

Query Complexity of Approximate Equilibria in Anonymous Games

Paul W. Goldberg^a, Stefano Turchetta^b

^a*Wolfson Building, Parks Road, University of Oxford, United Kingdom*

^b*KPMG Italy*

Abstract

We study the computation of Nash equilibria of anonymous games, via algorithms that use adaptive queries to a game's payoff function. We show that exact equilibria cannot be found via query-efficient algorithms, and exhibit a two-strategy, 3-player anonymous game whose exact equilibria require irrational numbers. We obtain positive results for known sub-classes of anonymous games.

Our main result is a new randomized query-efficient algorithm for approximate equilibria of two-strategy anonymous games that improves on the running time of previous algorithms. It is the first to obtain an inverse polynomial approximation in poly-time, and yields an efficient polynomial-time approximation scheme.

Keywords: Algorithms, Computational complexity, Game theory

1. Preliminaries

This paper studies *anonymous* games, in which a large number of players n share a constant number of pure strategies, and the payoff to a player depends on the number of players who use each strategy, but not their identities. Due to this property, these games have a polynomial (in n) size representation. Daskalakis and Papadimitriou [20] consider anonymous games and graphical games to be the two most important classes of concisely-represented multi-player games. Anonymous games appear frequently in practice, for

Email addresses: paul.goldberg@cs.ox.ac.uk (Paul W. Goldberg), stefano.turchetta@gmail.com (Stefano Turchetta)

example in voting systems, traffic routing, or auction settings. Although they have polynomial-sized representations, the representation may still be inconveniently large, making it desirable to work with algorithms that do not require all the data on a particular game of interest.

Query complexity is motivated in part by the observation that a game's entire payoff function may be syntactically cumbersome. It also leads to new results that distinguish the difficulty of alternative solution concepts. We assume that an algorithm has black-box access to the payoff function, via queries that specify an anonymized profile and return one or more of the players' payoffs.

1.1. Anonymous Games

A k -strategy anonymous game is a tuple $(n, k, \{u_j^i\}_{i \in [n], j \in [k]})$ that consists of n players, k pure strategies per player, and a utility function $u_j^i : \Pi_{n-1}^k \rightarrow [0, 1]$ for each player $i \in [n]$ and strategy $j \in [k]$, whose domain is the set $\Pi_{n-1}^k := \{(x_1, \dots, x_k) \in \mathbb{N}_0^k : \sum_{j \in [k]} x_j = n - 1\}$ of all possible ways to partition $n - 1$ players into the k strategies. The number of payoff values that define any such game is $n \cdot |\Pi_{n-1}^k| = O(n^k)$, i.e., polynomial in n if k is a constant, which we will always assume throughout this paper. In the special case of $k = 2$ (the setting of our main algorithm), u_j^i 's input is simply taken to be the number of players other than i that play strategy 1. The number of payoffs that define a 2-strategy game clearly is then $2n^2$. As indicated by u_j^i 's codomain, we make a standard assumption that all payoffs are normalized into the interval $[0, 1]$.

For two-strategy games [6, 7], the strategy played by player i can be represented by a Bernoulli random variable X_i indicating whether i plays strategy 1 (as opposed to strategy 2). Hence, a mixed strategy for i is the probability $p_i := \mathbb{E}[X_i]$ that i plays strategy 1. Let $X_{-i} := \sum_{\ell \in [n] \setminus \{i\}} X_\ell$ be the sum of all the random variables other than X_i . The expected utility obtained by player $i \in [n]$ for playing pure strategy $j \in \{1, 2\}$ against X_{-i} is

$$\mathbb{E}[u_j^i(X_{-i})] := \sum_{x=0}^{n-1} u_j^i(x) \cdot \Pr[X_{-i} = x].$$

For $k > 2$, let e_j denote the unit vector of length k with 1 at its j -th component. A mixed strategy of player i is represented using a random vector \mathcal{X}_i , which takes value e_j with the probability that i plays j . Letting p_j^i be the probability that player i plays strategy j , we then have $\mathbb{E}[\mathcal{X}_i] = (p_1^i, \dots, p_k^i)$.

Let $\mathcal{X}_{-i} := \sum_{\ell \in [n] \setminus \{i\}} \mathcal{X}_\ell$ be the sum of $n - 1$ such random vectors, where the subscript $-i$ denotes all players other than i . The expected utility obtained by player $i \in [n]$ for playing strategy $j \in [k]$ against \mathcal{X}_{-i} is

$$\mathbb{E}[u_j^i(\mathcal{X}_{-i})] := \sum_{x \in \Pi_{n-1}^k} u_j^i(x) \cdot \Pr[\mathcal{X}_{-i} = x].$$

Let $\mathcal{X} := (\mathcal{X}_i, \mathcal{X}_{-i})$. If i is playing a mixed strategy (p_1^i, \dots, p_k^i) , her expected payoff simply consists of a weighted average, i.e.,

$$\mathbb{E}[u^i(\mathcal{X})] := \sum_{j=1}^k p_j^i \cdot \mathbb{E}[u_j^i(\mathcal{X}_{-i})].$$

The probability mass function (p.m.f.) of \mathcal{X}_{-i} and X_{-i} respectively are a Poisson Multinomial Distribution and a Poisson Binomial Distribution. Both of them can be computed in polynomial time using dynamic programming (see e.g., [20]), i.e., expected utilities are computable in polynomial time.

1.2. Exact and Approximate Nash Equilibria.

With the above notation, we say that \mathcal{X}_i is a best-response if and only if $\mathbb{E}[u^i(\mathcal{X})] \geq \mathbb{E}[u_j^i(\mathcal{X}_{-i})]$ for all $j \in [k]$. A *Nash equilibrium* (NE) requires the players to be best-responding to each other; therefore, the above best-response condition must hold for every $i \in [n]$. This can be also viewed as no player having an incentive to deviate from her strategy. We consider a relaxation of NE, the notion of an ϵ -approximate Nash equilibrium (ϵ -NE), where every player's incentive to deviate is at most $\epsilon > 0$. We say that $(\mathcal{X}_i)_{i \in [n]}$, which represents a mixed-strategy profile, constitutes an ϵ -NE if for all $i \in [n]$ and all $j \in [k]$,

$$\mathbb{E}[u^i(\mathcal{X})] + \epsilon \geq \mathbb{E}[u_j^i(\mathcal{X}_{-i})].$$

This definition, however, does not prohibit allocating a small amount of probability to arbitrarily bad strategies. An ϵ -approximate well-supported Nash equilibrium (ϵ -WSNE) addresses this issue by forcing every player to place a positive amount of probability solely on ϵ -approximate best-responses, i.e., $(\mathcal{X}_i)_{i \in [n]}$ constitutes an ϵ -WSNE if for all $i \in [n]$, all $j \in [k]$, and all $\ell \in \text{supp}(\mathbb{E}[\mathcal{X}_i])$ ¹,

$$\mathbb{E}[u_\ell^i(\mathcal{X}_{-i})] + \epsilon \geq \mathbb{E}[u_j^i(\mathcal{X}_{-i})].$$

¹Given a vector v , $\text{supp}(v)$ is used to denote the support of v , i.e., $\text{supp}(v) = \{i : v_i > 0\}$.

Although an ϵ -WSNE is also an ϵ -NE, the converse need not be true.

1.2.1. Sub-classes of Anonymous Games

We investigate the query complexity of solving certain subclasses of anonymous games studied earlier in [8, 20].

An anonymous game is *symmetric* if for all $i, \ell \in [n]$, all $j \in [k]$, and all $x \in \Pi_{n-1}^k$, then $u_j^i(x) = u_j^\ell(x)$, i.e., all players share the same utility function. An anonymous game is *self-anonymous* if for all $i \in [n]$, all $j, \ell \in [k]$, and all $x \in \{y \in \Pi_{n-1}^k : y_\ell \neq 0\}$, then $u_j^i(x) = u_\ell^i(x + e_j - e_\ell)$, i.e., player i 's preferences depend on how all the n players are partitioned into the k strategies; therefore, i does not distinguish herself from the others. An anonymous game is *self-symmetric* if it is both symmetric and self-anonymous.

We also consider anonymous games whose payoff functions have a ‘‘Lipschitz property’’ that imposes a limit on the extent to which any player can affect any other player’s payoff by switching to a different strategy. An anonymous game is λ -*Lipschitz* (where λ is a non-negative parameter called the Lipschitz constant) if for all $i \in [n]$, $j \in [k]$, and all $x, y \in \Pi_{n-1}^k$, $|u_j^i(x) - u_j^i(y)| \leq \lambda \|x - y\|_1$, and $\|\cdot\|_1$ denotes the L_1 norm. Our algorithm for Lipschitz games (Section 3) is an ingredient of our main algorithm for general two-strategy anonymous games (Section 4).

1.3. Query-efficiency and Payoff Query Models.

Our general interest is in polynomial-time algorithms that find solutions of anonymous games, while checking just a small fraction of the $O(n^k)$ payoffs of an n -player, k -strategy game. The basic kind of query is a *single-payoff query* which receives as input a player $i \in [n]$, a strategy $j \in [k]$, and a partition $x \in \Pi_{n-1}^k$ of $n - 1$ players into the k strategies, and it returns the corresponding payoff $u_j^i(x)$.

The *query complexity* of an algorithm is the expected number of single-payoff queries that it needs in the worst case. Hence, an algorithm is query-efficient if its query complexity is $o(n^k)$.

A *profile query* (used in [25]) consists of an action profile $(a_1, \dots, a_n) \in [k]^n$ as input and outputs the payoffs that *every* player i obtains according to that profile. Clearly, any profile query can be simulated by a sequence of n single-payoff queries. Finally, an *all-players query* consists of a pair (j, x) for $j \in [k]$, $x \in \Pi_{n-1}^k$, and the response to (j, x) is the vector of all players’ utilities $(u_j^1(x), \dots, u_j^n(x))$.

We consider the cost of a query to be equal to the number of payoffs it returns; hence, a profile or an all-players query costs n single-payoff queries. We find that an algorithm being constrained to utilize profile queries may incur a linear loss in query-efficiency (cf. Section 2). Therefore, we focus on single-payoff and all-players queries, which better exploit the symmetries of anonymous games.

1.4. Related Work

In the last decade, there has been interest in the complexity of computing approximate Nash equilibria. A main reason is the **PPAD**-completeness results for computing an exact NE, for normal-form games [16, 10], where the latter paper extends the hardness also to fully polynomial-time approximation schemes (FPTAS). The possible existence of a FPTAS for anonymous games is an open problem, but [11] shows **PPAD**-completeness for finding ϵ -NE for anonymous games with 7 strategies, for exponentially small ϵ . The **FIXP**-completeness results of [24] for multiplayer games show an algebraic obstacle to the task of writing down a useful description of an exact equilibrium. On the other hand, there exists a subexponential-time algorithm to find an ϵ -NE in normal-form games [33], raising the well-known question of the possible existence of a PTAS for these games. A recent paper of Rubinfeld [34] rules out the possibility of a PTAS, assuming an Exponential Time Hypothesis for **PPAD**.

