



# Lipschitz Continuity and Approximate Equilibria

Argyrios Deligkas<sup>1</sup> · John Fearnley<sup>2</sup> · Paul Spirakis<sup>2,3</sup>

Received: 1 March 2017 / Accepted: 30 March 2020 / Published online: 25 April 2020  
© The Author(s) 2020

## Abstract

In this paper, we study games with continuous action spaces and non-linear payoff functions. Our key insight is that Lipschitz continuity of the payoff function allows us to provide algorithms for finding approximate equilibria in these games. We begin by studying Lipschitz games, which encompass, for example, all concave games with Lipschitz continuous payoff functions. We provide an efficient algorithm for computing approximate equilibria in these games. Then we turn our attention to penalty games, which encompass biased games and games in which players take risk into account. Here we show that if the penalty function is Lipschitz continuous, then we can provide a quasi-polynomial time approximation scheme. Finally, we study distance biased games, where we present simple strongly polynomial time algorithms for finding best responses in  $L_1$  and  $L_2^2$  biased games, and then use these algorithms to provide strongly polynomial algorithms that find  $2/3$  and  $5/7$  approximate equilibria for these norms, respectively.

**Keywords** Approximate equilibria · Lipschitz games · Penalty games · Biased games

---

Argyrios Deligkas and John Fearnley are supported by EPSRC Grant EP/L011018/1. The work of Paul Spirakis was supported partially by the Algorithmic Game Theory Project, (co-financed by the European Union European Social Fund) and by Greek National Funds, under the Research Program THALES. Also by the EU ERC Project ALGAME.

---

✉ John Fearnley  
john.fearnley@liverpool.ac.uk

<sup>1</sup> Industrial Engineering and Management, Technion, Haifa, Israel

<sup>2</sup> Department of Computer Science, University of Liverpool, Liverpool, UK

<sup>3</sup> Research Academic Computer Technology Institute (CTI), Patras, Greece

## 1 Introduction

Nash equilibria [29] are the central solution concept in game theory. However, recent advances have shown that computing an *exact* Nash equilibrium is PPAD-complete [8, 10], and so there are unlikely to be polynomial time algorithms for this problem. The hardness of computing exact equilibria has led to the study of *approximate* equilibria: while an exact equilibrium requires that all players have no incentive to deviate from their current strategy, an  $\epsilon$ -approximate equilibrium requires only that their incentive to deviate is less than  $\epsilon$ .

A fruitful line of work has developed studying the best approximations that can be found in polynomial-time for *bimatrix games*, which are two-player strategic form games. There, after a number of papers [5, 11, 12], the best known algorithm was given by Tsaknakis and Spirakis [32], who provide a polynomial time algorithm that finds a 0.3393-equilibrium. The existence of an FPTAS was ruled out by Chen et al. [8] unless  $\text{PPAD} = \text{P}$ . Recently, Rubinstein [31] proved that there is no PTAS for the problem, assuming the Exponential Time Hypothesis for PPAD. However, there is a *quasi-polynomial* approximation scheme given by Lipton et al. [27].

In a strategic form game, the game is specified by giving each player a finite number of strategies, and then specifying a table of payoffs that contains one entry for every possible combination of strategies that the players might pick. The players are allowed to use mixed strategies, and so ultimately the payoff function is a convex combination of the payoffs given in the table. However, some games can only be modelled in a more general setting where the action spaces are continuous, or the payoff functions are non-linear.

For example, Rosen's seminal work [30] considered *concave games*, where each player picks a vector from a convex set. The payoff to each player is specified by a function that satisfies the following condition: if every other player's strategy is fixed, then the payoff to a player is a concave function over his strategy space. Rosen proved that concave games always have an equilibrium. A natural subclass of concave games, studied by Caragiannis et al. [6], is the class of biased games. A biased game is defined by a strategic form game, a *base strategy* and a *penalty function*. The players play the strategic form game as normal, but they all suffer a penalty for deviating from their base strategy. This penalty can be a non-linear function, such as the  $L_2^2$  norm.

In this paper, we study the computation of approximate equilibria in such games. Our main observation is that Lipschitz continuity of the players' payoff functions (with respect to changes in the strategy space) allows us to provide algorithms that find approximate equilibria. Several papers have studied how the Lipschitz continuity of the players' payoff functions affects the existence, the quality, and the complexity of the equilibria of the underlying game. Azrieli and Shmaya [1] studied many player games and derived bounds for the Lipschitz constant of the payoff functions for the players that guarantees the existence of pure approximate equilibria for the game. We have to note though, that the games Azrieli and Shmaya study are significantly different from our games. In [1] the

Lipschitz coefficient refers to the payoff function of player  $i$  as a function of  $\mathbf{x}_{-i}$ , the strategy profile for the rest of the players. This means that the Lipschitz coefficient for each player  $i$  is defined as the Lipschitz constant of his utility function when  $x_i$  is fixed and  $\mathbf{x}_{-i}$  is a variable. In this paper, the Lipschitz coefficient refers to the payoff function of player  $i$  as a function of  $x_i$  when the  $\mathbf{x}_{-i}$  is fixed. We used this definition of the Lipschitz continuity in order to follow Rosen's definition of concave games that requires the payoff function of player  $i$  to be concave for every fixed strategy profile for the rest of the players. Daskalakis and Papadimitriou [13] proved that anonymous games possess pure approximate equilibria whose quality depends on the Lipschitz constant of the payoff functions and the number of pure strategies the players have and proved that these approximate equilibria can be computed in polynomial time. Furthermore, they gave a polynomial-time approximation scheme for anonymous games with many players and constant number of pure strategies. Babichenko [2] presented a best-reply dynamic for  $n$ -player binary-action Lipschitz anonymous games which reaches an approximate pure equilibrium in  $O(n \log n)$  steps. Deb and Kalai [14] studied how some variants of the Lipschitz continuity of the utility functions are sufficient to guarantee hindsight stability of equilibria.

## 1.1 Our Contribution

*$\lambda_p$ -Lipschitz Games* We begin by studying a very general class of games, where each player's strategy space is continuous, and represented by a convex set of vectors, and where the only restriction is that the payoff function is  $\lambda_p$ -Lipschitz continuous, that is to be Lipschitz continuous with respect to the  $L_p$  norm. This class is so general that exact equilibria, and even approximate equilibria may not exist. Nevertheless, we give an efficient algorithm that either outputs an  $\epsilon$ -equilibrium, or determines that the game has no exact equilibria. More precisely, for  $M$  player games with a strategy space defined as the convex hull of  $n$  vectors, that have  $\lambda_p$ -Lipschitz continuous payoff functions in the  $L_p$  norm, for  $p \geq 2$ , we either compute an  $\epsilon$ -equilibrium or determine that no exact equilibrium exists in time  $O(Mn^{Mk+l})$ , where  $k = O\left(\frac{\lambda^2 M p \gamma^2}{\epsilon^2}\right)$  and  $l = O\left(\frac{\lambda^2 p \gamma^2}{\epsilon^2}\right)$ , where  $\gamma = \max \|\mathbf{x}\|_p$  over all  $\mathbf{x}$  in the strategy space. Observe that this is a polynomial time algorithm when  $\lambda$ ,  $p$ ,  $\gamma$ ,  $M$ , and  $\epsilon$  are constant.

To prove this result, we utilize a recent result of Barman [4], which states that for every vector in a convex set, there is another vector that is  $\epsilon$  close to the original in the  $L_p$  norm, and is a convex combination of  $b$  points on the convex hull, where  $b$  depends on  $p$  and  $\epsilon$ , but does not depend on the dimension. Using this result, and the Lipschitz continuity of the payoffs, allows us to reduce the task of finding an  $\epsilon$ -equilibrium to checking only a small number of strategy profiles, and thus we get a brute-force algorithm that is reminiscent of the QPTAS given by Lipton et al. [27] for bimatrix games and by the QPTAS of Babichenko et al. [3] for many player games.

However, life is not so simple for us. Since we study a very general class of games, verifying whether a given strategy profile is an  $\epsilon$ -equilibrium is a non-trivial

task. It requires us to compute a *regret* for each player, which is the difference between the player's best response payoff and their actual payoff. Computing a best response in a bimatrix game is trivial, but for  $\lambda_p$ -Lipschitz games, it may be a hard problem. We get around this problem by instead giving an algorithm to compute *approximate* best responses. Hence we find *approximate* regrets, and it turns out that this is sufficient for our algorithm to work.

**Penalty Games** We then turn our attention to two-player *penalty games*. In these games, the players play a strategic form game, and their utility is the payoff achieved in the game *minus* a penalty. The penalty function can be an arbitrary function that depends on the player's strategy. This is a general class of games that encompasses a number of games that have been studied before. The biased games studied by Caragiannis et al. [6] are penalty games where the penalty is determined by the amount that a player deviates from a specified base strategy. The biased model was studied in the past by psychologists [33] and it is close to what they call *anchoring* [7, 24]. Anchoring is common in poker<sup>1</sup> and in fact there are several papers on poker that are reminiscent of anchoring [21–23]. In their seminal paper, Fiat and Papadimitriou [20] introduced a model for *risk prone* games, which resemble penalty games since the risk component can be encoded as a penalty. Mavronicolas and Monien [28] followed this line of research and provided results on the complexity of deciding if such games possess or not an equilibrium.

We again show that Lipschitz continuity helps us to find approximate equilibria. The only assumption that we make is that the penalty function is Lipschitz continuous in an  $L_p$  norm with  $p \geq 2$ . Again, this is a weak restriction, and it does not guarantee that exact equilibria exist. Even so, we give a quasi-polynomial time algorithm that either finds an  $\epsilon$ -equilibrium, or verifies that the game has no exact equilibrium.

Our result can be seen as a generalisation of the QPTAS given by Lipton et al. [27] for bimatrix games. Their approach is to show the existence of an approximate equilibrium with a logarithmic support. They proved this via the probabilistic method: if we know an exact equilibrium of a bimatrix game, then we can take logarithmically many samples from the strategies, and playing the sampled strategies uniformly will be an approximate equilibrium with positive probability. We take a similar approach, but since our games are more complicated, our proof is necessarily more involved. In particular, for Lipton et al. proving that the sampled strategies are an approximate equilibrium only requires showing that the expected payoff is close to the best response payoff. In penalty games, best response strategies are not necessarily pure, and so the events that we must consider are more complex.

**Distance Biased Games** Finally, we consider distance biased games, which form a subclass of penalty games that have been studied recently by Caragiannis et al. [6]. They showed that, under very mild assumptions on the bias function, biased games always have an exact equilibrium. Furthermore, for the case where the bias function is either the  $L_1$  norm, or the  $L_2^2$  norm, they give an exponential time algorithm for finding an exact equilibrium.

<sup>1</sup> <http://www.pokerology.com/articles/anchoring-bias/>.

Our results for penalty games already give a QPTAS for biased games, but we are also interested in whether there are polynomial-time algorithms that can find non-trivial approximations. We give a positive answer to this question for games where the bias is the  $L_1$  norm, or the  $L_2^2$  norm. We follow the well-known approach of Daskalakis et al. [12], who gave a simple algorithm for finding a 0.5-approximate equilibrium in a bimatrix game.

We show that this algorithm also works for biased games, although the generalisation is not entirely trivial. Again, this is because best responses cannot be trivially computed in biased games. For the  $L_1$  norm, best responses can be computed via linear programming, and for the  $L_2^2$  norm, best responses can be formulated as a quadratic program, and it turns out that this particular QP can be solved in polynomial time by the ellipsoid method. However, none of these algorithms are strongly polynomial. We show that, for each of the norms, best responses can be found by a simple strongly-polynomial combinatorial algorithm. We then analyse the quality of approximation provided by the technique of Daskalakis et al. [12]. We obtain a strongly polynomial algorithm for finding a  $2/3$  approximation in  $L_1$  biased games, and a strongly polynomial algorithm for finding a  $5/7$  approximation in  $L_2^2$  biased games.

