A theory of mind: Best responses to memory-one strategies. The limitations of extortion and restricted memory.

Nikoleta E. Glynatsi[1,*] and Vincent A. Knight[1]

[1] *Cardiff University, School of Mathematics, Cardiff, United Kingdom*
[*] *Corresponding author: Nikoleta E. Glynatsi, glynatsine@cardiff.ac.uk*

Abstract

Memory-one strategies are a set of Iterated Prisoner's Dilemma strategies that have been praised for their mathematical tractability and performance against single opponents. This manuscript investigates a theory of mind: *best response* memory-one strategies, as a multidimensional optimisation problem. We add to the literature that has shown that extortionate play is not always optimal by showing that optimal play is often not extortionate. We also provide evidence that memory-one strategies suffer from their limited memory in multi agent interactions and can be out performed by optimised strategies with longer memory.

The Prisoner's Dilemma (PD) is a two player game used in understanding the evolution of cooperative behaviour, formally introduced in [6]. 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 $T > R > P > S$ and $2R > T + S$, and the most common values used in the literature are $(R, P, S, T) = (3, 1, 0, 5)$ [4]. The numerical experiments of our manuscript are carried out using these payoff values. 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).

Memory-one strategies are a set of IPD strategies that have been studied thoroughly in the literature [24, 25], however, they have gained most of their attention when a certain subset of memory-one strategies was introduced in [26], the zero-determinant strategies (ZDs). In [27] it was stated that "Press and Dyson have fundamentally changed the viewpoint on the Prisoner's Dilemma". A special case of ZDs are extortionate strategies that choose their actions so that a linear relationship is forced between the players' score ensuring that they will always receive at least as much as their opponents. ZDs are indeed mathematically unique and are proven to be robust in pairwise interactions, however, their true effectiveness in tournaments and evolutionary dynamics has been questioned [3, 12, 13, 14, 18, 20].

In [26] the authors stated that "Only a player with a theory of mind about his opponent can do better, in which case Iterated Prisoner's Dilemma is an Ultimatum Game". The purpose of this work is to investigate the first part of this sentence, more specifically, to identify the best response strategy with a theory of mind of a given group of opponents. The outcomes of our work reinforce known results, namely that memory-one strategies must be forgiving to be evolutionarily stable [**?**, **?**] and that longer-memory strategies have a certain form of advantage over short memory strategies [**?**, **?**].

In particular, this work presents a closed form algebraic expression for the utility of a memory-one strategy against a given set of opponents, a compact method of identifying it's best response to that given set of opponents essentially: a theory of mind. The aim is to evaluate whether a best response memory-one strategy behaves in a zero-determinant way which in turn indicates whether it can be extortionate. We do this using a linear algebraic approach presented in [19]. This is done in tournaments with and without self interactions. Moreover, we introduce a framework that allows the comparison of an optimal memory-one strategy and an optimised strategy which has a larger memory.

To illustrate the analytical results obtained in this manuscript a number of numerical experiments are run. The source code for these experiments has been written in a sustainable manner [5]. It is open source (`https://github.com/Nikoleta-v3/Memory-size-in-the-prisoners-dilemma`) and tested which ensures the validity of the results. It has also been archived and can be found at [8].

1 Methods

One specific advantage of memory-one strategies is their mathematical tractability. They can be represented completely as an element of $\mathbb{R}^4_{[0,1]}$. This originates from [23] 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; both players cooperated (CC), the first player cooperated whilst the second player defected (CD), the first player defected whilst the second player cooperated (DC) and both players defected (DD). 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 [23] it was shown that it is not necessary to simulate the play of a strategy p against a memory-one opponent q. Rather this exact behaviour can be modeled as a stochastic process, and more specifically as a Markov chain whose corresponding transition matrix M is given by Eq. 2. The long run steady state probability vector v, which is the solution to $vM = v$, can be combined with the payoff matrices of Eq. 1 to give the expected payoffs for each player. More specifically, the utility for a memory-one strategy p against an opponent q, denoted as $u_q(p)$, is given by Eq. 3.

$$M = \begin{bmatrix} p_1q_1 & p_1(-q_1+1) & q_1(-p_1+1) & (-p_1+1)(-q_1+1) \\ p_2q_3 & p_2(-q_3+1) & q_3(-p_2+1) & (-p_2+1)(-q_3+1) \\ p_3q_2 & p_3(-q_2+1) & q_2(-p_3+1) & (-p_3+1)(-q_2+1) \\ p_4q_4 & p_4(-q_4+1) & q_4(-p_4+1) & (-p_4+1)(-q_4+1) \end{bmatrix} \tag{2}$$

$$u_q(p) = v \cdot (R, S, T, P). \tag{3}$$

This manuscript has explored the form of $u_q(p)$, to the authors knowledge no previous work has done this, and Theorem 1 states that $u_q(p)$ is given by a ratio of two quadratic forms [17].

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:*

$$u_q(p) = \frac{\frac{1}{2}pQp^T + cp + a}{\frac{1}{2}p\bar{Q}p^T + \bar{c}p + \bar{a}}, \tag{4}$$

where $Q, \bar{Q} \in \mathbb{R}^{4\times4}$ are square matrices defined by the transition probabilities of the opponent q_1, q_2, q_3, q_4 as follows:

$$Q = \begin{bmatrix} 0 & -(q_1-q_3)(Pq_2-P-Tq_4) & (q_1-q_2)(Pq_3-Sq_4) & (q_1-q_4)(Sq_2-S-Tq_3) \\ -(q_1-q_3)(Pq_2-P-Tq_4) & 0 & (q_2-q_3)(Pq_1-P-Rq_4) & -(q_3-q_4)(Rq_2-R-Tq_1+T) \\ (q_1-q_2)(Pq_3-Sq_4) & (q_2-q_3)(Pq_1-P-Rq_4) & 0 & (q_2-q_4)(Rq_3-Sq_1+S) \\ (q_1-q_4)(Sq_2-S-Tq_3) & -(q_3-q_4)(Rq_2-R-Tq_1+T) & (q_2-q_4)(Rq_3-Sq_1+S) & 0 \end{bmatrix}, \tag{5}$$

$$\bar{Q} = \begin{bmatrix} 0 & -(q_1-q_3)(q_2-q_4-1) & (q_1-q_2)(q_3-q_4) & (q_1-q_4)(q_2-q_3-1) \\ -(q_1-q_3)(q_2-q_4-1) & 0 & (q_2-q_3)(q_1-q_4-1) & (q_1-q_2)(q_3-q_4) \\ (q_1-q_2)(q_3-q_4) & (q_2-q_3)(q_1-q_4-1) & 0 & -(q_2-q_4)(q_1-q_3-1) \\ (q_1-q_4)(q_2-q_3-1) & (q_1-q_2)(q_3-q_4) & -(q_2-q_4)(q_1-q_3-1) & 0 \end{bmatrix}. \tag{6}$$

c and $\bar{c} \in \mathbb{R}^{4\times1}$ are similarly defined by:

$$c = \begin{bmatrix} q_1(Pq_2 - P - Tq_4) \\ -(q_3-1)(Pq_2 - P - Tq_4) \\ -Pq_1q_2 + Pq_2q_3 + Pq_2 - Pq_3 + Rq_2q_4 - Sq_2q_4 + Sq_4 \\ -Rq_2q_4 + Rq_4 + Sq_2q_4 - Sq_2 - Sq_4 + S + Tq_1q_4 - Tq_3q_4 + Tq_3 - Tq_4 \end{bmatrix}, \tag{7}$$

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

and the constant terms $a, \bar{a}$ are defined as $a = -Pq_2 + P + Tq_4$ and $\bar{a} = -q_2 + q_4 + 1$.