Daskalakis and Papadimitriou proved that anonymous games admit a PTAS and provided several improvements of its running time over the past few years. Their first algorithm [17] concerns two-strategy games and is based upon the quantization of the strategy space into nearby multiples of ϵ . This result was also extended to the multi-strategy case [18]. Daskalakis [14] subsequently gave an efficient PTAS whose running time is $\text{poly}(n) \cdot (1/\epsilon)^{O(1/\epsilon^2)}$, which relies on a better understanding of the structure of ϵ -equilibria in two-strategy anonymous games: There exists an ϵ -WSNE where either a small number of the players – at most $O(1/\epsilon^3)$ – randomize and the others play pure strategies, or whoever randomizes plays the same mixed strategy. Furthermore, Daskalakis and Papadimitriou [19] proved a lower bound on the running time needed by any *oblivious* algorithm, which lets the latter algorithm be essentially optimal. In the same article, they show that the lower bound can be broken by utilizing a non-oblivious algorithm, which has a running time for finding an ϵ -equilibrium in two-strategy anonymous games of $\text{poly}(n) \cdot (1/\epsilon)^{O(\log^2(1/\epsilon))}$. A complete proof is in [21]. This was improved (via

an improved upper bound on the cover size of Poisson binomial distributions) to $\text{poly}(n) \cdot (1/\epsilon)^{O(\log(1/\epsilon))}$ in [22], also for $k = 2$. It has been recently shown that also k -strategy anonymous games admit an efficient PTAS [15, 23].

We mention some related work on λ -Lipschitz games (as defined in Section 1.2.1). Any λ -Lipschitz k -strategy anonymous game is guaranteed to have an ϵ -approximate *pure* Nash equilibrium, with $\epsilon = O(\lambda k)$ [2, 20]. The convergence rate to a Nash equilibrium of best-reply dynamics in the context of two-strategy λ -Lipschitz anonymous games is studied by [32, 3]. In general, Lipschitz games need not be anonymous. Azrieli and Shmaya [2] study the behaviour of λ in terms of n for which there exist *pure* approximate equilibria of such games. [27] study the query complexity of searching for approximate equilibria of Lipschitz games (not necessarily anonymous).

Earlier work on the query complexity of classes of games includes the following. Fearnley et al. [25] presented the first series of results: they studied bimatrix games, graphical games, and congestion games on graphs. Similar to our negative result for exact equilibria of anonymous games, it was shown that a Nash equilibrium in a bimatrix game with k strategies per player requires k^2 queries, even in zero-sum games. However, more positive results arise if we move to ϵ -approximate Nash equilibria. Approximate equilibria of bimatrix games were studied in more detail in [26].

The query complexity of equilibria of unrestricted n -player games —where payoff functions are exponentially-large— was analyzed in [29, 4, 28, 9]. Hart and Nisan [29] showed that exponentially many deterministic queries are required to find a $\frac{1}{2}$ -approximate correlated equilibrium (CE) and that any randomized algorithm that finds an exact CE needs $2^{\Omega(n)}$ queries. Notice that lower bounds on correlated equilibria automatically apply to Nash equilibria. Goldberg and Roth [28] investigated in more detail the randomized query complexity of ϵ -CE and of the more demanding ϵ -well-supported CE. Babichenko [4] proved an exponential-in- n randomized lower bound for finding an ϵ -WSNE in n -player, k -strategy games, for constant $k = 10^4$ and $\epsilon = 10^{-8}$, extended by Chen et al [9] to an exponential lower bound for ϵ -Nash equilibria of binary-action games. Finally, Babichenko and Rubinfeld [5] show an exponential lower bound on the *communication complexity* of finding approximate Nash equilibria of n -player games. These exponential lower bounds do not hold in anonymous games, which can be fully revealed with a polynomial number of queries.

1.5. Our Results and their Significance; Organisation of the Paper

Section 2 explains the relationships between the query models we use. The main technical results begin in Section 3, where we present a query-efficient algorithm that finds an approximate *pure* Nash equilibrium in two-strategy λ -Lipschitz games (Algorithm 1; Theorem 3.1), which is subsequently used by Algorithm 2.

Section 4 presents our main result, Theorem 4.1. It shows that Algorithm 2 is a new randomized approximation scheme² for two-strategy anonymous games that differs conceptually from previous ones and offers new performance guarantees. It is query-efficient (using $o(n^2)$ queries) and has improved computational efficiency. It is the first PTAS for anonymous games that is polynomial in a setting where n and $1/\epsilon$ are polynomially related. In particular, its runtime is polynomial in n in a setting where $1/\epsilon$ may grow in proportion to $n^{1/4}$ and also has an improved polynomial dependence on n for all $\epsilon \geq n^{-1/4}$. In more detail, for any $\epsilon \geq n^{-1/4}$, the algorithm adaptively finds a $O(\epsilon)$ -NE with $\tilde{O}(\sqrt{n})$ (where we use $\tilde{O}(\cdot)$ to hide polylogarithmic factors) all-players queries (i.e., $\tilde{O}(n^{3/2})$ single payoffs) and runs in time $\tilde{O}(n^{3/2})$. This improves on the run-time of the algorithm of [20], which is $\text{poly}(n) \cdot (1/\epsilon)^{O(\log^2(1/\epsilon))}$ for a higher-degree polynomial in n . Recently, a follow-up paper to this one by Cheng et al. [13] extends our general approach to the k -strategy case. The idea is to make each player play a mixture of some strategy with the uniform distribution, which corresponds to a search over an appropriate class of Poisson multinomial distributions. [13] also obtain a quantitative improvement for the two-strategy case, via searching over a less constrained class of mixed strategies for the players.

Section 5 justifies our interest in approximate, rather than exact, Nash equilibria. We prove that even in two-strategy anonymous (indeed, self-anonymous) games, there does not exist a query-efficient algorithm for exact Nash equilibrium (Theorem 5.1). Alongside this, we provide an example of a three-player, two-strategy anonymous game whose unique Nash equilibrium needs all players to randomize with an irrational amount of probability (Theorem 5.2), answering a question posed in [20].

Section 6 focuses on the search for pure exact equilibria, in subclasses of

²To make Theorem 4.1 easier to read, we state it only for the best attainable approximation (i.e., $n^{-1/4}$); however, it is possible to set parameters to get any approximation $\epsilon \geq n^{-1/4}$. For details, see the proof of Theorem 4.1.

anonymous games where these are guaranteed to exist. We give tight bounds for two-strategy symmetric games, also k -strategy self-symmetric games, in the latter case via connecting them with the query complexity of searching for a local optimum of a function on a grid graph, allowing us to apply pre-existing results for that problem.

Section 7 is concerned with mixed-strategy equilibria of self-anonymous games. In contrast with our negative result (Theorem 5.1) for exact equilibria of these games, a uniform randomization over all the strategies is always an $O(n^{-1/2})$ -WSNE, for games having a constant number of strategies. Therefore, no payoff needs to be queried for this. Moreover, we give a reduction that maps any general two-strategy anonymous game G to a two-strategy self-anonymous game G' such that if an FPTAS for G' exists, then there also exists one for G . The possible existence of an FPTAS is the main open algorithmic question in the context of anonymous games, thus we show that—in the two-strategy case—the search for an answer to this question can focus on self-anonymous games. We conclude in Section 8.

2. Comparison of Query Models

It's convenient to present our algorithms in terms of all-player queries and profile queries, as defined in Section 1.3. Here we note some limits to the extent to which alternative queries can simulate each other. To query the payoffs resulting for a mixed-strategy profile p , notice that they can be approximated by randomly sampling pure profiles from p and using these as profile queries. Recall that one profile or one all-players query costs n single-payoff queries due to returning n payoffs. We use PR, AP, and SP to denote profile, all-players, and single-payoff queries, respectively.

2.1. Simulating Profile Queries

A PR query can be simulated by a sequence of n SP queries by requesting for each player the payoff associated with the strategy profile of the PR query.

A PR query can be simulated using at most k AP queries, as follows. Let $a = (a_1, \dots, a_n) \in [k]^n$ be a given profile query. Let $N_j \subseteq [n]$ be the players who play j in a . For all $j \in [k]$ such that $N_j \neq \emptyset$, we query the payoff for strategy j against the partition $(|N_1|, \dots, |N_j| - 1, \dots, |N_k|) \in \Pi_{n-1}^k$. The cost, therefore, increases at most by a factor of k .

2.2. Simulating Single-payoff Queries

A sequence of SP queries can obviously be simulated by a sequence of PR or AP queries having the same length. However, this approach increases the total cost by a factor of n since we receive the payoffs of every player. In general, it cannot be simulated by a shorter sequence, for example if every SP query is based on a different partition in Π_{n-1}^k .

2.3. Simulating All-players Queries

Clearly, any AP query can be simulated by n SP queries. We consider whether AP queries can be efficiently simulated by PR queries, and argue that this is generally not the case. Clearly, n profile queries suffice to obtain all the information returned by one AP query, and there exist examples where this upper bound is required. In section 2.4, we present such an example.

However, there are also cases in which an all-player query can be simulated by a constant number of profile queries. For instance, suppose we are dealing with a two-strategy anonymous game, and let $\alpha \in (0, 1)$ be a constant. If an AP query retrieves the payoff for playing strategy $j \in \{1, 2\}$ when αn players (in total) are playing strategy j , then $1/\alpha$ PR queries are enough to get all the information. Consider the following sequence of PR queries when $j = 1$. The case $j = 2$ is similar.

$$\underbrace{(1, \dots, 1)}_{\alpha n}, \underbrace{(2, \dots, 2)}_{(1-\alpha)n}, \underbrace{(2, \dots, 2)}_{\alpha n}, \underbrace{(1, \dots, 1)}_{\alpha n}, \underbrace{(2, \dots, 2)}_{(1-2\alpha)n}, \dots, \underbrace{(2, \dots, 2)}_{(1-\alpha)n}, \underbrace{(1, \dots, 1)}_{\alpha n}.$$

Clearly, every query lets αn players know their payoffs; therefore, $1/\alpha$ PR queries suffice to simulate an all-players query as specified above. The above simulation suggests that if an AP query algorithm asks the payoffs for strategy j subject to the constraint that the fraction of players using j is bounded away from 0, then it can be simulated by a PR query-algorithm whose cost is only increased by a constant factor.

