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

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

[1] *Cardiff University, School of Mathematics, Cardiff, United Kingdom*

Abstract

Memory-one strategies are a set of Iterated Prisoner's Dilemma strategies that have been acclaimed for their mathematical tractability and performance against single opponents. This manuscript investigates *best responses* to a collection of memory-one strategies as a multidimensional optimisation problem. Though extortionate memory-one strategies have gained much attention, we demonstrate that best response memory-one strategies do not behave in an extortionate way, and moreover, 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 their limited memory in multi agent interactions and can be out performed by longer memory strategies.

The Prisoner's Dilemma (PD) is a two player game used in understanding the evolution of cooperative behaviour, formally introduced in [5]. 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 $(R, P, S, T) = (3, 1, 0, 5)$ [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).

Memory-one strategies are a set of IPD strategies that have been studied thoroughly in the literature [22, 23], however, they have gained most of their attention when a certain subset of memory-one strategies was introduced in [24], the zero-determinant strategies (ZDs). In [25] 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 [2, 10, 11, 12, 16, 18].

In a similar fashion to [24] the purpose of this work is to consider a given memory-one strategy; however, whilst [24] found a way for a player to manipulate a given opponent, this work will consider a multidimensional optimisation approach to identify the best response to a given group of opponents. In particular, this work

presents a compact method of identifying the best response memory-one strategy against a given set of opponents, and evaluates whether it behaves in a zero-determinant way which in turn indicates whether it can be extortionate. This is also done in evolutionary settings. Moreover, we introduce a well designed framework that allows the comparison of an optimal memory one strategy and a more complex strategy which has a larger memory, and an identification of conditions for which defection is known to be stable; thus identifying environments where cooperation will not occur.

Methods and Results

Utility

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 [21] 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 [21] 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 Ep. 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.

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

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

This manuscript has explored the form of $u_q(p)$, to the authors knowledge no previous work has done this, and it proves that $u_q(p)$ is given by a ratio of two quadratic forms [15], (Theorem 2):

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

where $Q = Q(q), \bar{Q} = \bar{Q}(q) \in \mathbb{R}^{4\times4}$, $c = c(q)$ and $\bar{c} = \bar{c}(q) \in \mathbb{R}^{4\times1}$, $a = a(q)$ and $\bar{a} = \bar{a}(q) \in \mathbb{R}$.

This 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:

$$\frac{1}{N}\sum_{i=1}^{N} {u_q}^{(i)}(p) = \frac{\frac{1}{N}\sum_{i=1}^{N}(\frac{1}{2}pQ^{(i)}p^T + c^{(i)}p + a^{(i)}) \prod_{\substack{j=1 \\ j\neq i}}^{N}(\frac{1}{2}p\bar{Q}^{(j)}p^T + \bar{c}^{(j)}p + \bar{a}^{(j)})}{\prod_{i=1}^{N}(\frac{1}{2}p\bar{Q}^{(i)}p^T + \bar{c}^{(i)}p + \bar{a}^{(i)})}. \tag{5}$$

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 and evolutionary dynamics, and to explore the conditions under which defection dominates cooperation.

Stability of defection

An immediate result from our formulation can be obtained by evaluating the sign of Eq. 5's derivative at $p = (0, 0, 0, 0)$. If at that point the derivative is negative, then the utility of a player only decreases if they were to change their behaviour, and thus **defection at that point is stable**.

Lemma 1. *In a tournament of N players $\{q^{(1)}, q^{(2)}, \ldots, q^{(N)}\}$ for $q^{(i)} \in \mathbb{R}^4_{[0,1]}$ defection is stable if the transition probabilities of the opponents satisfy conditions Eq. 6 and Eq. 7.*

$$\sum_{i=1}^{N}(c^{(i)T}\bar{a}^{(i)} - \bar{c}^{(i)T}a^{(i)}) \leq 0 \tag{6}$$

while,

$$\sum_{i=1}^{N}\bar{a}^{(i)} \neq 0 \tag{7}$$

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 Eq. 22 gives:

$$\left.\frac{d\sum_{i=1}^{N} {u_q}^{(i)}(p)}{dp}\right|_{p=(0,0,0,0)} = \sum_{i=1}^{N}\frac{(c^{(i)T}\bar{a}^{(i)} - \bar{c}^{(i)T}a^{(i)})}{(\bar{a}^{(i)})^2} \tag{8}$$