The proof of Theorem 1 is given in Appendix 5.1. 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 N opponents is given by:

$$\frac{1}{N}\sum_{i=1}^{N} {u_q}^{(i)}(p) = \frac{\frac{1}{N}\sum\limits_{i=1}^{N}\left(\frac{1}{2}pQ^{(i)}p^T + c^{(i)}p + a^{(i)}\right) \prod\limits_{\substack{j=1 \\ j\neq i}}^{N}\left(\frac{1}{2}p\bar{Q}^{(j)}p^T + \bar{c}^{(j)}p + \bar{a}^{(j)}\right)}{\prod\limits_{i=1}^{N}\left(\frac{1}{2}p\bar{Q}^{(i)}p^T + \bar{c}^{(i)}p + \bar{a}^{(i)}\right)}. \tag{9}$$

Eq. (9) is the average score (using Eq. (4)) against the set of opponents.

Estimating the utility of a memory-one strategy against any number of opponents without simulating the interactions is the main result used in the rest of this manuscript. It will be used to obtain best response memory-one strategies in tournaments with and without self interactions in order to explore the limitations of extortion and restricted memory.

2 Results

Here we define *memory-one best response* strategies as a multi dimensional optimisation problem given by:

$$\begin{aligned} \max_{p} : & \sum_{i=1}^{N} {u_q}^{(i)}(p) \\ \text{such that} : & \ p \in \mathbb{R}_{[0,1]} \end{aligned} \tag{10}$$

Optimising this particular ratio of quadratic forms is not trivial. It can be verified empirically for the case of a single opponent that there exists at least one point for which the definition of concavity does not hold. The non concavity of $u(p)$ indicates multiple local optimal points. This is also intuitive. The best response against a cooperator, $q = (1, 1, 1, 1)$, is a defector $p^* = (0, 0, 0, 0)$. The strategies $p = (\frac{1}{2}, 0, 0, 0)$ and $p = (\frac{1}{2}, 0, 0, \frac{1}{2})$ are also best responses. The approach taken here is to introduce a compact way of constructing the discrete candidate set of all local optimal points, and evaluating the objective function Eq. 9. This gives the best response memory-one strategy. The approach is given in Theorem 2.

Theorem 2. *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 given 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:

$$S_q = \left\{ p \in \mathbb{R}^4 \left| \begin{array}{l} \bullet \quad p_j \in \{0, 1\} \quad \textit{and} \quad \dfrac{d}{dp_k} \displaystyle\sum_{i=1}^{N} u_q^{(i)}(p) = 0 \\ \qquad\qquad\qquad \textit{for all} \quad j \in J \quad \& \quad k \in K \quad \textit{for all} \quad J, K \\ \qquad\qquad\qquad \textit{where} \quad J \cap K = \emptyset \quad \textit{and} \quad J \cup K = \{1, 2, 3, 4\}. \\ \bullet \quad p \in \{0, 1\}^4 \end{array} \right. \right\}. \tag{11}$$

Note that there is no immediate way to find the zeros of $\frac{d}{dp} \sum\limits_{i=1}^{N} u_q(p)$ where,

$$\frac{d}{dp} \sum_{i=1}^{N} {u_q}^{(i)}(p) = \sum_{i=1}^{N} \frac{\left(pQ^{(i)} + c^{(i)}\right)\left(\frac{1}{2}p\bar{Q}^{(i)}p^T + \bar{c}^{(i)}p + \bar{a}^{(i)}\right)}{\left(\frac{1}{2}p\bar{Q}^{(i)}p^T + \bar{c}^{(i)}p + \bar{a}^{(i)}\right)^2} - \frac{\left(p\bar{Q}^{(i)} + \bar{c}^{(i)}\right)\left(\frac{1}{2}pQ^{(i)}p^T + c^{(i)}p + a^{(i)}\right)}{\left(\frac{1}{2}p\bar{Q}^{(i)}p^T + \bar{c}^{(i)}p + \bar{a}^{(i)}\right)^2} \tag{12}$$

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

$$\sum_{i=1}^{N} \left(pQ^{(i)} + c^{(i)}\right)\left(\frac{1}{2}p\bar{Q}^{(i)}p^T + \bar{c}^{(i)}p + \bar{a}^{(i)}\right) - \left(p\bar{Q}^{(i)} + \bar{c}^{(i)}\right)\left(\frac{1}{2}pQ^{(i)}p^T + c^{(i)}p + a^{(i)}\right) = 0, \quad \textit{while} \tag{13}$$

$$\sum_{i=1}^{N} \frac{1}{2}p\bar{Q}^{(i)}p^T + \bar{c}^{(i)}p + \bar{a}^{(i)} \neq 0. \tag{14}$$

The proof of Theorem 2 is given in Appendix 5.2. Finding best response memory-one strategies is analytically feasible using the formulation of Theorem 2 and resultant theory [16]. However, for large systems building the resultant becomes intractable. As a result, best responses will be estimated heuristically using a numerical method, suitable for problems with local optima, called Bayesian optimisation [22].

In several evolutionary settings such as Moran Processes self interactions are key. Previous work has identified interesting results such as the appearance of self recognition mechanisms when training strategies using evolutionary algorithms in Moran processes [18]. This aspect of reinforcement learning can be done for best response memory-one strategies by incorporating the strategy itself in the objective function as shown in Eq. 24. K is the number of self interactions that will take place.

$$\begin{aligned} \max_p : & \ \frac{1}{N} \sum_{i=1}^{N} {u_q}^{(i)}(p) + K u_p(p) \\ \text{such that} : & \ p \in \mathbb{R}_{[0,1]} \end{aligned} \tag{15}$$

For determining the memory-one best response with self interactions, an algorithmic approach is considered, called *best response dynamics*. The best response dynamics approach used in this manuscript is given by Algorithm 1.

Algorithm 1: Best response dynamics Algorithm

$p^{(t)} \leftarrow (1, 1, 1, 1)$;
while $p^{(t)} \neq p^{(t-1)}$ **do**
  $p^{(t+1)} = \text{argmax} \frac{1}{N} \sum\limits_{i=1}^{N} {u_q}^{(i)}(p^{(t)}) + K u_{p^{(t)}}(p^{(t)})$;
end

Using this approach it would be possible to create a memory-one best response strategy that updates on every generation of a Moran process to recalculate the optimal behaviour given the population. This extension of the "theory of mind" is an interesting avenue for future work.

In multi opponent settings, where the payoffs matter, strategies trying to exploit their opponents will suffer. Compared to ZDs, best response memory-one strategies, which have a theory of mind of their opponents, utilise their behaviour in order to gain the most from their interactions. The question that arises then is whether best response strategies are optimal because they behave in an extortionate way.

