PROCEEDINGS A

royalsocietypublishing.org/journal/rspa

Research

Cite this article: McAvoy A, Nowak MA. 2019 Reactive learning strategies for iterated games. *Proc. R. Soc. A* **475**: 20180819. http://dx.doi.org/10.1098/rspa.2018.0819

Received: 21 November 2018
Accepted: 29 January 2019

Subject Areas:
applied mathematics, mathematical modelling

Keywords:
adaptive strategy, iterated game, memory-one strategy, social dilemma

Author for correspondence:
Alex McAvoy
e-mail: alexmcavoy@fas.harvard.edu

Reactive learning strategies for iterated games

Alex McAvoy and Martin A. Nowak

Program for Evolutionary Dynamics, Harvard University, 1 Brattle Square, Suite 6, Cambridge, MA 02138, USA

AM, 0000-0002-9110-4635

In an iterated game between two players, there is much interest in characterizing the set of feasible pay-offs for both players when one player uses a fixed strategy and the other player is free to switch. Such characterizations have led to extortionists, equalizers, partners and rivals. Most of those studies use memory-one strategies, which specify the probabilities to take actions depending on the outcome of the previous round. Here, we consider 'reactive learning strategies', which gradually modify their propensity to take certain actions based on past actions of the opponent. Every linear reactive learning strategy, $\mathbf{p}^*$, corresponds to a memory one-strategy, $\mathbf{p}$, and vice versa. We prove that for evaluating the region of feasible pay-offs against a memory-one strategy, $\mathcal{C}(\mathbf{p})$, we need to check its performance against at most 11 other strategies. Thus, $\mathcal{C}(\mathbf{p})$ is the convex hull in $\mathbb{R}^2$ of at most 11 points. Furthermore, if $\mathbf{p}$ is a memory-one strategy, with feasible pay-off region $\mathcal{C}(\mathbf{p})$, and $\mathbf{p}^*$ is the corresponding reactive learning strategy, with feasible pay-off region $\mathcal{C}(\mathbf{p}^*)$, then $\mathcal{C}(\mathbf{p}^*)$ is a subset of $\mathcal{C}(\mathbf{p})$. Reactive learning strategies are therefore powerful tools in restricting the outcomes of iterated games.

1. Introduction

Since the discovery of zero-determinant strategies for iterated games by Press & Dyson [1], there has been a growing interest in the set of possible pay-offs that can be achieved against a fixed strategy. Imagine that Alice uses a particular strategy, while Bob can try out any conceivable strategy. The resulting set of pay-offs for both Alice and Bob define the 'feasible region' of Alice's strategy. If Alice uses a so-called zero-determinant strategy [1], then the feasible region is a line. In general, the feasible region is a two-dimensional

THE ROYAL SOCIETY PUBLISHING

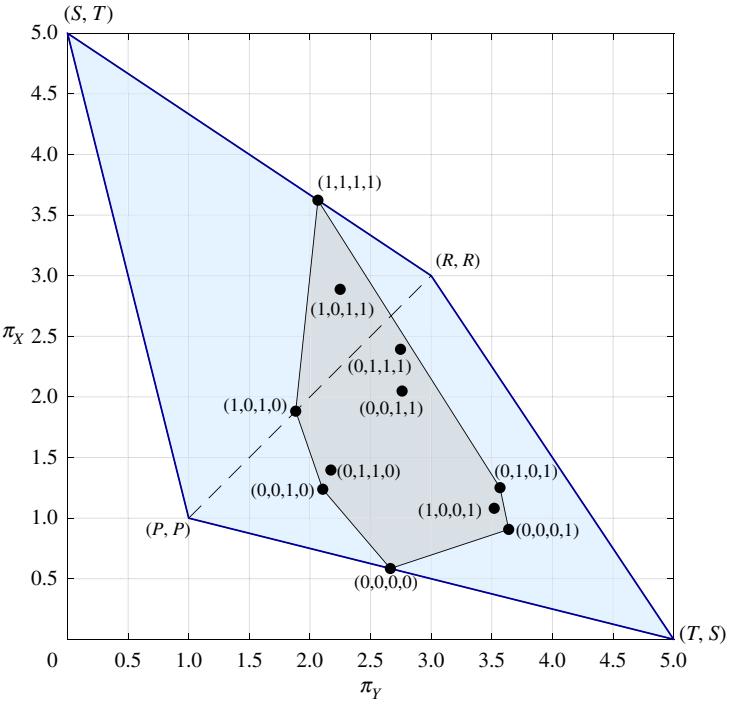

Figure 1. Feasible region (grey) for a strategy with $\mathbf{p}_{\bullet\bullet} = (0.7881, 0.8888, 0.4686, 0.0792)$ when $R = 3$, $S = 0$, $T = 5$ and $P = 1$. The light blue region depicts the set of all pay-off pairs that can be achieved in the iterated game, i.e. the convex hull of the points (R, R), (S, T), (P, P) and (T, S). The feasible region of $\mathbf{p}$ can be characterized as the convex hull of 11 points, corresponding to those opponent-strategies, $\mathbf{q}$, appearing next to each black dot. In this instance, five of these points already fall inside of the convex hull of the remaining six. However, one cannot remove one of these 11 points without destroying this characterization for some game-strategy pair. (Online version in colour.)

convex subset of the feasible pay-off region of the game (figure 1). Using the geometric intuition put forth by Press & Dyson [1], subsequent work has explored strategies that generate two-dimensional feasible regions, defined by linear inequalities rather than strict equations [2–4]. However, a general description of what this region looks like, as it relates to the type of strategy played, is currently not well understood. In this study, we characterize the feasible regions for the well-known class of memory-one strategies [5] and consider their relationships to those of a new class of 'reactive learning strategies'.

Iterated games have many applications across the social sciences and biology, and with them has come a proliferation of strategy classes of various complexities [6–10]. The type of strategy a player uses for dealing with repeated encounters depends on many factors, including the cognitive capacity of the player and the nature of the underlying 'one-shot' (or 'stage') games. In applications to theoretical biology, the most well-studied type of strategy is known as 'memory-one' because it takes into account the outcome of only the previous encounter when determining how to play in the next round [5,11]. This class of strategies, while forming only a small subset of all possible ways to play an iterated game [12], has several advantages over more complicated strategies. They permit rich behaviour in iterated play, such as punishment for exploitation and reward for cooperation [5,13–18]; but, owing to their simple memory requirements, they are also straightforward to implement in practice and analyse mathematically.

Memory, however, can apply to more than just the players' actions in the previous round. Since the action a player chooses in any particular encounter is typically chosen stochastically rather than deterministically, a player can also take into account *how* they chose their previous

action rather than just the result. In a social dilemma, for instance, each player chooses an action ('cooperate', C, or 'defect', D) in a given round and receives a pay-off for this action against that of the opponent. The distribution with which this action is chosen is referred to as a 'mixed action' and can be specified by a single number between 0 and 1, representing the tendency to cooperate. A standard memory-one strategy for player X is given by a five-tuple, $(p_0, p_{CC}, p_{CD}, p_{DC}, p_{DD})$, where p_0 is the probability of cooperation in the initial round and p_{xy} is the probability of cooperation following an outcome in which X uses action x and the opponent, Y, uses action y. We consider a variation on this theme, where instead of using x and y to determine the next mixed action, X uses the opponent's action, y, to update their own mixed action, $\sigma_X \in [0,1]$, that was used previously to generate x. We refer to a strategy of this form as a 'reactive learning strategy'.

Such a strategy is 'reactive' because it takes into account the realized action of just the opponent, and it is 'learning' because it adapts to this external stimulus. Like a memory-one strategy, a reactive learning strategy for X requires knowledge of information one round into the past, namely X's mixed action, σ_X, and Y's realized action, y. Unlike a memory-one strategy, in which the probability of cooperation is in the set $\{p_0, p_{CC}, p_{CD}, p_{DC}, p_{DD}\}$ in every round of the game, a reactive learning strategy can result in a broad range of cooperation tendencies for X over the duration of an iterated game. Moreover, these tendencies can be gradually changed over the course of many rounds, resulting (for example) in high probabilities of cooperation only after the opponent has demonstrated a sufficiently long history of cooperating. Punishment for defection can be similarly realized over a number of interactions. Remembering a probability, σ_X, and an action, y, instead of just two actions, x and y, can thus lead to more complex behaviours.

This adaptive approach to iterated games is similar to the Bush–Mosteller reinforcement learning algorithm [19–21], but there are important distinctions. For one, a reactive learning strategy does not necessarily reinforce behaviour resulting in higher pay-offs. Furthermore, it completely disregards the focal player's realized action, using only that of the opponent in the update mechanism. But there are certainly reactive learning strategies that are more closely related to reinforcement learning, and we give an example using a variation on the memory-one strategy tit-for-tat (TFT), which we call 'learning tit-for-tat (LTFT)'.

In this study, we establish some basic properties of reactive learning strategies relative to the memory-one space. We first characterize the feasible region of a memory-one strategy as the convex hull of at most 11 points. When then show that there is an embedding of the set of memory-one strategies in the set of reactive learning strategies with the following property: if $\mathbf{p}$ is a memory-one strategy and $\mathbf{p}^*$ is the corresponding reactive learning strategy, then the feasible region of $\mathbf{p}$ contains the feasible region of $\mathbf{p}^*$. Moreover, the image of the map $\mathbf{p} \mapsto \mathbf{p}^*$ is the set of *linear* reactive learning strategies, which consists of those strategies that send a player's mixed action, σ_X, to $\alpha\sigma_X + \beta$ for some $\alpha, \beta \in [0,1]$. As a consequence, if the goal of a player is to restrict the region of pay-offs attainable by the players, then this player should prefer using a linear reactive learning strategy over the corresponding memory-one strategy.

2. Memory-one strategies

Consider an iterated game between two players, X and Y. In every round, each player chooses an action from the set $\{C, D\}$ (cooperate or defect). They receive pay-offs based on the values in the matrix

$$\begin{array}{c} \\ C \\ D \end{array} \begin{array}{c} \begin{array}{cc} C & D \end{array} \\ \begin{pmatrix} R & S \\ T & P \end{pmatrix} \end{array}. \tag{2.1}$$

Over many rounds, these pay-offs are averaged to arrive at an expected pay-off for each player.

Whereas an action specifies the behaviour of a player in one particular encounter, a strategy specifies how a player behaves over the course of many encounters. One of the simplest and best-studied strategies for iterated games is a memory-one strategy [5], which for player X is defined as follows: for every $(x, y) \in \{C, D\}^2$ observed as action outcomes of a given round, X devises a mixed action $p_{xy} \in [0, 1]$ for the next round. The notation p_{xy} indicates that this mixed action depends on the (pure) actions of both players in the previous round, not how they arrived at those actions (e.g. by generating an action probabilistically). The term 'strategy' is reserved for the players' behaviours in the iterated game.

Let $\mathbf{Mem}_X^1$ be the space of all memory-one strategies for player X in an iterated game. With just two actions, C and D, we have $\mathbf{Mem}_X^1 = [0, 1] \times [0, 1]^4$, i.e. the space of all $(p_0, p_{CC}, p_{CD}, p_{DC}, p_{DD}) \in [0, 1]^5$. A pair of memory-one strategies, $\mathbf{p} := (p_0, p_{CC}, p_{CD}, p_{DC}, p_{DD})$ and $\mathbf{q} := (q_0, q_{CC}, q_{CD}, q_{DC}, q_{DD})$, for X and Y, respectively, yield a Markov chain on the space of all action pairs, $\{C, D\}^2$, whose transition matrix is

$$M(\mathbf{p}, \mathbf{q}) = \begin{array}{c} \\ CC \\ CD \\ DC \\ DD \end{array} \begin{array}{c} \begin{array}{cccc} CC & CD & DC & DD \end{array} \\ \begin{pmatrix} p_{CC}q_{CC} & p_{CC}(1-q_{CC}) & (1-p_{CC})q_{CC} & (1-p_{CC})(1-q_{CC}) \\ p_{CD}q_{DC} & p_{CD}(1-q_{DC}) & (1-p_{CD})q_{DC} & (1-p_{CD})(1-q_{DC}) \\ p_{DC}q_{CD} & p_{DC}(1-q_{CD}) & (1-p_{DC})q_{CD} & (1-p_{DC})(1-q_{CD}) \\ p_{DD}q_{DD} & p_{DD}(1-q_{DD}) & (1-p_{DD})q_{DD} & (1-p_{DD})(1-q_{DD}) \end{pmatrix} \end{array} \tag{2.2}$$

and whose initial distribution is $\mu_0 := (p_0q_0, p_0(1-q_0), (1-p_0)q_0, (1-p_0)(1-q_0))$. If $p_{xy}, q_{xy} \in (0, 1)$ for every $x, y \in \{C, D\}$, then this chain is ergodic and has a unique stationary distribution, $\mu(\mathbf{p}, \mathbf{q})$, which is independent of μ_0. In particular, the expected pay-offs, $\pi_X(\mathbf{p}, \mathbf{q}) = \mu(\mathbf{p}, \mathbf{q}) \cdot (R, S, T, P)$ and $\pi_Y(\mathbf{p}, \mathbf{q}) = \mu(\mathbf{p}, \mathbf{q}) \cdot (R, T, S, P)$, are independent of p_0 and q_0. In this case, π_X and π_Y are functions of just the response probabilities, $\mathbf{p}_{\bullet\bullet} := (p_{CC}, p_{CD}, p_{DC}, p_{DD})$ and $\mathbf{q}_{\bullet\bullet} := (q_{CC}, q_{CD}, q_{DC}, q_{DD})$.