## 2 Preliminaries

We start by fixing some notation. For each positive integer  $n$  we use  $[n]$  to denote the set  $\{1, 2, \dots, n\}$ , we use  $\Delta^n$  to denote the  $(n - 1)$ -dimensional simplex, and  $\|x\|_p^q$  to denote the  $(p, q)$ -norm of a vector  $x \in \mathbb{R}^d$ , i.e.  $\|x\|_p^q = (\sum_{i \in [d]} |x_i|^p)^{q/p}$ . When  $q = 1$ , then we will omit it for notation simplicity. Given a set  $X = \{x_1, x_2, \dots, x_n\} \subset \mathbb{R}^d$ , we use  $\text{conv}(X)$  to denote the convex hull of  $X$ . A vector  $y \in \text{conv}(X)$  is said to be  $k$ -uniform with respect to  $X$  if there exists a size  $k$  multiset  $S$  of  $[n]$  such that  $y = \frac{1}{k} \sum_{i \in S} x_i$ . When  $X$  is clear from the context we will simply say that a vector is  $k$ -uniform without mentioning that uniformity is with respect to  $X$ . We will use the notion of the  $\lambda_p$ -Lipschitz continuity.

**Definition 1** ( $\lambda_p$ -Lipschitz) A function  $f : A \rightarrow \mathbb{R}$ , is  $\lambda_p$ -Lipschitz continuous if for every  $x$  and  $y$  in  $A$ , it is true that  $|f(x) - f(y)| \leq \lambda \cdot \|x - y\|_p$ .

*Games and Strategies* A game with  $M$  players can be described by a set of available actions for each player and a utility function for each player that depends both on his chosen action and the actions the rest of the players chose. For each player  $i \in [M]$  we use  $S_i$  to denote his set of available actions and we call it his *strategy space*. We will use  $x_i \in S_i$  to denote a specific action chosen by player  $i$  and we will call it the *strategy* of player  $i$ , we use  $\mathbf{x} = (x_1, \dots, x_M)$  to denote a *strategy profile* of the game, and we will use  $\mathbf{x}_{-i}$  to denote the strategy profile where the player  $i$  is excluded, i.e.  $\mathbf{x}_{-i} = (x_1, \dots, x_{i-1}, x_{i+1}, \dots, x_M)$ . We use  $T_i(x_i, \mathbf{x}_{-i})$  to denote the utility of player  $i$  when he plays the strategy  $x_i$  and the rest of the players play according to the strategy profile  $\mathbf{x}_{-i}$ . We make the standard assumption that the utilities for the

players are in  $[0, 1]$ , i.e.  $T_i(x_i, \mathbf{x}_{-i}) \in [0, 1]$  for every  $i$  and every possible combination of  $x_i$  and  $\mathbf{x}_{-i}$ . A strategy  $\hat{x}_i$  is a *best response* against the strategy profile  $\mathbf{x}_{-i}$ , if  $T_i(\hat{x}_i, \mathbf{x}_{-i}) \geq T_i(x_i, \mathbf{x}_{-i})$  for all  $x_i \in S_i$ . The *regret* player  $i$  suffers under a strategy profile  $\mathbf{x}$  is the difference between the utility of his best response and his utility under  $\mathbf{x}$ , i.e.  $T_i(\hat{x}_i, \mathbf{x}_{-i}) - T_i(x_i, \mathbf{x}_{-i})$ .

An  $n \times n$  bimatrix game is a pair  $(R, C)$  of two  $n \times n$  matrices:  $R$  gives payoffs for the *row* player and  $C$  gives the payoffs for the *column* player. We make the standard assumption that all payoffs lie in the range  $[0, 1]$ . If  $\mathbf{x}$  and  $\mathbf{y}$  are mixed strategies for the row and the column player, respectively, then the expected payoff for the row player under strategy profile  $(\mathbf{x}, \mathbf{y})$  is given by  $\mathbf{x}^T R \mathbf{y}$  and for the column player by  $\mathbf{x}^T C \mathbf{y}$ .

## 2.1 Game Classes

In this section we define the classes of games studied in this paper. We will study  $\lambda_p$ -Lipschitz games, penalty games, biased games and distance biased games.

**$\lambda_p$ -Lipschitz Games** This is a very general class of games, where each player's strategy space is continuous, and represented by a convex set of vectors, and where the only restriction is that the payoff function is  $\lambda_p$ -Lipschitz continuous for some  $p \geq 2$ . This class is so general that exact equilibria, and even approximate equilibria may not exist.

Formally, an  $M$ -player  $\lambda_p$ -Lipschitz game  $\mathfrak{G}$  can be defined by the tuple  $(M, n, \lambda, p, \gamma, T)$  where:

- the strategy space  $S_i$  of player  $i$  is the convex hull of at most  $n$  vectors  $y_1, \dots, y_n$  in  $\mathbb{R}^d$ ;
- $T$  is a set of  $\lambda_p$ -Lipschitz continuous functions and each  $T_i(\mathbf{x}) \in T$ ;
- $\gamma$  is a parameter that intuitively shows how large the strategy space of the players is, formally  $\max_{x_i \in S_i} \|x_i\|_p \leq \gamma$  for every  $i \in [M]$ .

In what follows, we will assume that the Lipschitz continuity of a game,  $\lambda_p$ , is bounded by a number polylogarithmic in the size of the game  $\mathfrak{G}$ . Observe that in general this does not hold for normal form games, since there exist bimatrix games that are not constant Lipschitz continuous.

**Two-Player Penalty Games** A two-player penalty game  $\mathcal{P}$  is defined by a tuple  $(R, C, \mathbf{f}_r(\mathbf{x}), \mathbf{f}_c(\mathbf{y}))$ , where  $(R, C)$  is an  $n \times n$  bimatrix game and  $\mathbf{f}_r(\mathbf{x})$  and  $\mathbf{f}_c(\mathbf{y})$  are the penalty functions for the row and the column player respectively. The utilities for the players under a strategy profile  $(\mathbf{x}, \mathbf{y})$ , denoted by  $T_r(\mathbf{x}, \mathbf{y})$  and  $T_c(\mathbf{x}, \mathbf{y})$ , are given by

$$T_r(\mathbf{x}, \mathbf{y}) = \mathbf{x}^T R \mathbf{y} - \mathbf{f}_r(\mathbf{x}) \quad \text{and} \quad T_c(\mathbf{x}, \mathbf{y}) = \mathbf{x}^T C \mathbf{y} - \mathbf{f}_c(\mathbf{y})$$

where  $\mathbf{f}_r(\mathbf{x})$  and  $\mathbf{f}_c(\mathbf{y})$  are non negative functions.

We will use  $\mathcal{P}_{\lambda_p}$  to denote the set of two-player penalty games with  $\lambda_p$ -Lipschitz penalty functions. A special class of penalty games is obtained when  $\mathbf{f}_r(\mathbf{x}) = \mathbf{x}^T \mathbf{x}$  and  $\mathbf{f}_c(\mathbf{y}) = \mathbf{y}^T \mathbf{y}$ . We call these games as *inner product* penalty games.

**Two-Player Biased Games** This is a subclass of penalty games, where extra constraints are added to the penalty functions  $\bar{f}_r(\mathbf{x})$  and  $\bar{f}_c(\mathbf{y})$  of the players. In this class of games there is a *base strategy* and for each player and the penalty they receive is increasing with the distance between the strategy they choose and their base strategy. Formally, the row player has a base strategy  $\mathbf{p} \in \Delta^n$ , the column player has a base strategy  $\mathbf{q} \in \Delta^n$  and their strictly increasing penalty functions are defined as  $\bar{f}_r(\|\mathbf{x} - \mathbf{p}\|_r^s)$  and  $\bar{f}_c(\|\mathbf{y} - \mathbf{q}\|_m^l)$  respectively.

**Two-Player Distance Biased Games** This is a special class of biased games where the penalty function is a fraction of the distance between the base strategy of the player and his chosen strategy. Formally, a two player distance biased game  $\mathcal{B}$  is defined by a tuple  $(R, C, \mathbf{b}_r(\mathbf{x}, \mathbf{p}), \mathbf{b}_c(\mathbf{y}, \mathbf{q}), d_r, d_c)$ , where  $(R, C)$  is a bimatrix game,  $\mathbf{p} \in \Delta^n$  is a base strategy for the row player,  $\mathbf{q} \in \Delta^n$  is a base strategy for the column player,  $\mathbf{b}_r(\mathbf{x}, \mathbf{p}) = \|\mathbf{x} - \mathbf{p}\|_r^s$  and  $\mathbf{b}_c(\mathbf{y}, \mathbf{q}) = \|\mathbf{y} - \mathbf{q}\|_m^l$  are the penalty functions for the row and the column player respectively. The utilities for the players under a strategy profile  $(\mathbf{x}, \mathbf{y})$ , denoted by  $T_r(\mathbf{x}, \mathbf{y})$  and  $T_c(\mathbf{x}, \mathbf{y})$ , are given by

$$T_r(\mathbf{x}, \mathbf{y}) = \mathbf{x}^T R \mathbf{y} - d_r \cdot \mathbf{b}_r(\mathbf{x}, \mathbf{p}) \quad \text{and} \quad T_c(\mathbf{x}, \mathbf{y}) = \mathbf{x}^T C \mathbf{y} - d_c \cdot \mathbf{b}_c(\mathbf{y}, \mathbf{q}),$$

where  $d_r$  and  $d_c$  are non negative constants.

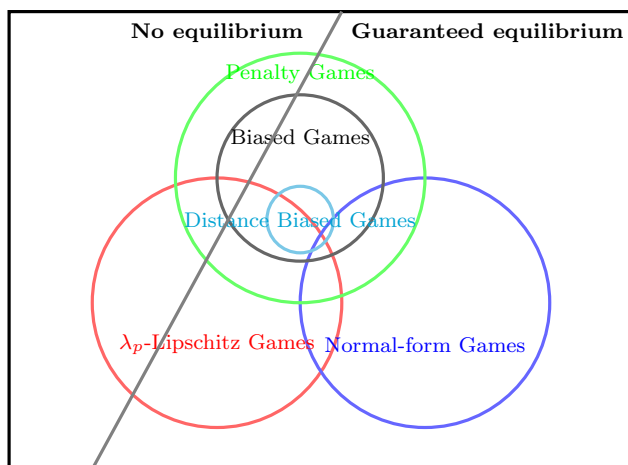
**Solution Concepts** A strategy profile is an equilibrium if no player can increase his utility by unilaterally changing his strategy. A relaxed version of this concept is the approximate equilibrium, or  $\epsilon$ -equilibrium, in which no player can increase his utility more than  $\epsilon$  by unilaterally changing his strategy. Formally, a strategy profile  $\mathbf{x}$  is an  $\epsilon$ -equilibrium in a game  $\mathcal{G}$  if for every player  $i \in [M]$  it holds that

$$T_i(x_i, \mathbf{x}_{-i}) \geq T_i(x'_i, \mathbf{x}_{-i}) - \epsilon \quad \text{for all } x'_i \in S_i.$$

## 2.2 Comparison Between the Classes of Games

Before we present our algorithms for computing approximate equilibria for Lipschitz games it would be useful to describe the differences between the classes of games and to state what the current status of the equilibrium existence for each class. Figure 1 shows the relation between the games' classes. It is well known that normal-form games possess an equilibrium [29], known as Nash equilibrium. On the other hand, Fiat and Papadimitriou [20] and Mavronicolas and Monien [28] studied games with penalty functions that capture risk. They showed that there exist games with no equilibrium and they proved that it is NP-complete to decide whether a game possess an equilibrium or not. Caragiannis et al. [6] studied biased games and proved that a large family of biased games possess an equilibrium. Distance biased games fall in this family and thus always possess an equilibrium. Finally, for  $\lambda_p$ -Lipschitz games it is an interesting open question whether they always possess an equilibrium, or if there are cases that do not possess an equilibrium.

The main difference between  $\lambda_p$ -Lipschitz games and penalty games is on the Lipschitz coefficient of the payoff functions. Our algorithm for  $\lambda_p$ -Lipschitz games



**Fig. 1** A map that depicts the relations between the games studied in this paper and our current knowledge for the equilibrium existence for each class

are efficient for “small” values of the Lipschitz coefficient, where small means a constant or polylogarithmic in the size of the game. Note however that normal form games and bimatrix games might have large Lipschitz coefficient, linear on the size of the game. So, our algorithm for  $\lambda_p$ -Lipschitz games cannot directly apply on penalty games, since the normal-form part of the game can have large Lipschitz coefficient. For that reason we present our second algorithm that tackles penalty games.