2.4. A Quadratic Cost Lower Bound in the Profile Model

We show that finding an ϵ -WSNE, for any $\epsilon < 1/2$, in a two-strategy anonymous game may require a linear number of profile queries, i.e., a cost of n^2 . Due to the fact that a two-strategy anonymous game incorporates $2n^2$ payoffs, such a lower bound rules out the possibility of a profile-query efficient algorithm for well-supported approximate equilibria.

Definition 2.1. Let \mathcal{D}_n be the following distribution over two-strategy n -player anonymous games where every player's payoff takes a value in $\{0, \frac{1}{2}, 1\}$. Let a player $h \in [n]$ be chosen uniformly at random. Let h 's and the other players' (ℓ denotes a typical player different from h) payoffs be defined as in Figure 1.

<table style="border-collapse: collapse; margin: auto;"> <tr> <td style="padding: 5px;">x</td> <td style="padding: 5px;">0</td> <td style="padding: 5px;">...</td> <td style="padding: 5px;">$n-2$</td> <td style="padding: 5px;">$n-1$</td> </tr> <tr> <td style="padding: 5px;">$u_1^h(x)$</td> <td style="border: 1px solid black; padding: 5px;">$\frac{1}{2}$</td> <td style="border: 1px solid black; padding: 5px;">...</td> <td style="border: 1px solid black; padding: 5px;">$\frac{1}{2}$</td> <td style="border: 1px solid black; padding: 5px;">$\frac{1}{2}$</td> </tr> <tr> <td style="padding: 5px;">$u_2^h(x)$</td> <td style="border: 1px solid black; padding: 5px;">0</td> <td style="border: 1px solid black; padding: 5px;">...</td> <td style="border: 1px solid black; padding: 5px;">0</td> <td style="border: 1px solid black; padding: 5px;">1</td> </tr> </table> <p style="text-align: center;">(a) h's payoff</p>	x	0	...	$n-2$	$n-1$	$u_1^h(x)$	$\frac{1}{2}$...	$\frac{1}{2}$	$\frac{1}{2}$	$u_2^h(x)$	0	...	0	1	<table style="border-collapse: collapse; margin: auto;"> <tr> <td style="padding: 5px;">x</td> <td style="padding: 5px;">0</td> <td style="padding: 5px;">...</td> <td style="padding: 5px;">$n-2$</td> <td style="padding: 5px;">$n-1$</td> </tr> <tr> <td style="padding: 5px;">$u_1^\ell(x)$</td> <td style="border: 1px solid black; padding: 5px;">$\frac{1}{2}$</td> <td style="border: 1px solid black; padding: 5px;">...</td> <td style="border: 1px solid black; padding: 5px;">$\frac{1}{2}$</td> <td style="border: 1px solid black; padding: 5px;">$\frac{1}{2}$</td> </tr> <tr> <td style="padding: 5px;">$u_2^\ell(x)$</td> <td style="border: 1px solid black; padding: 5px;">0</td> <td style="border: 1px solid black; padding: 5px;">...</td> <td style="border: 1px solid black; padding: 5px;">0</td> <td style="border: 1px solid black; padding: 5px;">0</td> </tr> </table> <p style="text-align: center;">(b) Other players' payoff</p>	x	0	...	$n-2$	$n-1$	$u_1^\ell(x)$	$\frac{1}{2}$...	$\frac{1}{2}$	$\frac{1}{2}$	$u_2^\ell(x)$	0	...	0	0
x	0	...	$n-2$	$n-1$																											
$u_1^h(x)$	$\frac{1}{2}$...	$\frac{1}{2}$	$\frac{1}{2}$																											
$u_2^h(x)$	0	...	0	1																											
x	0	...	$n-2$	$n-1$																											
$u_1^\ell(x)$	$\frac{1}{2}$...	$\frac{1}{2}$	$\frac{1}{2}$																											
$u_2^\ell(x)$	0	...	0	0																											

Figure 1: Definition of \mathcal{D}_n 's payoffs. x denotes the number of players who play strategy 1.

Remark 2.1. For $\epsilon < 1/2$, any randomized profile-query algorithm needs to make $\Omega(n)$ queries, i.e. $\Omega(n^2)$ payoffs, to find an ϵ -WSNE of a given two-strategy, n -player anonymous game.

Proof. Suppose $\epsilon < 1/2$, and consider an ϵ -WSNE of a game of the kind defined in Definition 2.1. In an ϵ -WNSE, every player ℓ must play pure strategy 1, and as a result, the “hidden” player h must play pure strategy 2. In order to find an ϵ -WNSE, it's necessary to discover which player has the payoff of 1 (when she plays 2 while all other players play 1). Any profile query that finds this payoff of 1 must take the form $(1, \dots, 1, 2, 1, \dots, 1) \in \{1, 2\}^n$.

We use Yao's minimax principle: assume that a game is sampled from \mathcal{D}_n of Definition 2.1, and consider the expected number of queries made by a deterministic algorithm, in order to find the payoff of 1. A query that does not find a payoff of 1 provides no new information to the algorithm, so queries of the form $(1, \dots, 1, 2, 1, \dots, 1) \in \{1, 2\}^n$ must be made until the 2 appears at h 's index. Due to h being chosen uniformly at random, a linear number of profile queries is required. \square

Remark 2.2. The all-players query $(2, n-1) \in \{1, 2\} \times \{0, \dots, n-1\}$ suffices to discover the pure Nash equilibrium of a game coming from \mathcal{D}_n .

3. Two-strategy λ -Lipschitz Games

A two-strategy anonymous game is λ -Lipschitz if for all $i \in [n]$, $j \in \{1, 2\}$, and $x, y \in \{0, \dots, n-1\}$ (recall that x and y denote the number of opponents who play strategy 1), it holds that $|u_j^i(x) - u_j^i(y)| \leq \lambda |x - y|$, where $\lambda \geq 0$ is the Lipschitz constant. We show how a solution can be efficiently found via a binary search on $\{0, \dots, n-1\}$. (This binary-search approach does not seem applicable to k -strategy games with $k > 2$.)

The following algorithms exploit our knowledge of the existence of *pure* approximate equilibria in Lipschitz games [2, 20]. Algorithm 1 is used in Section 4 as a subroutine of Algorithm 2 for general (not necessarily Lipschitz) anonymous two-strategy games. For notational convenience with regard to this subsequent result, in Algorithm 1 and Theorem 3.1 we use \bar{G} to denote the input game rather than G .

Definition 3.1. Let $(j, x) \in \{1, 2\} \times \{0, \dots, n-1\}$ be the input for an all-players query. For $\delta \geq 0$, a δ -accurate all-players query returns a tuple of values $(f_j^1(x), \dots, f_j^n(x))$ such that for all $i \in [n]$, $|u_j^i(x) - f_j^i(x)| \leq \delta$, i.e., they are within an additive δ of the correct payoffs $(u_j^1(x), \dots, u_j^n(x))$.

Theorem 3.1. Let $\bar{G} = (n, 2, \{\bar{u}_j^i\}_{i \in [n], j \in \{1, 2\}})$ be an n -player, two-strategy λ -Lipschitz anonymous game. Algorithm 1 finds a pure-strategy $3(\lambda + \delta)$ -NE using $4 \log n$ δ -accurate all-players payoff queries.

Proof. Consider the function $\phi : \{0, \dots, n-1\} \rightarrow \{-n, \dots, n\}$ defined in Algorithm 1. It can be readily checked that if $BR_1(0) = 0$ or $BR_1(n-1) = n$, then the solutions returned are correct. Alternatively, the algorithm has to find x such that $\phi(x) > 0$ and $\phi(x+1) \leq 0$, in the case when $\phi(0) > 0$ and $\phi(n-1) \leq 0$, and it is clear that $\log n$ evaluations of ϕ suffice to find x . Moreover, notice that a candidate x can be checked using the following 4 all-players queries: $(1, x)$, $(2, x)$, $(1, x+1)$, $(2, x+1)$. Hence, we need in total $4 \log n$ queries to find x .

The main task is to prove that given x satisfying these conditions, a pure profile \bar{p} can indeed be constructed in the way described, and that \bar{p} is indeed an approximate equilibrium.

Suppose for a contradiction that \bar{p} could not be constructed in the way described. For example, suppose that more than $x+1$ players are required to play 1 due to satisfying $\bar{u}_1^i(x) - \bar{u}_2^i(x) > 2\delta$. We argue that this would contradict that $\phi(x+1) \leq 0$. The fact that $\phi(x+1) \leq 0$ means that there

every player's utility function, so that it becomes λ -Lipschitz continuous for some λ . We smooth a utility function by requiring every player to use some amount of randomness. Specifically, for some small ζ we make every player place probability either ζ or $1 - \zeta$ onto strategy 1. Consequently, the expected payoff for player i is obtained by averaging her payoff values w.r.t. a sum of two binomial distributions, consisting of a discrete bell-shaped distribution whose standard deviation is at least $\zeta\sqrt{n}$.

We construct the smooth game \bar{G} in the following manner. The payoff received in \bar{G} by player i when x other players are playing strategy 1 is given by the expected payoff received in G by player i when x other players play 1 with probability $1 - \zeta$ and $n - 1 - x$ other players play 1 with probability ζ . This creates a λ -Lipschitz game \bar{G} with $\lambda = O(1/\zeta\sqrt{n})$.

Due to dealing with a two-strategy Lipschitz game, we can use the bisection method of Algorithm 1. If we were allowed to query \bar{G} directly, a logarithmic number of all-players queries would suffice. Unfortunately, this is not the case; thus, we need to simulate a query to \bar{G} with a small number of queries to the original game G . Those queries are randomly sampled from the mixed anonymous profile above, and we take enough samples to ensure we get good estimates of the payoffs in \bar{G} with sufficiently high probability.

Thus, we are able to find an approximate pure Nash equilibrium of \bar{G} with $\tilde{O}(\sqrt{n})$ all-players queries. This equilibrium is mapped back to G by letting the players who play strategy 1 in \bar{G} , play it with probability $1 - \zeta$ in G , and the ones who play strategy 2 in \bar{G} place probability ζ on strategy 1 in G . The quality of the approximation is proportional to $(\zeta + (\zeta\sqrt{n})^{-1})$.

Before presenting our main algorithm (Algorithm 2) and proving its efficiency (Theorem 4.1), we state the following lemmas that are used in the proof.

Lemma 4.1 ([20]). *Let X, Y be two random variables over $\{0, \dots, n\}$ such that $\|X - Y\|_{\text{TV}} \leq \delta$ (where $\|X - Y\|_{\text{TV}}$ denotes the total variation distance between X and Y , i.e., $1/2 \cdot \sum_{x=0}^n |\Pr[X = x] - \Pr[Y = x]|$). Let $f : \{0, \dots, n\} \rightarrow [0, 1]$. Then,*

$$\sum_{x=0}^n f(x) \cdot (\Pr[X = x] - \Pr[Y = x]) \leq 2\delta.$$

Lemma 4.2 (Simulation of a query to \bar{G} (Algorithm 2)). *Let $\delta, \tau > 0$. Let X be the sum of $n - 1$ Bernoulli random variables representing a mixed*

profile of $n - 1$ players in an n -player game G . Suppose we want to estimate, with additive error δ , the expected payoffs $\mathbb{E}[u_j^i(X)]$ for all $i \in [n], j \in \{1, 2\}$. This can be done with probability $\geq 1 - \tau$ using $(1/2\delta^2) \cdot \log(4n/\tau)$ all-players queries.