A useful way of thinking about a strategy is through its feasible region, i.e. the set of all possible pay-off pairs (for X and Y) that can be achieved against it. For any memory-one strategy $\mathbf{p}$ of X, let

$$\mathcal{C}(\mathbf{p}) := \left\{ \left(\pi_Y(\mathbf{p}, \mathbf{q}), \pi_X(\mathbf{p}, \mathbf{q}) \right) \right\}_{\mathbf{q} \in \mathbf{Mem}_X^1} \tag{2.3}$$

be this feasible region. (Note that, if X uses a memory-one strategy, then it suffices to assume that Y uses a memory-one strategy by the results of Press & Dyson [1].) This subset of the feasible region represents the 'geometry' of strategy $\mathbf{p}$ in the sense that it captures all possible pay-off pairs against an opponent.

In this section, we show that the feasible region for $\mathbf{p} \in \mathbf{Mem}_X^1$ with $\mathbf{p}_{\bullet\bullet} \in (0, 1)^4$ is characterized by playing $\mathbf{p}$ against the following 11 strategies: $(0, 0, 0, 0)$, $(0, 0, 0, 1)$, $(0, 0, 1, 0)$, $(0, 0, 1, 1)$, $(0, 1, 0, 1)$, $(0, 1, 1, 0)$, $(0, 1, 1, 1)$, $(1, 0, 0, 1)$, $(1, 0, 1, 0)$, $(1, 0, 1, 1)$ and $(1, 1, 1, 1)$. In other words, $\mathcal{C}(\mathbf{p})$ is the convex hull of 11 points (figure 1). Therefore, any $\mathbf{p} \in \mathbf{Mem}_X^1$ generates a simple polygon in $\mathbb{R}^2$ whose number of extreme points is uniformly bounded over all game-strategy pairs, $((R, S, T, P), \mathbf{p})$.

Lemma 2.1. *For $\mathbf{q} \in \boldsymbol{Mem}_X^1$ and $x, y \in \{C, D\}$, let $(\mathbf{q}; q_{xy} = q'_{xy})$ be the strategy obtained from $\mathbf{q}$ by changing q_{xy} to $q'_{xy} \in [0, 1]$. If $\mathbf{p}_{\bullet\bullet} \in (0, 1)^4$, $\mathbf{q} \in \boldsymbol{Mem}_X^1$ and $x, y \in \{C, D\}$, then the point $(\pi_Y(\mathbf{p}, \mathbf{q}), \pi_X(\mathbf{p}, \mathbf{q}))$ falls on the line joining $(\pi_Y(\mathbf{p}, (\mathbf{q}; q_{xy} = 0)), \pi_X(\mathbf{p}, (\mathbf{q}; q_{xy} = 0)))$ and $(\pi_Y(\mathbf{p}, (\mathbf{q}; q_{xy} = 1)), \pi_X(\mathbf{p}, (\mathbf{q}; q_{xy} = 1)))$.*

Proof. Let $\mathbf{p}_{\bullet\bullet} \in (0, 1)^4$ and $\mathbf{q} \in \mathbf{Mem}_X^1$. Since the transition matrix of equation (2.2) is just 4×4, one can directly solve for its stationary distribution, $\mu(\mathbf{p}, \mathbf{q})$ (e.g. by using Gaussian elimination

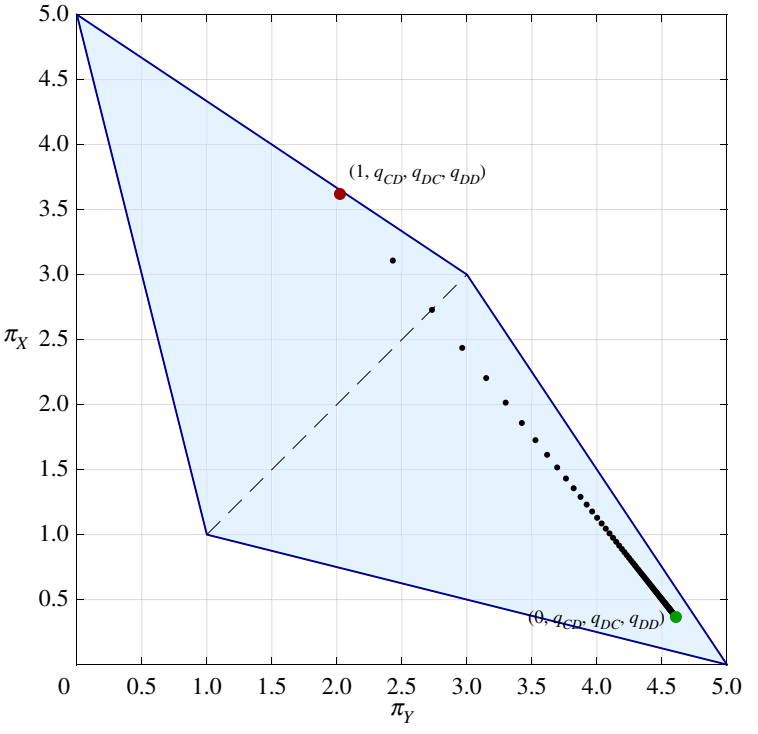

Figure 2. The set of points $(\pi_Y(\mathbf{p}, \mathbf{q}), \pi_X(\mathbf{p}, \mathbf{q}))$, where $\mathbf{p}_{\bullet\bullet} = (0.7876, 0.9856, 0.4095, 0.0301)$ and $\mathbf{q}_{\bullet\bullet} = (q_{CC}, 0.9963, 0.0166, 0.9879)$ as q_{CC} varies between 0 (green) and 1 (red) in uniform increments of 0.01. The resulting points all fall along a line; however, they are not uniformly distributed even though the distribution of q_{CC} is uniform. Parameters: $R = 3, S = 0, T = 5$ and $P = 1$. (Online version in colour.)

or the determinant formula of Press & Dyson [1]). For example, suppose that $x = y = C$. Then, with

$$L(q_{CC}) := \frac{(1 - q_{CC})}{1 + q_{CC} \left(\frac{\begin{array}{c} p_{CC} - p_{CC}p_{CD} + p_{CC}p_{DD} - p_{CC}q_{CD} + p_{CC}q_{DD} + p_{DC}q_{CD} - p_{DD}q_{DD} + p_{CC}p_{CD}q_{CD} \\ - p_{CC}p_{CD}q_{DD} - p_{CC}p_{DD}q_{CD} - p_{CD}p_{DC}q_{CD} - p_{CC}p_{DC}q_{DC} + p_{CC}p_{DC}q_{DD} \\ + p_{CC}p_{DD}q_{DC} + p_{CD}p_{DC}q_{DC} - p_{CD}p_{DD}q_{DC} + p_{DC}p_{DD}q_{CD} + p_{CD}p_{DD}q_{DD} - p_{DC}p_{DD}q_{DD} \end{array}}{\begin{array}{c} p_{DD} - p_{CD} - q_{CD} + q_{DD} + p_{CD}q_{CD} + p_{CD}q_{DC} + p_{DC}q_{CD} - p_{CD}q_{DD} - p_{DD}q_{CD} \\ - p_{DC}q_{DC} + p_{DC}q_{DD} + p_{DD}q_{DC} - p_{DD}q_{DD} - p_{CC}p_{CD}q_{DC} - p_{CD}p_{DC}q_{CD} + p_{CD}p_{DC}q_{DC} \\ + p_{CC}p_{DD}q_{DD} + p_{DC}p_{DD}q_{CD} - p_{DC}p_{DD}q_{DD} - p_{CD}q_{CD}q_{DC} + p_{DC}q_{CD}q_{DC} + p_{CD}q_{DC}q_{DD} \\ - p_{DD}q_{DC}q_{DD} + p_{CC}p_{CD}q_{CD}q_{DC} - p_{CC}p_{DC}q_{CD}q_{DC} - p_{CC}p_{CD}q_{DC}q_{DD} + p_{CC}p_{DC}q_{CD}q_{DD} \\ - p_{CC}p_{DD}q_{CD}q_{DD} - p_{CD}p_{DC}q_{CD}q_{DD} - p_{CD}p_{DD}q_{CD}q_{DC} + p_{CD}p_{DD}q_{CD}q_{DD} \\ + p_{CC}p_{DD}q_{DC}q_{DD} + p_{CD}p_{DC}q_{DC}q_{DD} + p_{DC}p_{DD}q_{CD}q_{DC} - p_{DC}p_{DD}q_{DC}q_{DD} + 1 \end{array}} \right)}, \tag{2.4}$$

one has

$$\begin{aligned} (\pi_Y(\mathbf{p}, \mathbf{q}), \pi_X(\mathbf{p}, \mathbf{q})) &= L(q_{CC}) (\pi_Y(\mathbf{p}, (\mathbf{q}; q_{CC} = 0)), \pi_X(\mathbf{p}, (\mathbf{q}; q_{CC} = 0))) \\ &\quad + (1 - L(q_{CC})) (\pi_Y(\mathbf{p}, (\mathbf{q}; q_{CC} = 1)), \pi_X(\mathbf{p}, (\mathbf{q}; q_{CC} = 1))). \end{aligned} \tag{2.5}$$

Provided $(\pi_Y(\mathbf{p}, (\mathbf{q}; q_{CC} = 0)), \pi_X(\mathbf{p}, (\mathbf{q}; q_{CC} = 0))) \neq (\pi_Y(\mathbf{p}, (\mathbf{q}; q_{CC} = 1)), \pi_X(\mathbf{p}, (\mathbf{q}; q_{CC} = 1)))$, we also have $L(0) = 1$ and $L(1) = 0$. Moreover, one can check that, under this condition, $L'(q_{CC})$ is nowhere equal to 0, and $0 \leqslant L(q_{CC}) \leqslant 1$ for every $q_{CC} \in [0, 1]$. The other cases with $x, y \in \{C, D\}$ are analogous. ■

Remark 2.2. Even when q_{xy} is uniformly distributed between 0 and 1, the corresponding points in the feasible region need not be uniformly distributed between the endpoints corresponding to $q_{xy} = 0$ and $q_{xy} = 1$, respectively (figure 2). This result is therefore somewhat different from

the analogous situation of playing against a mixed action in a stage game, where, for a pay-off function $u : S_X \times S_Y \to \mathbb{R}^2$ and mixed action $\sigma_X \in \Delta(S_X)$ and $\sigma_Y \in \Delta(S_Y)$, one has $u(\sigma_X, \sigma_Y) = \int_{y \in S_Y} u(\sigma_X, y)\, \mathrm{d}\sigma_Y(y)$ due to linearity.