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

Consider a population for which defection is known to be stable. In that population all the members will over time adopt the same behaviour; thus in such population cooperation will never take over. This is demonstrated in Fig. 1. These have been simulated using [1] an open source research framework for the study of the IPD.

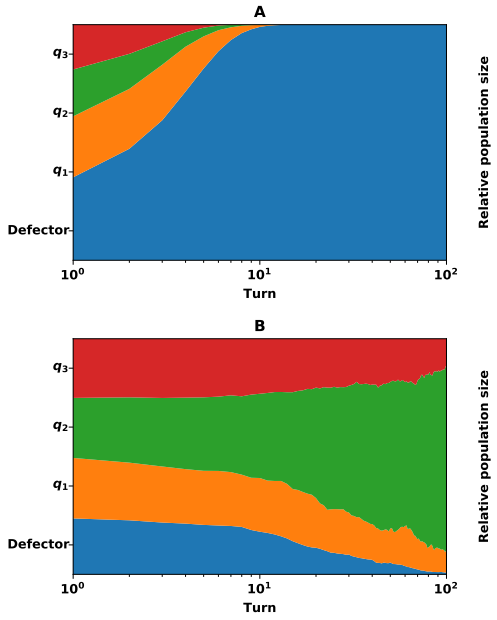

Figure 1: A. For $q_1 = (0.22199, 0.87073, 0.20672, 0.91861)$, $q_2 = (0.48841, 0.61174, 0.76591, 0.51842)$ and $q_3 = (0.2968, 0.18772, 0.08074, 0.73844)$, Eq. 6 and Eq. 7 hold and Defector takes over the population. B. For $q_1 = (0.96703, 0.54723, 0.97268, 0.71482)$, $q_2 = (0.69773, 0.21609, 0.97627, 0.0062)$ and $q_3 = (0.25298, 0.43479, 0.77938, 0.19769)$, Eq. 6 fails and Defector does not take over the population.

Best response memory-one strategies

As discussed ZDs have been acclaimed for their robustness against a single opponent. ZDs are evidence that extortion works in pairwise interactions, their behaviour ensures that the strategies will never lose a game. However, this paper argues that in multi opponent interactions, 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.

To answer this question, we initially define *memory-one best response* strategies as a multi dimensional optimisation problem given by:

$$\begin{aligned} &\max_{p} : \sum_{i=1}^{N} {u_q}^{(i)}(p) \\ &\text{such that} : p \in \mathbb{R}_{[0,1]} \end{aligned} \tag{9}$$

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. 5. This gives the best response memory-one strategy. The approach is given in Theorem 3.

Finding best response memory-one strategies is analytically feasible using the formulation of Theorem 3 and resultant theory [14]. 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 [20].

This is extended to evolutionary settings. In these settings self interactions are key. Self interactions can be incorporated in the formulation that has been used so far. More specifically, the optimisation problem of Eq. 26 is extended to include self interactions:

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

For determining the memory-one best response in an evolutionary setting, 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**
  $p^{(t+1)} = \text{argmax} \frac{1}{N} \sum_{i=1}^{N} {u_q}^{(i)}(p^{(t+1)}) + u_p^{(t)}(p^{(t+1)})$;
end

The results of this section use Bayesian optimisation to generate a data set of best response memory-one strategies, in tournaments and evolutionary dynamics whilst $N = 2$. The data set is available at [6]. It contains a total of 1000 trials corresponding to 1000 different instances of a best response strategy in tournaments and evolutionary dynamics. For each trial a set of 2 opponents is randomly generated and the memory-one best responses against them is found.

The source code used in this manuscript has been written in a sustainable manner [4]. 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 [7].