Proof. Suppose we draw a set of N random samples $\{Z_1, \dots, Z_N\}$ from the probability distribution of X (which can be done by computing each Z_i as a sum of 0/1 outcomes of $n - 1$ biased coin flips. For each Z_ℓ and each $j \in \{1, 2\}$ we can make an all-players query that tells us, for every player, that payoff obtained by that player for playing j when Z_i other players play 1. So, a total of $2N$ queries are made.

Let $\hat{U}_j^i := 1/N \cdot \sum_{\ell=1}^N u_j^i(Z_\ell)$ denote our estimate of $\mathbb{E}[u_j^i(X)]$. Then $\mathbb{E}[\hat{U}_j^i] = \mathbb{E}[u_j^i(X)]$.

We can now use Hoeffding's inequality to get that

$$\Pr \left[\left| \hat{U}_j^i - \mathbb{E}[u_j^i(X)] \right| \geq \delta \right] \leq 2 \exp(-2\delta^2 N).$$

Since there are $2n$ quantities we desire to estimate (2 strategies per player) within additive error δ , we require a failure probability of at most $\tau/2n$, so that with a union bound we get that the estimates have additive error $\leq \delta$ with probability at least $1 - \tau$. Thus, we need that $2 \exp(-2\delta^2 N) \leq \tau/2n$, which is satisfied for $N \geq (1/2\delta^2) \cdot \log(4n/\tau)$. \square

Lemma 4.3. *Let Y be a binomial random variable, $Y \sim B(n, p)$ for $p \in [\zeta, 1 - \zeta]$, where $0 < \zeta < \frac{1}{2}$. Then, the probability value at Y 's mode is at most $\frac{e}{2\pi\zeta\sqrt{n}} \left(1 + \frac{1}{\zeta n}\right)$, i.e., $O\left(\frac{1}{\zeta\sqrt{n}}\right)$.*

Proof. The mode of Y is either $\lfloor np \rfloor$ or $\lfloor np \rfloor + 1$. We bound the ratio $\Pr[Y = \lfloor np \rfloor + 1] / \Pr[Y = \lfloor np \rfloor]$, then we apply Stirling's bound on the value $\Pr[Y = \lfloor np \rfloor]$. We bound the ratio as follows, where we use $\{a\}$ to

denote the fractional part of a .

$$\begin{aligned}
\frac{\binom{n}{\lfloor np \rfloor + 1} \cdot p^{\lfloor np \rfloor + 1} \cdot (1-p)^{n - \lfloor np \rfloor - 1}}{\binom{n}{\lfloor np \rfloor} \cdot p^{\lfloor np \rfloor} \cdot (1-p)^{n - \lfloor np \rfloor}} &= \frac{p}{1-p} \cdot \frac{\lfloor np \rfloor! \cdot (n - \lfloor np \rfloor)!}{(\lfloor np \rfloor + 1)! \cdot (n - \lfloor np \rfloor - 1)!} \\
&= \frac{p}{1-p} \cdot \frac{n - \lfloor np \rfloor}{\lfloor np \rfloor + 1} \\
&= \frac{p}{1-p} \cdot \frac{n - np + \{np\}}{\lfloor np \rfloor + 1} \\
&= \frac{p}{1-p} \cdot \left(\frac{(1-p)n}{\lfloor np \rfloor + 1} + \frac{\{np\}}{\lfloor np \rfloor + 1} \right) \\
&\leq \frac{np}{np} + \frac{p \cdot \{np\}}{(1-p) \cdot np} \\
&< 1 + \frac{1}{\zeta n}.
\end{aligned}$$

In the second to last step we used the fact that $\lfloor np \rfloor + 1 \geq np$. In the last one we used both $\{np\} < 1$ and $1-p \geq \zeta$. Next we bound the value at $x = \lfloor np \rfloor$ using Stirling's bounds.

$$\begin{aligned}
\Pr[Y = x] &= \binom{n}{x} \cdot p^x \cdot (1-p)^{n-x} \\
&\leq \frac{e \cdot n^{n+1/2} \cdot e^{-n} \cdot p^x \cdot (1-p)^{n-x}}{\sqrt{2\pi} \cdot x^{x+1/2} \cdot e^{-x} \cdot \sqrt{2\pi} \cdot (n-x)^{n-x+1/2} \cdot e^{-(n-x)}} \\
&= \frac{e}{2\pi} \cdot \frac{n^{n+1/2} \cdot p^x \cdot (1-p)^{n-x}}{(\lfloor np \rfloor)^{x+1/2} \cdot (n - \lfloor np \rfloor)^{n-x+1/2}} \\
&= \frac{e}{2\pi} \cdot \frac{n^{n+1/2} \cdot p^{\lfloor np \rfloor} \cdot (1-p)^{n - \lfloor np \rfloor}}{n^{\lfloor np \rfloor + 1/2} \cdot n^{n - \lfloor np \rfloor + 1/2} \cdot p^{\lfloor np \rfloor + 1/2} \cdot (1-p)^{n - \lfloor np \rfloor + 1/2}} \\
&= \frac{e}{2\pi} \cdot \frac{1}{\sqrt{p(1-p)} \cdot \sqrt{n}} \\
&\leq \frac{e}{2\pi} \cdot \frac{1}{\zeta \sqrt{n}} = O\left(\frac{1}{\zeta \sqrt{n}}\right).
\end{aligned}$$

If we combine $\Pr[Y = m]/\Pr[Y = x] < 1 + \frac{1}{\zeta n}$ with $\Pr[Y = x] \leq \frac{e}{2\pi} \cdot \frac{1}{\zeta \sqrt{n}}$, it follows that $\Pr[Y = m] \leq \frac{e}{2\pi \zeta \sqrt{n}} (1 + \frac{1}{\zeta n}) = O(\frac{1}{\zeta \sqrt{n}})$, concluding the proof. \square

Lemma 4.4. Let $Z := \sum_{i=1}^n Z_i$ be the sum of n independent 0-1 random variables such that for some $0 < \zeta \leq 1/2$, $\mathbb{E}[Z_i] \in \{\zeta, 1 - \zeta\}$ for all $i \in [n]$. Then, the probability value at Z 's mode is $O\left(\frac{1}{\zeta\sqrt{n}}\right)$.

Proof. $Z = X + Y$ for binomial random variables X, Y with $X \sim B(n_X, \zeta)$ and $Y \sim B(n_Y, 1 - \zeta)$, where $n_X + n_Y = n$. Z 's probability mass function can be written as

$$\Pr[Z = i] = \sum_{x=0}^{n_X} \Pr[Z = i | X = x] \cdot \Pr[X = x] = \sum_{x=0}^{n_X} \Pr[X = x] \cdot \Pr[Y = i - x].$$

Since $n_X + n_Y = n$, we have $\max\{n_X, n_Y\} \geq n/2 = \Omega(n)$. Assume n_X be the maximum. Then, by Lemma 4.3, $\Pr[X = x] \leq O\left(\frac{1}{\zeta\sqrt{n_X}}\right)$ for all $x = 0, \dots, n_X$. Hence,

$$\begin{aligned} \Pr[Z = i] &\leq \sum_{x=0}^{n_X} O\left(\frac{1}{\zeta\sqrt{n_X}}\right) \cdot \Pr[Y = i - x] \\ &= O\left(\frac{1}{\zeta\sqrt{n_X}}\right) \cdot \sum_{x=0}^{n_X} \Pr[Y = i - x] \\ &\leq O\left(\frac{1}{\zeta\sqrt{n_X}}\right) = O\left(\frac{1}{\zeta\sqrt{n}}\right). \end{aligned}$$

□

Lemma 4.5. Let $X^{(j,n)} := \sum_{i \in [n]} X_i$ denote the sum of n independent 0/1 random variables where $\mathbb{E}[X_i] = 1 - \zeta$ for all $i \leq j$, and $\mathbb{E}[X_i] = \zeta$ for all $i > j$. Then, for all $j \in [n]$, we have $\|X^{(j-1,n)} - X^{(j,n)}\|_{\text{TV}} = O\left(\frac{1}{\zeta\sqrt{n}}\right)$.

Proof. We use the following recursive formula for the probability mass function of a Poisson Binomial Distribution, as described in [30]. Due to this not depending on j , we use $X^{(*,n)}$ to denote a sum of n independent 0/1 random variables whose expectations can potentially be all different. Then if p_n is the expectation of the n -th variable,

$$\Pr[X^{(*,n)} = i] = (1 - p_n) \cdot \Pr[X^{(*,n-1)} = i] + p_n \cdot \Pr[X^{(*,n-1)} = i - 1]. \quad (1)$$

We want to bound the total variation distance between $X^{(j-1,n)}$ and $X^{(j,n)}$, i.e.,

$$\|X^{(j-1,n)} - X^{(j,n)}\|_{\text{TV}} = \frac{1}{2} \sum_{i=1}^n |\Pr[X^{(j-1,n)} = i] - \Pr[X^{(j,n)} = i]|.$$

Note that $X^{(j-1,n)}$ and $X^{(j,n)}$ differ only by how one coin flip is biased. Thus, we can use (1) to write

$$\Pr[X^{(j-1,n)} = i] = (1 - \zeta) \cdot \Pr[X^{(j-1,n-1)} = i] + \zeta \cdot \Pr[X^{(j-1,n-1)} = i - 1], \text{ and}$$

$$\Pr[X^{(j,n)} = i] = \zeta \cdot \Pr[X^{(j-1,n-1)} = i] + (1 - \zeta) \cdot \Pr[X^{(j-1,n-1)} = i - 1].$$

If we combine these with the total variation distance expression, we have the following expression for variation distance:

$$\begin{aligned} & \frac{1}{2} \sum_{i=1}^{n-1} |(1 - 2\zeta) \cdot \Pr[X^{(j,n-1)} = i] - (1 - 2\zeta) \cdot \Pr[X^{(j,n-1)} = i - 1]| \\ &= \frac{(1 - 2\zeta)}{2} \sum_{i=1}^{n-1} |\Pr[X^{(j,n-1)} = i] - \Pr[X^{(j,n-1)} = i - 1]|. \end{aligned}$$