Proposition 2.3. *For any* $\mathbf{p} \in Mem_X^1$ *with* $\mathbf{p}_{\bullet\bullet} \in (0,1)^4$, $\mathcal{C}(\mathbf{p})$ *is the convex hull of the following* 11 *points*:

$$\begin{pmatrix} \pi_X^{(0,0,0,0)} \\ \pi_Y^{(0,0,0,0)} \end{pmatrix} = \begin{pmatrix} \frac{P - Pp_{CD} + Sp_{DD}}{p_{DD} - p_{CD} + 1} \\ \frac{P - Pp_{CD} + Tp_{DD}}{p_{DD} - p_{CD} + 1} \end{pmatrix}; \tag{2.6a}$$

$$\begin{pmatrix} \pi_X^{(0,0,0,1)} \\ \pi_Y^{(0,0,0,1)} \end{pmatrix} = \begin{pmatrix} \frac{P + T - Pp_{CD} + Rp_{DD} + Sp_{DC} - Tp_{CD} - Tp_{DD} - Rp_{CD}p_{DD} + Sp_{CC}p_{DD} - Sp_{DC}p_{DD} + Tp_{CD}p_{DD}}{p_{DC} - 2p_{CD} + p_{CC}p_{DD} - p_{DC}p_{DD} + 2} \\ \frac{P + S - Pp_{CD} + Rp_{DD} - Sp_{CD} - Sp_{DD} + Tp_{DC} - Rp_{CD}p_{DD} + Sp_{CD}p_{DD} + Tp_{CC}p_{DD} - Tp_{DC}p_{DD}}{p_{DC} - 2p_{CD} + p_{CC}p_{DD} - p_{DC}p_{DD} + 2} \end{pmatrix}; \tag{2.6b}$$

$$\begin{pmatrix} \pi_X^{(0,0,1,0)} \\ \pi_Y^{(0,0,1,0)} \end{pmatrix} = \begin{pmatrix} \frac{P - Pp_{DC} + Sp_{DD} + Tp_{DD} - Pp_{CC}p_{CD} + Pp_{CD}p_{DC} + Rp_{CD}p_{DD} - Tp_{CD}p_{DD}}{2p_{DD} - p_{DC} - p_{CC}p_{CD} + p_{CD}p_{DC} + 1} \\ \frac{P - Pp_{DC} + Sp_{DD} + Tp_{DD} - Pp_{CC}p_{CD} + Pp_{CD}p_{DC} + Rp_{CD}p_{DD} - Sp_{CD}p_{DD}}{2p_{DD} - p_{DC} - p_{CC}p_{CD} + p_{CD}p_{DC} + 1} \end{pmatrix}; \tag{2.6c}$$

$$\begin{pmatrix} \pi_X^{(0,0,1,1)} \\ \pi_Y^{(0,0,1,1)} \end{pmatrix} = \begin{pmatrix} \frac{P + T - Pp_{DC} + Rp_{DD} + Sp_{DC} - Tp_{DD} - Pp_{CC}p_{CD} + Pp_{CD}p_{DC} + Rp_{CD}p_{DC} - Rp_{DC}p_{DD} + Sp_{CC}p_{DD} - Sp_{DC}p_{DD} - Tp_{CC}p_{CD} + Tp_{CC}p_{DD}}{2(p_{CC}p_{DD} - p_{CC}p_{CD} + p_{CD}p_{DC} - p_{DC}p_{DD} + 1)} \\ \frac{P + S - Pp_{DC} + Rp_{DD} - Sp_{DD} + Tp_{DC} - Pp_{CC}p_{CD} + Pp_{CD}p_{DC} + Rp_{CD}p_{DC} - Rp_{DC}p_{DD} - Sp_{CC}p_{CD} + Sp_{CC}p_{DD} + Tp_{CC}p_{DD} - Tp_{DC}p_{DD}}{2(p_{CC}p_{DD} - p_{CC}p_{CD} + p_{CD}p_{DC} - p_{DC}p_{DD} + 1)} \end{pmatrix}; \tag{2.6d}$$

$$\begin{pmatrix} \pi_X^{(0,1,0,1)} \\ \pi_Y^{(0,1,0,1)} \end{pmatrix} = \begin{pmatrix} \frac{T + Pp_{DC} + Rp_{DC} - Tp_{CD} - Tp_{DD} - Pp_{CD}p_{DC} - Rp_{CD}p_{DC} + Sp_{CC}p_{DC} + Tp_{CD}p_{DD}}{2p_{DC} - p_{CD} - p_{DD} + p_{CC}p_{DC} - 2p_{CD}p_{DC} + p_{CD}p_{DD} + 1} \\ \frac{S + Pp_{DC} + Rp_{DC} - Sp_{CD} - Sp_{DD} - Pp_{CD}p_{DC} - Rp_{CD}p_{DC} + Sp_{CD}p_{DD} + Tp_{CC}p_{DC}}{2p_{DC} - p_{CD} - p_{DD} + p_{CC}p_{DC} - 2p_{CD}p_{DC} + p_{CD}p_{DD} + 1} \end{pmatrix}; \tag{2.6e}$$

$$\begin{pmatrix} \pi_X^{(0,1,1,0)} \\ \pi_Y^{(0,1,1,0)} \end{pmatrix} = \begin{pmatrix} \frac{Pp_{DC} + Tp_{DD} - Pp_{CC}p_{DC} + Rp_{DC}p_{DD} + Sp_{DC}p_{DD} - Tp_{CD}p_{DD}}{p_{DC} + p_{DD} - p_{CC}p_{DC} - p_{CD}p_{DD} + 2p_{DC}p_{DD}} \\ \frac{Pp_{DC} + Sp_{DD} - Pp_{CC}p_{DC} + Rp_{DC}p_{DD} - Sp_{CD}p_{DD} + Tp_{DC}p_{DD}}{p_{DC} + p_{DD} - p_{CC}p_{DC} - p_{CD}p_{DD} + 2p_{DC}p_{DD}} \end{pmatrix}; \tag{2.6f}$$

$$\begin{pmatrix} \pi_X^{(0,1,1,1)} \\ \pi_Y^{(0,1,1,1)} \end{pmatrix} = \begin{pmatrix} \frac{T + Pp_{DC} + Rp_{DC} - Tp_{DD} - Pp_{CC}p_{DC} + Sp_{CC}p_{DC} - Tp_{CC}p_{CD} + Tp_{CC}p_{DD}}{2p_{DC} - p_{DD} - p_{CC}p_{CD} + p_{CC}p_{DD} + 1} \\ \frac{S + Pp_{DC} + Rp_{DC} - Sp_{DD} - Pp_{CC}p_{DC} - Sp_{CC}p_{CD} + Sp_{CC}p_{DD} + Tp_{CC}p_{DC}}{2p_{DC} - p_{DD} - p_{CC}p_{CD} + p_{CC}p_{DD} + 1} \end{pmatrix}; \tag{2.6g}$$

$$\begin{pmatrix} \pi_X^{(1,0,0,1)} \\ \pi_Y^{(1,0,0,1)} \end{pmatrix} = \begin{pmatrix} -\frac{P + T - Pp_{CC} - Pp_{CD} + Rp_{DD} + Sp_{DC} - Tp_{CC} - Tp_{CD} + Pp_{CC}p_{CD} - Rp_{CD}p_{DD} - Sp_{CC}p_{DC} + Tp_{CC}p_{CD}}{2p_{CC} + 2p_{CD} - p_{DC} - p_{DD} - 2p_{CC}p_{CD} + p_{CC}p_{DC} + p_{CD}p_{DD} - 2} \\ -\frac{P + S - Pp_{CC} - Pp_{CD} + Rp_{DD} - Sp_{CC} - Sp_{CD} + Tp_{DC} + Pp_{CC}p_{CD} - Rp_{CD}p_{DD} + Sp_{CC}p_{CD} - Tp_{CC}p_{DC}}{2p_{CC} + 2p_{CD} - p_{DC} - p_{DD} - 2p_{CC}p_{CD} + p_{CC}p_{DC} + p_{CD}p_{DD} - 2} \end{pmatrix}; \tag{2.6h}$$

$$\begin{pmatrix} \pi_X^{(1,0,1,0)} \\ \pi_Y^{(1,0,1,0)} \end{pmatrix} = \begin{pmatrix} \frac{P - Pp_{CC} - Pp_{DC} + Sp_{DD} + Tp_{DD} + Pp_{CC}p_{DC} + Rp_{CD}p_{DD} - Sp_{CC}p_{DD} - Tp_{CC}p_{DD}}{2p_{DD} - p_{DC} - p_{CC} + p_{CC}p_{DC} - 2p_{CC}p_{DD} + p_{CD}p_{DD} + 1} \\ \frac{P - Pp_{CC} - Pp_{DC} + Sp_{DD} + Tp_{DD} + Pp_{CC}p_{DC} + Rp_{CD}p_{DD} - Sp_{CC}p_{DD} - Tp_{CC}p_{DD}}{2p_{DD} - p_{DC} - p_{CC} + p_{CC}p_{DC} - 2p_{CC}p_{DD} + p_{CD}p_{DD} + 1} \end{pmatrix}; \tag{2.6i}$$

$$\begin{pmatrix} \pi_X^{(1,0,1,1)} \\ \pi_Y^{(1,0,1,1)} \end{pmatrix} = \begin{pmatrix} \frac{P + T - Pp_{CC} - Pp_{DC} + Rp_{DD} + Sp_{DC} - Tp_{CC} + Pp_{CC}p_{DC} + Rp_{CD}p_{DC} - Rp_{DC}p_{DD} - Sp_{CC}p_{DC}}{p_{DD} - 2p_{CC} + p_{CD}p_{DC} - p_{DC}p_{DD} + 2} \\ \frac{P + S - Pp_{CC} - Pp_{DC} + Rp_{DD} - Sp_{CC} + Tp_{DC} + Pp_{CC}p_{DC} + Rp_{CD}p_{DC} - Rp_{DC}p_{DD} - Tp_{CC}p_{DC}}{p_{DD} - 2p_{CC} + p_{CD}p_{DC} - p_{DC}p_{DD} + 2} \end{pmatrix}; \tag{2.6j}$$

and

$$\begin{pmatrix} \pi_X^{(1,1,1,1)} \\ \pi_Y^{(1,1,1,1)} \end{pmatrix} = \begin{pmatrix} \frac{T + Rp_{DC} - Tp_{CC}}{p_{DC} - p_{CC} + 1} \\ \frac{S + Rp_{DC} - Sp_{CC}}{p_{DC} - p_{CC} + 1} \end{pmatrix}. \tag{2.6k}$$

Proof. Press & Dyson [1] show that if X uses a memory-one strategy, $\mathbf{p}$, then any strategy of the opponent, $\mathbf{y}$, can be replaced by a memory-one strategy, $\mathbf{q}$, without changing the pay-offs to X and Y; thus, if X uses a memory-one strategy, one may assume without a loss of generality that Y also uses a memory-one strategy. If $\mathbf{p}_{\bullet\bullet} \in (0,1)^4$ and $\mathbf{q} \in \mathbf{Mem}_X^1$, the fact that $(\pi_Y(\mathbf{p}, \mathbf{q}), \pi_X(\mathbf{p}, \mathbf{q}))$ can be written as a convex combination of the 16 points $\{(\pi_Y(\mathbf{p}, \mathbf{q}'), \pi_X(\mathbf{p}, \mathbf{q}'))\}_{\mathbf{q}'_{\bullet\bullet} \in \{0,1\}^4}$ then follows immediately from lemma 2.1. Moreover, the points corresponding to $(0,0,0,0)$, $(0,1,0,0)$ and $(1,0,0,0)$ are the same, as are the points corresponding to $(1,1,0,1)$, $(1,1,1,0)$ and $(1,1,1,1)$; thus, we can eliminate four points. Furthermore, we can remove the point associated with $(1,1,0,0)$

Table 1. For each point, $\boldsymbol{\pi}_{X,Y}^{(i_1,i_2,i_3,i_3)}$, the feasible region $\mathcal{C}(\mathbf{p})$ cannot (in general) be expressed as the convex hull of the remaining 10 points different from $\boldsymbol{\pi}_{X,Y}^{(i_1,i_2,i_3,i_3)}$. That is, each row gives (i) one of the 11 points of which $\mathcal{C}$ is the convex hull and (ii) an example of a game-strategy pair for which $\boldsymbol{\pi}_{X,Y}^{(i_1,i_2,i_3,i_3)}$ is an extreme point of $\mathcal{C}(\mathbf{p})$.