The results of this section use Bayesian optimisation to generate a data set of best response memory-one strategies for $N = 2$ opponents. The data set is available at [7]. It contains a total of 1000 trials corresponding to 1000 different instances of a best response strategy in tournaments with and without self interactions. For each trial a set of 2 opponents is randomly generated and the memory-one best response against them is found. In order to investigate whether best responses behave in an extortionate matter the SSE method described in [19] is used. More specifically, in [19] the point x^*, in the space of memory-one strategies, that is the nearest extortionate strategy to a given strategy p is given by,

$$x^* = \left(C^T C\right)^{-1} C^T \bar{p} \tag{16}$$

where $\bar{p} = (p_1 - 1, p_2 - 1, p_3, p_4)$ and

$$
C = \begin{bmatrix} R-P & R-P \\ S-P & T-P \\ T-P & S-P \\ 0 & 0 \end{bmatrix}. \tag{17}
$$

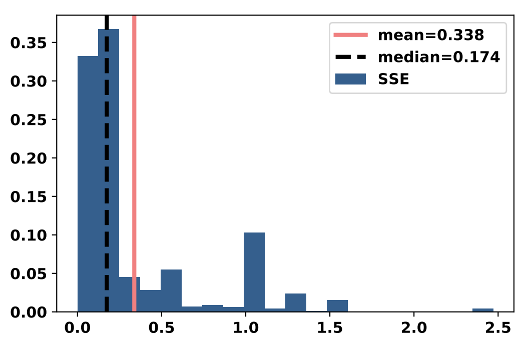
Once this closest ZDs is found, the squared norm of the remaining error is referred to as sum of squared errors of prediction (SSE):

$$
\text{SSE} = \bar{p}^T\bar{p} - \bar{p}C\left(C^TC\right)^{-1}C^T\bar{p} = \bar{p}^T\bar{p} - \bar{p}Cx^* \tag{18}
$$

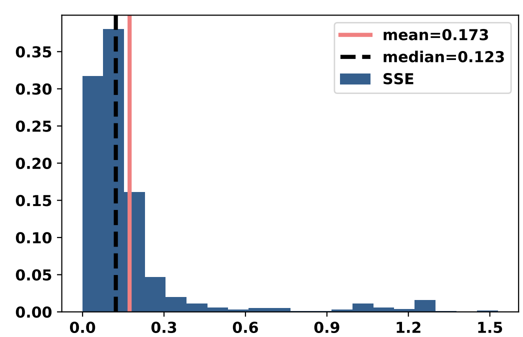
Thus, SSE is defined as how far a strategy is from behaving as a ZD. A high SSE implies a non extortionate behaviour. The distributions of SSE for the best response in tournaments ($N = 2$) with and without self interactions (with $K = 1$) are given in Figure 1. Moreover, a statistical summary of the SSE distributions is given in Table 1.

	mean	std	5%	50%	95%	max	median	skew	kurt
Tournament without self interactions	0.34	0.40	0.028	0.17	1.05	2.47	0.17	1.87	3.60
Tournament with self interactions	0.17	0.23	0.01	0.12	0.67	1.53	0.12	3.42	1.92

Table 1: SSE of best response memory-one when $N = 2$



(a) SEE distribution for best response in tournaments without self interactions.



(b) SEE distribution for best response in tournaments with self interactions.

Figure 1: SEE distributions for best response in tournaments without and with self interactions.

For the best response in tournaments that do not include self interactions the distribution of SSE is skewed to the left, indicating that the best response does exhibit ZDs behaviour and so could be extortionate, however, the best response is not uniformly a ZDs. A positive measure of skewness and kurtosis, and a mean of 0.34 indicate a heavy tail to the right. Therefore, in several cases the strategy is not trying to extort its opponents. In [11] a similar behaviour is refereed to as the *partner strategy*. The partner strategy aims to share the payoff for mutual cooperation, but it is ready to fight back when being exploited. The partner strategy was designed, but the best responses which are defined by their opponents seem to exhibit the same behaviour.

Similarly, when considering self interactions, the distribution of SSE for the best response strategy has skewness and kurtosis that indicate a heavy tail to the right. This indicates that evolutionary stable memory-one strategies need to more adaptable than a ZDs, and aim for mutual cooperation as well as exploitation which is in line with the results of [11] where their strategy was designed to adapt and was shown to be evolutionary stable. The findings of this work show that an optimal strategy acts in the same way.

The difference between best responses in tournaments without and with self interactions is further explored by Fig. 2. Though, no statistically significant differences have been found, from Fig. 2, it seems that the best response that incorporate self interactions has a higher median p_2; which corresponds to the probability of cooperating after receiving a defection. Thus, they are more likely to forgive after being tricked. This is due to the fact that they could be playing against themselves, and they need to be able to forgive so that future cooperation can occur.

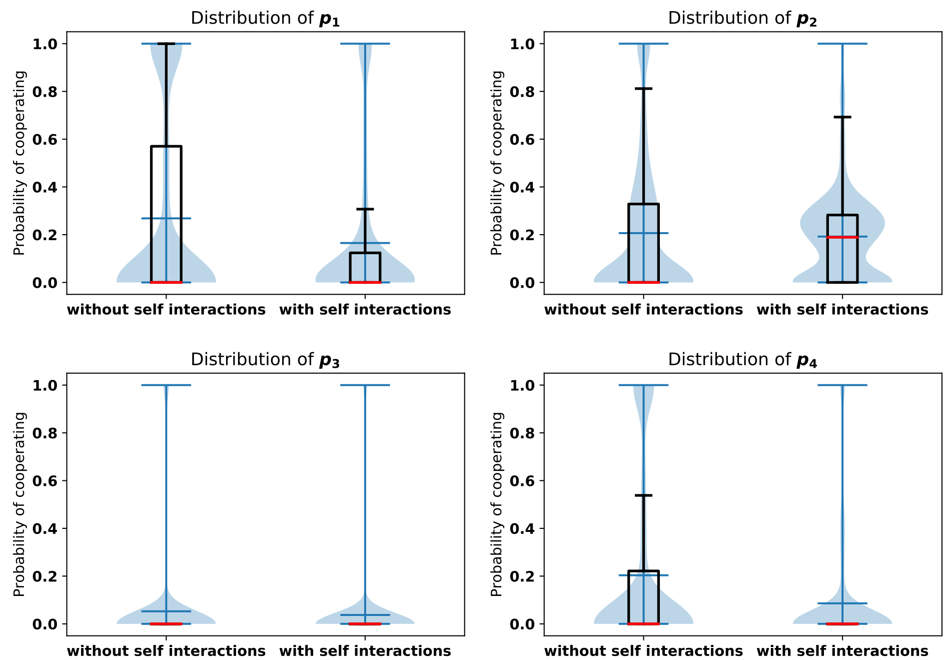


Figure 2: Distributions of p^* for best responses in tournaments and evolutionary settings. The medians, denoted as $\bar{p}^*$, for tournaments are $\bar{p}^* = (0, 0, 0, 0)$, and for evolutionary settings $\bar{p}^* = (0, 0.19, 0, 0)$.