### 3 Approximate Equilibria in $\lambda_p$ -Lipschitz Games

In this section, we give an algorithm for computing approximate equilibria in  $\lambda_p$ -Lipschitz games. Note that our definition of a  $\lambda_p$ -Lipschitz game does not guarantee that an equilibrium always exists. Our technique can be applied *irrespective* of whether an exact equilibrium exists. If an exact equilibrium does exist, then our technique will always find an  $\epsilon$ -equilibrium. If an exact equilibrium does not exist, then our algorithm either finds an  $\epsilon$ -equilibrium or reports that the game does not have an exact equilibrium.

In order to derive our algorithm we will utilize the following theorem of Barman [4]. Intuitively, Barman’s theorem states that we can approximate any point  $\mu$  in the convex hull of  $n$  points using a uniform point  $\mu'$  that needs only “few” samples from  $\mu$  to construct it.

**Theorem 2** (Barman [4]) *Given a set of vectors  $Z = \{z_1, z_2, \dots, z_n\} \subset \mathbb{R}^d$ , let  $\text{conv}(Z)$  denote the convex hull of  $Z$ . Furthermore, let  $\gamma := \max_{z \in Z} \|z\|_p$  for some  $2 \leq p < \infty$ . For every  $\epsilon > 0$  and every  $\mu \in \text{conv}(Z)$ , there exists a  $\frac{4py^2}{\epsilon^2}$ -uniform vector  $\mu' \in \text{conv}(Z)$  such that  $\|\mu - \mu'\|_p \leq \epsilon$ .*



Combining Theorem 2 with the Definition 1 we get the following lemma.

**Lemma 3** *Let  $Z = \{z_1, z_2, \dots, z_n\} \subset \mathbb{R}^d$ , let  $f : \text{conv}(Z) \rightarrow \mathbb{R}$  be a  $\lambda_p$ -Lipschitz continuous function for some  $2 \leq p < \infty$ , let  $\epsilon > 0$  and let  $k = \frac{4\lambda^2 p \gamma^2}{\epsilon^2}$ , where  $\gamma := \max_{z \in Z} \|z\|_p$ . Furthermore, let  $f(\mathbf{z}^*)$  be the optimum value of  $f$ . Then we can compute a  $k$ -uniform point  $\mathbf{z}' \in \text{conv}(Z)$  in time  $O(n^k)$ , such that  $|f(\mathbf{z}^*) - f(\mathbf{z}')| < \epsilon$ .*

**Proof** From Theorem 2 we know that for the chosen value of  $k$  there exists a  $k$ -uniform point  $\mathbf{z}'$  such that  $\|\mathbf{z}' - \mathbf{z}^*\|_p < \epsilon/\lambda$ . Since the function  $f(\mathbf{z})$  is  $\lambda_p$ -Lipschitz continuous, we get that  $|f(\mathbf{z}') - f(\mathbf{z}^*)| < \epsilon$ . In order to compute this point we have to exhaustively evaluate the function  $f$  in all  $k$ -uniform points and choose the point with the maximum value. Since there are  $\binom{n+k-1}{k} = O(n^k)$  possible  $k$ -uniform points, the lemma follows.  $\square$

We now prove our result about  $\lambda_p$ -Lipschitz games. In what follows we will study a  $\lambda_p$ -Lipschitz game  $\mathfrak{G} := (M, n, \lambda, p, \gamma, \mathcal{T})$ . Assuming the existence of an exact equilibrium, we establish the existence of a  $k$ -uniform approximate equilibrium in the game  $\mathfrak{G}$ , where  $k$  depends on  $M, \lambda, p$  and  $\gamma$ . Note that  $\lambda$  depends heavily on  $p$  and the utility functions for the players.

Since by the definition of  $\lambda_p$ -Lipschitz games the strategy space  $S_i$  for every player  $i$  is the convex hull of  $n$  vectors  $y_1, \dots, y_n$  in  $\mathbb{R}^d$ , any  $x_i \in S_i$  can be written as a convex combination of  $y_j$ s. Hence,  $x_i = \sum_{j=1}^n \alpha_j y_j$ , where  $\alpha_j \geq 0$  for every  $j \in [n]$  and  $\sum_{j=1}^n \alpha_j = 1$ . Then,  $\alpha = (\alpha_1, \dots, \alpha_n)$  is a probability distribution over the vectors  $y_1, \dots, y_n$ , i.e. vector  $y_j$  is drawn with probability  $\alpha_j$ . Thus, we can sample a strategy  $x_i$  by the probability distribution  $\alpha$ .

So, let  $\mathbf{x}^*$  be an equilibrium for  $\mathfrak{G}$  and let  $\mathbf{x}'$  be a sampled uniform strategy profile from  $\mathbf{x}^*$ . For each player  $i$  we define the following events

$$\begin{aligned} \phi_i &= \left\{ \left| T_i(x'_i, \mathbf{x}'_{-i}) - T_i(x_i^*, \mathbf{x}_{-i}^*) \right| < \epsilon/2 \right\}, \\ \pi_i &= \left\{ T_i(x_i, \mathbf{x}'_{-i}) < T_i(x'_i, \mathbf{x}'_{-i}) + \epsilon \right\}, \quad \text{for all possible } x_i \\ \psi_i &= \left\{ \|x'_i - x_i^*\|_p < \frac{\epsilon}{2M\lambda} \right\} \quad \text{for some } p > 1. \end{aligned}$$

Notice that if all the events  $\pi_i$  occur at the same time, then the sampled profile  $\mathbf{x}'$  is an  $\epsilon$ -equilibrium. We will show that if for a player  $i$  the events  $\phi_i$  and  $\bigcap_j \psi_j$  hold, then the event  $\pi_i$  is also true.

**Lemma 4** *For all  $i \in [M]$  it holds that  $\bigcap_{j \in [M]} \psi_j \cap \phi_i \subseteq \pi_i$ .*

**Proof** Suppose that both events  $\phi_i$  and  $\bigcap_j \psi_j$  hold. We will show that the event  $\pi_i$  must be true too. Let  $x_i$  be an arbitrary strategy, let  $\mathbf{x}_{-i}^*$  be a strategy profile for the rest of the players, and let  $\mathbf{x}'_{-i}$  be a sampled strategy profile from  $\mathbf{x}_{-i}^*$ . Since we assume that the events  $\psi_j$  are true for all  $j$  and since  $\|\mathbf{x}'_{-i} - \mathbf{x}_{-i}^*\|_p \leq \sum_{j \neq i} \|x'_j - x_j^*\|_p$  we get that

$$\begin{aligned}
\|\mathbf{x}'_{-i} - \mathbf{x}^*_{-i}\|_p &\leq \sum_{j \neq i} \|x'_j - x^*_j\|_p \\
&\leq \sum_{j \neq i} \frac{\epsilon}{2M\lambda} \\
&\leq \frac{\epsilon}{2\lambda}.
\end{aligned}$$

Furthermore, since by assumption the utility functions for the players are  $\lambda_p$ -Lipschitz continuous we have that

$$|T_i(x_i, \mathbf{x}'_{-i}) - T_i(x_i, \mathbf{x}^*_{-i})| \leq \frac{\epsilon}{2}.$$

This means that

$$\begin{aligned}
T_i(x_i, \mathbf{x}'_{-i}) &\leq T_i(x_i, \mathbf{x}^*_{-i}) + \frac{\epsilon}{2} \\
&\leq T_i(x^*_i, \mathbf{x}^*_{-i}) + \frac{\epsilon}{2}
\end{aligned} \tag{1}$$

since the strategy profile  $(x^*_i, \mathbf{x}^*_{-i})$  is an equilibrium of the game. Furthermore, since by assumption the event  $\phi_i$  is true, we get that

$$T_i(x^*_i, \mathbf{x}^*_{-i}) < T_i(x'_i, \mathbf{x}'_{-i}) + \frac{\epsilon}{2}. \tag{2}$$

Hence, if we combine the inequalities (1) and (2) we get that  $T_i(x_i, \mathbf{x}'_{-i}) < T_i(x'_i, \mathbf{x}'_{-i}) + \epsilon$  for all possible  $x_i$ . Thus, if the events  $\phi_i$  and  $\psi_j$  for every  $j \in [M]$  hold, then the event  $\pi_i$  holds too.  $\square$

We are ready to prove the main result of the section.

**Theorem 5** *In any  $\lambda_p$ -Lipschitz game  $\mathfrak{G} = (M, n, \lambda, p, \gamma, T)$  that possesses an equilibrium and any  $\epsilon > 0$ , there is a  $k$ -uniform strategy profile, with  $k = \frac{16M^2\lambda^2p\gamma^2}{\epsilon^2}$  that is an  $\epsilon$ -equilibrium.*

**Proof** In order to prove the claim, it suffices to show that there is a strategy profile where every player plays a  $k$ -uniform strategy, for the chosen value of  $k$ , such that the events  $\pi_i$  hold for all  $i \in [M]$ . Since the utility functions in  $\mathfrak{G}$  are  $\lambda_p$ -Lipschitz continuous it holds that  $\bigcap_{i \in [n]} \psi_i \subseteq \bigcap_{i \in [n]} \phi_i$ . Furthermore, combining this with Lemma 4 we get that  $\bigcap_{i \in [n]} \psi_i \subseteq \bigcap_{i \in [n]} \pi_i$ . Thus, if the event  $\psi_i$  is true for every  $i \in [n]$ , then the event  $\bigcap_{i \in [n]} \pi_i$  is true as well.

Then, from Theorem 2 we get that for each  $i \in [M]$  there is a  $\frac{16M^2\lambda^2p\gamma^2}{\epsilon^2}$ -uniform point  $x'_i$  such that the event  $\psi_i$  occurs with positive probability. The claim follows.  $\square$

Theorem 5 establishes the existence of a  $k$ -uniform approximate equilibrium, but this does not immediately give us our algorithm. The obvious approach is to perform a brute force check of all  $k$ -uniform strategies, and then output the one that provides the best approximation. There is a problem with this, however, since computing the

quality of approximation requires us to compute the regret for each player, which in turn requires us to compute a best response for each player. Computing an exact best response in a Lipschitz game is a hard problem in general, since we make no assumptions about the utility functions of the players. Fortunately, it is sufficient to instead compute an *approximate* best response for each player, and Lemma 3 can be used to do this. So we can get the following corollary

**Corollary 6** *Let  $\mathbf{x}$  be a strategy profile for a  $\lambda_p$ -Lipschitz game  $\mathfrak{G} = (M, n, \lambda, p, \gamma, \mathcal{T})$ , and let  $\hat{x}_i$  be a best response for player  $i$  against the profile  $\mathbf{x}_{-i}$ . There is a  $\frac{4\lambda^2 p \gamma^2}{\epsilon^2}$ -uniform strategy  $x'_i$  that is an  $\epsilon$ -best response against  $\mathbf{x}_{-i}$ .*

Our goal is to *approximate* the approximation guarantee for a given strategy profile. More formally, given a strategy profile  $\mathbf{x}$  that is an  $\epsilon$ -equilibrium, and a constant  $\delta > 0$ , we want an algorithm that outputs a number within the range  $[\epsilon - \delta, \epsilon + \delta]$ . Corollary 6 allows us to do this. For a given strategy profile  $\mathbf{x}$ , we first compute  $\delta$ -approximate best responses for each player, then we can use these to compute  $\delta$ -approximate regrets for each player. The maximum over the  $\delta$ -approximate regrets then gives us an approximation of  $\epsilon$  with a tolerance of  $\delta$ . This is formalised in the following algorithm.

#### Algorithm 1. Evaluation of approximation guarantee

**Input:** A strategy profile  $\mathbf{x}$  for  $\mathfrak{G}$ , and a constant  $\delta > 0$ .

**Output:** An additive  $\delta$ -approximation of the approximation guarantee  $\alpha(\mathbf{x})$  for the strategy profile  $\mathbf{x}$ .

1. Set  $l = \frac{4\lambda^2 p \gamma^2}{\delta^2}$ .
2. For every player  $i \in [M]$ 
  - (a) For every  $l$ -uniform strategy  $x'_i$  of player  $i$  compute  $T_i(x'_i, \mathbf{x}_{-i})$ .
  - (b) Set  $m^* = \max_{x'_i} T_i(x'_i, \mathbf{x}_{-i})$ .
  - (c) Set  $\mathcal{R}_i(\mathbf{x}) = m^* - T_i(x_i, \mathbf{x}_{-i})$ .
3. Set  $\alpha(\mathbf{x}) = \max_{i \in [M]} \mathcal{R}_i(\mathbf{x})$ .
4. Return  $\alpha(\mathbf{x})$ .