point	$\left(\begin{smallmatrix} R & S \\ T & P \end{smallmatrix}\right)$	$\mathbf{p}_{\bullet\bullet}$
$\boldsymbol{\pi}_{X,Y}^{(0,0,0,0)}$	$\begin{pmatrix} 4.5953 & -3.5001 \\ -0.1798 & 4.4972 \end{pmatrix}$	(0.0347, 0.8913, 0.9873, 0.1164)
$\boldsymbol{\pi}_{X,Y}^{(0,0,0,1)}$	$\begin{pmatrix} 3.5909 & 3.7183 \\ 3.1091 & 2.6508 \end{pmatrix}$	(0.3420, 0.5591, 0.0468, 0.9941)
$\boldsymbol{\pi}_{X,Y}^{(0,0,1,0)}$	$\begin{pmatrix} 0.1150 & 1.2677 \\ -2.8725 & 1.4290 \end{pmatrix}$	(0.8937, 0.9211, 0.6995, 0.0052)
$\boldsymbol{\pi}_{X,Y}^{(0,0,1,1)}$	$\begin{pmatrix} -0.1523 & 1.7642 \\ -3.3334 & -3.9907 \end{pmatrix}$	(0.5319, 0.4107, 0.9805, 0.0823)
$\boldsymbol{\pi}_{X,Y}^{(0,1,0,1)}$	$\begin{pmatrix} 2.1084 & 0.4235 \\ 4.5449 & -4.5716 \end{pmatrix}$	(0.3897, 0.6428, 0.2422, 0.0300)
$\boldsymbol{\pi}_{X,Y}^{(0,1,1,0)}$	$\begin{pmatrix} 2.5627 & -2.5701 \\ -4.1353 & 4.0437 \end{pmatrix}$	(0.7502, 0.7603, 0.9999, 0.3161)
$\boldsymbol{\pi}_{X,Y}^{(0,1,1,1)}$	$\begin{pmatrix} 0.0600 & 1.1524 \\ 2.8660 & 1.3631 \end{pmatrix}$	(0.1145, 0.9494, 0.7587, 0.9214)
$\boldsymbol{\pi}_{X,Y}^{(1,0,0,1)}$	$\begin{pmatrix} -4.4025 & 1.6813 \\ -2.9162 & 1.1664 \end{pmatrix}$	(0.9629, 0.0020, 0.2554, 0.8444)
$\boldsymbol{\pi}_{X,Y}^{(1,0,1,0)}$	$\begin{pmatrix} 0.1167 & 2.5125 \\ -0.3462 & -4.6919 \end{pmatrix}$	(0.4121, 0.4373, 0.5380, 0.8915)
$\boldsymbol{\pi}_{X,Y}^{(1,0,1,1)}$	$\begin{pmatrix} -0.3787 & 1.1357 \\ 1.5417 & 2.7617 \end{pmatrix}$	(0.2570, 0.5191, 0.1293, 0.9332)
$\boldsymbol{\pi}_{X,Y}^{(1,1,1,1)}$	$\begin{pmatrix} -1.8211 & -3.2300 \\ -4.6281 & -0.4609 \end{pmatrix}$	(0.0009, 0.4996, 0.4362, 0.9653)

because it lies on the line connecting the points associated with $(0,0,0,0)$ and $(1,1,1,1)$. One can easily check that the remaining 11 points have the following property: if point i is removed, then there exist R, S, T, P and $\mathbf{p}$ for which $\mathcal{C}(\mathbf{p})$ is not the convex hull of the 10 points different from i (table 1). Thus, for a general $\mathbf{p}$ and pay-off matrix, all 11 of these points are required. ■

Remark 2.4. $\mathbf{p}$ enforces a linear pay-off relationship if and only if these 11 points are collinear.

Remark 2.5. One needs all 11 of these points for general R, S, T, P and $\mathbf{p}$. However, for any particular game-strategy pair, it is often the case that several of these points are unnecessary because they lie within the convex hull of some other subset of these 11 points; they are typically not all extreme points of $\mathcal{C}(\mathbf{p})$.

3. Reactive learning strategies

In a traditional memory-one strategy, X's probability of playing C depends on the realized actions of the two players, x and y. However, X can observe more than just their pure action against the opponent's; they also know how they arrived at x (i.e. they know the mixed action, σ_X, that resulted in x in the previous round). Of course, X need not be able to see Y's mixed action, but

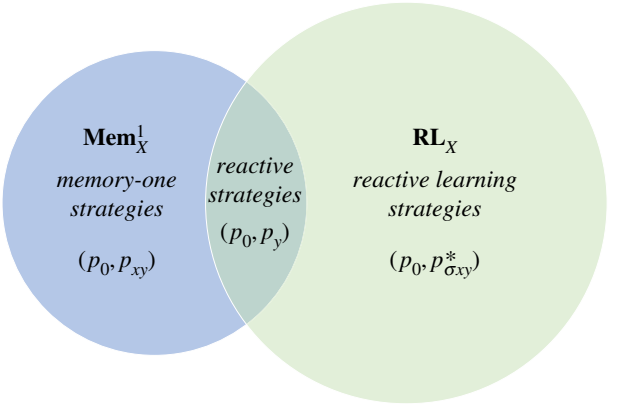

Figure 3. The space of memory-one strategies, $\mathbf{Mem}_X^1$, as it relates to the space of reactive learning strategies, $\mathbf{RL}_X$. Both sets contain the space of reactive strategies [22], which take into account only the last move, y, of the opponent. Whereas a memory-one strategy takes into account the last pure action of X as well, x, a reactive learning strategy uses X's last *mixed* action, $\sigma_X \in [0, 1]$. After each round, a reactive learning strategy uses y to update X's probability of cooperating. $\mathbf{RL}_X$ is 'larger' than $\mathbf{Mem}_X^1$ in the sense that there is an injective map $\mathbf{Mem}_X^1 \to \mathbf{RL}_X$ that is not surjective. (Online version in colour.)

they can still observe the pure action Y played. Therefore, an alternative notion of a memory-one strategy for player X could be defined as follows: after X plays $\sigma_X \in [0, 1]$ and Y plays y, X then chooses a new action based on the distribution $p^*_{\sigma_X y} \in [0, 1]$. In this formulation, p^* is a map from $[0, 1] \times \{C, D\}$ to $[0, 1]$. We refer to such a map, p^*, together with X's initial probability of playing C, p_0, as a 'reactive learning strategy' for player X (figure 3).

In other words, in contrast to $\mathbf{Mem}_X^1 = [0, 1] \times [0, 1]^4$, which can be alternatively described as

$$\mathbf{Mem}_X^1 = [0, 1] \times \left\{ p : \{C, D\} \times \{C, D\} \to [0, 1] \right\}, \tag{3.1}$$

we define the space of reactive learning strategies as

$$\mathbf{RL}_X := [0, 1] \times \left\{ p^* : [0, 1] \times \{C, D\} \to [0, 1] \right\}, \tag{3.2}$$

where $[0, 1]$ indicates the space of mixed actions for X and $\{C, D\}$ indicates the action space for Y. Although $[0, 1]$ is a much larger space than $\{C, D\}$, the updates of mixed actions can be easier to specify using reactive learning strategies since they allow for adaptive modification of an existing mixed action (without the need to devise a new mixed action from scratch after every observed history of play).

Example 3.1. Suppose that player X starts by playing C and D with equal probability, i.e. $p_0 = 1/2$. For fixed $\eta \in [0, 1]$ (the 'learning rate'), cooperation from the opponent leads to $p^*_{\sigma_X C} = (1 - \eta)\sigma_X + \eta$ while defection leads to $p^*_{\sigma_X D} = (1 - \eta)\sigma_X$. Thus, a long pattern of exploitation by Y leads X to defect more often. On the other hand, X does not immediately forgive such behaviour but rather requires Y to cooperate repeatedly to bring X back up to higher levels of cooperation. For example, if X starts with p_0 and Y defects ℓ times in a row, then X subsequently cooperates with probability $(1 - \eta)^\ell p_0$. In order to bring X's probability of cooperation above p_0 once again, Y must then cooperate for T rounds, where

$$T \geqslant \frac{\log\left((1 - p_0)/(1 - (1 - \eta)^\ell p_0)\right)}{\log(1 - \eta)}. \tag{3.3}$$

We refer to this strategy as LTFT because it pushes a player's cooperation probability in the direction of the opponent's last move (figure 4). In this way, a reactive learning strategy can encode more complicated behaviour than a memory-one strategy. Conversely, memory-one

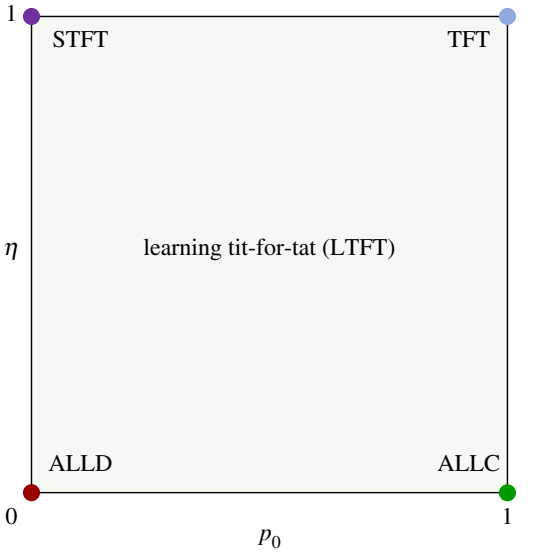

Figure 4. 'Learning tit-for-tat (LTFT)', an analogue of tit-for-tat (TFT) within the space of reactive learning strategies. LTFT is the function of two parameters, p_0 (the initial mixed action) and η (the learning rate). Player X initially plays C with probability p_0. In all subsequent rounds, if X played C with probability σ_X and Y played C (resp. D) in the previous round, in the next round X plays C with probability $p^*_{\sigma_X C} = (1-\eta)\sigma_X + \eta$ (resp. $p^*_{\sigma_X D} = (1-\eta)\sigma_X$). At the corners lie the strategies ALLD (always defect), ALLC (always cooperate), TFT (tit-for-tat) and STFT (suspicious tit-for-tat).

strategies can also encode behaviour not captured by reactive learning strategies, which we discuss further in §3c.

(a) Linear reactive learning strategies

A pertinent question at this point is whether there is a 'natural' map from $\mathbf{Mem}^1_X$ to $\mathbf{RL}_X$. Let $(p_0, \mathbf{p}_{\bullet\bullet}) = (p_0, p_{CC}, p_{CD}, p_{DC}, p_{DD})$ be a memory-one strategy. If (p'_0, p^*) is the corresponding reactive learning strategy, then the first requirement we impose is $p'_0 = p_0$. If $\sigma_X = 1$, then X plays C with probability one. It is therefore reasonable to insist that $p^*_{1y} = p_{Cy}$. Similarly, X plays D with probability one when $\sigma_X = 0$, and we insist that $p^*_{0y} = p_{Dy}$. Suppose now that σ_X and σ'_X are two mixed actions for X. If Y plays $y \in \{C, D\}$, then the responses for X corresponding to σ_X and σ'_X are $p^*_{\sigma_X y}$ and $p^*_{\sigma'_X y}$, respectively. If X plays σ_X with probability $w \in [0,1]$ and σ'_X with probability $1-w$, then it is also natural to insist that the response is $p^*_{\sigma_X y}$ with probability w and $p^*_{\sigma'_X y}$ with probability $1-w$. Thus, for any $\sigma_X \in [0,1]$ and $y \in \{C, D\}$, with these requirements p^* can be written uniquely in terms of $\mathbf{p}_{\bullet\bullet}$ as

$$p^*_{\sigma_X y} = \sigma_X p^*_{1y} + (1-\sigma_X)\, p^*_{0y} = \sigma_X p_{Cy} + (1-\sigma_X)\, p_{Dy}. \tag{3.4}$$

Using this map, one can naturally identify $\mathbf{Mem}^1_X$ with the set of *linear* reactive learning strategies, $\mathbf{LRL}_X \subseteq \mathbf{RL}_X$, consisting of those functions $p^* : [0,1] \times \{C, D\} \to [0,1]$ for which there exist $a, b, c, d \in \mathbb{R}$ with

$$p^*_{\sigma_X C} = \sigma_X a + (1-\sigma_X)\, c \tag{3.5a}$$

and

$$p^*_{\sigma_X D} = \sigma_X b + (1-\sigma_X)\, d. \tag{3.5b}$$

Clearly, any such a, b, c, d must lie in $[0,1]$ since $p^*_{\sigma_X y} \in [0,1]$ for every $\sigma_X \in [0,1]$ and $y \in \{C, D\}$.

Under this correspondence, the strategy of example 3.1 has parameters $(1/2, 1, 1-\eta, \eta, 0)$. But note that this map, $\mathbf{Mem}^1_X \to \mathbf{RL}_X$, is not surjective due to the fact that not every reactive learning

strategy is linear. For example, if $(a,b,c,d) \in [0,1]^4$ and $p^* \in \mathbf{RL}_X$ is the quadratic response function defined by

$$p^*_{\sigma_X C} := (\sigma_X)^2 a + \left(1 - (\sigma_X)^2\right) c \tag{3.6a}$$

and

$$p^*_{\sigma_X D} := (\sigma_X)^2 b + \left(1 - (\sigma_X)^2\right) d, \tag{3.6b}$$

then there exists no $(p_{CC}, p_{CD}, p_{DC}, p_{DD}) \in [0,1]^4$ mapping to p^* provided $a \neq c$ or $b \neq d$.