To investigate the above findings more formally a Moran process will be considered. In a population of N total individuals of two types, the probability that K individuals of a given type can take over the population, the fixation probability x_K is given by:

$$x_k = \frac{1 + \sum_{j=1}^{K-1} \prod_{i=1}^{j} \gamma_i}{1 + \sum_{j=1}^{N-1} \prod_{i=1}^{j} \gamma_i}$$

where:

$$\gamma_i = \frac{p_{K,K-1}}{p_{K,K+1}}.$$

In the case considered here K is the number of best response players in the population, so that $N-K$ is the number of opponents.

The difference proposed here is to allow the best response player to act dynamically: adjusting their probability vector at every generation. In essence using the theory of mind to find the best response to not only the opponent but also the distribution of the population. Thus for every value of K there is a different best response player.

In the case of the best response player the transition probabilities depend on the payoff matrix $A^{(K)}$ where:

- $A_{11}^{(K)}$ is the long run utility of the best response player against itself.
- $A_{12}^{(K)}$ is the long run utility of the best response player against the opponent.
- $A_{11}^{(K)}$ is the long run utility of the opponent against the best response player.
- $A_{11}^{(K)}$ is the long run utility of the opponent against itself.

The matrix $A^{(K)}$ is calculated using Eq. (4) for each value of K once the best response dynamics algorithm has calculated the best response player.

Using this, the total utilities/fitnesses for each player can be written down:

$$f_1^{(K)} = (K-1)A_{11}^{(K)} + (N-K)A_{12}^{(K)}$$

$$f_2^{(K)} = (K)A_{21}^{(K)} + (N-K-1)A_{22}^{(K)}$$

where $f_1^{(K)}$ is the fitness of the best response player, and $f_2^{(K)}$ is the fitness of the opponent.

Using this:

$$p_{K,K-1} = \frac{(N-K)f_2^{(K)}}{Kf_1^{(K)} + (N-K)f_2^{(K)}}\frac{K}{N}$$

and:

$$p_{K,K+1} = \frac{Kf_1^{(K)}}{Kf_1^{(K)} + (N-K)f_2^{(K)}}\frac{(N-K)}{N}$$

which are all that are required to calculate x_K.

Figure **??** shows the results of an analysis of x_K for dynamically updating players. This is obtained over data points. The ratio shows a relatively high performance compared to a non dynamic best response strategy the mean x_K over all values of K and all experiments is given by and in some cases this dynamic updating results in a 25% increase in the absorption probability. As denoted before it is clear that the best response strategy does not have a low SSE this is further compondounded by the ratio being above one showing that in many cases the dynamic strategy benefits from its ability to adapt.

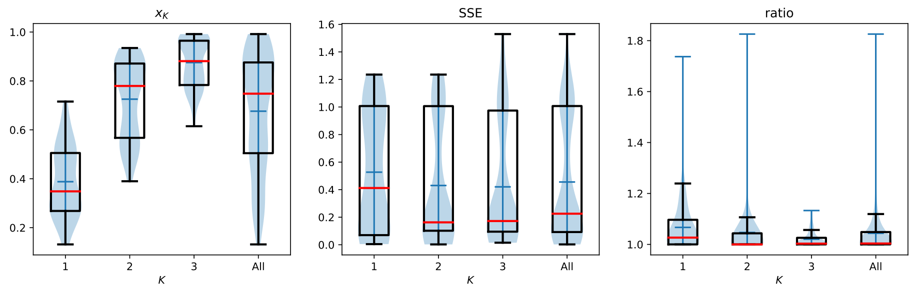


Figure 3: Results for the best response player in a dynamic Moran process. The ratio is taken as the ratio of x_k of the dynamically updating player to the fixation probability of a best response player that does not update as the population density changes.

The other main finding presented in [**?**] was that short memory of the strategies was all that was needed. We argue that the second limitation of ZDs in multi opponent interactions is that of their restricted memory. To demonstrate the effectiveness of memory in the IPD we explore a best response longer-memory strategy against a given set of memory-one opponents, and compare it's performance to that of a memory-one best response.

In [9], a strategy called *Gambler* which makes probabilistic decisions based on the opponent's n_1 first moves, the opponent's m_1 last moves and the player's m_2 last moves was introduced. In this manuscript Gambler with parameters: $n_1 = 2, m_1 = 1$ and $m_2 = 1$ is used as a longer-memory strategy. By considering the opponent's first two moves, the opponents last move and the player's last move, there are only 16 $(4 \times 2 \times 2)$ possible outcomes that can occur, furthermore, Gambler also makes a probabilistic decision of cooperating in the opening move. Thus, Gambler is a function $f : \{\text{C, D}\} \to [0, 1]_{\mathbb{R}}$. This can be hard coded as an element of $[0, 1]_{\mathbb{R}}^{16+1}$, one probability for each outcome plus the opening move. Hence, compared to Eq. 24, finding an optimal Gambler is a 17 dimensional problem given by:

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

Note that Eq. 9 can not be used here for the utility of Gambler, and actual simulated players are used. This is done using [1] with 500 turns and 200 repetitions, moreover, Eq. 19 is solved numerically using Bayesian optimisation.

Similarly to previous sections, a large data set has been generated with instances of an optimal Gambler and a memory-one best response, available at [7]. Estimating a best response Gambler (17 dimensions) is computational more expensive compared to a best response memory-one (4 dimensions). As a result, the analysis of this section is based on a total of 152 trials. As before, for each trial $N = 2$ random opponents have been selected.

The ratio between Gambler's utility and the best response memory-one strategy's utility has been calculated and its distribution in given in Fig. 3. It is evident from Fig. 3 that Gambler always performs as well as the best response memory-one strategy and often performs better. There are no points where the ratio value is less than 1, thus Gambler never performed less than the best response memory-one strategy and in places outperforms it. However, against two memory-one opponents Gambler's performance is better than the optimal memory-one strategy. This is evidence that in the case of multiple opponents, having a shorter memory is limiting.

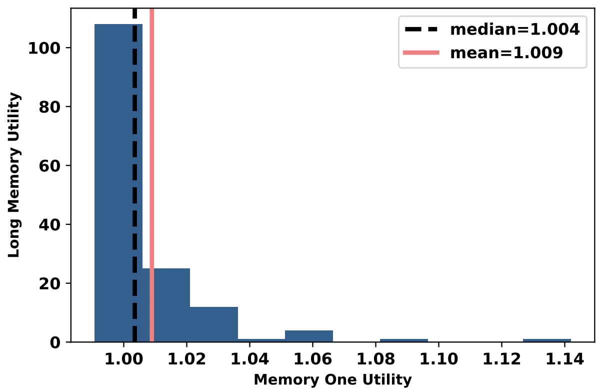


Figure 4: The ratio between the utilities of Gambler and best response memory-one strategy for 152 different pair of opponents.

3 Discussion

This manuscript has considered *best response* strategies in the IPD game, and more specifically, *memory-one best responses*. It has proven that:

- The utility of a memory-one strategy against a set of memory-one opponents can be written as a sum of ratios of quadratic forms (Theorem 1).
- There is a compact way of identifying a memory-one best response to a group of opponents through a search over a discrete set (Theorem 2).

Note that Theorem 2 which does not only have game theoretic novelty, but also the mathematical novelty of solving quadratic ratio optimisation problems where the quadratics are non concave.