By definition of $X^{(j,n-1)}$ and Lemma 4.4 we know that $\Pr[X^{(j-1,n-1)} = i] \leq O(\frac{1}{\zeta\sqrt{n-1}})$ for any $i \in \{0, \dots, n-1\}$. Let m be the mode of $X^{(j-1,n-1)}$. Due to $X^{(j-1,n-1)}$ being unimodal, we can split the sum $\sum_{i=0}^{n-1} |\Pr[X^{(j-1,n-1)} = i] - \Pr[X^{(j-1,n-1)} = i - 1]|$ into two summations over $\{0, \dots, m\}$ and $\{m + 1, \dots, n - 1\}$ where $\Pr[X^{(j-1,n-1)} = i]$ is, respectively, increasing or decreasing and, hence, remove the absolute value operator. Consider the summation over $\{0, \dots, m\}$; the other case is symmetric. Then,

$$\begin{aligned} & \sum_{i=1}^m \Pr[X^{(j-1,n-1)} = i] - \Pr[X^{(j-1,n-1)} = i - 1] \\ &= (\Pr[X^{(j-1,n-1)} = 1] - \Pr[X^{(j-1,n-1)} = 0]) + \dots \\ & \dots + (\Pr[X^{(j-1,n-1)} = m] - \Pr[X^{(j-1,n-1)} = m - 1]) \\ &= \Pr[X^{(j-1,n-1)} = m] - \Pr[X^{(j-1,n-1)} = 0] \leq \Pr[X^{(j-1,n-1)} = m]. \end{aligned}$$

Summing up both the increasing and decreasing side bounds, we get that

$$\begin{aligned} & \sum_{i=0}^{n-1} |\Pr[X^{(j-1,n-1)} = i] - \Pr[X^{(j-1,n-1)} = i - 1]| \leq \\ & 2 \cdot \Pr[X^{(j-1,n-1)} = m] = O\left(\frac{1}{\zeta\sqrt{n-1}}\right). \end{aligned}$$

Substituting back to the variation distance expression and observing that $1 - 2\zeta < 1$, we obtain

$$\|X^{(j-1,n)} - X^{(j,n)}\|_{\text{TV}} \leq O\left(\frac{1}{\zeta\sqrt{n-1}}\right) = O\left(\frac{1}{\zeta\sqrt{n}}\right),$$

as in the statement of the Lemma. \square

Definition 4.1. Let $G = (n, 2, \{u_j^i\}_{i \in [n], j \in \{1,2\}})$ be an anonymous game. For $\zeta > 0$, the ζ -smoothed version of G is a game $\bar{G} = (n, 2, \{\bar{u}_j^i\}_{i \in [n], j \in \{1,2\}})$ defined as follows. Let $X_{-i}^{(x)} := \sum_{\ell \neq i} X_\ell$ denote the sum of $n-1$ Bernoulli random variables where x of them have expectation equal to $1 - \zeta$, and the remaining ones have expectation equal to ζ . The payoff $\bar{u}_j^i(x)$ obtained by player $i \in [n]$ for playing strategy $j \in \{1, 2\}$ against $x \in \{0, \dots, n-1\}$ is

$$\bar{u}_j^i(x) := \sum_{y=0}^{n-1} u_j^i(y) \cdot \Pr[X_{-i}^{(x)} = y] = \mathbb{E}\left[u_j^i\left(X_{-i}^{(x)}\right)\right].$$

Theorem 4.1. Let $G = (n, 2, \{u_j^i\}_{i \in [n], j \in \{1,2\}})$ be an anonymous game. For ϵ satisfying $1/\epsilon = O(n^{1/4})$, Algorithm 2 can be used to find (with probability $\geq \frac{3}{4}$) an ϵ -NE of G , using $O(\sqrt{n} \cdot \log^2 n)$ all-players queries (hence, $O(n^{3/2} \cdot \log^2 n)$ single-payoff queries) in time $O(n^{3/2} \cdot \log^2 n)$.

Proof. Set ζ equal to ϵ and let \bar{G} be the ζ -smoothed version of G . We claim that \bar{G} is a λ -Lipschitz game for $\lambda = O((\zeta\sqrt{n})^{-1})$. Let $X_{-i}^{(x)}$ be as in Definition 4.1. By Lemma 4.5, $\|X_{-i}^{(x-1)} - X_{-i}^{(x)}\|_{\text{TV}} \leq O\left(\frac{1}{\zeta\sqrt{n}}\right)$ for all $x \in [n-1]$. Then by Lemma 4.1, we have

$$|\bar{u}_j^i(x-1) - \bar{u}_j^i(x)| \leq O\left(\frac{1}{\zeta\sqrt{n}}\right).$$

Theorem 3.1 shows that Algorithm 1 finds a pure-strategy $3(\lambda + \delta)$ -WSNE of \bar{G} , using $O(\log n)$ δ -accurate all-players queries. Thus, Algorithm 1 finds a $O(\frac{1}{\zeta\sqrt{n}} + \delta)$ -WSNE of \bar{G} , where δ is the additive accuracy of queries.

Despite not being allowed to query \bar{G} directly, we can simulate any δ -accurate query to \bar{G} with a set of randomized all-players queries to G . This is done in the body of Algorithm 2. By Lemma 4.2, for $\tau > 0$, $(1/2\delta^2) \log(4n/\tau)$

Algorithm 2: Approximate NE general payoffs

Data: ϵ ; query access to utility function u of n -player anonymous game G ; parameters τ (failure probability), δ (accuracy of queries).

Result: $O(\epsilon)$ -NE of G .

begin

Set $\zeta = \epsilon$. Let \bar{G} be the ζ -smoothed version of G , as in Definition 4.1.

// By Lemma 4.1 and Lemma 4.5 it follows that
// \bar{G} is λ -Lipschitz for $\lambda = O(1/\zeta\sqrt{n})$.

Apply Algorithm 1 to \bar{G} , simulating each all-players δ -accurate query to \bar{G} using multiple queries according to Lemma 4.2.

Let \bar{p} be the obtained pure profile solution to \bar{G} .

Construct p by replacing probabilities of 0 in \bar{p} with ζ and probabilities of 1 with $1 - \zeta$.

return p .

end

randomized queries to G correctly simulate a δ -accurate query to \bar{G} with probability $\geq 1 - \tau$.

In total, the algorithm makes $O(\log n \cdot (1/\delta^2) \cdot \log(n/\tau))$ all-players payoff queries to G . With a union bound over the $4 \log n$ simulated queries to \bar{G} , this works with probability $1 - 4\tau \log n$.

Once we find this pure-strategy $O\left(\frac{1}{\zeta\sqrt{n}} + \delta\right)$ -WSNE of \bar{G} , the last part of Algorithm 2 maps the pure output profile to a mixed one where whoever plays 1 in \bar{G} places probability $(1 - \zeta)$ on 1, and whoever plays 2 in \bar{G} places probability ζ on 1. It is easy to verify that the regret experienced by player i (that is, the difference in payoff between i 's payoff and i 's best-response payoff) in G is at most ζ more than the one she experiences in \bar{G} .

The extra additive ζ to the regret of players means that we have an ϵ -NE of G with $\epsilon = O(\zeta + \delta + \frac{1}{\zeta\sqrt{n}})$. The query complexity thus is $O(\log n \cdot (1/\delta^2) \cdot \log(n/\tau))$.

Setting $\delta = 1/\sqrt[4]{n}$, $\zeta = 1/\sqrt[4]{n}$, $\tau = 1/16 \log n$, we find an $O(1/\sqrt[4]{n})$ -Nash equilibrium using $O(\sqrt{n} \cdot \log^2 n)$ all-players queries with probability at least $3/4$. We remark that the above parameters can be chosen to satisfy any given approximation guarantee $\epsilon \geq n^{-1/4}$, i.e., simply find solutions to the equation

$\epsilon = \zeta + \delta + (\zeta\sqrt{n})^{-1}$. This allows for a family of algorithms parameterized by ϵ , for $\epsilon \in [n^{-1/4}, 1)$, thus an approximation scheme.

The runtime is equal to the number of single-payoff queries and can be calculated as follows. Calculating the value of $\phi(i)$ in Algorithm 1 takes $O(n\sqrt{n} \log n)$. We make $O(\sqrt{n} \log n)$ queries to G to simulate one in \bar{G} , and once we gather all the information, we need an additional linear time factor to count the number of players whose best response is 1. The fact that the above part is performed at every step of the binary search implies a total running time of $O(n^{3/2} \cdot \log^2 n)$ for Algorithm 1. Algorithm 2 simply invokes Algorithm 1 and only needs linear time to construct the profile p ; thus, it runs in the same time. \square

5. Negative Results for Exact Nash Equilibria

We lower-bound the number of single-payoff queries (the least constrained query model) needed to find an exact NE in an anonymous game. We construct games in which any algorithm must query most of the payoffs in order to determine what strategies form a NE. These challenging games are constructed to have the key property that in any Nash equilibrium, a constant fraction of the n players must randomize.

Example 5.1. Let G be the following two-strategy, n -player anonymous game. Let n be even. Half of the players have a utility function as shown by the top side (a) of Figure 2, and the remaining half as at (b).³

Lemma 5.1. *In any Nash equilibrium of G as defined in Example 5.1, at least $n/2$ players must use mixed strategies.*

Proof. Consider a NE of G and let p_i be the probability that player i plays 1, in that equilibrium.

Let $P_1^i(s)$ denote the expected payoff of player i for playing strategy 1, minus the expected payoff of i for playing 2, in the (possibly mixed) strategy

³This game is somewhat reminiscent of the *Village Versus Beach* game of Kalai [31], and both games have equilibria where all players fully randomise. However, the Village Versus Beach game also has a pure-strategy Nash equilibrium if the number of men/women is even, and if the number is odd, there's a Nash equilibrium where one man and one woman randomise uniformly, and the others play pure strategies. The game defined here is designed such that a constant fraction of players must mix, in *any* exact Nash equilibrium (Lemma 5.1).

x	0	1	2	...	$n-2$	$n-1$
$u_1^i(x)$	$\frac{1}{2} - (\frac{1}{2} - \frac{1}{2n})$	$\frac{1}{2} - (\frac{1}{2} - \frac{3}{2n})$	$\frac{1}{2} - (\frac{1}{2} - \frac{5}{2n})$...	$\frac{1}{2} + (\frac{1}{2} - \frac{3}{2n})$	$\frac{1}{2} + (\frac{1}{2} - \frac{1}{2n})$
$u_2^i(x)$	$\frac{1}{2}$	$\frac{1}{2}$	$\frac{1}{2}$...	$\frac{1}{2}$	$\frac{1}{2}$

(a) Payoff table for “majority-seeking” player i

x	0	1	2	...	$n-2$	$n-1$
$u_1^i(x)$	$\frac{1}{2} + (\frac{1}{2} - \frac{1}{2n})$	$\frac{1}{2} + (\frac{1}{2} - \frac{3}{2n})$	$\frac{1}{2} + (\frac{1}{2} - \frac{5}{2n})$...	$\frac{1}{2} - (\frac{1}{2} - \frac{3}{2n})$	$\frac{1}{2} - (\frac{1}{2} - \frac{1}{2n})$
$u_2^i(x)$	$\frac{1}{2}$	$\frac{1}{2}$	$\frac{1}{2}$...	$\frac{1}{2}$	$\frac{1}{2}$

(b) Payoff table for “minority-seeking” player i

Figure 2: Majority-minority game G 's payoffs. There are $\frac{n}{2}$ majority-seeking players and $\frac{n}{2}$ minority-seeking players. x denotes the number of players other than i who play 1. The key feature of the payoff function is that for a majority-seeking player, the advantage of playing 2 as opposed to 1 decreases linearly in x , and increases linearly in x for a minority-seeking player.

profile s . If, in a pure profile, x other players play strategy 1, then for a majority player i , $P_1^i(s) = (\frac{x}{n} - \frac{1}{2} + \frac{1}{2n})$. By linearity of expectations, if in a mixed strategy s , x is the expected number of other players who play 1, then $P_1^i(s) = (\frac{x}{n} - \frac{1}{2} + \frac{1}{2n})$. Consequently, the incentive for a majority player i' to play 1 is $\sum_{i \neq i'} \frac{p_i}{n} - \frac{n-1}{2n}$. Furthermore, the incentive for a minority player i' to play 2 is $\sum_{i \neq i'} \frac{p_i}{n} - \frac{n-1}{2n}$.