In order to investigate whether best responses behave in an extortionate matter the SSE method [17] is used. In [17] it is proven that all extortionate ZDs reside on a triangular plane. For a given p, a strategy x^* is defined as the nearest ZDs. The distance between the two strategies is explicitly calculated and referred to as the sum of squared errors of prediction (SSE); which corresponds to how far p is from behaving as a ZDs. Thus, a high SSE implies non ZD, which in turn implies a non extortionate behaviour. The SSE method has been applied to the data set. A statistics summary of the SSE distribution for the best response in tournaments and evolutionary dynamics is given in Table 1.

	mean	std	5%	50%	95%	max	median	skew	kurt
Tournament	0.34	0.40	0.028	0.17	1.05	2.47	0.17	1.87	3.60
Evolutionary Setting	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 $N = 2$

For the best response in tournaments 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 indicates a heavy tail to the right. Therefore, in several cases the strategy is not trying to extort its opponents. Similarly the evolutionary best response strategy does not behave uniformly extortionately. A larger value of both the kurtosis and the skewness of the SSE distribution indicates that in evolutionary settings a memory-one best response is even more adaptable.

The difference between best responses in tournaments and in evolutionary settings is further explored by Fig. 2. Though, no statistically significant differences have been found, from Fig. 2, it seems that evolutionary best response 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.

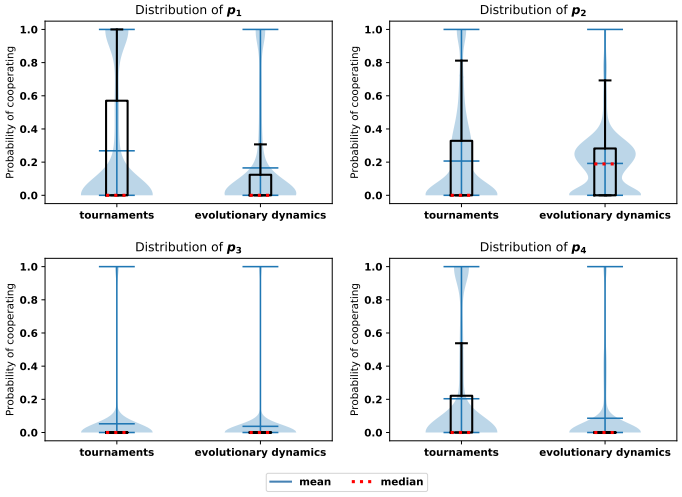

Figure 2: Distributions of p^* for best responses in tournaments and evolutionary settings. The medians, denoted as $\bar{p}^*$, for tournaments are $\bar{p}^* = (0, 0, 0, 0)$, and for evolutionary settings $\bar{p}^* = (0, 0.19, 0, 0)$.

Longer memory best responses

This section focuses on the memory size of strategies. The effectiveness of memory in the IPD has been previously explored in the literature, however, no one has compared the performance of longer-memory strategies to memory-one best responses.

In [8], 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. 26, finding an optimal Gambler is a 17 dimensional problem given by:

$$\begin{aligned} &\max_{p} : \sum_{i=1}^{N} {U_q}^{(i)}(f) \\ &\text{such that} : f \in \mathbb{R}^{17}_{[0,1]} \end{aligned} \tag{11}$$

Note that Eq. 5 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. 11 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 [6]. 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. For each trial two random opponents have been selected. The 152 pair of opponents are a sub set of the opponents used in the previous section.

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. This seems to be at odd with the result of [24] that against a memory-one opponent having a longer memory will not give a strategy any advantage. 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 two opponents having a shorter memory is limiting and this is potentially another example of the advantages of adaptability.