Moreover Theorem 1, allows us to obtain a condition for which in an environment of memory-one opponents defection is the stable choice, based only on the coefficients of the opponents, as stated in Lemma 3.

Lemma 3. *In a tournament of N players $\{q^{(1)}, q^{(2)}, \dots, q^{(N)}\}$ for $q^{(i)} \in \mathbb{R}^4_{[0,1]}$ defection is stable if the transition probabilities of the opponents satisfy conditions Eq. 30 and Eq. 31.*

$$\sum_{i=1}^{N} (c^{(i)T} \bar{a}^{(i)} - \bar{c}^{(i)T} a^{(i)}) \leq 0 \tag{20}$$

while,

$$\sum_{i=1}^{N} \bar{a}^{(i)} \neq 0 \tag{21}$$

The proof of Lemma 3 is given in Appendix **??** and a numerical simulation demonstrating the result is given in Fig. 4.

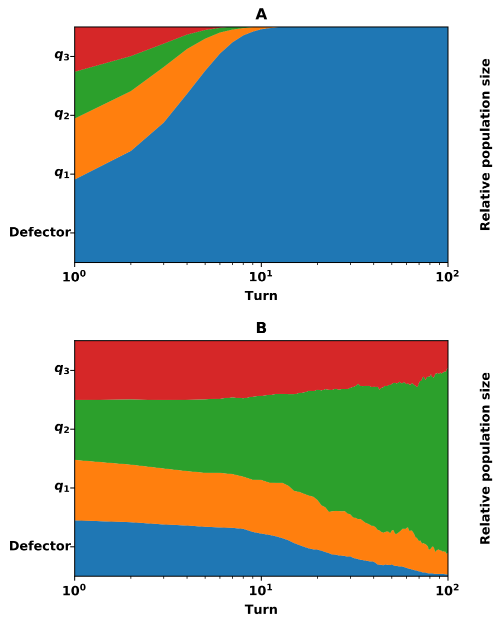


Figure 5: A. For $q_1 = (0.22199, 0.87073, 0.20672, 0.91861)$, $q_2 = (0.48841, 0.61174, 0.76591, 0.51842)$ and $q_3 = (0.2968, 0.18772, 0.08074, 0.73844)$, Eq. 30 and Eq. 31 hold and Defector takes over the population. B. For $q_1 = (0.96703, 0.54723, 0.97268, 0.71482)$, $q_2 = (0.69773, 0.21609, 0.97627, 0.0062)$ and $q_3 = (0.25298, 0.43479, 0.77938, 0.19769)$, Eq. 30 fails and Defector does not take over the population. These results have been obtained by using [1] an open source research framework for the study of the IPD.

The empirical results have shown that the performance and the evolutionary stability of memory-one strategies rely on adaptability and not on extortion, and that memory-one strategies' performance is limited by their memory in cases where they interact with multiple opponents.

These results were mainly to investigate the behaviour of memory-one strategies and their limitations. A large data set which contained best responses in tournaments whilst including or not self interactions for $N = 2$ was generated and is archived in [7]. This allowed us to investigate their respective behaviours, and whether it was extortionate acts that made them the most favorable strategies. It was shown that it was not extortion but adaptability that allowed the strategies to gain the most from their interactions. In settings with self interactions there is some evidence that it is more likely to forgive after being tricked.

All the empirical results presented in this manuscript have been for the case of $N = 2$. In future work we would consider larger values of N, however, we believe that for larger values of N the results that have been presented here would only be more evident. In addition, we would investigate potential theoretical results for the best responses dynamics algorithm discussed. Another interesting avenue would be to study the Moran process with a dynamically updating best response.

By specifically exploring the entire space of memory-one strategies to identify the best strategy for a variety of situations, this work adds to the literature casting doubt on the effectiveness of ZDs, highlights the importance of adaptability and provides a framework for the continued understanding of these important questions.

4 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 [10].
- The Matplotlib library for visualisation [15].
- The SymPy library for symbolic mathematics [21].
- The Numpy library for data manipulation [28].

References

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

[2] Karim M Abadir and Jan R Magnus. *Matrix algebra*, volume 1. Cambridge University Press, 2005.

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

[4] Robert Axelrod and William D. Hamilton. The evolution of cooperation. *Science*, 211(4489):1390–1396, 1981.

[5] Fabien CY Benureau and Nicolas P Rougier. Re-run, repeat, reproduce, reuse, replicate: transforming code into scientific contributions. *Frontiers in neuroinformatics*, 11:69, 2018.

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

[7] Nikoleta E. Glynatsi. Raw data for: "Stability of defection, optimisation of strategies and the limits of memory in the Prisoner's Dilemma.", September 2019.

[8] Nikoleta E. Glynatsi and Vince Knight. Nikoleta-v3/Memory-size-in-the-prisoners-dilemma: initial release. November 2019.

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

[10] Tim Head, MechCoder, Gilles Louppe, Iaroslav Shcherbatyi, fcharras, Zé Vinícius, cmmalone, Christopher Schröder, nel215, Nuno Campos, Todd Young, Stefano Cereda, Thomas Fan, rene rex, Kejia (KJ) Shi, Justus Schwabedal, carlosdanielcsantos, Hvass-Labs, Mikhail Pak, SoManyUsernamesTaken, Fred Callaway, Loïc Estève, 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.

[11] Christian Hilbe, Krishnendu Chatterjee, and Martin A Nowak. Partners and rivals in direct reciprocity. *Nature human behaviour*, 2(7):469–477, 2018.

[12] Christian Hilbe, Martin Nowak, and Karl Sigmund. Evolution of extortion in iterated prisoner's dilemma games. *Proceedings of the National Academy of Sciences*, page 201214834, 2013.

[13] Christian Hilbe, Martin Nowak, and Arne Traulsen. Adaptive dynamics of extortion and compliance. *PLOS ONE*, 8(11):1–9, 11 2013.

[14] Christian Hilbe, Arne Traulsen, and Karl Sigmund. Partners or rivals? strategies for the iterated prisoner's dilemma. *Games and Economic Behavior*, 92:41 – 52, 2015.

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

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

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

[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 prisoner's dilemma. *PLOS ONE*, 13(10):1–33, 10 2018.

[19] Vincent A. Knight, Marc Harper, Nikoleta E. Glynatsi, and Jonathan Gillard. Recognising and evaluating the effectiveness of extortion in the iterated prisoner's dilemma. *CoRR*, abs/1904.00973, 2019.

[20] 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.

[21] Aaron Meurer, Christopher Smith, Mateusz Paprocki, Ondřej Čertík, Sergey Kirpichev, Matthew Rocklin, AMiT Kumar, Sergiu Ivanov, Jason Moore, Sartaj Singh, Thilina Rathnayake, Sean Vig, Brian Granger, Richard Muller, Francesco Bonazzi, Harsh Gupta, Shivam Vats, Fredrik Johansson, Fabian Pedregosa, and Anthony Scopatz. Sympy: symbolic computing in python. *PeerJ Computer Science*, 3, 2017.

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

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

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

[25] 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.

[26] William H. Press and Freeman 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.

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

[28] Stefan Van Der Walt, S. Chris Colbert, and Gael Varoquaux. The NumPy array: a structure for efficient numerical computation. *Computing in Science & Engineering*, 13(2):22–30, 2011.