(b) Stationary distributions

Suppose that (p_0, p^*) and (q_0, q^*) are reactive learning strategies for X and Y, respectively. These strategies generate a Markov chain on the (infinite) space $\{C,D\}^2 \times [0,1]^2$ with transition probabilities between $((x,y),(\sigma_X,\sigma_Y)), ((x',y'),(p^*_{\sigma_X y}, q^*_{\sigma_Y x})) \in \{C,D\}^2 \times [0,1]^2$ given by

$$P_{\left((x,y),(\sigma_X,\sigma_Y)\right)\to\left((x',y'),\left(p^*_{\sigma_X y}, q^*_{\sigma_Y x}\right)\right)} := \begin{cases} p^*_{\sigma_X y} q^*_{\sigma_Y x} & x' = C,\ y' = C, \\ p^*_{\sigma_X y}\left(1 - q^*_{\sigma_Y x}\right) & x' = C,\ y' = D, \\ \left(1 - p^*_{\sigma_X y}\right) q^*_{\sigma_Y x} & x' = D,\ y' = C, \\ \left(1 - p^*_{\sigma_X y}\right)\left(1 - q^*_{\sigma_Y x}\right) & x' = D,\ y' = D. \end{cases} \tag{3.7}$$

To simplify notation, we can also denote the right-hand side of this equation by $p^*_{\sigma_X y}(x') q^*_{\sigma_Y x}(y')$.

If ν is a stationary distribution of this chain, then, for any $((x,y),(\sigma_X,\sigma_Y)) \in \{C,D\}^2 \times [0,1]^2$,

$$\begin{aligned} \nu\Big((x,y),(\sigma_X,\sigma_Y)\Big) &= \int_{\substack{((x',y'),(\sigma'_X,\sigma'_Y)) \\ (p^*_{\sigma'_X y'}, q^*_{\sigma'_Y x'}) = (\sigma_X,\sigma_Y)}} P_{\left((x',y'),(\sigma'_X,\sigma'_Y)\right)\to\left((x,y),(\sigma_X,\sigma_Y)\right)} \, \mathrm{d}\nu\Big(\left(x',y'\right),\left(\sigma'_X,\sigma'_Y\right)\Big) \\ &= \int_{\substack{((x',y'),(\sigma'_X,\sigma'_Y)) \\ (p^*_{\sigma'_X y'}, q^*_{\sigma'_Y x'}) = (\sigma_X,\sigma_Y)}} \sigma_X(x)\,\sigma_Y(y)\, \mathrm{d}\nu\Big(\left(x',y'\right),\left(\sigma'_X,\sigma'_Y\right)\Big). \end{aligned} \tag{3.8}$$

In general, ν is difficult to give explicitly. However, it is possible to understand the marginal distributions on σ_X and σ_Y in more detail (see appendix A). In any case, having an explicit formula for ν is not necessary for obtaining our main result on feasible pay-off regions, which we turn to in the next section.

(c) Feasible pay-off regions

By looking at the feasible region of a strategy, we uncover a nice relationship between a memory-one strategy, $\mathbf{p}$, and its corresponding (linear) reactive learning strategy, $\mathbf{p}^*$. Namely, for every $\mathbf{p} \in \mathbf{Mem}^1_X$, we have $\mathcal{C}(\mathbf{p}^*) \subseteq \mathcal{C}(\mathbf{p})$. In this section, we give a proof of this fact and illustrate some of its consequences.

For $t \geqslant 1$, let $\mathcal{H}_t = (\{C,D\}^2)^t$ be the history of play from time 0 through time $t-1$ [12]. When $t = 0$, $\mathcal{H}_0 = \{\varnothing\}$, where $\varnothing$ denotes the 'empty' history, indicating that no play came before the present encounter. A behavioural strategy for a player specifies, for every possible history of play, a probability of using C in the next encounter. That is, if $\mathcal{H} := \sqcup_{t \geqslant 0} \mathcal{H}_t$, then a behavioural strategy is a map $\mathcal{H} \to [0,1]$. The following lemma shows that when considering the feasible region of a memory-one or reactive learning strategy, one can assume without a loss of generality that the opponent is playing a Markov strategy.

Lemma 3.2. *Let $\mathcal{M} \subseteq \mathcal{B}$ be the set of all Markov strategies, i.e.*

$$\mathcal{M} := \Big\{ \mathbf{y} : \{1,2,\dots\} \times \{C,D\}^2 \to [0,1] \Big\}. \tag{3.9}$$

For any $\mathbf{x} \in \boldsymbol{Mem}^1_X \cup \boldsymbol{RL}_X$, we have $\mathcal{C}(\mathbf{x}) = \{(\pi_Y(\mathbf{x},\mathbf{y}), \pi_X(\mathbf{x},\mathbf{y}))\}_{\mathbf{y} \in \mathcal{M}}$.

Proof. When $\mathbf{p} \in \mathbf{Mem}_X^1$, the lemma follows from [1, appendix A]. Specifically, when X plays $\mathbf{p} \in \mathbf{Mem}_X^1$ against $\mathbf{y} \in \mathcal{B}$, consider the time-$t$ distributions μ_t on $\{C,D\}^2$ and $\overline{\mu}_t$ on $\mathcal{H}_t$. For $(x_{t+1}, y_{t+1}) \in \{C,D\}^2$,

$$\begin{aligned}\mu_{t+1}(x_{t+1}, y_{t+1}) &= \sum_{h_{t+1} \in \mathcal{H}_{t+1}} p_{x_t y_t}(x_{t+1})\, \mathbf{y}_{h_{t+1}}(y_{t+1})\, \overline{\mu}_{t+1}(h_{t+1}) \\ &= \sum_{h_{t+1} \in \mathcal{H}_{t+1}} p_{x_t y_t}(x_{t+1})\, \mathbf{y}_{(h_t,(x_t,y_t))}(y_{t+1})\, \overline{\mu}_{t+1}(h_{t+1}) \\ &= \sum_{(x_t,y_t)\in\{C,D\}^2} p_{x_t y_t}(x_{t+1}) \sum_{h_t \in \mathcal{H}_t} \mathbf{y}_{(h_t,(x_t,y_t))}(y_{t+1})\, \mu_t(x_t, y_t \mid h_t)\, \overline{\mu}_t(h_t). \end{aligned} \tag{3.10}$$

Therefore, the same sequence of distributions $\{\mu_t\}_{t\geqslant 0}$ arises when Y uses the Markov strategy defined by

$$q_{x_t y_t}^{t+1}(y_{t+1}) := \frac{\sum_{h_t \in \mathcal{H}_t} \mathbf{y}_{(h_t,(x_t,y_t))}(y_{t+1})\, \mu_t(x_t, y_t \mid h_t)\, \overline{\mu}_t(h_t)}{\sum_{h_t \in \mathcal{H}_t} \mu_t(x_t, y_t \mid h_t)\, \overline{\mu}_t(h_t)}. \tag{3.11}$$

If $p^* : [0,1] \times \{C,D\} \to [0,1]$ is a reactive learning strategy that X uses against $\mathbf{y} \in \mathcal{B}$, then for every $t \geqslant 0$ there are distributions ν_t on $\{C,D\}^2$, χ_t on $[0,1]$, and $\overline{\nu}_t$ on $\mathcal{H}_t \times [0,1]$. For $(x_{t+1}, y_{t+1}) \in \{C,D\}^2$,

$$\begin{aligned}&\nu_{t+1}(x_{t+1}, y_{t+1}) \\ &= \int_{(h_{t+1},\sigma_X^t)\in\mathcal{H}_{t+1}\times[0,1]} p^*_{\sigma_X^t y_t}(x_{t+1}) \mathbf{y}_{h_{t+1}}(y_{t+1})\, \mathrm{d}\overline{\nu}_{t+1}(h_{t+1}, \sigma_X^t) \\ &= \sum_{(x_t,y_t)\in\{C,D\}^2} \int_{\sigma_X^t\in[0,1]} p^*_{\sigma_X^t y_t}(x_{t+1}) \\ &\quad \times \int_{(h_t,\sigma_X^{t-1})\in\mathcal{H}_t\times[0,1]} \mathbf{y}_{(h_t,(x_t,y_t))}(y_{t+1})\, \mathrm{d}\chi_t(\sigma_X^t \mid (h_t,(x_t,y_t)), \sigma_X^{t-1})\, \mathrm{d}\overline{\nu}_t(h_t, \sigma_X^{t-1}). \end{aligned} \tag{3.12}$$

Consider the Markov strategy for Y with $q_0 := y_\varnothing$ and $q_{x_0 y_0}^1(y_1) := \mathbf{y}_{(x_0,y_0)}(y_1)$. For $t \geqslant 1$, let

$$q_{x_t y_t}^{t+1}(y_{t+1}) := \frac{\int_{\sigma_X^t\in[0,1]} p^*_{\sigma_X^t y_t}(x_{t+1}) \int_{(h_t,\sigma_X^{t-1})\in\mathcal{H}_t\times[0,1]} \mathbf{y}_{(h_t,(x_t,y_t))}(y_{t+1})\, \mathrm{d}\chi_t(\sigma_X^t \mid (h_t,(x_t,y_t)), \sigma_X^{t-1})\, \mathrm{d}\overline{\nu}_t(h_t, \sigma_X^{t-1})}{\int_{\sigma_X^t\in[0,1]} p^*_{\sigma_X^t y_t}(x_{t+1})\, \mathrm{d}\chi_t(\sigma_X^t \mid x_t, y_t)\nu_t(x_t, y_t)}. \tag{3.13}$$

If ν_t' and χ_t' are the analogues of ν_t and χ_t for p^* against $\{\mathbf{q}^t\}_{t\geqslant 1}$, then clearly $\nu_t = \nu_t'$ and $\chi_t = \chi_t'$ for $t = 0, 1$. Suppose that for some $t \geqslant 1$, we have $\nu_t = \nu_t'$ and $\chi_t = \chi_t'$. It follows, then, that at time $t+1$,

$$\begin{aligned}\nu_{t+1}'(x_{t+1}, y_{t+1}) &= \sum_{(x_t,y_t)\in\{C,D\}^2} q_{x_t y_t}^{t+1}(y_{t+1}) \int_{\sigma_X^t\in[0,1]} p^*_{\sigma_X^t y_t}(x_{t+1})\, \mathrm{d}\chi_t'(\sigma_X^t \mid x_t, y_t)\, \nu_t'(x_t, y_t) \\ &= \sum_{(x_t,y_t)\in\{C,D\}^2} q_{x_t y_t}^{t+1}(y_{t+1}) \int_{\sigma_X^t\in[0,1]} p^*_{\sigma_X^t y_t}(x_{t+1})\, \mathrm{d}\chi_t(\sigma_X^t \mid x_t, y_t)\, \nu_t(x_t, y_t) \\ &= \nu_{t+1}(x_{t+1}, y_{t+1}), \end{aligned} \tag{3.14}$$

which gives the desired result for $\mathbf{x} \in \mathbf{RL}_X$. ∎

This lemma leads to a straightforward proof of our main result:

Theorem 3.3. *$\mathcal{C}(\mathbf{p}^*) \subseteq \mathcal{C}(\mathbf{p})$ for every $\mathbf{p} \in \mathbf{Mem}_X^1$.*

Proof. By lemma 3.2, for $\mathbf{x} \in \mathbf{RL}_X$, we may assume the opponent's strategy is Markovian, meaning that it has a memory of one round into the past but can depend on the current round, t. This dependence on t distinguishes a Markov strategy from a memory-one strategy, the latter of

which also has memory of one round into the past but is independent of t. We denote by $\mathcal{M}$ the set of all Markov strategies (equation (3.9)).

Let $\mathbf{p}^*$ be a linear reactive learning strategy for X and suppose that $\mathbf{y} \in \mathcal{M}$. For every $t \geqslant 0$, these strategies generate a distribution v_t^* over $\{C,D\}^2 \times [0,1]$. For any strategy $\mathbf{q}$ against $\mathbf{p}$, there is a sequence of distributions μ_t on $\{C,D\}^2$ generated by these two strategies. We prove the proposition by finding $\{\mathbf{q}^t\}_{t\geqslant 1} \in \mathcal{M}$ such that $\mu_t(x_t,y_t) = v_t^*(\{(x_t,y_t)\} \times [0,1])$ for every $(x_t,y_t) \in \{C,D\}^2$ and $t \geqslant 0$.