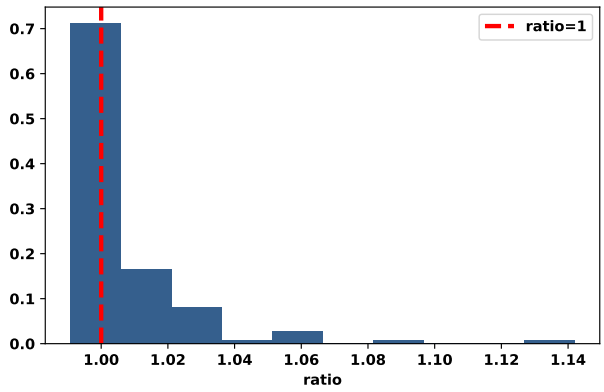

Figure 3: The ratio between the utilities of Gambler and best response memory-one strategy for 152 different pair of opponents.

Discussion

This manuscript has considered *best response* strategies in the IPD game, and more specifically, *memory-one best responses*. It has proven that there is a compact way of identifying a memory-one best response to a group of opponents, and moreover it obtained a condition for which in an environment of memory-one opponents defection is the stable choice, based only on the coefficients of the opponents. The later parts of this paper focused on a series of empirical results, where it was shown that the performance and the evolutionary stability of memory-one strategies rely on adaptability and not on extortion. Finally, it was shown that memory-one strategies' performance is limited by their memory in cases where they interact with multiple opponents.

Following the work described in [21], where it was shown that the utility between two memory-one strategies can be estimated by a Markov stationary state, we proved that the utilities can be written as a ration of two quadratic forms in R^4, Theorem 2. This was extended to include multiple opponents, as the IPD is commonly studied in such situations. This formulation allowed us to introduce an approach for identifying memory-one best responses to any number of opponents; Theorem 3. This does not only have game theoretic novelty, but also a mathematical novelty of solving quadratic ratio optimisation problems where the quadratics are non concave. The results were used to define a condition for which defection is known to be stable.

This manuscript presented several experimental results. All data for the results is archived in [6]. 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 and in evolutionary settings for $N = 2$ was generated. This allowed us to investigate their respective behaviours, and whether it was extortionate acts that made them the most favorable strategies. However, it was shown that it was not extortion but adaptability that allowed the strategies to gain the most from their interactions. In evolutionary settings it was shown that the best response strategy was even more adaptable, and there is some evidence that it is more likely to forgive after being tricked. Moreover, the performance of memory-one strategies was put against the performance of a longer memory strategy called Gambler. There were several cases where Gambler would outperform the memory-one strategy, however, a memory-one strategy never managed to outperform a Gambler. This result occurred whilst considering a Gambler with a sufficiently larger memory but not a sufficiently larger amount of information regarding the game.

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 evolutionary best responses dynamics algorithm discussed.

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

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 [9], the Matplotlib library for visualisation [13], the SymPy library for symbolic mathematics [19] and the Numpy library for data manipulation [26].

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

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

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

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

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

[8] 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.

[9] 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.

[10] 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.

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

[12] 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.

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

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

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

[16] 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.

[17] 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.

[18] 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.

[19] 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.

[20] 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.

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

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

[23] 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.

[24] 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.

[25] 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.

[26] 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.

Appendix

Theorem 2

Theorem 2. *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:*

$$u_q(p) = \frac{\frac{1}{2}pQp^T + cp + a}{\frac{1}{2}p\bar{Q}p^T + \bar{c}p + \bar{a}}, \tag{12}$$

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

$$Q = \begin{bmatrix} 0 & -(q_1-q_3)(q_2-5q_4-1) & q_3(q_1-q_2) & -5q_3(q_1-q_4) \\ -(q_1-q_3)(q_2-5q_4-1) & 0 & (q_2-q_3)(q_1-3q_4-1) & (q_3-q_4)(5q_1-3q_2-2) \\ q_3(q_1-q_2) & (q_2-q_3)(q_1-3q_4-1) & 0 & 3q_3(q_2-q_4) \\ -5q_3(q_1-q_4) & (q_3-q_4)(5q_1-3q_2-2) & 3q_3(q_2-q_4) & 0 \end{bmatrix}, \tag{13}$$

