisoners dilemma. *Proceedings of the National Academy of Sciences*, 109(26):10134–10135, 2012.

[31] 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.

A Appendix Tables

The memory one strategies used in the computer tournament described in [30] are given by Table 5.

	Name	Memory one representation	Reference
1	Cooperator	$(1, 1, 1, 1)$	[3]
2	Defector	$(0, 0, 0, 0)$	[3]
3	Random	$(\frac{1}{2}, \frac{1}{2}, \frac{1}{2}, \frac{1}{2})$	[3]
4	Tit for Tat	$(1, 0, 1, 0)$	[3]
5	Grudger	$(1, 0, 0, 0)$	[20]
6	Generous Tit for Tat	$(1, \frac{1}{3}, 1, \frac{1}{3})$	[25]
7	Win Stay Lose Shift	$(1, 0, 0, 1)$	[26]
8	ZDGTFT2	$(1, \frac{1}{8}, 1, \frac{1}{4})$	[30]
9	ZDExtort2	$(\frac{8}{9}, \frac{1}{2}, \frac{1}{3}, 0)$	[30]
10	Hard Joss	$(\frac{9}{10}, 0, \frac{9}{10}, 0)$	[30]

Table 5: List of strategies used in the tournament described in [30].