5 Appendix

5.1 Theorem 1 Proof

The utility of a memory one player p against an opponent q, $u_q(p)$, can be written as a ratio of two quadratic forms on R^4.

Proof. It was discussed that $u_q(p)$ it is the product of the steady state vector v and the PD payoffs,

$$u_q(p) = v \cdot (R, S, T, P).$$

The steady state vector which is the solution to $vM = v$ is given by

$$v = \Bigg[\frac{p_2p_3(q_2q_4 - q_3q_4) + p_2p_4(q_2q_3 - q_2q_4 - q_3 + q_4) + p_3p_4(-q_2q_3 + q_3q_4) - p_3q_2q_4 + p_4q_4(q_2 - 1)}{\bar{v}},$$
$$\frac{p_1p_3(q_1q_4 - q_2q_4) + p_1p_4(-q_1q_2 + q_1 + q_2q_4 - q_4) + p_3p_4(q_1q_2 - q_1q_4 - q_2 + q_4) + p_3q_4(q_2 - 1) - p_4q_2(q_4 + 1) + p_4(q_4 - 1)}{\bar{v}},$$
$$\frac{-p_1p_2(q_1q_4 - q_3q_4) - p_1p_4(-q_1q_3 + q_3q_4) + p_1q_1q_4 - p_2p_4(q_1q_3 - q_1q_4 - q_3 + q_4) - p_2q_4(q_3 + 1) - p_4q_4(q_1 + q_3) - p_4(q_3 + q_4) - q_4}{\bar{v}},$$
$$\frac{p_1p_2(q_1q_2 - q_1 - q_2q_3 + q_3) + p_1p_3(-q_1q_3 + q_2q_3) - p_1q_1(q_2 + 1) + p_2p_3(-q_1q_2 + q_1q_3 + q_2 - q_3) + p_2(q_3q_2 - q_2 - q_3 - 1) + p_3(q_1q_2 - q_3q_2 - q_2 - q_3) + q_2 - 1}{\bar{v}}\Bigg],$$

where,

$$\begin{aligned}\bar{v} = \; & p_1p_2(q_1q_2 - q_1q_4 - q_1 - q_2q_3 + q_3q_4 + q_3) - p_1p_3(q_1q_3 - q_1q_4 - q_2q_3 + q_2q_4) - p_1p_4(q_1q_2 - q_1q_3 - q_1 - q_2q_4 + q_3q_4 + q_4) - \\ & p_1q_1(q_2 + q_4 + 1) + p_2p_3(-q_1q_2 + q_1q_3 + q_2q_4 + q_2 - q_3q_4 - q_3) + p_2p_4(-q_1q_3 + q_1q_4 + q_2q_3 - q_2q_4) + p_2q_2(q_3 - 1) - p_2q_3(q_4 - 1) + \\ & p_2(q_4 + 1) + p_3p_4(q_1q_2 - q_1q_4 - q_2q_3 - q_2 + q_3q_4 + q_4) + p_3q_2q_1(-p_3 - 1) + p_3(q_3 - q_4) - p_4(q_1q_4 + q_2 + q_3q_4 - q_3 + q_4 - 1) + \\ & q_2 - q_4 - 1\end{aligned}$$

The dot product of $v \cdot (R, S, T, P)$ gives,

$$\begin{aligned}u_q(p) = & \frac{R\left(p_2p_3(q_2q_4 - q_3q_4) + p_2p_4(q_2q_3 - q_2q_4 - q_3 + q_4) + p_3p_4(-q_2q_3 + q_3q_4) - p_3q_2q_4 + p_4q_4(q_2 - 1)\right)}{\bar{v}} + \\ & \frac{S\left(p_1p_3(q_1q_4 - q_2q_4) + p_1p_4(-q_1q_2 + q_1 + q_2q_4 - q_4) + p_3p_4(q_1q_2 - q_1q_4 - q_2 + q_4) + p_3q_4(q_2 - 1) - p_4q_2(q_4 + 1) + p_4(q_4 - 1)\right)}{\bar{v}} + \\ & \frac{T\left(-p_1p_2(q_1q_4 - q_3q_4) - p_1p_4(-q_1q_3 + q_3q_4) + p_1q_1q_4 - p_2p_4(q_1q_3 - q_1q_4 - q_3 + q_4) - p_2q_4(q_3 + 1) - p_4q_4(q_1 + q_3) - p_4(q_3 + q_4) - q_4\right)}{\bar{v}} + \\ & \frac{P\left(p_1(p_2(q_1q_2 - q_1 - q_2q_3 + q_3) + p_3(-q_1q_3 + q_2q_3) - q_1(q_2 + 1)) + p_2p_3((-q_1q_2 + q_1q_3 + q_2 - q_3) + (q_3q_2 - q_2 - q_3 - 1))\right)}{\bar{v}} + \\ & \frac{P\left(p_3(q_1q_2 - q_3q_2 - q_2 - q_3) + q_2 - 1\right)}{\bar{v}} \implies\end{aligned}$$

$$u_q(p) = \left(\frac{\begin{array}{c}-p_1p_2(q_1 - q_3)(Pq_2 - P - Tq_4) + p_1p_3(q_1 - q_2)(Pq_3 - Sq_4) + p_1p_4(q_1 - q_4)(Sq_2 - S - Tq_3) + p_2p_3(q_2 - q_3)(Pq_1 - P - Rq_4) - \\ p_2p_4(q_3 - q_4)(Rq_2 - R - Tq_1 + T) + p_3p_4(q_2 - q_4)(Rq_3 - Sq_1 + S) + p_1q_1(Pq_2 - P - Tq_4) - p_2(q_3 - 1)(Pq_2 - P - Tq_4) + \\ p_3(-Pq_1q_2 + Pq_2q_3 + Pq_2 - Pq_3 + Rq_2q_4 - Sq_2q_4 + Sq_4) + p_4(-Rq_2q_4 + Rq_4 + Sq_2q_4 - Sq_2 - Sq_4 + S + Tq_1q_4 - Tq_3q_4 + Tq_3 - Tq_4) \\ -Pq_2 + P + Tq_4\end{array}}{\begin{array}{c}p_1p_2(q_1q_2 - q_1q_4 - q_1 - q_2q_3 + q_3q_4 + q_3) + p_1p_3(-q_1q_3 + q_1q_4 + q_2q_3 - q_2q_4) + p_1p_4(-q_1q_2 + q_1q_3 + q_1 + q_2q_4 - q_3q_4 - q_4) + \\ p_2p_3(-q_1q_2 + q_1q_3 + q_2q_4 + q_2 - q_3q_4 - q_3) + p_2p_4(-q_1q_3 + q_1q_4 + q_2q_3 - q_2q_4) + p_3p_4(q_1q_2 - q_1q_4 - q_2q_3 - q_2 + q_3q_4 + q_4) + \\ p_1(-q_1q_2 + q_1q_4 + q_1) + p_2(q_2q_3 - q_2 - q_3q_4 - q_3 + q_4 + 1) + p_3(q_1q_2 - q_2q_3 - q_2 + q_3 - q_4) + p_4(-q_1q_4 + q_2 + q_3q_4 - q_3 + q_4 - 1) + \\ q_2 - q_4 - 1\end{array}}\right).$$