$$\bar{Q} = \begin{bmatrix} 0 & -(q_1-q_3)(q_2-q_4-1) & (q_1-q_2)(q_3-q_4) & (q_1-q_4)(q_2-q_3-1) \\ -(q_1-q_3)(q_2-q_4-1) & 0 & (q_2-q_3)(q_1-q_4-1) & (q_1-q_2)(q_3-q_4) \\ (q_1-q_2)(q_3-q_4) & (q_2-q_3)(q_1-q_4-1) & 0 & -(q_2-q_4)(q_1-q_3-1) \\ (q_1-q_4)(q_2-q_3-1) & (q_1-q_2)(q_3-q_4) & -(q_2-q_4)(q_1-q_3-1) & 0 \end{bmatrix}. \tag{14}$$

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

$$c = \begin{bmatrix} q_1\,(q_2 - 5q_4 - 1) \\ -\,(q_3 - 1)\,(q_2 - 5q_4 - 1) \\ -q_1q_2 + q_2q_3 + 3q_2q_4 + q_2 - q_3 \\ 5q_1q_4 - 3q_2q_4 - 5q_3q_4 + 5q_3 - 2q_4 \end{bmatrix}, \tag{15}$$

$$\bar{c} = \begin{bmatrix} q_1\,(q_2 - q_4 - 1) \\ -\,(q_3 - 1)\,(q_2 - q_4 - 1) \\ -q_1q_2 + q_2q_3 + q_2 - q_3 + q_4 \\ q_1q_4 - q_2 - q_3q_4 + q_3 - q_4 + 1 \end{bmatrix}, \tag{16}$$

and the constant terms $a, \bar{a}$ *are defined as* $a = -q_2 + 5q_4 + 1$ *and* $\bar{a} = -q_2 + q_4 + 1$.

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

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

More specifically, with $(R, P, S, T) = (3, 1, 0, 5)$

$$u_q(p) = \begin{pmatrix} p_1p_2(q_1q_2 - 5q_1q_4 - q_1 - q_2q_3 + 5q_3q_4 + q_3) + p_1p_3(-q_1q_3 + q_2q_3) + p_1p_4(5q_1q_3 - 5q_3q_4) + p_3p_4(-3q_2q_3 + 3q_3q_4) + \\ p_2p_3(-q_1q_2 + q_1q_3 + 3q_2q_4 + q_2 - 3q_3q_4 - q_3) + p_2p_4(-5q_1q_3 + 5q_1q_4 + 3q_2q_3 - 3q_2q_4 + 2q_3 - 2q_4) + \\ p_1(-q_1q_2 + 5q_1q_4 + q_1) + p_2(q_2q_3 - q_2 - 5q_3q_4 - q_3 + 5q_4 + 1) + p_3(q_1q_2 - q_2q_3 - 3q_2q_4 - q_2 + q_3) + \\ \dfrac{p_4(-5q_1q_4 + 3q_2q_4 + 5q_3q_4 - 5q_3 + 2q_4) + q_2 - 5q_4 - 1}{p_1p_2(q_1q_2 - q_1q_4 - q_1 - q_2q_3 + q_3q_4 + q_3) + p_1p_3(-q_1q_3 + q_1q_4 + q_2q_3 - q_2q_4) + p_1p_4(-q_1q_2 + q_1q_3 + q_1 + q_2q_4 - q_3q_4 - q_4) + \\ p_2p_3(-q_1q_2 + q_1q_3 + q_2q_4 + q_2 - q_3q_4 - q_3) + p_2p_4(-q_1q_3 + q_1q_4 + q_2q_3 - q_2q_4) + p_3p_4(q_1q_2 - q_1q_4 - q_2q_3 - q_2 + q_3q_4 + q_4) + \\ p_1(-q_1q_2 + q_1q_4 + q_1) + p_2(q_2q_3 - q_2 - q_3q_4 - q_3 + q_4 + 1) + p_3(q_1q_2 - q_2q_3 - q_2 + q_3 - q_4) + p_4(-q_1q_4 + q_2 + q_3q_4 - q_3 + q_4 - 1) + \\ q_2 - q_4 - 1 \end{pmatrix}. \tag{17}$$