Let χ_t be the (marginal) distribution on $\sigma_X^t \in [0,1]$ at time t. For $y_t \in \{C,D\}$, denote by $\chi_t(\cdot \mid y_t)$ this distribution conditioned on Y using action y_t at time t. For $t \geqslant 0$, consider the strategy with $q_0 := y_\varnothing$ and

$$q_{Cy_t}^{t+1}(y_{t+1}) := \frac{\displaystyle\int_{\sigma_X^t\in[0,1]} \sigma_X^t \left(\sigma_X^t y_{Cy_t}^{t+1}(y_{t+1}) + (1-\sigma_X^t)\, y_{Dy_t}^{t+1}(y_{t+1})\right) \mathrm{d}\chi_t(\sigma_X^t \mid y_t)}{\displaystyle\int_{\sigma_X^t\in[0,1]} \sigma_X^t \,\mathrm{d}\chi_t(\sigma_X^t \mid y_t)} \tag{3.15a}$$

and

$$q_{Dy_t}^{t+1}(y_{t+1}) := \frac{\displaystyle\int_{\sigma_X^t\in[0,1]} (1-\sigma_X^t) \left(\sigma_X^t y_{Cy_t}^{t+1}(y_{t+1}) + (1-\sigma_X^t)\, y_{Dy_t}^{t+1}(y_{t+1})\right) \mathrm{d}\chi_t(\sigma_X^t \mid y_t)}{\displaystyle\int_{\sigma_X^t\in[0,1]} (1-\sigma_X^t)\,\mathrm{d}\chi_t(\sigma_X^t \mid y_t)}. \tag{3.15b}$$

Clearly, $\mu_0(x_0,y_0) = v_0^*(\{(x_0,y_0)\} \times [0,1])$ for every $(x_0,y_0) \in \{C,D\}^2$. Suppose, for some $t \geqslant$, that $\mu_t(x_t,y_t) = v_t^*(\{(x_t,y_t)\} \times [0,1])$ for every $(x_t,y_t) \in \{C,D\}^2$. For $(x_{t+1},y_{t+1}) \in \{C,D\}^2$, we then have

$$\begin{aligned}
&\mu_{t+1}(x_{t+1},y_{t+1}) \\
&\quad= \sum_{(x_t,y_t)\in\{C,D\}^2} p_{x_t y_t}(x_{t+1})\, q_{x_t y_t}^{t+1}(y_{t+1})\, \mu_t(x_t,y_t) \\
&\quad= \sum_{y_t\in\{C,D\}} \left(p_{Cy_t}(x_{t+1})\, q_{Cy_t}^{t+1}(y_{t+1})\, \mu_t(C,y_t) + p_{Dy_t}(x_{t+1})\, q_{Dy_t}^{t+1}(y_{t+1})\, \mu_t(D,y_t)\right) \\
&\quad= \sum_{y_t\in\{C,D\}} p_{Cy_t}(x_{t+1}) \int_{\sigma_X^t\in[0,1]} \sigma_X^t \left(\sigma_X^t y_{Cy_t}^{t+1}(y_{t+1}) + (1-\sigma_X^t)\, y_{Dy_t}^{t+1}(y_{t+1})\right) \mathrm{d}\chi_t(\sigma_X^t \mid y_t) \\
&\qquad+ \sum_{y_t\in\{C,D\}} p_{Dy_t}(x_{t+1}) \int_{\sigma_X^t\in[0,1]} (1-\sigma_X^t) \left(\sigma_X^t y_{Cy_t}^{t+1}(y_{t+1}) + (1-\sigma_X^t)\, y_{Dy_t}^{t+1}(y_{t+1})\right) \mathrm{d}\chi_t(\sigma_X^t \mid y_t) \\
&\quad= \sum_{(x_t,y_t)\in\{C,D\}^2} y_{x_t y_t}^{t+1}(y_{t+1}) \int_{\sigma_X^t\in[0,1]} \left(\sigma_X^t p_{Cy_t} + (1-\sigma_X^t)\, p_{Dy_t}\right) \mathrm{d}v_t^*\left(\{(x_t,y_t)\} \times \{\sigma_X^t\}\right) \\
&\quad= \sum_{(x_t,y_t)\in\{C,D\}^2} \int_{\sigma_X^t\in[0,1]} p_{\sigma_X^t y_t}^*(x_{t+1})\, y_{x_t y_t}^{t+1}(y_{t+1})\, \mathrm{d}v_t^*\left(\{(x_t,y_t)\} \times \{\sigma_X^t\}\right) \\
&\quad= v_{t+1}^*\left(\{(x_{t+1},y_{t+1})\} \times [0,1]\right).
\end{aligned} \tag{3.16}$$

Therefore, by induction and the definition of expected pay-off in an iterated game, $\mathcal{C}(\mathbf{p}^*) \subseteq \mathcal{C}(\mathbf{p})$. ■

As a consequence of theorem 3.3, we see that $\mathbf{p}^*$ enforces a linear pay-off relationship [1] whenever $\mathbf{p}$ does. However, the converse need not hold; figure 5(*b*) gives an example in which X's pay-off is a function of Y's when X uses $\mathbf{p}^*$ but not when X uses $\mathbf{p}$. Although this example illustrates an extreme case of when the pay-off region collapses, perhaps the most interesting behaviour is illustrated by figure 5*a*,*c*,*d*. In these examples, we focus on the pay-off regions that can be obtained against memory-one opponents. Using $\mathbf{p}^*$ instead of $\mathbf{p}$ can both bias pay-offs in favour of X and limit potential losses against a spiteful opponent.

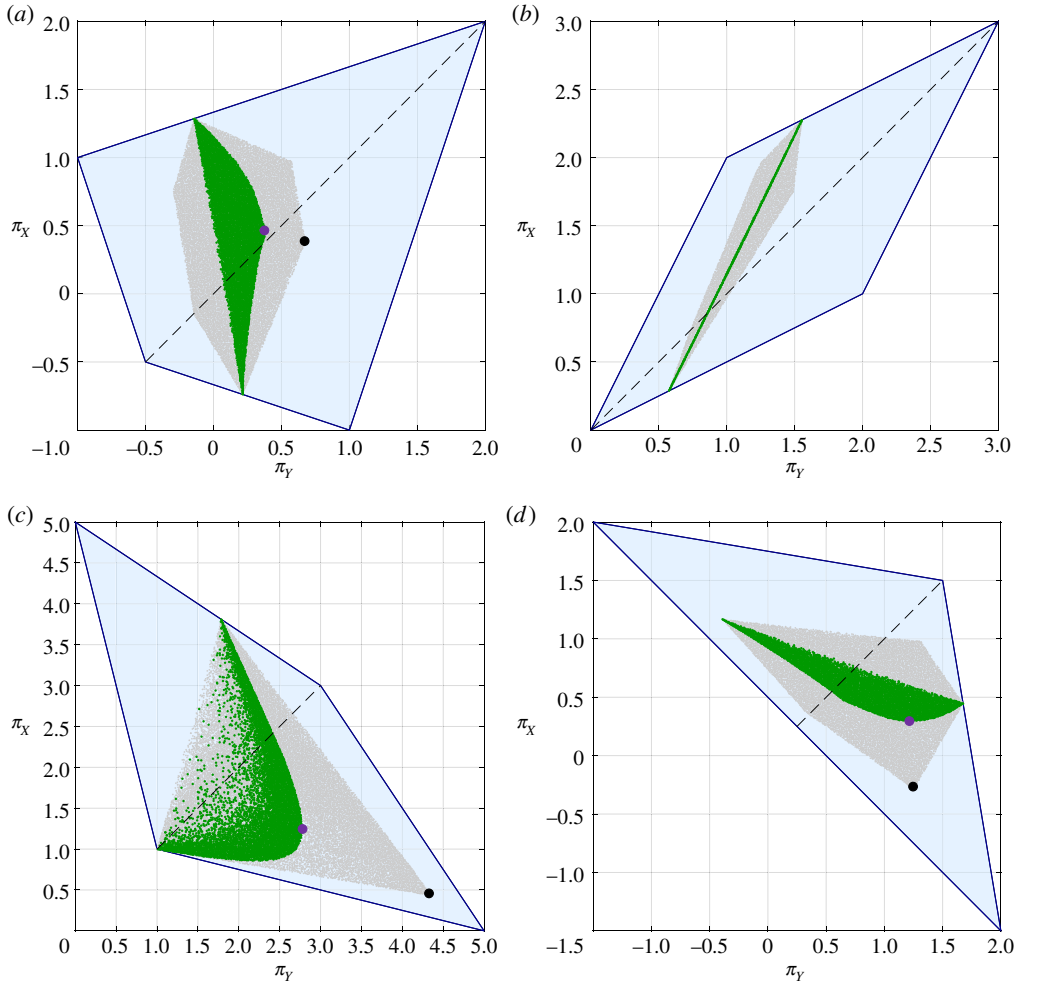

Figure 5. Simulated pay-offs against a fixed memory-one strategy, $\mathbf{p}$ (grey), and its corresponding reactive learning strategy, $\mathbf{p}^*$ (green), as the opponent plays 10^5 randomly chosen strategies $\mathbf{q} \in \mathbf{Mem}_X^1$. (*a*) If the opponent is greedy and wishes to optimize his or her own pay-off only, then upon exploring the space $\mathbf{Mem}_X^1$ for sufficiently long, the pay-offs will end up at the black point when X uses $\mathbf{p}$ and at the magenta point when X uses $\mathbf{p}^*$. In this scenario, $\mathbf{p}$ favours Y having a higher pay-off than X, while $\mathbf{p}^*$ favours X having a higher pay-off than Y. Thus, $\mathbf{p}^*$ extorts a pay-off-maximizing opponent while $\mathbf{p}$ is more generous. (*b*) The pay-offs against $\mathbf{p}^*$ (green) can fall along a line even when those against $\mathbf{p}$ (grey) form a two-dimensional region. In (*c*), by using $\mathbf{p}^*$ instead of $\mathbf{p}$, X can limit the pay-off the opponent receives from the black point to the magenta point. Similarly, in (*d*), X can limit the potential 'punishment' incurred from Y. When X uses $\mathbf{p}$, the opponent can choose a strategy that gives X a negative pay-off (black point). When X uses $\mathbf{p}^*$, no such strategy of the opponent exists, and the worst pay-off X can possibly receive is positive (magenta point). The parameters used are (*a*) $\mathbf{p} = (0.90, 0.50, 0.01, 0.20, 0.90)$ and $R = 2$, $S = -1$, $T = 1$ and $P = 1/2$; (*b*) $\mathbf{p} = (1.0000, 0.6946, 0.0354, 0.1168, 0.3889)$ and $R = 3$, $S = 1$, $T = 2$ and $P = 0$; (*c*) $\mathbf{p} = (0.8623, 0.6182, 0.9528, 0.5601, 0.0001)$ and $R = 3$, $S = 0$, $T = 5$ and $P = 1$; and (*d*) $\mathbf{p} = (0.5626, 0.2381, 0.7236, 0.9537, 0.1496)$ and $R = 1/2$, $S = -3/2$, $T = 2$ and $P = 3/2$. Each coordinate of $\mathbf{q}$ is chosen independently from an arcsine (i.e. Beta(1/2, 1/2)) distribution. (Online version in colour.)

For a memory-one strategy $\mathbf{p} \in \mathbf{Mem}_X^1$, we can ask how the region $\{(\pi_Y(\mathbf{p}, \mathbf{q}), \pi_X(\mathbf{p}, \mathbf{q}))\}_{\mathbf{q} \in \mathbf{Mem}_X^1}$ compares to $\{(\pi_Y(\mathbf{p}^*, \mathbf{q}^*), \pi_X(\mathbf{p}^*, \mathbf{q}^*))\}_{\mathbf{q} \in \mathbf{Mem}_X^1}$. In other words, does the map $\mathbf{p} \mapsto \mathbf{p}^*$ transform the feasible region of a strategy when the opponents are also subjected to this map? Figure 6 demonstrates that this map can significantly distort the distribution of pay-offs within the feasible region.