Let us consider the numerator of $u_q(p)$. The cross product terms p_ip_j are given by,

$$-p_1p_2(q_1-q_3)(Pq_2-P-Tq_4)+p_1p_3(q_1-q_2)(Pq_3-Sq_4)+p_1p_4(q_1-q_4)(Sq_2-S-Tq_3)+$$
$$p_2p_3(q_2-q_3)(Pq_1-P-Rq_4)-p_2p_4(q_3-q_4)(Rq_2-R-Tq_1+T)+p_3p_4(q_2-q_4)(Rq_3-Sq_1+S)$$

This can be re written in a matrix format given by Eq. 20.

$$(p_1,p_2,p_3,p_4)\frac{1}{2}\begin{bmatrix} 0 & -(q_1-q_3)(Pq_2-P-Tq_4) & (q_1-q_2)(Pq_3-Sq_4) & (q_1-q_4)(Sq_2-S-Tq_3) \\ -(q_1-q_3)(Pq_2-P-Tq_4) & 0 & (q_2-q_3)(Pq_1-P-Rq_4) & -(q_3-q_4)(Rq_2-R-Tq_1+T) \\ (q_1-q_2)(Pq_3-Sq_4) & (q_2-q_3)(Pq_1-P-Rq_4) & 0 & (q_2-q_4)(Rq_3-Sq_1+S) \\ (q_1-q_4)(Sq_2-S-Tq_3) & -(q_3-q_4)(Rq_2-R-Tq_1+T) & (q_2-q_4)(Rq_3-Sq_1+S) & 0 \end{bmatrix}\begin{pmatrix} p_1 \\ p_2 \\ p_3 \\ p_4 \end{pmatrix} \tag{22}$$

Similarly, the linear terms are given by,

$$p_1q_1(Pq_2-P-Tq_4)-p_2(q_3-1)(Pq_2-P-Tq_4)+p_3(-Pq_1q_2+Pq_2q_3+Pq_2-Pq_3+Rq_2q_4-Sq_2q_4+Sq_4)+$$
$$p_4(-Rq_2q_4+Rq_4+Sq_2q_4-Sq_2-Sq_4+S+Tq_1q_4-Tq_3q_4+Tq_3-Tq_4)$$

and the expression can be written using a matrix format as Eq. 21.

$$(p_1,p_2,p_3,p_4)\begin{bmatrix} q_1(Pq_2-P-Tq_4) \\ -(q_3-1)(Pq_2-P-Tq_4) \\ -Pq_1q_2+Pq_2q_3+Pq_2-Pq_3+Rq_2q_4-Sq_2q_4+Sq_4 \\ -Rq_2q_4+Rq_4+Sq_2q_4-Sq_2-Sq_4+S+Tq_1q_4-Tq_3q_4+Tq_3-Tq_4 \end{bmatrix} \tag{23}$$

Finally, the constant term of the numerator, which is obtained by substituting $p=(0,0,0,0)$, is given by Eq. 22.

$$-Pq_2+P+Tq_4 \tag{24}$$

Combining Eq. 20, Eq. 21 and Eq. 22 gives that the numerator of $u_q(p)$ can be written as,

$$\frac{1}{2}p\begin{bmatrix} 0 & -(q_1-q_3)(Pq_2-P-Tq_4) & (q_1-q_2)(Pq_3-Sq_4) & (q_1-q_4)(Sq_2-S-Tq_3) \\ -(q_1-q_3)(Pq_2-P-Tq_4) & 0 & (q_2-q_3)(Pq_1-P-Rq_4) & -(q_3-q_4)(Rq_2-R-Tq_1+T) \\ (q_1-q_2)(Pq_3-Sq_4) & (q_2-q_3)(Pq_1-P-Rq_4) & 0 & (q_2-q_4)(Rq_3-Sq_1+S) \\ (q_1-q_4)(Sq_2-S-Tq_3) & -(q_3-q_4)(Rq_2-R-Tq_1+T) & (q_2-q_4)(Rq_3-Sq_1+S) & 0 \end{bmatrix}p^T+$$
$$\begin{bmatrix} q_1(Pq_2-P-Tq_4) \\ -(q_3-1)(Pq_2-P-Tq_4) \\ -Pq_1q_2+Pq_2q_3+Pq_2-Pq_3+Rq_2q_4-Sq_2q_4+Sq_4 \\ -Rq_2q_4+Rq_4+Sq_2q_4-Sq_2-Sq_4+S+Tq_1q_4-Tq_3q_4+Tq_3-Tq_4 \end{bmatrix}p-Pq_2+P+Tq_4$$

and equivalently as,

$$\frac{1}{2}pQp^T+cp+a$$

where $Q\in\mathbb{R}^{4\times4}$ is a square matrix defined by the transition probabilities of the opponent q_1,q_2,q_3,q_4 as follows:

$$
\boldsymbol{Q} = \begin{bmatrix} 0 & -(q_1-q_3)(Pq_2-P-Tq_4) & (q_1-q_2)(Pq_3-Sq_4) & (q_1-q_4)(Sq_2-S-Tq_3) \\ -(q_1-q_3)(Pq_2-P-Tq_4) & 0 & (q_2-q_3)(Pq_1-P-Rq_4) & -(q_3-q_4)(Rq_2-R-Tq_1+T) \\ (q_1-q_2)(Pq_3-Sq_4) & (q_2-q_3)(Pq_1-P-Rq_4) & 0 & (q_2-q_4)(Rq_3-Sq_1+S) \\ (q_1-q_4)(Sq_2-S-Tq_3) & -(q_3-q_4)(Rq_2-R-Tq_1+T) & (q_2-q_4)(Rq_3-Sq_1+S) & 0 \end{bmatrix},
$$

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

$$
\boldsymbol{c} = \begin{bmatrix} q_1(Pq_2-P-Tq_4) \\ -(q_3-1)(Pq_2-P-Tq_4) \\ -Pq_1q_2+Pq_2q_3+Pq_2-Pq_3+Rq_2q_4-Sq_2q_4+Sq_4 \\ -Rq_2q_4+Rq_4+Sq_2q_4-Sq_2-Sq_4+S+Tq_1q_4-Tq_3q_4+Tq_3-Tq_4 \end{bmatrix},
$$

and $a = -Pq_2 + P + Tq_4$.

The same process is done for the denominator. $\square$

5.2 Theorem 2 Proof

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 given 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:

$$
S_q = \left\{ p \in \mathbb{R}^4 \;\middle|\; \begin{array}{l} \bullet \quad p_j \in \{0,1\} \quad \text{and} \quad \dfrac{d}{dp_k} \displaystyle\sum_{i=1}^{N} u_q^{(i)}(p) = 0 \\ \qquad\qquad\qquad\;\; \text{for all} \quad j \in J \quad \& \quad k \in K \quad \text{for all} \quad J, K \\ \qquad\qquad\qquad\;\; \text{where} \quad J \cap K = \emptyset \quad \text{and} \quad J \cup K = \{1,2,3,4\}. \\ \bullet \quad p \in \{0,1\}^4 \end{array} \right\}. \tag{25}
$$

Proof. The optimisation problem of Eq. 24

$$
\begin{aligned} \max_p : & \sum_{i=1}^{N} {u_q}^{(i)}(p) \\ \text{such that} : & \; p \in \mathbb{R}_{[0,1]} \end{aligned} \tag{26}
$$

can be written as:

$$
\begin{aligned}
\max_{p} :\ & \sum_{i=1}^{N} {u_q}^{(i)}(p) \\
\text{such that} : p_i &\leq 1 \text{ for } \in \{1,2,3,4\} \\
-p_i &\leq 0 \text{ for } \in \{1,2,3,4\}
\end{aligned} \tag{27}
$$

The optimisation problem has two inequality constraints and regarding the optimality this means that:

- either the optimum is away from the boundary of the optimization domain, and so the constraints plays no role;
- or the optimum is on the constraint boundary.