Utilising the above algorithm by setting  $\delta = \epsilon$ , we can now produce an algorithm that finds an approximate equilibrium in Lipschitz games; hence, in what follows in this section we assume that  $\delta = \epsilon$ . The algorithm checks all  $k$ -uniform strategy profiles, using the value of  $k$  given by Theorem 5, and for each one, evaluates the quality approximation using the algorithm given above.

**Algorithm 2.**  $3\epsilon$ -equilibrium for  $\lambda_p$ -Lipschitz game  $\mathfrak{L}$

**Input:** Game  $\mathfrak{L}$  and  $\epsilon > 0$ .

**Output:** An  $3\epsilon$ -equilibrium for  $\mathfrak{L}$ .

1. Set  $k > \frac{16\lambda^2 Mp\gamma^2}{\epsilon^2}$ .
2. For every  $k$ -uniform strategy profile  $\mathbf{x}'$ 
  - (a) Evaluate the approximation of  $\alpha(\mathbf{x}')$  via Algorithm 1.
  - (b) If  $\alpha(\mathbf{x}')$  is less than  $2\epsilon$ , return  $\mathbf{x}'$ .

If the algorithm returns a strategy profile  $\mathbf{x}$ , then it must be a  $3\epsilon$ -equilibrium. This is because we check that an  $\epsilon$ -approximation of  $\alpha(\mathbf{x})$  is less than  $2\epsilon$ , and therefore  $\alpha(\mathbf{x}) \leq 3\epsilon$ . Secondly, we argue that if the game has an exact equilibrium, then this procedure will always output a  $3\epsilon$ -approximate equilibrium. From Theorem 5 we know that if  $k > \frac{16\lambda^2 Mp\gamma^2}{\epsilon^2}$ , then there is a  $k$ -uniform strategy profile  $\mathbf{x}$  that is an  $\epsilon$ -equilibrium for  $\mathfrak{L}$ . When we apply our approximate regret algorithm to  $\mathbf{x}$ , to find an  $\epsilon$ -approximation of  $\alpha(\mathbf{x})$ , the algorithm will return a number that is less than  $2\epsilon$ , hence  $\mathbf{x}$  will be returned by the algorithm.

To analyse the running time, observe that there are  $\binom{n+k-1}{k} = O(n^k)$  possible  $k$ -uniform strategies for each player, thus  $O(n^{Mk})$   $k$ -uniform strategy profiles. Furthermore, our regret approximation algorithm runs in time  $O(Mn^l)$ , where  $l = \frac{4\lambda^2 p\gamma^2}{\epsilon^2}$ . Hence, we get the next theorem.

**Theorem 7** *Given a  $\lambda_p$ -Lipschitz game  $\mathfrak{L} = (M, n, \lambda, p, \gamma, T)$  that possesses an equilibrium and any constant  $\epsilon > 0$ , a  $3\epsilon$ -equilibrium can be computed in time  $O(Mn^{Mk+l})$ , where  $k = O\left(\frac{\lambda^2 Mp\gamma^2}{\epsilon^2}\right)$  and  $l = O\left(\frac{\lambda^2 p\gamma^2}{\epsilon^2}\right)$ .*

Although it might be hard to decide whether a game has an equilibrium, our algorithm can be applied in any  $\lambda_p$ -Lipschitz game. If the game does not possess an exact equilibrium then our algorithm may find an approximate equilibrium, but if it fails to do so, then the contrapositive of Theorem 7 implies that the game does not possess an exact equilibrium.

**Theorem 8** *For any  $\lambda_p$ -Lipschitz game  $\mathfrak{L} = (M, n, \lambda, p, \gamma, T)$  and any  $\epsilon > 0$  in time  $O(Mn^{Mk+l})$ , we can either compute a  $3\epsilon$ -equilibrium, or decide that  $\mathfrak{L}$  does not possess an exact equilibrium, where  $k = O\left(\frac{\lambda^2 Mp\gamma^2}{\epsilon^2}\right)$  and  $l = O\left(\frac{\lambda^2 p\gamma^2}{\epsilon^2}\right)$ .*

## 4 A Quasi-polynomial Algorithm for Two-Player Penalty Games

In this section we present an algorithm that, for any  $\epsilon > 0$ , can compute an  $\epsilon$ -equilibrium for any penalty game in  $\mathcal{P}_{\lambda_p}$  (penalty games with  $\lambda_p$ -Lipschitz penalty functions) that posses an exact equilibrium in quasi-polynomial time. For the algorithm, we take the same approach as we did in the previous section for  $\lambda_p$ -Lipschitz games: we show that if an exact equilibrium exists, then a  $k$ -uniform approximate equilibrium always exists too, and provide a brute-force search algorithm for finding it. Once again, since best response computation may be hard for this class of games, we must provide an approximation algorithm for finding the quality of an approximate equilibrium.

We first focus on penalty games that posses an exact equilibrium. In what follows we will follow the literature and we will assume that the payoffs of the players are in  $[0, 1]$  So, let  $(\mathbf{x}^*, \mathbf{y}^*)$  be an equilibrium of the game and let  $(\mathbf{x}', \mathbf{y}')$  be a  $k$ -uniform strategy profile sampled from this equilibrium. We define the following four events:

$$\begin{aligned}\phi_r &= \{ |T_r(\mathbf{x}', \mathbf{y}') - T_r(\mathbf{x}^*, \mathbf{y}^*)| < \epsilon/2 \} \\ \pi_r &= \{ T_r(\mathbf{x}, \mathbf{y}') < T_r(\mathbf{x}', \mathbf{y}') + \epsilon \} \text{ for all } \mathbf{x} \\ \phi_c &= \{ |T_c(\mathbf{x}', \mathbf{y}') - T_c(\mathbf{x}^*, \mathbf{y}^*)| < \epsilon/2 \} \\ \pi_c &= \{ T_c(\mathbf{x}', \mathbf{y}) < T_c(\mathbf{x}', \mathbf{y}') + \epsilon \} \text{ for all } \mathbf{y}.\end{aligned}$$

The goal is to derive a value for  $k$  such that all the four events above are true, or equivalently  $Pr(\phi_r \cap \pi_r \cap \phi_c \cap \pi_r) > 0$ .

Note that in order to prove that  $(\mathbf{x}', \mathbf{y}')$  is an  $\epsilon$ -equilibrium we *only* have to consider the events  $\pi_r$  and  $\pi_c$ . Nevertheless, as we show in Lemma 9, the events  $\phi_r$  and  $\phi_c$  are crucial in our analysis. The proof of the main theorem boils down to the events  $\phi_r$  and  $\phi_c$ .

We will focus only on the row player, since the same analysis can be applied to the column player. Firstly we study the event  $\pi_r$ .

**Lemma 9** *For all penalty games it holds that  $Pr(\pi_r^c) \leq n \cdot e^{-\frac{k\epsilon^2}{2}} + Pr(\phi_r^c)$ .*

**Proof** We begin by introducing the following auxiliary events for all  $i \in [n]$

$$\psi_{ri} = \left\{ R_i \mathbf{y}' < R_i \mathbf{y}^* + \frac{\epsilon}{2} \right\}.$$

We prove how the events  $\psi_{ri}$  and the event  $\phi_r$  are related with the event  $\pi_r$ . Assume that the event  $\phi_r$  and the events  $\psi_{ri}$  for all  $i \in [n]$  are true. Let  $\mathbf{x}$  be any mixed strategy for the row player. Since by assumption  $R_i \mathbf{y}' < R_i \mathbf{y}^* + \frac{\epsilon}{2}$  and since  $\mathbf{x}$  is a probability distribution, it holds that  $\mathbf{x}^T R \mathbf{y}' < \mathbf{x}^T R \mathbf{y}^* + \frac{\epsilon}{2}$ . If we subtract  $\bar{f}_r(\mathbf{x})$  from each side we get that  $\mathbf{x}^T R \mathbf{y}' - \bar{f}_r(\mathbf{x}) < \mathbf{x}^T R \mathbf{y}^* - \bar{f}_r(\mathbf{x}) + \frac{\epsilon}{2}$ . This means that  $T_r(\mathbf{x}, \mathbf{y}') < T_r(\mathbf{x}, \mathbf{y}^*) + \frac{\epsilon}{2}$  for all  $\mathbf{x}$ . But we know that  $T_r(\mathbf{x}, \mathbf{y}^*) \leq T_r(\mathbf{x}^*, \mathbf{y}^*)$  for all  $\mathbf{x} \in \Delta^n$ , since  $(\mathbf{x}^*, \mathbf{y}^*)$  is an equilibrium. Thus, we get that  $T_r(\mathbf{x}, \mathbf{y}') < T_r(\mathbf{x}^*, \mathbf{y}^*) + \frac{\epsilon}{2}$  for all possible  $\mathbf{x}$ . Furthermore, since the event  $\phi_r$  is true too, we get that  $T_r(\mathbf{x}, \mathbf{y}') < T_r(\mathbf{x}', \mathbf{y}') + \epsilon$ . Thus, if the

events  $\phi_r$  and  $\psi_{ri}$  for all  $i \in [n]$  are true, then the event  $\pi_r$  must be true as well. Formally,  $\phi_r \cap_{i \in [n]} \psi_{ri} \subseteq \pi_r$ . Thus,  $Pr(\pi_r^c) \leq Pr(\phi_r^c) + \sum_i \psi_{ri}^c$ . Observe,  $R_i \mathbf{y}'$  is a sum of  $k$  independent random variables of expected value  $R_i \mathbf{y}^*$ . Each one of these random variables can take value in  $[0, 1]$ . Hence, we can use the Hoeffding bound to get that  $Pr(\psi_{ri}^c) \leq e^{-\frac{ke^2}{2}}$  for all  $i \in [n]$ . Our claim follows.  $\square$

With Lemma 9 in hand, we can see that in order to compute a value for  $k$  it is sufficient to study the event  $\phi_r$ . We introduce the following auxiliary events that we will study separately:  $\phi_{ru} = \{|\mathbf{x}'^T R \mathbf{y}' - \mathbf{x}^{*T} R \mathbf{y}^*| < \epsilon/4\}$  and  $\phi_{rb} = \{|\mathbf{f}_r(\mathbf{x}') - \mathbf{f}_r(\mathbf{x}^*)| < \epsilon/4\}$ . It is easy to see that if both  $\phi_{rb}$  and  $\phi_{ru}$  are true, then the event  $\phi_r$  must be true too. So we have  $\phi_{rb} \cap \phi_{ru} \subseteq \phi_r$ . The following lemma was essentially proven in [27], but we include it for reasons of completeness of our paper.

**Lemma 10**  $Pr(\phi_{ru}^c) \leq 4e^{-\frac{ke^2}{32}}$ .

**Proof** Recall,  $\phi_{ru} = \{|\mathbf{x}'^T R \mathbf{y}' - \mathbf{x}^{*T} R \mathbf{y}^*| < \epsilon/4\}$ . We define the events

$$\psi_1 = \left\{ \left| \mathbf{x}'^T R \mathbf{y}^* - \mathbf{x}^{*T} R \mathbf{y}^* \right| < \epsilon/8 \right\} \quad \psi_2 = \left\{ \left| \mathbf{x}'^T R \mathbf{y}' - \mathbf{x}'^T R \mathbf{y}^* \right| < \epsilon/8 \right\}.$$

Note that if both events  $\psi_1$  and  $\psi_2$  are true, then  $\phi_{ru}$  holds as well. Since  $\mathbf{x}'$  is sampled from  $\mathbf{x}^*$ ,  $\mathbf{x}'^T R \mathbf{y}^*$  can be seen as the sum of  $k$  independent random variables each with expected value  $\mathbf{x}^{*T} R \mathbf{y}^*$ . In addition, each random variable can take values in  $[0, 1]$ .

Hence, we can apply the Hoeffding bound and get that  $Pr(\psi_1^c) \leq 2e^{-\frac{ke^2}{32}}$ . With similar arguments we can prove that  $Pr(\psi_2^c) \leq 2e^{-\frac{ke^2}{32}}$ , therefore  $Pr(\phi_{ru}^c) \leq 4e^{-\frac{ke^2}{32}}$   $\square$

In addition, we must prove an upper bound on the event  $\phi_{rb}^c$ . To do so, we will use the following lemma which was proven by Barman [4].<sup>2</sup>

**Lemma 11** (Barman [4]) *Given a set of vectors  $Z = \{z_1, z_2, \dots, z_n\} \subset \mathbb{R}^d$ , let  $\text{conv}(Z)$  denote the convex hull of  $Z$ , and let  $2 \leq p < \infty$ . Furthermore, let  $\mu \in \text{conv}(Z)$  and let  $\mu'$  be a  $k$ -uniform vector sampled from  $\mu$ . Then,  $E[\|\mu' - \mu^*\|_p] \leq \frac{2\sqrt{p}}{\sqrt{k}}$ .*

We are ready to prove the last crucial lemma for our theorem.