Suppose a majority player i' mixes with probability $p_{i'} \in (0, 1)$. Notice that $\sum_{i \neq i'} p_i = \frac{n-1}{2}$. The expected number of users of strategy 1 differs from the expected number of users of strategy 2 by less than 1. This means that no majority player may use a pure strategy; if he did, he would have an incentive to use the opposite strategy. It follows that all majority players must use mixed strategies.

Suppose a minority player i' who plays a mixed strategy uses $p_{i'} \in (0, 1)$. Suppose, in addition, all majority players play pure strategies. In that case, as before, any majority player would want to switch. So in this case, all majority players must mix, as before.

Finally, suppose all players play pure strategies. If strategies 1 and 2 both have the same number of users, then all majority players will want to switch. Alternatively, if, say, strategy 1 is used by more than $n/2$ players, it will be

being used by a minority player who will want to switch. Thus, all majority players must use mixed strategies.

This means that (for each majority player) at least $n/2$ payoffs need to be known for the computation of expected utilities. If one of these payoffs is unknown, there remains the possibility that it could take an alternative value that results in that player no longer being indifferent between 1 and 2, and having an incentive to deviate. \square

Theorem 5.1. *A randomised single-payoff query algorithm needs to query $\Omega(n^2)$ payoffs in expectation in order to find an exact Nash equilibrium of an n -player game, even for two-strategy self-anonymous games.*

Proof. Suppose we know that the game of interest is one of $2n^2 + 1$ candidate games, one of which is G (of Example 5.1), and the other $2n^2$ games are versions of G in which one of the $2n^2$ payoffs has been perturbed slightly (a different payoff being perturbed for each separate game). Suppose furthermore that the game of interest has been selected uniformly at random from one of these $2n^2 + 1$ candidates.

Consider a deterministic algorithm A that attempts to compute an exact equilibrium based on queries. If A is run on G , we know from Lemma 5.1 that A will output a Nash equilibrium N in which at least $n/2$ players mix. Suppose that when A is run on G , A queries only $n^2/100$ of G 's payoffs. If instead A gets a randomly-selected game from the candidate set, with probability $> \frac{9}{10}$, A will receive a perturbed game and also the queries will not show that it is perturbed. There is, separately, a probability at least $\frac{1}{8}$ that the player whose has a perturbed payoff will be one who plays a mixed strategy in N , and the perturbation will give him an incentive to deviate. (With probability just under $\frac{1}{2}$, one of the players who mix in N will get a perturbed payoff. With probability $\geq \frac{1}{2}$, that payoff will be associated with a partition of the other players that, in N , has positive probability of occurring, since from Lemma 5.1 at least $n/2$ players must mix, so $n/2$ partitions have positive probability.) Putting these together, with probability $> \frac{1}{100}$ the solution found by A is inexact.

If we treat inexact solutions as having cost 1 and exact solutions as having cost 0, the expected cost of A is $> \frac{1}{100}$. A can be assumed to be the best deterministic algorithm, so by Yao's minimax principle, the expected cost of any randomised algorithm is also at least $\frac{1}{100}$, for a worst-case input.

Finally, G is not self-anonymous, but we can invoke Theorem 7.2, which reduces the problem of computing equilibria of anonymous games, to the

computation of equilibria of self-anonymous games. □

5.1. A game whose solution must have irrational numbers

Daskalakis and Papadimitriou [20] note as an open problem, the question of whether there is a two-strategy anonymous game whose Nash equilibria require players to mix with irrational probabilities. The following example shows that such a game does indeed exist, even with just 3 players. In the context of the present paper, it is a further motivation for our focus on approximate rather than exact Nash equilibria.

Example 5.2. Consider the following anonymous game represented in normal form in Figure 3 and in its anonymous form in Figure 4. We show that in the unique equilibrium, the row, the column, and the matrix players must randomize respectively with probabilities

$$p_r = \frac{1}{12}(\sqrt{241} - 7), \quad p_c = \frac{1}{16}(\sqrt{241} - 7), \quad p_m = \frac{1}{36}(23 - \sqrt{241}).$$

		1	2			1	2
1	(1, 0, 1)	(1, $\frac{1}{2}$, 0)		1	(1, 0, 0)	(0, $\frac{1}{4}$, $\frac{1}{2}$)	
2	(0, 0, 0)	($\frac{1}{2}$, $\frac{1}{4}$, 0)		2	($\frac{1}{2}$, 1, $\frac{1}{2}$)	(1, 0, 1)	
		1				2	

Figure 3: The three-player two-strategy anonymous game in normal form. A payoff tuple (a, b, c) represents the row, the column, and the matrix players' payoff, respectively.

	x	0	1	2		x	0	1	2		x	0	1	2
$u_1^r(x)$	0	1	1		$u_1^c(x)$	1	0	0		$u_1^m(x)$	0	0	1	
$u_2^r(x)$	1	$\frac{1}{2}$	0		$u_2^c(x)$	0	$\frac{1}{4}$	$\frac{1}{2}$		$u_2^m(x)$	1	$\frac{1}{2}$	0	

(a) r 's payoff table (b) c 's payoff table (c) m 's payoff table

Figure 4: The three-player two-strategy anonymous game represented in the anonymous compact form, where x denotes the number of other players playing strategy 1.

Theorem 5.2. *There exists a three-player, two-strategy anonymous game that has a unique Nash equilibrium where all the players must randomize with irrational probabilities.*

Proof. We use the game in Example 5.2, which is represented in anonymous form in Figure 4. It is easy to check that the game admits no pure Nash equilibrium. Let r, c, m denote the row, column, and matrix player, respectively. Further, let p_i denote the amount of probability that $i \in \{r, c, m\}$ allocates to strategy 1. Suppose, for the moment, that the game admits only fully-mixed equilibria. Then, these can be found by solving the following system of equations, which results from making everyone indifferent.

$$\begin{cases} \frac{1}{2} \cdot (p_c \cdot (1 - p_m) + p_m \cdot (1 - p_c)) + p_c \cdot p_m = (1 - p_m) \cdot (1 - p_c) \\ (1 - p_r) \cdot (1 - p_m) = \frac{1}{4} \cdot (p_r \cdot (1 - p_m) + p_m \cdot (1 - p_r)) + \frac{1}{2} \cdot p_r \cdot p_m \\ p_r \cdot p_c = (1 - p_r) \cdot (1 - p_c) + \frac{1}{2} \cdot (p_r \cdot (1 - p_c) + p_c \cdot (1 - p_r)), \end{cases}$$

which reduces to

$$\begin{cases} \frac{3}{2} \cdot (p_c + p_m) - p_c \cdot p_m = 1 \\ \frac{5}{4} \cdot (p_r + p_m) - p_r \cdot p_m = 1 \\ \frac{1}{2} \cdot (p_r + p_c) + p_r \cdot p_c = 1, \end{cases}$$

and whose unique solution in the interval $[0, 1]$ is

$$p_r = \frac{1}{12}(\sqrt{241} - 7) \approx 0.71, \quad p_c = \frac{1}{16}(\sqrt{241} - 7) \approx 0.53, \\ p_m = \frac{1}{36}(23 - \sqrt{241}) \approx 0.21.$$

Now we show that everybody must indeed randomize in order to be in equilibrium. Suppose we fix player r to play 1. Given this, it is easy to see that both c and m must play 2, making r unhappy and willing to move to strategy 2. If we fix r to play 2, then m plays 2 and c plays 1. However, r would be better off deviating to strategy 1. Similar arguments can prove that we cannot fix c to any of the two strategies nor m to play 1. The most interesting case is when we fix m to play 2. Given this, we must set $p_r = \frac{4}{5}$ and $p_c = \frac{2}{3}$ in order for r and c to be in equilibrium between each other. Player m 's expected payoff for playing 1 is $\frac{2}{3} \cdot \frac{4}{5} = \frac{8}{15}$, which is larger than what she gets for playing 2, i.e., $(\frac{1}{3} \cdot \frac{1}{5}) + \frac{1}{2} \cdot (\frac{4}{15} + \frac{2}{15}) = \frac{4}{15}$. Hence, m cannot play pure 2, and the game has a unique fully-mixed Nash equilibrium where all players must randomize with irrational probabilities. \square

6. Pure Equilibria of Symmetric Games

We turn our attention to subclasses of anonymous games studied in [8], with an interest in positive results to contrast with the negative ones of Section 5. We present query-efficient algorithms to find *pure* Nash equilibria (PNE) of 2-strategy symmetric anonymous games, also k -strategy self-symmetric games. (The existence of PNE of 2-strategy symmetric anonymous games is pointed out in [8]; see the discussion following Theorem 2, see also [12]. As noted in [8], self-symmetric games are a special case of common-payoff, or pure-coordination games, and so have pure Nash equilibria.) Since in symmetric games every player shares the same utility function, the following results are presented in terms of single-payoff queries: a typical query identifies $u_j(x)$, the payoff to any player who plays $j \in \{1, 2\}$ while x other players play 1.

Proposition 6.1. *A pure Nash equilibrium of any 2-strategy n -player symmetric game can be found with $O(\log n)$ single-payoff queries.*

Proof. Algorithm 3 uses a binary search approach to find the pure Nash equilibrium (PNE) promised by the existence argument of [8, 12] mentioned at the start of this Section 6.