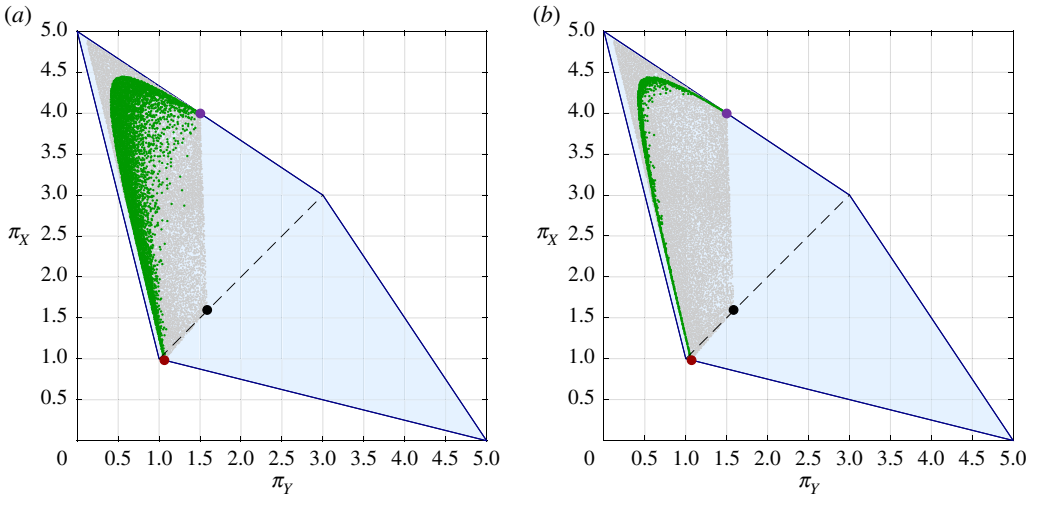

Figure 6. Distortions in the distribution of pay-offs against reactive learning strategies. In both panels, the grey region is formed by playing 10^5 randomly chosen strategies $\mathbf{q} \in \mathbf{Mem}_X^1$ against a fixed strategy $\mathbf{p} \in \mathbf{Mem}_X^1$. The green region in (*a*) arises from simulating the pay-offs of $\mathbf{p}^*$ against 10^5 strategies $\mathbf{q} \in \mathbf{Mem}_X^1$. In (*b*), this same reactive learning strategy, $\mathbf{p}^*$, is simulated against 10^5 strategies $\mathbf{q}^* \in \mathbf{RL}_X$ for $\mathbf{q} \in \mathbf{Mem}_X^1$. In both panels, the optimal outcome for *Y* is the black point when *X* uses $\mathbf{p}$ and the magenta point when *X* uses $\mathbf{p}^*$. The magenta point represents a much better outcome for *X* and only a slightly worse outcome for *Y* than the black point, indicating that $\mathbf{p}^*$ is highly extortionate relative to $\mathbf{p}$ when played against a pay-off-maximizing opponent. In both panels, the parameters are $\mathbf{p} = (0.50, 0.99, 0.40, 0.01, 0.01)$ and $R = 3$, $S = 0$, $T = 5$ and $P = 1$. Each coordinate of $\mathbf{q}$ is chosen independently from an arcsine (i.e. Beta(1/2, 1/2)) distribution. (Online version in colour.)

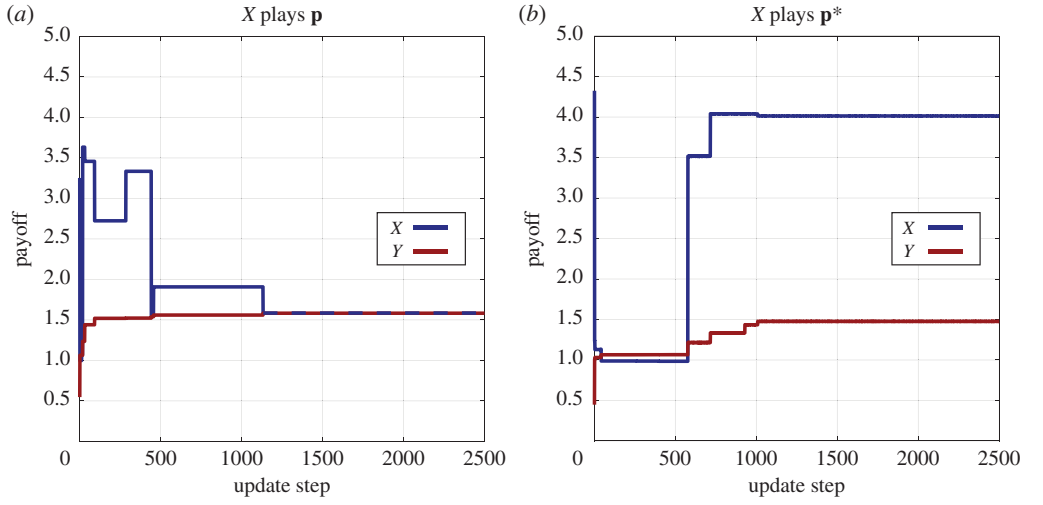

Figure 7. (*a*) Optimization against a memory-one strategy and (*b*) the corresponding reactive learning strategy. In each panel, *X*'s strategy is fixed with parameters $\mathbf{p} = (0.50, 0.99, 0.40, 0.01, 0.01)$. *Y* chooses an initial memory-one strategy, $\mathbf{q}$, from an arcsine distribution. At each update step, *Y* samples another strategy, $\mathbf{q}'$, from the same distribution. If *Y*'s pay-off for playing $\mathbf{q}'$ against *X* exceeds that of playing $\mathbf{q}$ against *X*, then *Y* replaces his or her current strategy with $\mathbf{q}'$. Otherwise, $\mathbf{q}'$ is discarded and *Y* retains $\mathbf{q}$. Over time, this process generates a sequence of pay-off pairs for *X* and *Y*, shown in (*a*,*b*). Relative to $\mathbf{p}$, the reactive learning strategy $\mathbf{p}^*$ is highly extortionate. (Online version in colour.)

(d) Optimization through mutation

Suppose that *X* uses a fixed reactive learning strategy, $\mathbf{p}^*$, for some $\mathbf{p} \in \mathrm{Mem}_X^1$. Starting from some random memory-one strategy, $\mathbf{q}$, the opponent might seek to optimize his or her pay-off through

a series of mutations. In other words, Y is subjected to the following process. First, sample a new strategy $\mathbf{q}' \in \mathbf{Mem}_X^1$. If the pay-off to Y for $\mathbf{q}'$ against $\mathbf{p}^*$ exceeds that of $\mathbf{q}$ against $\mathbf{p}^*$, switch to $\mathbf{q}'$; otherwise, retain $\mathbf{q}$. This step then repeats until Y has a sufficiently high pay-off (or else has not changed strategies in some fixed number of steps). From figure 6, one expects this process to give different results from the same update scheme when X plays the memory-one strategy $\mathbf{p}$ instead of $\mathbf{p}^*$.

As expected, figure 7 shows that this optimization process behaves quite differently against $\mathbf{p}^*$ as it does against $\mathbf{p}$. Whereas using $\mathbf{p}$ in this example results in equitable outcomes, using $\mathbf{p}^*$ gives X a much higher pay-off than Y, indicating extortionate behaviour. One can also imagine other optimization procedures (not covered here), such as when $\mathbf{q}'$ is always sufficiently close to $\mathbf{q}$ (i.e. local mutations). When X uses $\mathbf{p}^*$, a path from the red point to the magenta point in figure 6 through random local sampling of $\mathbf{q}$ typically requires Y to initially accept lower pay-offs. If Y uses $\mathbf{q}^*$ instead of $\mathbf{q}$, as in figure 6*b*, this effect is amplified.

4. Discussion

Our primary focus has been on the feasible region generated by a fixed strategy. This approach to studying X's strategy is inspired by the 'zero-determinant' strategies of Press & Dyson [1], which enforce linear subsets of the feasible region. This perspective has also been expanded to cover so-called partner and rival strategies [2–4], which have proven extremely useful in understanding repeated games from an evolutionary perspective. The feasible region of a memory-one strategy, $\mathbf{p}$, is quite simple and can be characterized as the convex hull of at most 11 points. Furthermore, these points are all straightforward to write down explicitly in terms of the pay-off matrix and the entries of $\mathbf{p}$ (see equation (2.6)). The feasible region of a reactive learning strategy, in terms of its boundary and extreme points, is evidently more complicated in general.

Both memory-one and reactive learning strategies contain the set of all reactive strategies. For every memory-one strategy, $\mathbf{p}$, there exists a corresponding linear reactive learning strategy, $\mathbf{p}^*$, and this correspondence defines an injective map $\mathbf{Mem}_X^1 \to \mathbf{RL}_X$. In general, however, $\mathbf{p}$ cannot be identified with its image, $\mathbf{p}^*$, unless $\mathbf{p}$ is reactive. We make this claim formally using the geometry of a strategy within the feasible region, $\mathcal{C}(\mathbf{p})$, which captures all possible pay-off pairs against an opponent. For any memory-one strategy, we have $\mathcal{C}(\mathbf{p}^*) \subseteq \mathcal{C}(\mathbf{p})$. Therefore, reactive learning strategies generally allow a player to impose greater control over where pay-offs fall within the feasible region than do traditional memory-one strategies. As illustrated in figure 5*a*, this added control can prevent a greedy, self-pay-off-maximizing opponent from obtaining more than X when X uses $\mathbf{p}^*$, even when such an opponent receives an unfair share of the pay-offs when X uses $\mathbf{p}$ instead. The proof of the containment $\mathcal{C}(\mathbf{p}^*) \subseteq \mathcal{C}(\mathbf{p})$ also extends to discounted games, where each pay-off unit received t rounds into the future is valued at δ^t units at present for some 'discounting factor', $\delta \in [0, 1]$.

Another property of the map $\mathbf{Mem}_X^1 \to \mathbf{RL}_X$ sending $\mathbf{p}$ to $\mathbf{p}^*$ is that it distorts the distribution of pay-offs within the feasible region. Since $\mathbf{Mem}_X^1$ can be identified with the space of linear reactive learning strategies under this map, it is natural to compare the region of possible pay-offs when $\mathbf{p}$ plays against memory-one strategies to the one obtained from when $\mathbf{p}^*$ plays against linear reactive learning strategies. These distortions, as illustrated in figure 6, are particularly relevant when X plays against an opponent who is using a process such as simulated annealing to optimize pay-off. One can see from this example that if Y initially has a low pay-off, then with localized strategy exploration they must be willing to accept lower pay-offs before they find a strategy that improves their initial pay-off. This concern is not relevant when Y can simply compute the best response to X's strategy, but it is highly pertinent to evolutionary settings in which the opponent's strategy is obtained through mutation and selection rather than 'computation'.

Reactive learning strategies are also more intuitive than memory-one strategies in some ways. Rather than being a dictionary of mixed actions based on all possible observed outcomes, a reactive learning strategy is simply an algorithm for updating one's tendency to choose a certain action. It, therefore, allows a player to alter their behaviour (mixed action) over time in

response to various stimuli (actions of the opponent). This strategic approach to iterated games is reminiscent of both the Bush–Mosteller model [19] and the weighted majority algorithm [23], although traditionally these models are not studied through the pay-off regions they generate in iterated games. There are several interesting directions for future research in this area. For one, we have mainly considered the space of linear reactive learning strategies, but the space $\mathbf{RL}_X$ is much larger and could potentially exhibit complicated evolutionary dynamics. Furthermore, one could relax the condition that these strategies be reactive and allow them to use X's realized action in addition to X's mixed action. But even without these complications, we have seen that linear reactive learning strategies have quite interesting relationships to traditional memory-one strategies.

Data accessibility. This article does not contain any additional data.

Authors' contributions. All authors designed research, performed research and wrote the paper.

Competing interests. We declare we have no competing interests.

Funding. The authors gratefully acknowledge support from the Lifelong Learning Machines program from DARPA/MTO. Research was sponsored by the Army Research Laboratory (ARL) and was accomplished under cooperative agreement no. W911NF-18-2-0265. The views and conclusions contained in this document are those of the authors and should not be interpreted as representing the official policies, either expressed or implied, of the Army Research Laboratory or the US Government. The US Government is authorized to reproduce and distribute reprints for Government purposes notwithstanding any copyright notation herein.

Acknowledgements. The authors are grateful to Krishnendu Chatterjee, Christian Hilbe and Joshua Plotkin for many helpful conversations and for feedback on earlier versions of this work.

Appendix A. Convergence of mixed actions

Suppose that X and Y use strategies (p_0, p^*) and (q_0, q^*), respectively. Let $\sigma_X^0 = p_0$ and $\sigma_Y^0 = q_0$ be the initial distributions on $\{C, D\}$ for X and Y, respectively. If these distributions are known at time $t \geqslant 0$, then, *on average*, the corresponding distributions at time $t+1$ are given by the system of equations

$$\sigma_X^{t+1} := \sigma_Y^t p^*_{\sigma_X^t C} + \left(1 - \sigma_Y^t\right) p^*_{\sigma_X^t D} \tag{A 1a}$$

and

$$\sigma_Y^{t+1} := \sigma_X^t q^*_{\sigma_Y^t C} + \left(1 - \sigma_X^t\right) q^*_{\sigma_Y^t D}. \tag{A 1b}$$

This system suggests a fixed-point analysis to determine whether the sequence $\{(\sigma_X^t, \sigma_Y^t)\}_{t \geqslant 0}$ converges.