**Lemma 12**  $Pr(\phi_{rb}^c) \leq \frac{8\lambda\sqrt{p}}{\epsilon\sqrt{k}}$ .

**Proof** Since we assume that the penalty function  $\mathbf{f}_r(\mathbf{x}')$  is  $\lambda_p$ -Lipschitz continuous the event  $\phi_{rb}$  can be replaced by the event  $\phi_{rb'} = \{\|\mathbf{x}' - \mathbf{x}^*\|_p < \epsilon/4\lambda\}$ . It is easy to

<sup>2</sup> This was part of the proof of Theorem 2 in [4].

see that  $\phi_{rb} \subseteq \phi_{rb'}$ . Then, from Lemma 11 we get that  $E[\|\mathbf{x}' - \mathbf{x}^*\|_p] \leq \frac{2\sqrt{p}}{\sqrt{k}}$ . Thus, using Markov's inequality we get that

$$\begin{aligned} Pr\left(\|\mathbf{x}' - \mathbf{x}^*\|_p \geq \frac{\epsilon}{4\lambda}\right) &\leq \frac{E[\|\mathbf{x}' - \mathbf{x}^*\|_p]}{\frac{\epsilon}{4\lambda}} \\ &\leq \frac{8\lambda\sqrt{p}}{\epsilon\sqrt{k}}. \end{aligned}$$

□

Let us define the event  $GOOD = \phi_r \cap \phi_c \cap \pi_r \cap \pi_c$ . To prove our theorem it suffices to prove that  $Pr(GOOD) > 0$ . Notice that for the events  $\phi_c$  and  $\pi_c$  the same analysis as for  $\phi_r$  and  $\pi_r$  can be used. Then, using Lemmas 9, 12 and 10 we get that  $Pr(GOOD^c) < 1$  for the chosen value of  $k$ .

**Theorem 13** *For any equilibrium  $(\mathbf{x}^*, \mathbf{y}^*)$  of a penalty game from the class  $\mathcal{P}_{\lambda_p}$ , any  $\epsilon > 0$ , and any  $k \in \frac{\Omega(\lambda^2 \log n)}{\epsilon^2}$ , there exists a  $k$ -uniform strategy profile  $(\mathbf{x}', \mathbf{y}')$  that:*

1.  $(\mathbf{x}', \mathbf{y}')$  is an  $\epsilon$ -equilibrium for the game,
2.  $|T_r(\mathbf{x}', \mathbf{y}') - T_r(\mathbf{x}^*, \mathbf{y}^*)| < \epsilon/2$ ,
3.  $|T_c(\mathbf{x}', \mathbf{y}') - T_c(\mathbf{x}^*, \mathbf{y}^*)| < \epsilon/2$ .

**Proof**

$$\begin{aligned} Pr(GOOD^c) &\leq Pr(\phi_r^c) + Pr(\pi_r^c) + Pr(\phi_c^c) + Pr(\pi_c^c) \\ &\leq 2(Pr(\phi_r^c) + Pr(\pi_r^c)) \\ &\leq 2\left(2Pr(\phi_r^c) + n \cdot e^{-\frac{ke^2}{2}}\right) \quad (\text{from Lemma 9}) \\ &\leq 2\left(2Pr(\phi_{ru}^c) + 2Pr(\phi_{rb'}^c) + n \cdot e^{-\frac{ke^2}{2}}\right) \\ &\leq 2\left(4e^{-\frac{ke^2}{32}} + \frac{16\lambda\sqrt{p}}{\epsilon\sqrt{k}} + n \cdot e^{-\frac{ke^2}{2}}\right) \quad (\text{from Lemma 12}) \\ &< 1 \quad \text{for } k = \frac{c \cdot \lambda^2 \cdot \log n}{\epsilon^2} \text{ and } c \cdot \lambda^2 < 2. \end{aligned}$$

Hence,  $Pr(GOOD) > 0$  and our claim follows. □

Theorem 13 establishes the *existence* of a  $k$ -uniform strategy profile  $(\mathbf{x}', \mathbf{y}')$  that is an  $\epsilon$ -equilibrium, but as before, we must provide an efficient method for approximating the quality of approximation provided by a given strategy profile. To do so, we first give the following lemma, which shows that approximate best responses can be computed in quasi-polynomial time for penalty games.

**Lemma 14** *Let  $(\mathbf{x}, \mathbf{y})$  be a strategy profile for a penalty game  $\mathcal{P}_{\lambda}$ , and let  $\hat{\mathbf{x}}$  be a best response against  $\mathbf{y}$ . There is an  $l$ -uniform strategy  $\mathbf{x}'$ , with  $l = \frac{\Omega(\lambda^2 \sqrt{p})}{\epsilon^2}$ , that is an  $\epsilon$ -best response against  $\mathbf{y}$ , i.e.  $T_r(\hat{\mathbf{x}}, \mathbf{y}) < T_r(\mathbf{x}', \mathbf{y}) + \epsilon$ .*

**Proof** We will prove that  $|T_r(\hat{\mathbf{x}}, \mathbf{y}) - T_r(\mathbf{x}', \mathbf{y})| < \epsilon$  which implies our claim. Let  $\phi_1 = \{|\hat{\mathbf{x}}^T R \mathbf{y} - \mathbf{x}'^T R \mathbf{y}| \leq \epsilon/2\}$  and  $\phi_2 = \{|\hat{f}_r(\hat{\mathbf{x}}) - \hat{f}_r(\mathbf{x}')| < \epsilon/2\}$ . Notice that Lemma 12 does not use anywhere the fact that  $\mathbf{x}^*$  is an equilibrium strategy, thus it holds even if  $\mathbf{x}^*$  is replaced by  $\hat{\mathbf{x}}$ . Thus,  $Pr(\phi_2^c) \leq \frac{4\lambda\sqrt{p}}{\epsilon\sqrt{k}}$ . Furthermore, using similar analysis as in Lemma 10, we can prove that  $Pr(\phi_1^c) \leq 4e^{-\frac{k\epsilon^2}{32}}$  and using similar arguments as in the proof of Theorem 13 it can be easily proved that for the chosen value of  $l$  it holds that  $Pr(\phi_1^c) + Pr(\phi_2^c) < 1$ , thus the events  $\phi_1$  and  $\phi_2$  occur with positive probability and our claim follows.  $\square$

Given this lemma, we can reuse Algorithm 1, but with  $l = \frac{\Omega(\lambda^2 \sqrt{p})}{\epsilon^2}$ , to provide an algorithm that approximates the quality of approximation of a given strategy profile. Then, we can reuse Algorithm 2 with  $k = \frac{\Omega(\lambda^2 \log n)}{\epsilon^2}$  to provide a quasi-polynomial time algorithm that finds approximate equilibria in penalty games. Notice again that our algorithm can be applied in games in which it is computationally hard to verify whether an exact equilibrium exists. Our algorithm either will compute an approximate equilibrium or it will fail to find one, in which case the game does not possess an exact equilibrium.

**Theorem 15** *In any penalty game in  $\mathcal{P}_{\lambda_p}$  and any  $\epsilon > 0$ , in quasi polynomial time we can either compute a  $3\epsilon$ -equilibrium, or decide that it does not possess an exact equilibrium.*

## 5 Distance Biased Two-Player Games

In this section, we focus on two particular classes of distance biased games, and we provide strongly polynomial-time algorithms for computing approximate equilibria for them. We focus on the following two penalty functions:

- $L_1$  penalty:  $b_r(\mathbf{x}, \mathbf{p}) = \|\mathbf{x} - \mathbf{p}\|_1 = \sum_i |x_i - p_i|$ .
- $L_2$  penalty:  $b_r(\mathbf{x}, \mathbf{p}) = \|\mathbf{x} - \mathbf{p}\|_2^2 = \sum_i (x_i - p_i)^2$ .



Our approach is to follow the well-known technique of [12] that finds a 0.5-NE in a bimatrix game. The algorithm that we will use for all three penalty functions is given below.

### Algorithm 3. The Base Algorithm

1. Compute a best response  $\mathbf{y}^*$  against  $\mathbf{p}$ .
2. Compute a best response  $\mathbf{x}$  against  $\mathbf{y}^*$ .
3. Set  $\mathbf{x}^* = \delta \cdot \mathbf{p} + (1 - \delta) \cdot \mathbf{x}$ , for some  $\delta \in [0, 1]$ .
4. Return the strategy profile  $(\mathbf{x}^*, \mathbf{y}^*)$ .

While this is a well-known technique for bimatrix games, note that it cannot immediately be applied to penalty games. This is because the algorithm requires us to compute two best response strategies, and while computing a best-response is trivial in bimatrix games, this is not the case for penalty games. Best responses for  $L_1$  penalties can be computed in polynomial time via linear programming, and for  $L_2^2$  penalties, the ellipsoid algorithm can be applied. However, these methods do not provide strongly polynomial algorithms.

In this section, for each of the penalty functions, we develop a simple combinatorial algorithm for computing best response strategies. Our algorithms are strongly polynomial. Then, we determine the quality of the approximate equilibria given by the base algorithm when our best response techniques are used. In what follows we make the common assumption that the payoffs of the underlying bimatrix game  $(R, C)$  are in  $[0, 1]$ . Furthermore, we use  $R_i$  to denote the  $i$ -th row of the matrix  $R$ .

## 5.1 A 2/3-Approximation Algorithm for $L_1$ -Biased Games

We start by considering  $L_1$ -biased games. Suppose that we want to compute a best-response for the row player against a fixed strategy  $\mathbf{y}$  of the column player. We will show that best response strategies in  $L_1$ -biased games have a very particular form: if  $b$  is the best response strategy in the (unbiased) bimatrix game  $(R, C)$ , then the best-response places all of its probability on  $b$  *except* for a certain set of rows  $S$  where it is too costly to shift probability away from  $\mathbf{p}$ . The rows  $i \in S$  will be played with  $\mathbf{p}_i$  to avoid taking the penalty for deviating.

The characterisation for whether it is too expensive to shift away from  $\mathbf{p}$  is given by the following lemma.

**Lemma 16** *Let  $j$  be a pure strategy, let  $k$  be a pure strategy with  $\mathbf{p}_k > 0$ , and let  $\mathbf{x}$  be a strategy with  $\mathbf{x}_k = \mathbf{p}_k$  and  $\mathbf{x}_j \geq \mathbf{p}_j$ . The utility for the row player increases when we shift probability from  $k$  to  $j$  if and only if  $R_j \mathbf{y} - R_k \mathbf{y} - 2d_r > 0$ .*

**Proof** Recall, the payoff the row player gets from pure strategy  $i$  when he plays  $\mathbf{x}$  against strategy  $\mathbf{y}$  is  $\mathbf{x}_i \cdot R_i \mathbf{y} - d_r \cdot |\mathbf{x}_i - \mathbf{p}_i|$ . Suppose now that we shift  $\delta$

probability from  $k$  to  $j$ , where  $\delta \in (0, \mathbf{p}_k]$ . Then the utility for the row player is equal to  $T_r(\mathbf{x}, \mathbf{y}) + \delta \cdot (R_j \mathbf{y} - R_k \mathbf{y} - 2d_r)$ , where the final term is the penalty for shifting  $\delta$  probability from  $k$  and adding it this probability to  $j$ . Thus, the utility for the row player increases under this shift if and only if  $R_j \mathbf{y} - R_k \mathbf{y} - 2d_r > 0$ .  $\square$

With Lemma 16 in hand, we can give an algorithm for computing a best response.

**Algorithm 4.** Best Response Algorithm for  $L_1$  penalty

1. Set  $S = 0$ .
2. Compute a pure best response  $b$  against  $\mathbf{y}$  in the unbiased bimatrix game  $(R, C)$ .
3. For each index  $i \neq b$  in the range  $1 \leq i \leq n$ :
  - (a) If  $R_b \mathbf{y} - R_i \mathbf{y} - 2d_r \leq 0$ , then set  $\mathbf{x}_i = \mathbf{p}_i$  and  $S = S + \mathbf{p}_i$ .
  - (b) Otherwise set  $\mathbf{x}_i = 0$
4. Set  $\mathbf{x}_b = 1 - S$ .
5. Return  $\mathbf{x}$ .