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

$$p_1p_2(q_1q_2 - 5q_1q_4 - q_1 - q_2q_3 + 5q_3q_4 + q_3) + p_1p_3(-q_1q_3 + q_2q_3) + p_1p_4(5q_1q_3 - 5q_3q_4) + p_3p_4(-3q_2q_3 + 3q_3q_4) + \\ p_2p_3(-q_1q_2 + q_1q_3 + 3q_2q_4 + q_2 - 3q_3q_4 - q_3) + p_2p_4(-5q_1q_3 + 5q_1q_4 + 3q_2q_3 - 3q_2q_4 + 2q_3 - 2q_4).$$

This can be re written in a matrix format given by Eq. 18.

$$(p_1, p_2, p_3, p_4)\frac{1}{2}\begin{bmatrix} 0 & -\,(q_1 - q_3)\,(q_2 - 5q_4 - 1) & q_3\,(q_1 - q_2) & -5q_3\,(q_1 - q_4) \\ -\,(q_1 - q_3)\,(q_2 - 5q_4 - 1) & 0 & (q_2 - q_3)\,(q_1 - 3q_4 - 1) & (q_3 - q_4)\,(5q_1 - 3q_2 - 2) \\ q_3\,(q_1 - q_2) & (q_2 - q_3)\,(q_1 - 3q_4 - 1) & 0 & 3q_3\,(q_2 - q_4) \\ -5q_3\,(q_1 - q_4) & (q_3 - q_4)\,(5q_1 - 3q_2 - 2) & 3q_3\,(q_2 - q_4) & 0 \end{bmatrix}\begin{pmatrix} p_1 \\ p_2 \\ p_3 \\ p_4 \end{pmatrix} \tag{18}$$

Similarly, the linear terms are given by,

$$p_1(-q_1q_2 + 5q_1q_4 + q_1) + p_2(q_2q_3 - q_2 - 5q_3q_4 - q_3 + 5q_4 + 1) + p_3(q_1q_2 - q_2q_3 - 3q_2q_4 - q_2 + q_3) + \\ p_4(-5q_1q_4 + 3q_2q_4 + 5q_3q_4 - 5q_3 + 2q_4).$$

and the expression can be written using a matrix format as Eq. 19.

$$(p_1, p_2, p_3, p_4) \begin{bmatrix} q_1\,(q_2 - 5q_4 - 1) \\ -(q_3 - 1)\,(q_2 - 5q_4 - 1) \\ -q_1 q_2 + q_2 q_3 + 3q_2 q_4 + q_2 - q_3 \\ 5q_1 q_4 - 3q_2 q_4 - 5q_3 q_4 + 5q_3 - 2q_4 \end{bmatrix} \tag{19}$$

Finally, the constant term of the numerator, which is obtained by substituting $p = (0, 0, 0, 0)$, is given by Eq. 20.

$$q_2 - 5q_4 - 1 \tag{20}$$

Combining Eq. 18, Eq. 19 and Eq. 20 gives that the numerator of $u_q(p)$ can be written as,

$$\frac{1}{2} p \begin{bmatrix} 0 & -(q_1 - q_3)\,(q_2 - 5q_4 - 1) & q_3\,(q_1 - q_2) & -5q_3\,(q_1 - q_4) \\ -(q_1 - q_3)\,(q_2 - 5q_4 - 1) & 0 & (q_2 - q_3)\,(q_1 - 3q_4 - 1) & (q_3 - q_4)\,(5q_1 - 3q_2 - 2) \\ q_3\,(q_1 - q_2) & (q_2 - q_3)\,(q_1 - 3q_4 - 1) & 0 & 3q_3\,(q_2 - q_4) \\ -5q_3\,(q_1 - q_4) & (q_3 - q_4)\,(5q_1 - 3q_2 - 2) & 3q_3\,(q_2 - q_4) & 0 \end{bmatrix} p^T + \\ \begin{bmatrix} 0 & -(q_1 - q_3)\,(q_2 - 5q_4 - 1) & q_3\,(q_1 - q_2) & -5q_3\,(q_1 - q_4) \\ -(q_1 - q_3)\,(q_2 - 5q_4 - 1) & 0 & (q_2 - q_3)\,(q_1 - 3q_4 - 1) & (q_3 - q_4)\,(5q_1 - 3q_2 - 2) \\ q_3\,(q_1 - q_2) & (q_2 - q_3)\,(q_1 - 3q_4 - 1) & 0 & 3q_3\,(q_2 - q_4) \\ -5q_3\,(q_1 - q_4) & (q_3 - q_4)\,(5q_1 - 3q_2 - 2) & 3q_3\,(q_2 - q_4) & 0 \end{bmatrix} p + q_2 - 5q_4 - 1$$