Suppose that $(\sigma_X, \sigma_Y) \in [0,1]^2$ is a fixed point of this system, i.e.

$$\sigma_X = \sigma_Y p^*_{\sigma_X C} + (1 - \sigma_Y)\, p^*_{\sigma_X D} \tag{A 2a}$$

and

$$\sigma_Y = \sigma_X q^*_{\sigma_Y C} + (1 - \sigma_X)\, q^*_{\sigma_Y D}. \tag{A 2b}$$

We consider this system for two types of linear reactive learning strategies: those coming from reactive strategies and those coming from general memory-one strategies under the map $\mathbf{Mem}_X^1 \to \mathbf{RL}_X$.

We first consider reactive strategies of the form (p_C, p_D), where p_C (resp. p_D) is the probability a player uses C after the opponent played C (resp. D). Let (p_C, p_D) and (q_C, q_D) be fixed strategies for X and Y. For these reactive strategies, the system equation (A1) takes the form

$$\sigma_X^{t+1} := \sigma_Y^t p_C + \left(1 - \sigma_Y^t\right) p_D \tag{A 3a}$$

and

$$\sigma_Y^{t+1} := \sigma_X^t q_C + \left(1 - \sigma_X^t\right) q_D. \tag{A 3b}$$

One can easily check that this dynamical system has a unique fixed point, which Hofbauer & Sigmund [24] refer to as the 'asymptotic C-level' of (p_C, p_D) against (q_C, q_D), and which is given explicitly by

$$\sigma_X = \frac{p_C q_D + p_D (1 - q_D)}{1 - (p_C - p_D)(q_C - q_D)} \tag{A 4a}$$

and

$$\sigma_Y = \frac{p_D q_C + (1 - p_D) q_D}{1 - (p_C - p_D)(q_C - q_D)}. \tag{A 4b}$$

Furthermore, we have the following, straightforward convergence result.

Proposition A.1. *If $(p_C, p_D), (q_C, q_D) \in (0,1)^2$, and if $(\sigma_X, \sigma_Y) \in (0,1)^2$ is given by equation* (A 4)*, then*

$$\lim_{t\to\infty} (\sigma_X^t, \sigma_Y^t) = (\sigma_X, \sigma_Y) \tag{A 5}$$

for any initial condition, $(p_0, q_0) \in [0,1]^2$.

Proof. For $(p_C, p_D), (q_C, q_D) \in (0,1)^2$, consider the map

$$\begin{aligned} f : [0,1]^2 &\longrightarrow [0,1]^2 \\ : \begin{pmatrix} x \\ y \end{pmatrix} &\longmapsto \begin{pmatrix} y p_C + (1-y) p_D \\ x q_C + (1-x) q_D \end{pmatrix}. \end{aligned} \tag{A 6}$$

For $(x,y), (x',y') \in [0,1]^2$, we have

$$f(x,y) - f(x',y') = \begin{pmatrix} (y - y')(p_C - p_D) \\ (x - x')(q_C - q_D) \end{pmatrix}. \tag{A 7}$$

It follows that $\|f(x,y) - f(x',y')\| \leqslant \lambda \|(x,y) - (x',y')\|$, where $\lambda := \max\{|p_C - p_D|, |q_C - q_D|\} < 1$. By the contraction mapping theorem, there is then a unique fixed point $(\sigma_X, \sigma_Y) \in [0,1]^2$ such that

$$\lim_{t\to\infty} f^t (p_0, q_0) = (\sigma_X, \sigma_Y) \tag{A 8}$$

for any $(p_0, q_0) \in [0,1]^2$. It is straightforward to check that equation (A 4) is a fixed point of equation (A 3). ∎

In particular, if $\mu := (\sigma_X\sigma_Y, \sigma_X(1-\sigma_Y), (1-\sigma_X)\sigma_Y, (1-\sigma_X)(1-\sigma_Y))$, then a straightforward calculation shows that μ is the stationary distribution of $M((p_C, p_D, p_C, p_D), (q_C, q_D, q_C, q_D))$ (equation (2.2)).

Remark A.2. Proposition A.1 need not hold if p_y and q_x are not strictly between 0 and 1. For example, when X and Y both play TFT, f is a simple involution with $f(x,y) = (y,x)$, which preserves distance.

Consider now the case of general memory-one strategies with $\mathbf{p}_{\bullet\bullet} := (p_{CC}, p_{CD}, p_{DC}, p_{DD})$ for X and $\mathbf{q}_{\bullet\bullet} := (q_{CC}, q_{CD}, q_{DC}, q_{DD})$ for Y. For these strategies, the system defined by equation (A1) has the form

$$\sigma_X^{t+1} := \sigma_Y^t (\sigma_X^t p_{CC} + (1 - \sigma_X^t) p_{DC}) + (1 - \sigma_Y^t)(\sigma_X^t p_{CD} + (1 - \sigma_X^t) p_{DD}) \tag{A 9a}$$

and

$$\sigma_Y^{t+1} := \sigma_X^t (\sigma_Y^t q_{CC} + (1 - \sigma_Y^t) q_{DC}) + (1 - \sigma_X^t)(\sigma_Y^t q_{CD} + (1 - \sigma_Y^t) q_{DD}). \tag{A 9b}$$

In the spirit of proposition A.1, for fixed $\mathbf{p}_{\bullet\bullet}, \mathbf{q}_{\bullet\bullet} \in (0,1)^4$, we could consider the map

$$\begin{aligned} F : [0,1]^2 &\longrightarrow [0,1]^2 \\ : \begin{pmatrix} x \\ y \end{pmatrix} &\longmapsto \begin{pmatrix} y(x p_{CC} + (1-x) p_{DC}) + (1-y)(x p_{CD} + (1-x) p_{DD}) \\ x(y q_{CC} + (1-y) q_{DC}) + (1-x)(y q_{CD} + (1-y) q_{DD}) \end{pmatrix} \end{aligned} \tag{A 10}$$

and analyse its fixed points. At this point, however, a couple of remarks are in order:

(i) F need not be a contraction, even when $\mathbf{p}_{\bullet\bullet}$ and $\mathbf{q}_{\bullet\bullet}$ have entries strictly between 0 and 1. For example, with $\mathbf{p}_{\bullet\bullet} = (0.9566, 0.2730, 0.0056, 0.0095)$ and $\mathbf{q}_{\bullet\bullet} = (0.9922, 0.0918, 0.3217, 0.0054)$,

$$\begin{aligned} 0.0441 &= \|F(0.7404, 0.6928) - F(0.8241, 0.8280)\| \\ &> \|(0.7404, 0.6928) - (0.8241, 0.8280)\| = 0.0253. \end{aligned} \tag{A 11}$$

We would conjecture that this map is an *eventual* contraction, in which case the convergence result of proposition A.1 still holds (although the explicit formulae for σ_X and σ_Y differ from equation (A4)).

(ii) A fixed point of F, (σ_X, σ_Y), even when it exists and is unique, generally does not have the property that $\mu(\mathbf{p}, \mathbf{q}) = (\sigma_X\sigma_Y, \sigma_X(1-\sigma_Y), (1-\sigma_X)\sigma_Y, (1-\sigma_X)(1-\sigma_Y))$, where μ is the stationary distribution of equation (2.2). Furthermore, the long-run mean-frequency distribution on $\{C, D\}^2$ can be distinct from *both* of these distributions, including when the opponent plays $\mathbf{q}$ against $\mathbf{p}^*$ and when they play $\mathbf{q}^*$ against $\mathbf{p}^*$. An example of when these four distributions are pairwise distinct is easy to write down, e.g. $\mathbf{p} = (0.01, 0.01, 0.01, 0.99, 0.01)$ and $\mathbf{q} = (0.99, 0.99, 0.01, 0.99, 0.99)$. All four distributions coincide when $\mathbf{p}$ and $\mathbf{q}$ are both reactive, but in general they can be distinct.

References

1. Press WH, Dyson FJ. 2012 Iterated prisoner's dilemma contains strategies that dominate any evolutionary opponent. *Proc. Natl Acad. Sci. USA* **109**, 10 409–10 413. (doi:10.1073/pnas.1206569109)
2. Akin E. 2015 What you gotta know to play good in the iterated prisoner's dilemma. *Games* **6**, 175–190. (doi:10.3390/g6030175)
3. Hilbe C, Traulsen A, Sigmund K. 2015 Partners or rivals? Strategies for the iterated prisoner's dilemma. *Games Econ. Behav.* **92**, 41–52. (doi:10.1016/j.geb.2015.05.005)
4. Hilbe C, Chatterjee K, Nowak MA. 2018 Partners and rivals in direct reciprocity. *Nat. Human Behav.* **2**, 469–477. (doi:10.1038/s41562-018-0320-9)
5. Nowak M, Sigmund K. 1993 A strategy of win-stay, lose-shift that outperforms tit-for-tat in the prisoner's dilemma game. *Nature* **364**, 56–58. (doi:10.1038/364056a0)
6. Axelrod R. 1984 *The evolution of cooperation*. New York, NY: Basic Books.
7. Lehrer E. 1988 Repeated games with stationary bounded recall strategies. *J. Econ. Theory* **46**, 130–144. (doi:10.1016/0022-0531(88)90153-6)
8. Hauert C, Schuster HG. 1997 Effects of increasing the number of players and memory size in the iterated Prisoner's Dilemma: a numerical approach. *Proc. R. Soc. Lond. B* **264**, 513–519. (doi:10.1098/rspb.1997.0073)
9. Nowak MA. 2006 Five rules for the evolution of cooperation. *Science* **314**, 1560–1563. (doi:10.1126/science.1133755)
10. Hilbe C, Martinez-Vaquero LA, Chatterjee K, Nowak MA. 2017 Memory-*n* strategies of direct reciprocity. *Proc. Natl Acad. Sci. USA* **114**, 4715–4720. (doi:10.1073/pnas.1621239114)
11. Baek SK, Jeong H-C, Hilbe C, Nowak MA. 2016 Comparing reactive and memory-one strategies of direct reciprocity. *Sci. Rep.* **6**, 25676. (doi:10.1038/srep25676)
12. Fudenberg D, Tirole J. 1991 *Game theory*. Cambridge, MA: MIT Press.
13. Posch M. 1999 Win–stay, lose–shift strategies for repeated games—memory length, aspiration levels and noise. *J. Theor. Biol.* **198**, 183–195. (doi:10.1006/jtbi.1999.0909)
14. Dal Bó P. 2005 Cooperation under the shadow of the future: experimental evidence from infinitely repeated games. *Am. Econ. Rev.* **95**, 1591–1604. (doi:10.1257/000282805775014434)
15. Nowak MA. 2006 *Evolutionary dynamics: exploring the equations of life*. Cambridge, MA: Belknap Press.
16. Barlo M, Carmona G, Sabourian H. 2009 Repeated games with one-memory. *J. Econ. Theory* **144**, 312–336. (doi:10.1016/j.jet.2008.04.003)
17. Dal Bó P, Fréchette GR. 2011 The evolution of cooperation in infinitely repeated games: experimental evidence. *Am. Econ. Rev.* **101**, 411–429. (doi:10.1257/aer.101.1.411)
18. Stewart AJ, Plotkin JB. 2016 Small groups and long memories promote cooperation. *Sci. Rep.* **6**, 26889. (doi:10.1038/srep26889)

19. Bush RR, Mosteller F. 1953 A stochastic model with applications to learning. *Ann. Math. Stat.* **24**, 559–585. (doi:10.1214/aoms/1177728914)
20. Roth AE, Erev I. 1995 Learning in extensive-form games: experimental data and simple dynamic models in the intermediate term. *Games Econ. Behav.* **8**, 164–212. (doi:10.1016/s0899-8256(05)80020-x)
21. Izquierdo LR, Izquierdo SS. 2008 Dynamics of the Bush–Mosteller learning algorithm in 2x2 games. In *Reinforcement learning*. I-Tech Education and Publishing.
22. Nowak M, Sigmund K. 1990 The evolution of stochastic strategies in the prisoner's dilemma. *Acta Applicandae Math.* **20**, 247–265. (doi:10.1007/bf00049570)
23. Littlestone N, Warmuth MK. 1989 The weighted majority algorithm. In *30th Annual Symp. on Foundations of Computer Science, Research Triangle Park, NC, USA, 30 October–1 November 1989*, pp. 256–261. (doi:10.1109/sfcs.1989.63487)
24. Hofbauer J, Sigmund K. 1998 *Evolutionary games and population dynamics*. Cambridge, UK: Cambridge University Press. (doi:10.1017/cbo9781139173179)