**Lemma 17** *Algorithm 4 correctly computes a best response against  $\mathbf{y}$ .*

**Proof** Assume that we start from the (mixed) strategy  $\mathbf{p}$ . We will consider the cases depending whether  $\mathbf{p}$  is a best response or not. If  $\mathbf{p}$  is a best response, then the player cannot increase his payoff by changing his strategy. Hence, from Lemma 16 we know that for all pairs of pure strategies  $i$  and  $j$ , it holds that  $R_j \mathbf{y} - R_i \mathbf{y} - 2d_r \leq 0$ . Observe that in this case, our algorithm will produce  $\mathbf{p}$  due to Step 3(a).

If  $\mathbf{p}$  is not a best response, then we can get a best response by shifting probability mass from  $\mathbf{p}$ . We can partition the pure strategies of the player into two categories: those that shifting probability increases the payoff and those that do not; this partition can be computed via Lemma 16. Again, in Step 3(a) of the algorithm all strategies that belong to the second category are identified and no probability is shifted. In addition, if we are able to shift probability away from a strategy  $k$ , then we should obviously shift it all of this probability mass to a best response strategy for the (unbiased) bimatrix game, since this strategy maximizes the increase in the payoff. Step 3(b) guarantees that all the probability mass is shifted away from any such pure strategy. Step 4 guarantees that all this probability is placed on the correct pure strategy. Finally, since  $\mathbf{p}$  is a probability distribution and we simply shift its probability mass, we get that  $\mathbf{x}$  is indeed a probability distribution.  $\square$

Our characterisation has a number of consequences. Firstly, it can be seen that if  $d_r \geq 1/2$ , then there is no profitable shift of probability between any two pure strategies, since  $0 \leq R_i \mathbf{y} \leq 1$  for all  $i \in [n]$ . Thus, we get the following corollary.

**Corollary 18** *If  $d_r \geq 1/2$ , then the row player always plays the strategy  $\mathbf{p}$  irrespectively from which strategy his opponent plays, i.e.  $\mathbf{p}$  is a dominant strategy.*

Moreover, since we can compute a best response in polynomial time we get the next theorem.

**Theorem 19** *In biased games with  $L_1$  penalty functions and  $\max\{d_r, d_c\} \geq 1/2$ , an equilibrium can be computed in polynomial time.*

**Proof** Assume that  $d_r \geq 1/2$ . From Corollary 18 we get that the row player will play his base strategy  $\mathbf{p}$ . Then we can use Algorithm 4 to compute a best response against  $\mathbf{p}$  for the column player. This profile will be an equilibrium for the game since no player can increase his payoff by unilaterally changing his strategy.  $\square$

Finally, using the characterization of best responses we can see that there is a connection between the equilibria of the distance biased game and approximate well supported Nash equilibria (WSNE) of the underlying bimatrix game. An  $\epsilon$ -WSNE of a bimatrix game  $(R, C)$  is a strategy profile  $(\mathbf{x}, \mathbf{y})$  where every player plays with positive probability only pure strategies that are  $\epsilon$ -best responses; formally, for every  $i$  such that  $\mathbf{x}_i > 0$  it holds that  $R_i \mathbf{y} \geq \max_k R_k \mathbf{y} - \epsilon$  and for every  $j$  such that  $\mathbf{y}_j > 0$  it holds that  $C_j^T \mathbf{x} \geq \max_k C_k^T \mathbf{x} - \epsilon$

**Theorem 20** *Let  $\mathcal{B} = (R, C, \mathbf{b}_r(\mathbf{x}, \mathbf{p}), \mathbf{b}_c(\mathbf{y}, \mathbf{q}), d_r, d_c)$  be a distance biased game with  $L_1$  penalties and let  $d := \max\{d_r, d_c\}$ . Any equilibrium of  $\mathcal{B}$  is a  $2d$ -WSNE for the bimatrix game  $(R, C)$ .*

**Proof** Let  $(\mathbf{x}^*, \mathbf{y}^*)$  be an equilibrium for  $\mathcal{B}$ . From the best response Algorithm for  $L_1$  penalty games we can see that  $\mathbf{x}_i^* > 0$  if and only if  $R_b \cdot \mathbf{y}^* - R_i \cdot \mathbf{y}^* - 2d_r \leq 0$ , where  $b$  is a pure best response against  $\mathbf{y}^*$ . This means that for every  $i \in [n]$  with  $\mathbf{x}_i^* > 0$ , it holds that  $R_i \cdot \mathbf{y}^* \geq \max_{j \in [n]} R_j \cdot \mathbf{y}^* - 2d$ . Similarly, it holds that  $C_i^T \cdot \mathbf{x}^* \geq \max_{j \in [n]} C_j^T \cdot \mathbf{x}^* - 2d$  for all  $i \in [n]$  with  $\mathbf{y}_i^* > 0$ . This is the definition of a  $2d$ -WSNE for the bimatrix game  $(R, C)$ .  $\square$

*Approximation algorithm* We now analyse the approximation guarantee provided by the base algorithm for  $L_1$ -biased games. So, let  $(\mathbf{x}^*, \mathbf{y}^*)$  be the strategy profile returned by the base algorithm. Since we have already shown that exact equilibria can be found in games with either  $d_c \geq 1/2$  or  $d_r \geq 1/2$ , we will assume that both  $d_c$  and  $d_r$  are less than  $1/2$ , since this is the only interesting case.

We start by considering the regret of the row player. The following lemma will be used in the analysis of both our approximation algorithms.

**Lemma 21** *Under the strategy profile  $(\mathbf{x}^*, \mathbf{y}^*)$  the regret for the row player is at most  $\delta$ .*

**Proof** Notice that for all  $i \in [n]$  we have

$$|\delta \mathbf{p}_i + (1 - \delta) \mathbf{x}_i - \mathbf{p}_i| = (1 - \delta) |\mathbf{x}_i - \mathbf{p}_i|,$$

hence  $\|\mathbf{x}^* - \mathbf{p}\|_1 = (1 - \delta) \|\mathbf{x} - \mathbf{p}\|_1$ . Furthermore, notice that  $\sum_i ((1 - \delta) \mathbf{x}_i + \delta \mathbf{p}_i - \mathbf{p}_i)^2 = (1 - \delta)^2 \|\mathbf{x} - \mathbf{p}\|_2^2$ , thus  $\|\mathbf{x}^* - \mathbf{p}\|_2^2 \leq (1 - \delta) \|\mathbf{x} - \mathbf{p}\|_2^2$ . Hence, for the payoff for the row player it holds  $T_r(\mathbf{x}^*, \mathbf{y}^*) \geq \delta \cdot T_r(\mathbf{p}, \mathbf{y}^*) + (1 - \delta) \cdot T_r(\mathbf{x}, \mathbf{y}^*)$  and his regret under the strategy profile  $(\mathbf{x}^*, \mathbf{y}^*)$  is

$$\begin{aligned} \mathcal{R}^r(\mathbf{x}^*, \mathbf{y}^*) &= \max_{\tilde{\mathbf{x}}} T_r(\tilde{\mathbf{x}}, \mathbf{y}^*) - T_r(\mathbf{x}^*, \mathbf{y}^*) \\ &= T_r(\mathbf{x}, \mathbf{y}^*) - T_r(\mathbf{x}^*, \mathbf{y}^*) \quad (\text{since } \mathbf{x} \text{ is a best response against } \mathbf{y}^*) \\ &\leq \delta (T_r(\mathbf{x}, \mathbf{y}^*) - T_r(\mathbf{p}, \mathbf{y}^*)) \\ &\leq \delta \left( \text{since } \max_{\mathbf{x}} T_r(\mathbf{x}, \mathbf{y}^*) \leq 1 \text{ and } T_r(\mathbf{p}, \mathbf{y}^*) \geq 0 \right). \end{aligned}$$

□

Next, we consider the regret of the column player. Observe that the precondition of  $d_c \cdot \mathbf{b}_c(\mathbf{y}^*, \mathbf{q}) \leq 1$  always holds, since we have  $\|\mathbf{y}^* - \mathbf{q}\|_1 \leq 2$ , thus  $d_c \cdot \mathbf{b}_c(\mathbf{y}^*, \mathbf{q}) \leq 1$  since we are only interested in the case where  $d_c \leq 1/2$ .

**Lemma 22** *If  $d_c \cdot \mathbf{b}_c(\mathbf{y}^*, \mathbf{q}) \leq 1$ , then under strategy profile  $(\mathbf{x}^*, \mathbf{y}^*)$  the column player suffers at most  $2 - 2\delta$  regret.*

**Proof** The regret of the column player under the strategy profile  $(\mathbf{x}^*, \mathbf{y}^*)$  is

$$\begin{aligned} \mathcal{R}^c(\mathbf{x}^*, \mathbf{y}^*) &= \max_{\mathbf{y}} T_c(\mathbf{x}^*, \mathbf{y}) - T_c(\mathbf{x}^*, \mathbf{y}^*) \\ &= \max_{\mathbf{y}} \{ (1 - \delta) T_c(\mathbf{x}, \mathbf{y}) + \delta T_c(\mathbf{p}, \mathbf{y}) \} - (1 - \delta) T_c(\mathbf{x}, \mathbf{y}^*) - \delta T_c(\mathbf{p}, \mathbf{y}^*) \\ &\leq (1 - \delta) \left( \max_{\mathbf{y}} T_c(\mathbf{x}^*, \mathbf{y}) - T_c(\mathbf{x}, \mathbf{y}^*) \right) (\text{since } \mathbf{y}^* \text{ is a best response against } \mathbf{p}) \\ &\leq (1 - \delta) (1 + d_c \cdot \mathbf{b}_c(\mathbf{y}^*, \mathbf{q})) \quad \left( \text{since } \max_{\mathbf{x}} T_c(\mathbf{x}^*, \mathbf{y}) \leq 1 \right) \\ &\leq (1 - \delta) \cdot 2 \quad (\text{since } d_c \cdot \mathbf{b}_c(\mathbf{y}^*, \mathbf{q}) \leq 1). \end{aligned}$$

□

To complete the analysis, we must select a value for  $\delta$  that equalises the two regrets. It can easily be verified that setting  $\delta = 2/3$  ensures that  $\delta = 2 - 2\delta$ , and so we have the following theorem.

**Theorem 23** *In biased games with  $L_1$  penalties a  $2/3$ -equilibrium can be computed in polynomial time.*

## 5.2 A 5/7-Approximation Algorithm for $L_2^2$ -Biased Games

We now turn our attention to biased games with an  $L_2^2$  penalty. Again, we start by giving a combinatorial algorithm for finding a best response. Throughout this section, we fix  $\mathbf{y}$  as a column player strategy, and we will show how to compute a best response for the row player.

Best responses in  $L_2^2$ -biased games can be found by solving a quadratic program, and actually this particular quadratic program can be solved via the ellipsoid algorithm [26]. We will give a simple combinatorial algorithm that produces a closed formula for the solution. Hence, we will obtain a strongly polynomial time algorithm for finding best responses.

Our algorithm can be applied on  $L_2^2$  penalty functions and any value  $d_r$ , but for notation simplicity we describe our method for  $d_r = 1$ . Furthermore, we define  $\alpha_i := R_i \mathbf{y} + 2\mathbf{p}$ , and we call  $\alpha_i$  as the payoff of pure strategy  $i$ . Then, the utility for the row player can be written as  $T_r(\mathbf{x}, \mathbf{y}) = \sum_{i=1}^n \mathbf{x}_i \cdot \alpha_i - \sum_{i=1}^n \mathbf{x}_i^2 - \mathbf{p}^T \mathbf{p}$ . Notice that the term  $\mathbf{p}^T \mathbf{p}$  is a constant and it does not affect the solution of the best response; so we can exclude it from our computations. Thus, a best response for the row player against strategy  $\mathbf{y}$  is the solution of the following quadratic program

$$\begin{aligned} & \text{maximize} && \sum_{i=1}^n \mathbf{x}_i \cdot \alpha_i - \sum_{i=1}^n \mathbf{x}_i^2 \\ & \text{subject to} && \sum_{i=1}^n \mathbf{x}_i = 1 \\ & && \mathbf{x}_i \geq 0 \quad \text{for all } i \in [n]. \end{aligned}$$

Observe that since  $\sum \mathbf{x}_i^2$  is a strictly convex function, therefore the solution of the problem above is unique, hence there is a unique best response against any strategy.