To see that a PNE exists, suppose that x players play 1 in some pure profile. Label x blue if $u_1(x) > u_2(x)$, and green otherwise. If 0 is green, this corresponds to an equilibrium where all players play 2, and if $n - 1$ is blue, we have an equilibrium where all players play 1. In each iteration we start with an interval whose left end is blue and whose right end is green. We query the middle of the interval; if it is blue we proceed with the right half of the interval, otherwise the left half. After $O(\log n)$ iterations we end up with an interval $\{x, x + 1\}$ with $u_i(x) > u_2(x)$ and $u_2(x + 1) \geq u_1(x + 1)$, therefore $x + 1$ is an equilibrium. \square

Moving to k -strategy self-symmetric games, this is a special case of pure coordination games where every pure-strategy profile yields the same utility to all players [8]. The utility function of such a game is of the form $u : \Pi_n^k \rightarrow [0, 1]$, denoting the utility that *all* players get for strategy profile Π_n^k . Such a game possesses a pure Nash equilibrium corresponding to a local maximum of u . The following results identify the deterministic and randomized payoff query complexity of finding a PNE in k -strategy self-symmetric games, for any constant k . This is done by relating it to the problem of finding a local

Algorithm 3: SymmetricPNE

Data: Query access to n -player game: $u_j(x)$ is a player's utility for playing $j \in \{1, 2\}$ when x other players play 1.

Result: The number of players m playing strategy 1 in a PNE.

begin

 | **return** search($0, n - 1$)

end

Procedure search(α, β)

 | $m := \lfloor \frac{\alpha + \beta}{2} \rfloor$

 | **if** $m = \alpha \vee m = \beta$ **then**

 | **return** m

 | **end**

 | Use queries to identify: $u_1(m - 1), u_2(m - 1), u_1(m), u_2(m)$

 | **if** $u_1(m - 1) \geq u_2(m - 1)$ and $u_1(m) \leq u_2(m)$ **then**

 | **return** m

 | **end**

 | **if** $u_1(m - 1) < u_2(m - 1)$ **then**

 | $\beta := m$

 | **else**

 | $\alpha := m$

 | **end**

 | **return** search(α, β)

optimum of a real-valued function f on a regular grid graph (Lemma 6.1), about which much is known, and then applying those pre-existing results. Of interest to us is the d -dimensional regular grid graph; $[n]^d$ denotes the graph on $[n] \times \dots \times [n]$, where edges are only present between two vertices that differ by 1 in a single coordinate, and other coordinates are equal.

Lemma 6.1. *For any constant k , the query complexity (deterministic or randomised) of searching for a pure Nash equilibrium of k -strategy n -player self-symmetric games, is within a constant factor of the query complexity of searching for a local optimum of the grid graph $[n]^{k-1}$.*

Proof. Given a self-symmetric game G , let $\mathcal{G}(G)$ be the graph of pure-strategy profiles of G , whose edges are single-player deviations. We seek a vertex of $\mathcal{G}(G)$ which forms a local optimum of u .

We start by reducing the problem of searching for an optimum of a function f of the vertices of $[n]^{k-1}$ to the search for a PNE of a k -strategy self-symmetric game G . First, extend f to the domain $(\{0\} \cup [n])^{k-1}$ in such a way that no local optimum of f occurs at a point with a zero coordinate. Assume that the codomain of f is $[0, 1]$. Construct G with kn players and k strategies as follows. Let $s = (x_1, \dots, x_k)$ be a typical anonymous pure profile of G , where x_j denotes the number of players who play strategy j . If $0 \leq x_j \leq n$ for each $j < k$, set $u(s)$ equal to $f(p)$, where p is the corresponding point in $[n]^{k-1}$. For the remaining profiles (the red area of Figure 5, depicting the case $k = 2$), we assign them negative utilities in such a way that none of them are local optima of u . Consequently, any PNE of G corresponds to a local optimum of f , so the query complexity of G is at least the query complexity of finding a local optimum of f . Note that the description size of G is larger than that of f by just a linear factor (the value of the k -th entry of a pure profile of G is determined by the previous $k - 1$ entries).

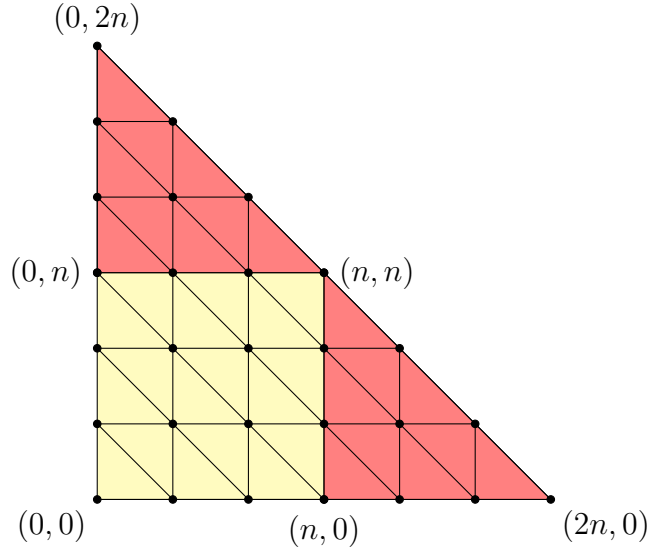


Figure 5: The graph when $k = 3$. The yellow area shows the embedded grid. The irrelevant points for the local maxima computation are in red.

Next we reduce the query-based search for a PNE of an n -player k -strategy G to the problem of locally optimizing a function f on a k -dimensional grid graph. We use the graph $[n]^{k-1}$ where the first $k - 1$ coordinates of a profile in G are mapped to the corresponding vertex of $[n]^{k-1}$, and f maps

the remaining vertices of $[n]^{k-1}$ to negative values in such a way that none of them are solutions. Reducing in this direction, we have to deal with non-grid edges of $\mathcal{G}(G)$ whose presence may cause some profile not to be a PNE (the diagonal edges in Figure 5). Such edges correspond to a player switching from a strategy in $\{1, \dots, k-1\}$ to another strategy in $\{1, \dots, k-1\}$ (as opposed to switching to/from strategy k , which correspond to the grid edges). We deal with this a careful designing of the function f in terms of u . Each vertex v of $\mathcal{G}(G)$ belongs to a set of triangles $\{v, v', v''\}$ for which $\{v', v''\}$ is a non-grid edge. To begin with, set f equal to u everywhere. Then for every vertex v , and every such triangle $\{v, v', v''\}$, if $u(v) < u(v')$ and $u(v) < u(v'')$, $f(v)$ is re-set equal to $(u(v') + u(v''))/2$, and if v belongs to multiple such triangles, the largest such value is used.

This construction of f from u gets rid of local optima that may arise from deleting the diagonal edges (for the purpose of embedding in $[n]^{k-1}$). Any remaining local optima of f correspond to local optima of u that are still local optimal of u with the diagonal edges included, hence PNE. For constant k , a query to f can be computed by a constant number of queries to u , so the query complexities (as functions of n) are linearly related. \square

Corollary 6.1. *The randomized query complexity of searching for PNE of self-symmetric games is $\Theta(n^{(k-1)/2})$ for constant $k \geq 5$.*

Proof. We use Lemma 6.1 and a result of Zhang [35] that the randomized query complexity of optimizing a function on the grid $[n]^d$ is $\Theta(n^{d/2})$ for $d \geq 4$. The statement follows from the fact that we have $d = k - 1$. \square

Corollary 6.2. *The deterministic query complexity of searching for PNE of self-symmetric games is $\Theta(n^{k-2})$ for constant $k > 2$.*

Proof. We use Lemma 6.1 and Theorem 3 of Althöfer and Klaus-Uwe [1], which identifies upper and lower bounds for the query complexity of deterministically optimizing a function on the grid $[n]^d$ that are both proportional to n^{d-1} . Again, the statement follows from having $d = k - 1$. (We note that in [1], the roles of n and k are reversed relative to here.) \square

7. Self-anonymous Games

We show that in self-anonymous games, if every player uses the uniform distribution over her actions, this mixed profile constitutes a $O(1/\sqrt{n})$ -WSNE. This means that no query is needed to find any such approximation.

We provide an inductive proof on the constant number of actions k . First, we demonstrate that this holds for two-strategy games and subsequently utilize this result as the base case of the induction. We prove the following lemma below although, in fact, it also follows from the combination of Lemma 4.1 and Lemma 4.5 with $\epsilon = 1/2$.

Lemma 7.1. *In any two-strategy n -player self-anonymous game, the mixed-strategy profile $s = (\frac{1}{2}, \dots, \frac{1}{2})$ is an $O(1/\sqrt{n})$ -WSNE.*

Proof. We will show that for any player $i \in [n]$, we have

$$\left| \mathbb{E}[u_1^i(X_{-i})] - \mathbb{E}[u_2^i(X_{-i})] \right| \leq \frac{e}{\pi} \cdot \frac{1}{\sqrt{n-1}}. \quad (2)$$

To prove (2), we analyse the expected value to player 1 of strategy 1 minus that of strategy 2 as follows:

$$\begin{aligned} & \sum_{x=0}^{n-1} (u_1^i(x) - u_2^i(x)) \cdot \Pr[X_{-i} = x] \\ = & \sum_{x=0}^{n-2} (u_2^i(x+1) - u_2^i(x)) \cdot \Pr[X_{-i} = x] + (u_1^i(n-1) - u_2^i(n-1)) \cdot \Pr[X_{-i} = n-1] \\ = & \sum_{x=0}^{n-2} (u_2^i(x+1) - u_2^i(x)) \cdot \binom{n-1}{x} \cdot \frac{1}{2^{n-1}} + (u_1^i(n-1) - u_2^i(n-1)) \cdot \frac{1}{2^{n-1}} \\ = & \frac{1}{2^{n-1}} \left(\sum_{x=1}^{n-1} u_2^i(x) \cdot \left(\binom{n-1}{x-1} - \binom{n-1}{x} \right) + (u_1^i(n-1) - u_2^i(0)) \right), \end{aligned}$$

where in the second step we applied the self-anonymity property, in the third step we used the definition of the p.m.f. of the binomial distribution, and in the fourth one we simply rearranged terms. Since the utility function outputs values in $[0, 1]$, then $u_1^i(n-1) - u_2^i(0) \leq 1$. Moreover, for all $x = 1, \dots, \frac{n-1}{2}$, we have $\binom{n-1}{x-1} - \binom{n-1}{x} < 0$, and strictly positive for the remaining values. Thus, in the worst case we have

$$u_2^i(x) = \begin{cases} 0 & \text{if } x \in \{1, \dots, \frac{n-1}{2}\} \\ 1 & \text{if } x \in \{\frac{n-1}{2} + 1, \dots, n-1\}, \end{cases}$$

reducing the above expression to be at most

$$\begin{aligned}
& \frac{1}{2^{n-1}} \left(\sum_{x=\frac{n-1}{2}+1}^{n-1} \left(\binom{n-1}{x-1} - \binom{n-1}{x} \right) + 1 \right) \\
&= \frac{1}{2^{n-1}} \left(\sum_{x=\frac{n-1}{2}+1}^{n-1} \binom{n-1}{x-1} - \sum_{x=\frac{n-1}{2}+1}^{n-2} \binom{n-1}{x} \right) \\
&= \frac{1}{2^{n-1}} \left(\sum_{x=\frac{n-1}{2}}^{n-2} \binom{n-1}{x} - \sum_{x=\frac{n-1}{2}+1}^{n-2} \binom{n-1}{x} \right) = \frac{1}{2^{n-1}} \binom{n-1}{\frac{n-1}{2}} = \frac{1}{2^{n-1}} \cdot \frac{(n-1)!}{\left(\left(\frac{n-1}{2}\right)!\right)^2}.
\end{aligned}$$