and equivalently as,

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

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

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

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

$$c = \begin{bmatrix} q_1\,(q_2 - 5q_4 - 1) \\ -(q_3 - 1)\,(q_2 - 5q_4 - 1) \\ -q_1 q_2 + q_2 q_3 + 3q_2 q_4 + q_2 - q_3 \\ 5q_1 q_4 - 3q_2 q_4 - 5q_3 q_4 + 5q_3 - 2q_4 \end{bmatrix},$$

and $a = -q_2 + 5q_4 + 1$.

The same process is done for the denominator. □

Theorem 3

Theorem 3. *The optimal behaviour of a memory-one strategy player $p^* \in \mathbb{R}^4_{[0,1]}$ against a set of N opponents $\{q^{(1)}, q^{(2)}, \ldots, q^{(N)}\}$ 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:

$$S_q = \left\{ p \in \mathbb{R}^4 \middle| \begin{array}{ll} \bullet & p_j \in \{0,1\} \quad and \quad \frac{d}{dp_k} \sum_{i=1}^{N} u_q^{(i)}(p) = 0 \\ & \qquad\qquad for\ all \quad j \in J \quad \& \quad k \in K \quad for\ all \quad J, K \\ & \qquad\qquad where \quad J \cap K = \emptyset \quad and \quad J \cup K = \{1,2,3,4\}. \\ \bullet & p \in \{0,1\}^4 \end{array} \right\}. \tag{21}$$

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

$$\frac{d}{dp} \sum_{i=1}^{N} {u_q}^{(i)}(p) = \sum_{i=1}^{N} \frac{\left(pQ^{(i)} + c^{(i)}\right)\left(\frac{1}{2} p\bar{Q}^{(i)} p^T + \bar{c}^{(i)} p + \bar{a}^{(i)}\right)}{\left(\frac{1}{2} p\bar{Q}^{(i)} p^T + \bar{c}^{(i)} p + \bar{a}^{(i)}\right)^2} - \frac{\left(p\bar{Q}^{(i)} + \bar{c}^{(i)}\right)\left(\frac{1}{2} pQ^{(i)} p^T + c^{(i)} p + a^{(i)}\right)}{\left(\frac{1}{2} p\bar{Q}^{(i)} p^T + \bar{c}^{(i)} p + \bar{a}^{(i)}\right)^2} \tag{22}$$

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

$$\sum_{i=1}^{N} \left(pQ^{(i)} + c^{(i)}\right)\left(\frac{1}{2} p\bar{Q}^{(i)} p^T + \bar{c}^{(i)} p + \bar{a}^{(i)}\right) - \left(p\bar{Q}^{(i)} + \bar{c}^{(i)}\right)\left(\frac{1}{2} pQ^{(i)} p^T + c^{(i)} p + a^{(i)}\right) = 0, \quad while \tag{23}$$

$$\sum_{i=1}^{N} \frac{1}{2} p\bar{Q}^{(i)} p^T + \bar{c}^{(i)} p + \bar{a}^{(i)} \neq 0. \tag{24}$$

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

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

where S_q is given by:

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

The optimisation problem of Eq. 26

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

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 solution is constructed as:

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

This set is denoted as S_q and the optimal solution to Eq. 26 is the point from S_q for which the utility is maximised. □