In what follows, without loss of generality, we assume that  $\alpha_1 \geq \dots \geq \alpha_n$ . That is, the pure strategies are ordered according to their payoffs. In the next lemma we prove that in every best response, if a player plays pure strategy  $l$  with positive probability, then he must play every pure strategy  $k$  with  $k < l$  with positive probability.

**Lemma 24** *In every best response  $\mathbf{x}^*$  if  $\mathbf{x}_l^* > 0$  then  $\mathbf{x}_k^* > 0$  for all  $k < l$ . In addition, if  $k < l$  and  $\mathbf{x}_l^* > 0$ , then  $\mathbf{x}_k^* = \mathbf{x}_l^* + \frac{\alpha_k - \alpha_l}{2}$ .*

**Proof** For the sake of contradiction suppose that there is a best response  $\mathbf{x}^*$  and a  $k < l$  such that  $\mathbf{x}_l^* > 0$  and  $\mathbf{x}_k^* = 0$ . Let us denote  $M = \sum_{i \neq \{l, k\}} \alpha_i \cdot \mathbf{x}_i^* - \sum_{i \neq \{l, k\}} \mathbf{x}_i^{*2}$ . Suppose now that we shift some probability, denoted by  $\delta$ , from pure strategy  $l$  to pure strategy  $k$ . Then the utility for the row player is  $T_r(\mathbf{x}^*, \mathbf{y}) = M + \alpha_l \cdot (\mathbf{x}_l^* - \delta) - (\mathbf{x}_l^* - \delta)^2 + \alpha_k \cdot \delta - \delta^2$ , which is maximized for  $\delta = \frac{\alpha_k - \alpha_l + 2\mathbf{x}_l^*}{4}$ . Notice that  $\delta > 0$  since  $\alpha_k \geq \alpha_l$  and  $\mathbf{x}_l^* > 0$ , thus the row player can increase his utility by assigning positive probability to pure strategy  $k$  which contradicts the fact that  $\mathbf{x}^*$  is a best response. To prove the second claim, we observe

that, using exactly the same arguments as above, the payoff of the row player will decrease if we try to shift probability between pure strategies  $k$  and  $l$ , when  $\mathbf{x}_l^* > 0$  and  $\mathbf{x}_k^* = \mathbf{x}_l^* + \frac{\alpha_k - \alpha_l}{2}$ .  $\square$

Lemma 24 implies that there are only  $n$  possible supports that a best response can use. In addition, the second part of the lemma means that for all  $i \in [k]$  we get

$$\mathbf{x}_i = \frac{1}{2} \left( \alpha_i - \frac{\sum_{j=1}^k \alpha_j - 2}{k} \right). \quad (3)$$

So, our algorithm does the following. It loops through all  $n$  candidate supports for a best response. For each one, it uses Equation (3) to determine the probabilities, and then checks whether these form a probability distribution, and thus if this is a feasible response. If it is, then it is saved in a list of feasible responses, otherwise it is discarded. After all  $n$  possibilities have been checked, the feasible response with the highest payoff is then returned.

#### Algorithm 5. Best Response Algorithm for $L_2^2$ penalty

1. For  $i = 1 \dots n$ 
  - (a) Set  $\mathbf{x}_1 \geq \dots \geq \mathbf{x}_i > 0$  and  $\mathbf{x}_{i+1} = \dots = \mathbf{x}_n = 0$ .
  - (b) Check if there is a feasible response under these constraints.
  - (c) If so, add it to the list of feasible responses.
2. Among the feasible responses choose one with the highest utility.

### 5.2.1 Approximation Algorithm

We now show that the base algorithm gives a  $5/7$ -approximation when applied to  $L_2^2$ -penalty games. For the row player's regret, we can use Lemma 21 to show that the regret is bounded by  $\delta$ . However, for the column player's regret, things are more involved. We will show that the regret of the column player is at most  $2.5 - 2.5\delta$ . That analysis depends on the maximum entry of the base strategy  $\mathbf{q}$  and more specifically on whether  $\max_k \mathbf{q}_k \leq 1/2$  or not.

**Lemma 25** *If  $\max_k \{\mathbf{q}_k\} \leq 1/2$ , then the regret the column player suffers under strategy profile  $(\mathbf{x}^*, \mathbf{y}^*)$  is at most  $2.5 - 2.5\delta$ .*

**Proof** Note that when  $\max_k \{\mathbf{q}_k\} \leq 1/2$ , then  $\mathbf{b}_c = \|\mathbf{y} - \mathbf{q}\|_2^2 \leq 1.5$  for all possible  $\mathbf{y}$ . Then, using similar analysis as in Lemma 22, along with the fact that  $d_c \cdot \mathbf{b}_c(\mathbf{y}^*, \mathbf{q}) \leq 2$  for  $L_2^2$  penalties and the assumption that  $d_c = 1$ , we get that

$$\begin{aligned}
 \mathcal{R}^c(\mathbf{x}^*, \mathbf{y}^*) &= \max_{\mathbf{y}} T_c(\mathbf{x}^*, \mathbf{y}) - T_c(\mathbf{x}^*, \mathbf{y}^*) \\
 &= \max_{\mathbf{y}} \left\{ (1 - \delta)T_c(\mathbf{x}, \mathbf{y}) + \delta T_c(\mathbf{p}, \mathbf{y}) \right\} - (1 - \delta)T_c(\mathbf{x}, \mathbf{y}^*) - \delta T_c(\mathbf{p}, \mathbf{y}^*) \\
 &\leq (1 - \delta) \left( \max_{\mathbf{y}} T_c(\mathbf{x}^*, \mathbf{y}) - T_c(\mathbf{x}, \mathbf{y}^*) \right) \text{ (since } \mathbf{y}^* \text{ is a best response against } \mathbf{p}) \\
 &\leq (1 - \delta)(1 + d_c \cdot \mathbf{b}_c(\mathbf{y}^*, \mathbf{q})) \quad \left( \text{since } \max_{\mathbf{x}} T_c(\mathbf{x}^*, \mathbf{y}) \leq 1 \right) \\
 &\leq (1 - \delta) \cdot 2.5 \quad \left( \text{since } d_c \cdot \mathbf{b}_c(\mathbf{y}^*, \mathbf{q}) \leq 1 \right).
 \end{aligned}$$

□

For the case where there is a  $k$  such that  $\mathbf{q}_k > 1/2$  a more involved analysis is needed. The first goal is to prove that under *any* strategy  $\mathbf{y}^*$  that is a best response against  $\mathbf{p}$  the pure strategy  $k$  is played with positive probability. In order to prove that, first we show that there is a feasible response against strategy  $\mathbf{p}$  where pure strategy  $k$  is played with positive probability. In what follows we denote  $\beta_i := C_i^T \mathbf{p} + 2\mathbf{q}_i$ .

**Lemma 26** *Let  $\mathbf{q}_k > 1/2$  for some  $k \in [n]$ . Then there is a feasible response where pure strategy  $k$  is played with positive probability.*

**Proof** Note that  $\beta_k > 1$  since by assumption  $\mathbf{q}_k > 1/2$ . Recall from Equation (3) that in a feasible response  $\mathbf{y}$  it holds that  $\mathbf{y}_i = \frac{1}{2} \left( \beta_i - \frac{\sum_{j=1}^k \beta_j - 2}{k} \right)$ .

In order to prove the claim it is sufficient to show that  $\mathbf{y}_k > 0$  when we set  $\mathbf{y}_i > 0$  for all  $i \in [k]$ . Or equivalently, to show that  $\beta_k - \frac{\sum_{j=1}^k \beta_j - 2}{k} = \frac{1}{k} \left( (k-1)\beta_k + 2 - \sum_{j=1}^{k-1} \beta_j \right) > 0$ . But,

$$\begin{aligned}
 (k-1)\beta_k + 2 - \sum_{j=1}^{k-1} \beta_j &> k + 1 - \sum_{j=1}^{k-1} (C^T \mathbf{x} + 2\mathbf{q}_j) \quad (\text{since } \beta_k > 1) \\
 &\geq k + 1 - (k-1) - \sum_{j=1}^{k-1} 2\mathbf{q}_j \\
 &\geq 2\mathbf{q}_k \quad (\text{since } \mathbf{q} \in \Delta^n) \\
 &> 0.
 \end{aligned}$$

The claim follows. □

Next it is proven that the utility of the column player is increasing when he adds pure strategies in his support that yield payoff greater than 1, i.e.  $\beta_i > 1$ .

**Lemma 27** Let  $\mathbf{y}^k$  and  $\mathbf{y}^{k+1}$  be two feasible responses with support size  $k$  and  $k + 1$  respectively, where  $\beta_{k+1} > 1$ . Then  $T_c(\mathbf{x}, \mathbf{y}^{k+1}) > T_c(\mathbf{x}, \mathbf{y}^k)$ .

**Proof** Let  $\mathbf{y}^k$  be a feasible response with support size  $k$  for the column player against strategy  $\mathbf{p}$  and let  $\lambda(k) := \frac{\sum_{j=1}^k \beta_j - 2}{2k}$ . Then the utility of the column player when he plays  $\mathbf{y}^k$  can be written as

$$\begin{aligned} T_c(\mathbf{x}, \mathbf{y}^k) &= \sum_{i=1}^n \mathbf{y}_i^k \cdot \beta_i - \sum_{i=1}^n (\mathbf{x}_i^k)^2 - \mathbf{q}^T \mathbf{q} \\ &= \frac{1}{4} \sum_{i=1}^k \beta_i^2 - k \cdot (\lambda(k))^2 - \mathbf{q}^T \mathbf{q}. \end{aligned}$$

The goal now is to prove that  $T_c(\mathbf{x}, \mathbf{y}^{k+1}) - T_c(\mathbf{x}, \mathbf{y}^k) > 0$ . By the previous analysis for  $T_c(\mathbf{x}, \mathbf{y}^k)$  and if  $A := \sum_{i=1}^k \beta_i - 2$ , then

$$\begin{aligned} T_c(\mathbf{x}, \mathbf{y}^{k+1}) - T_c(\mathbf{x}, \mathbf{y}^k) &= \frac{1}{4} \sum_{i=1}^{k+1} \beta_i^2 - (k+1)(\lambda(k+1))^2 - \frac{1}{4} \sum_{i=1}^k \beta_i^2 + k \cdot (\lambda(k))^2 \\ &= \frac{1}{4(k+1)} (k\beta_{k+1}^2 + A^2 - 2A\beta_{k+1}) \\ &> \frac{1}{4(k+1)} (k + A^2 - 2A) \quad (\text{since } 1 < \beta_{k+1} \leq 2 \text{ and } A > k - 2) \\ &> \frac{1}{4(k+1)} (k^2 - 5k + 8) \quad (\text{since } A > k - 2) \\ &> 0. \end{aligned}$$

□

Notice that  $\beta_k \geq 2\mathbf{p}_k > 1$ . Thus, the utility of the feasible response that assigns positive probability to pure strategy  $k$  is strictly greater than the utility of any feasible responses that does not assign probability to  $k$ . Thus strategy  $k$  is always played in a best response. Hence, the next lemma follows.

**Lemma 28** If there is a  $k \in [n]$  such that  $\mathbf{q}_k > 1/2$ , then in every best response  $\mathbf{y}^*$  the pure strategy  $k$  is played with positive probability.

Using now Lemma 28 we can provide a better bound for the regret the column player suffers, since in every best response  $\mathbf{y}^*$  the pure strategy  $k$  is played with positive probability. In order to prove our claim we first prove the following lemma.