Thus, the following three cases must be considered:

Case 1: The solution is on the boundary and any of the possible combinations for $p_i \in \{0,1\}$ for $i \in \{1,2,3,4\}$ are candidate optimal solutions.

Case 2: The optimum is away from the boundary of the optimization domain and the interior solution p^* necessarily satisfies the condition $\frac{d}{dp}\sum_{i=1}^{N} u_q(p^*) = 0$.

Case 3: The optimum is away from the boundary of the optimization domain but some constraints are equalities. The candidate solutions in this case are any combinations of $p_j \in \{0,1\}$ and $\frac{d}{dp_k}\sum_{i=1}^{N} u_q^{(i)}(p) = 0$ forall $j \in J$ & $k \in K$ forall J, K where $J \cap K = \emptyset$ and $J \cup K = \{1,2,3,4\}$.

Combining cases 1-3 a set of candidate solutions, denoted as S_q, is constructed as:

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

The derivative of $\sum_{i=1}^{N} u_q^{(i)}(p)$ calculated using the following property (see [2] for details):

$$
\frac{dxAx^T}{dx} = 2Ax. \tag{28}
$$

Using property (26):

$$
\frac{d}{dp}\frac{1}{2}pQp^T + cp + a = pQ + c \text{ and } \frac{d}{dp}\frac{1}{2}p\bar{Q}p^T + \bar{c}p + \bar{a} = p\bar{Q} + \bar{c}. \tag{29}
$$

Note that the derivative of cp is c and the constant disappears. Combining these it can be proven that:

$$
\begin{aligned}
\frac{d}{dp}\sum_{i=1}^{N} u_q^{(i)}(p) &= \sum_{i=1}^{N} \frac{\frac{d}{dp}(\frac{1}{2}pQ^{(i)}p^T + c^{(i)}p + a^{(i)})(\frac{1}{2}pQ^{\overline{(i)}}p^T + c^{\overline{(i)}}p + a^{\overline{(i)}}) - \frac{d}{dp}(\frac{1}{2}pQ^{\overline{(i)}}p^T + c^{\overline{(i)}}p + a^{\overline{(i)}})(\frac{1}{2}pQ^{(i)}p^T + c^{(i)}p + a^{(i)})}{(\frac{1}{2}pQ^{\overline{(i)}}p^T + c^{\overline{(i)}}p + a^{\overline{(i)}})^2} \\
&= \sum_{i=1}^{N} \frac{(pQ^{(i)} + c^{(i)}+)(\frac{1}{2}pQ^{\overline{(i)}}p^T + c^{\overline{(i)}}p + a^{\overline{(i)}})}{(\frac{1}{2}pQ^{\overline{(i)}}p^T + c^{\overline{(i)}}p + a^{\overline{(i)}})^2} - \frac{(pQ^{\overline{(i)}} + c^{\overline{(i)}})(\frac{1}{2}pQ^{(i)}p^T + c^{(i)}p + a^{(i)})}{(\frac{1}{2}pQ^{\overline{(i)}}p^T + c^{\overline{(i)}}p + a^{\overline{(i)}})^2}
\end{aligned}
$$

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

$$
\sum_{i=1}^{N} \left(pQ^{(i)} + c^{(i)}\right)\left(\frac{1}{2}p\bar{Q}^{(i)}p^T + \bar{c}^{(i)}p + \bar{a}^{(i)}\right) - \left(p\bar{Q}^{(i)} + \bar{c}^{(i)}\right)\left(\frac{1}{2}pQ^{(i)}p^T + c^{(i)}p + a^{(i)}\right) = 0, \quad \textit{while} \tag{30}
$$

$$
\sum_{i=1}^{N} \frac{1}{2}p\bar{Q}^{(i)}p^T + \bar{c}^{(i)}p + \bar{a}^{(i)} \neq 0. \tag{31}
$$

The optimal solution to Eq. 24 is the point from S_q for which the utility is maximised. □

6 Lemma 3 Proof

In a tournament of N players $\{q^{(1)}, q^{(2)}, \dots, q^{(N)}\}$ for $q^{(i)} \in \mathbb{R}^4_{[0,1]}$ defection is stable if the transition probabilities of the opponents satisfy conditions Equation (30) and Equation (31).

$$
\sum_{i=1}^{N} (c^{(i)T}\bar{a}^{(i)} - \bar{c}^{(i)T}a^{(i)}) \leq 0 \tag{32}
$$

while,

$$
\sum_{i=1}^{N} \bar{a}^{(i)} \neq 0 \tag{33}
$$

Proof. For defection to be stable the derivative of the utility at the point $p = (0, 0, 0, 0)$ must be negative.

Substituting $p = (0, 0, 0, 0)$ in,

$$
\frac{d}{dp}\sum_{i=1}^{N} {u_q}^{(i)}(p) = \sum_{i=1}^{N} \frac{\left(pQ^{(i)} + c^{(i)}\right)\left(\frac{1}{2}p\bar{Q}^{(i)}p^T + \bar{c}^{(i)}p + \bar{a}^{(i)}\right)}{\left(\frac{1}{2}p\bar{Q}^{(i)}p^T + \bar{c}^{(i)}p + \bar{a}^{(i)}\right)^2} - \frac{\left(p\bar{Q}^{(i)} + \bar{c}^{(i)}\right)\left(\frac{1}{2}pQ^{(i)}p^T + c^{(i)}p + a^{(i)}\right)}{\left(\frac{1}{2}p\bar{Q}^{(i)}p^T + \bar{c}^{(i)}p + \bar{a}^{(i)}\right)^2} \tag{34}
$$

gives:

$$\left.\frac{d\sum_{i=1}^{N} {u_q}^{(i)}(p)}{dp}\right|_{p=(0,0,0,0)} = \sum_{i=1}^{N} \frac{(c^{(i)}\bar{a}^{(i)} - \bar{c}^{(i)}a^{(i)})}{(\bar{a}^{(i)})^2} \tag{35}$$

The sign of the numerator $\sum_{i=1}^{N}(c^{(i)}\bar{a}^{(i)} - \bar{c}^{(i)}a^{(i)})$ can vary based on the transition probabilities of the opponents. The denominator can not be negative, and otherwise is always positive. Thus the sign of the derivative is negative if and only if $\sum_{i=1}^{N}(c^{(i)}\bar{a}^{(i)} - \bar{c}^{(i)}a^{(i)}) \leq 0$. $\square$