**Lemma 29**  $\mathbf{y}^{*T} \mathbf{y}^* - 2\mathbf{y}_k^* \mathbf{q}_k \leq 1 - 2\mathbf{q}_k$ .

**Proof** Notice from (3) that for all  $i$  we get  $\mathbf{y}_i = \mathbf{y}_k + \frac{1}{2}(\beta_i - \beta_k)$ . Using that we can write the term  $\mathbf{y}^T \mathbf{y} = \sum_i \mathbf{y}_i^2$  as follows for a when  $\mathbf{y}$  has support size  $s$



$$\begin{aligned}\sum_{i=1}^s \mathbf{y}_i^2 &= \mathbf{y}_k^2 + \sum_{i \neq k} \mathbf{y}_i^2 \\ &= \mathbf{y}_k^2 + \sum_{i \neq k} \left( \mathbf{y}_k + \frac{1}{2}(\beta_i - \beta_k) \right)^2 \\ &= s\mathbf{y}_k^2 + \left( \sum_{i \neq k} (\beta_i - \beta_k) \right) \mathbf{y}_k + \frac{1}{4} \sum_{i \neq k} (\beta_k - \beta_i)^2.\end{aligned}$$

Then we can see that  $\mathbf{y}^{*T} \mathbf{y} - 2\mathbf{y}_k^{*T} \mathbf{q}_k$  is increasing as  $\mathbf{y}_k^*$  increases, since we know from Lemma 28 that  $\mathbf{y}_k^* > 0$ . This becomes clear if we take the partial derivative of  $\mathbf{y}^{*T} \mathbf{y}^* - 2\mathbf{y}_k^{*T} \mathbf{q}_k$  with respect to  $\mathbf{y}_k^*$  which is equal to

$$\begin{aligned}2s\mathbf{y}_k^* + \sum_{i \neq k} (\beta_i - \beta_k) - 2\mathbf{q}_k &= 2s\mathbf{y}_k^* + \sum_{i \neq k} 2(\mathbf{y}_i^* - \mathbf{y}_k^*) - 2\mathbf{q}_k \quad \left( \text{since } \mathbf{y}_i = \mathbf{y}_k + \frac{1}{2}(\beta_i - \beta_k) \right) \\ &= 2s\mathbf{y}_k^* + 2 \sum_{i \neq k} \mathbf{y}_i^* - 2(s-1)\mathbf{y}_k^* - 2\mathbf{q}_k \\ &= 2 \sum_{i=1}^s \mathbf{y}_i^* - 2\mathbf{q}_k \\ &= 2 - 2\mathbf{q}_k \\ &\geq 0 \quad (\text{since } \mathbf{y}_k^* > 0).\end{aligned}$$

Thus, the value of  $\mathbf{y}^{*T} \mathbf{y}^* - 2\mathbf{y}_k^{*T} \mathbf{q}_k$  is maximized when  $\mathbf{y}_k^* = 1$  and our claim follows.  $\square$

**Lemma 30** *Let  $\mathbf{y}^*$  be a best response when there is a pure strategy  $k$  with  $\mathbf{q}_k > 1/2$ . Then the regret for the column player under strategy profile  $(\mathbf{x}^*, \mathbf{y}^*)$  is bounded by  $2 - 2\delta$ .*

**Proof** Observe that by the definition of Algorithm 3 we get that the regret for the column player under the produced strategy profile is

$$\begin{aligned}\mathcal{R}^c(\mathbf{x}^*, \mathbf{y}^*) &\leq (1 - \delta) \left( \max_{\tilde{\mathbf{y}} \in \Delta} \{ \hat{\mathbf{x}}^T C \tilde{\mathbf{y}} \} + 2\tilde{\mathbf{y}}^T \mathbf{q}_k - 2\mathbf{y}^{*T} \mathbf{q} + \mathbf{y}^{*T} \mathbf{y}^* \right) \\ &\leq (1 - \delta) (1 + 2\mathbf{q}_k - 2\mathbf{y}^{*T} \mathbf{q} + \mathbf{y}^{*T} \mathbf{y}^*).\end{aligned}\tag{4}$$

From Lemma 29 we get  $\mathbf{y}^{*T} \mathbf{y}^* - 2\mathbf{y}^{*T} \mathbf{q} \leq 1 - 2\mathbf{q}_k$ . Thus, from (4) we get that  $\mathcal{R}^c(\mathbf{x}^*, \mathbf{y}^*) \leq 2 - 2\delta$ .  $\square$

Recall now that the regret for the row player is bounded by  $\delta$ , so if we optimize with respect to  $\delta$  the regrets are equal for  $\delta = 2/3$ . Thus, the next theorem follows, since when there is a  $k$  with  $\mathbf{q}_k > 1/2$  the Algorithm 1 produces a  $2/3$ -equilibrium. Hence, combining this with Lemma 25 we have that Theorem 31 follows for  $\delta = 5/7$ .

**Theorem 31** *In biased games with  $L_2^2$  penalties a  $5/7$ -equilibrium can be computed in polynomial time.*

## 6 Conclusions

We have studied games with infinite action spaces, and non-linear payoff functions. We have shown that Lipschitz continuity of the payoff function can be exploited to provide approximation algorithms. For  $\lambda_p$ -Lipschitz games, Lipschitz continuity of the payoff function allows us to provide an efficient algorithm for finding approximate equilibria. For penalty games, Lipschitz continuity of the penalty function allows us to provide a QPTAS. Finally, we provided strongly polynomial approximation algorithms for  $L_1$  and  $L_2^2$  distance biased games.

Several open questions stem from our paper. The most important one is to decide whether  $\lambda_p$ -Lipschitz games always possess an exact equilibrium. Another interesting feature is that we cannot verify efficiently in all penalty games whether a given strategy profile is an equilibrium, and so it seems questionable whether PPAD can capture the full complexity of penalty games. Is FIXP [18] the correct class for these problems, or is it the newly-introduced class BU [16]? On the other side, for the distance biased games that we studied in this paper, we have shown that we can decide in polynomial time if a strategy profile is an equilibrium. Is the equilibrium computation problem PPAD-complete for the two classes of games we studied? Are there any subclasses of penalty games, e.g. when the underlying normal form game is zero sum, that are easy to solve? Can we utilize the recent result of [15] to derive QPTASs for other families of games?

Another obvious direction is to derive better polynomial time approximation guarantees under for biased games. We believe that the optimization approach introduced by Tsaknakis and Spirakis [32] might tackle this problem. This approach has recently been successfully applied to polymatrix games [17]. A similar analysis might be applied to games with  $L_1$  penalties, which would lead to a constant approximation guarantee similar to the bound of 0.5 that was established in that paper. The other known techniques that compute approximate Nash equilibria [5] and approximate well supported Nash equilibria [9, 19, 25] solve a zero sum bimatrix game in order to derive the approximate equilibrium, and there is no obvious way to generalise this approach in penalty games.

**Acknowledgements** We would like to thank the anonymous reviewers for their valuable suggestions that significantly improved the paper.

**Open Access** This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if changes were made. The images or other third party material in this article are included in the article's Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit <http://creativecommons.org/licenses/by/4.0/>.

## References

1. Azrieli, Y., Shmaya, E.: Lipschitz games. *Math. Oper. Res.* **38**(2), 350–357 (2013)
2. Babichenko, Y.: Best-reply dynamics in large binary-choice anonymous games. *Games Econ. Behav.* **81**, 130–144 (2013)
3. Babichenko, Y., Barman, S., Peretz, R.: Simple approximate equilibria in large games. In: *Proceedings of EC*, pp. 753–770 (2014)
4. Barman, S.: Approximating Nash equilibria and dense bipartite subgraphs via an approximate version of Caratheodory’s theorem. In: *Proceedings of STOC 2015*, pp. 361–369 (2015)
5. Bosse, H., Byrka, J., Markakis, E.: New algorithms for approximate Nash equilibria in bimatrix games. *Theor. Comput. Sci.* **411**(1), 164–173 (2010)
6. Caragiannis, I., Kurokawa, D., Procaccia, A.D.: Biased games. In: *Proceedings of AAAI*, pp. 609–615 (2014)
7. Chapman, G.B., Johnson, E.J.: Anchoring, activation, and the construction of values. *Organ. Behav. Hum. Decis. Process.* **79**(2), 115–153 (1999)
8. Chen, X., Deng, X., Teng, S.-H.: Settling the complexity of computing two-player Nash equilibria. *J. ACM* **56**(3), 14:1–14:57 (2009)
9. Czumaj, A., Deligkas, A., Fasoulakis, M., Fearnley, J., Jurdzinski, M., Savani, R.: Distributed methods for computing approximate equilibria. *Algorithmica* **81**(3), 1205–1231 (2019)
10. Daskalakis, C., Goldberg, P.W., Papadimitriou, C.H.: The complexity of computing a Nash equilibrium. *SIAM J. Comput.* **39**(1), 195–259 (2009)
11. Daskalakis, C., Mehta, A., Papadimitriou, C.H.: Progress in approximate Nash equilibria. In: *Proceedings of EC*, pp. 355–358 (2007)
12. Daskalakis, C., Mehta, A., Papadimitriou, C.H.: A note on approximate Nash equilibria. *Theor. Comput. Sci.* **410**(17), 1581–1588 (2009)
13. Daskalakis, C., Papadimitriou, C.H.: Approximate Nash equilibria in anonymous games. *J. Econ. Theory* **156**, 207–245 (2015)
14. Deb, J., Kalai, E.: Stability in large Bayesian games with heterogeneous players. *J. Econ. Theory* **157**(C), 1041–1055 (2015)
15. Deligkas, A., Fearnley, J., Melissourgos, T., Spirakis, P.G.: Approximating the existential theory of the reals. In: *Web and Internet Economics—14th International Conference, WINE 2018*, Oxford, UK, December 15–17, 2018, *Proceedings*, pp. 126–139 (2018)
16. Deligkas, A., Fearnley, J., Melissourgos, T., Spirakis, P.G.: Computing exact solutions of consensus halving and the Borsuk–Ulam theorem. *CoRR*, [arXiv:1903.03101](https://arxiv.org/abs/1903.03101) (2019)
17. Deligkas, A., Fearnley, J., Savani, R., Spirakis, P.: Computing approximate Nash equilibria in polymatrix games. *Algorithmica* **77**(2), 487–514 (2017)
18. Etessami, K., Yannakakis, M.: On the complexity of Nash equilibria and other fixed points. *SIAM J. Comput.* **39**(6), 2531–2597 (2010)
19. Fearnley, J., Goldberg, P.W., Savani, R., Sørensen, T.B.: Approximate well-supported Nash equilibria below two-thirds. In: *SAGT*, pp. 108–119 (2012)
20. Fiat, A., Papadimitriou, C.H.: When the players are not expectation maximizers. In: *SAGT*, pp. 1–14 (2010)
21. Ganzfried, S., Sandholm, T.: Game theory-based opponent modeling in large imperfect-information games. In: *10th International Conference on Autonomous Agents and Multiagent Systems—Volume 2, AAMAS ’11*, pp. 533–540 (2011)
22. Ganzfried, S., Sandholm, T.: Safe opponent exploitation. In: *Proceedings of EC, EC ’12*, pp. 587–604. ACM, New York (2012)
23. Johanson, M., Zinkevich, M., Bowling, M.: Computing robust counter-strategies. In: Platt, J., Koller, D., Singer, Y., Roweis, S. (eds.) *Advances in Neural Information Processing Systems*, vol. 20, pp. 721–728. Curran Associates, Inc., Red Hook (2008)
24. Kahneman, D.: Reference points, anchors, norms, and mixed feelings. *Organ. Behav. Hum. Decis. Process.* **51**(2), 296–312 (1992)
25. Kontogiannis, S.C., Spirakis, P.G.: Well supported approximate equilibria in bimatrix games. *Algorithmica* **57**(4), 653–667 (2010)
26. Kozlov, M., Tarasov, S., Khachiyan, L.: The polynomial solvability of convex quadratic programming. *USSR Comput. Math. Math. Phys.* **20**(5), 223–228 (1980)

27. Lipton, R.J., Markakis, E., Mehta, A.: Playing large games using simple strategies. In: EC, pp. 36–41 (2003)
28. Mavronicolas, M., Monien, B.: The complexity of equilibria for risk-modeling valuations. *Theor. Comput. Sci.* **634**, 67–96 (2016)
29. Nash, J.: Non-cooperative games. *Ann. Math.* **54**(2), 286–295 (1951)
30. Rosen, J.B.: Existence and uniqueness of equilibrium points for concave  $n$ -person games. *Econometrica* **33**(3), 520–534 (1965)
31. Rubinstein, A.: Settling the complexity of computing approximate two-player Nash equilibria. In: 2016 IEEE 57th Annual Symposium on Foundations of Computer Science (FOCS), pp. 258–265. IEEE (2016)
32. Tsaknakis, H., Spirakis, P.G.: An optimization approach for approximate Nash equilibria. *Internet Math.* **5**(4), 365–382 (2008)
33. Tversky, A., Kahneman, D.: Judgment under uncertainty: heuristics and biases. *Science* **185**(4157), 1124–1131 (1974)

**Publisher's Note** Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.