By Stirling's bounds, we know that $\sqrt{2\pi} \cdot n^{n+1/2} \cdot e^{-n} \leq n! \leq e \cdot n^{n+1/2} \cdot e^{-n}$. Hence,

$$\begin{aligned}
\frac{1}{2^{n-1}} \cdot \frac{(n-1)!}{\left(\left(\frac{n-1}{2}\right)!\right)^2} &\leq \frac{1}{2^{n-1}} \cdot \frac{e \cdot (n-1)^{n-1+\frac{1}{2}} \cdot e^{-(n-1)}}{\left(\sqrt{2\pi} \cdot \left(\frac{n-1}{2}\right)^{\frac{n-1+1}{2}} \cdot e^{-\left(\frac{n-1}{2}\right)}\right)^2} \\
&= \frac{1}{2^{n-1}} \cdot \frac{e \cdot (n-1)^{n-\frac{1}{2}} \cdot e^{-(n-1)}}{2\pi \cdot \left(\frac{n-1}{2}\right)^n \cdot e^{-(n-1)}} = \frac{e}{\pi} \cdot \frac{1}{\sqrt{n-1}}.
\end{aligned}$$

This gives the required upper bound on the value of strategy 1 minus that of strategy 2. The value of strategy 2 minus that of strategy 1 has the same analysis and upper bound, which gives us (2). \square

Theorem 7.1. *For constant k , in any k -strategy n -player self-anonymous game letting every player randomize uniformly is a $O(1/\sqrt{n})$ -WSNE.*

Proof. We proceed by induction on the number of strategies k where the base case $k = 2$ follows by Lemma 7.1. Suppose that it holds that in any $(k-1)$ -strategy n -player self-anonymous game, every player $i \in [n]$ mixing uniformly is a $O(1/\sqrt{n})$ -WSNE. We show that this holds also for k strategies.

Let G_k be a k -strategy self-anonymous game. Moreover, let $X_i^{(\ell)}$ be a random variable indicating whether player i plays strategy ℓ . Let $X_{-i}^{(k)} := \sum_{j \neq i} X_j^{(k)}$ denote the number of players other than i playing strategy k in G_k . We observe that $\mathbb{E}[X_{-i}^{(k)}] = \frac{n-1}{k}$, so by Chernoff bounds, we have that

$$\Pr \left[X_{-i}^{(k)} \geq \frac{2}{k} (n-1) \right] \leq e^{-\frac{n-1}{3k^2}}, \quad (3)$$

thus, exponentially small in n for constant k . We bound the difference in player i 's utility between two strategies, say, 1 and 2, which is

$$\sum_{\substack{x_1, \dots, x_k: \\ x_1 + \dots + x_k = n}} (u_1^i(x_1, \dots, x_k) - u_2^i(x_1, \dots, x_k)) \cdot \Pr \left[X_{-i}^{(1)} = x_1, \dots, X_{-i}^{(k)} = x_k \right],$$

where the utility function u_j^i takes as input the number x_m of players playing strategy m , for all $m = 1, \dots, k$.

Next we decompose this into the contributions to the expected difference between i 's utility for 1 and 2, arising from the event that x_k players play strategy k :

$$\sum_{x_k=0}^n \Pr \left[X_{-i}^{(k)} = x_k \right] \sum_{\substack{x_1, \dots, x_{k-1}: \\ x_1 + \dots + x_{k-1} = n - x_k}} (u_1^i(x_2, \dots, x_k) - u_2^i(x_2, \dots, x_k)) \cdot \Pr \left[X_{-i}^{(2)} = x_2, \dots, X_{-i}^{(k-1)} = x_{k-1} \mid X_{-i}^{(k)} = x_k \right].$$

Observe that when we sum over x_2, \dots, x_{k-1} but not x_k , we are dealing with an $(n - x_k)$ -player $(k - 1)$ -strategy game G_{k-1} where all $(n - x_k)$ players are still randomizing uniformly among the $k - 1$ strategies. To see this, we could think of fixing the identities of the x_k players playing strategy k to be $\{n - x_k + 1, \dots, n\}$ but G_{k-1} is anonymous, i.e., invariant under permutations of the players. By induction hypothesis, we can bound the difference in payoff in G_{k-1} by $O\left(\frac{1}{\sqrt{n - x_k}}\right)$. We can, therefore, write the difference in utilities between 1 and 2 as

$$\begin{aligned} & \sum_{x_k=0}^{n-1} \Pr \left[X_{-i}^{(k)} = x_k \right] \cdot O\left(\frac{1}{\sqrt{n - x_k}}\right) \\ & \leq \sum_{x_k=0}^{\frac{2}{k}(n-1)} \Pr \left[X_{-i}^{(k)} = x_k \right] \cdot O\left(\frac{1}{\sqrt{n - 2/k}}\right) + \sum_{x_k=\frac{2}{k}(n-1)+1}^{n-1} e^{-\frac{n-1}{3k^2}} \cdot 1 \\ & \leq O\left(\sqrt{\frac{k}{kn - 2}}\right) + \frac{k-2}{k} n \cdot e^{-\frac{n-1}{3k^2}} = O\left(\frac{1}{\sqrt{n}}\right). \end{aligned}$$

where the first inequality used Equation (3). We should perhaps note that in summing over big-oh expressions, the hidden constant in the notation is

the same for all terms in the summation (namely, one that applies for $k - 1$ strategy games, which exists by inductive hypothesis). As long as k is a constant, we have the above claimed upper bound. In addition, notice that if k is not a constant, the result no longer holds since the proof adds a constant factor of $k!$ into the big-oh notation. \square

Corollary 7.1. *For any constant k , no payoff queries are needed to find a $O(1/\sqrt{n})$ -approximate equilibrium in k -strategy n -player self-anonymous games.*

Theorem 7.2 extends an idea of Lemma 1 of [8], which reduces the computation of an exact equilibrium of a two-strategy anonymous game to a two-strategy self-anonymous game. Theorem 7.2 goes further, showing that to some extent approximations can be preserved. We apply it to prove Corollary 7.2, that the problem of finding an FPTAS in two-strategy anonymous games reduces to finding an FPTAS in two-strategy self-anonymous games. Below, we also discuss why Theorem 7.2 does not extend straightforwardly to more than two strategies.

Theorem 7.2. *Let $G := (n, 2, \{u_j^i\}_{i \in [n], j \in [2]})$ be a two-strategy anonymous game having payoffs in $[0, 1]$, and let $\bar{G} := (n, 2, \{\bar{u}_j^i\}_{i \in [n], j \in [2]})$ be the following self-anonymous game.*

- $\bar{u}_2^i(0)$ is set equal to $\frac{1}{2}$,
- for all $x \in \{0, \dots, n - 1\}$, $\bar{u}_1^i(x)$ is set equal to $\bar{u}_2^i(x) + \frac{u_1^i(x) - u_2^i(x)}{2^n}$,
- for all $x \in \{0, \dots, n - 1\}$, $\bar{u}_2^i(x + 1)$ is set equal to $\bar{u}_1^i(x)$.

(Self-anonymity of \bar{G} follows from $\bar{u}_2^i(x + 1) = \bar{u}_1^i(x)$ for all x .)

Then, a mixed strategy profile $s = (p_1, \dots, p_n)$ is an ϵ -approximate Nash equilibrium in G if and only if it is an $\frac{\epsilon}{2^n}$ -approximate Nash equilibrium in \bar{G} .

Proof. First of all, it is straightforward to check that the construction of \bar{G} ensures all payoffs lie in $[0, 1]$. Let X_j denote a Bernoulli random variable indicating whether player j plays strategy 1, whose expectation is $\mathbb{E}[X_j] := p_j$ for all $j \in [n]$, and let $X_{-i} := \sum_{j \neq i} X_j$.

\implies : We now show that if a strategy profile $s = (p_1, \dots, p_n)$ is an ϵ -approximate Nash equilibrium in G , then it is an $\frac{\epsilon}{2^n}$ -approximate Nash equilibrium in \bar{G} . Assume for contradiction that s is not an $\frac{\epsilon}{2^n}$ -approximate Nash

equilibrium in \bar{G} . Hence, there must be some player $i \in [n]$ that gains strictly more than $\frac{\epsilon}{2n}$ by deviating to either strategy 1 or 2, i.e.,

$$\exists i \in [n] : p_i \cdot \mathbb{E}[\bar{u}_1^i(X_{-i})] + (1 - p_i) \cdot \mathbb{E}[\bar{u}_2^i(X_{-i})] + \frac{\epsilon}{2n} < \mathbb{E}[\bar{u}_\ell^i(X_{-i})],$$

for some $\ell \in [2]$. We make a case distinction on ℓ .

Case $\ell = 1$: We have that

$$(1 - p_i) \cdot \mathbb{E}[\bar{u}_2^i(X_{-i})] + \frac{\epsilon}{2n} < (1 - p_i) \cdot \mathbb{E}[\bar{u}_1^i(X_{-i})],$$

which by definition is equivalent to

$$(1 - p_i) \cdot \sum_{x=0}^{n-1} \bar{u}_2^i(x) \cdot \Pr[X_{-i} = x] + \frac{\epsilon}{2n} < (1 - p_i) \cdot \sum_{x=0}^{n-1} \bar{u}_1^i(x) \cdot \Pr[X_{-i} = x].$$

If we apply our payoff transformation rule on the RHS, we have

$$\begin{aligned} & (1 - p_i) \cdot \sum_{x=0}^{n-1} \bar{u}_2^i(x) \cdot \Pr[X_{-i} = x] + \frac{\epsilon}{2n} < \\ & (1 - p_i) \cdot \sum_{x=0}^{n-1} \left(\bar{u}_2^i(x) + \frac{u_1^i(x) - u_2^i(x)}{2n} \right) \cdot \Pr[X_{-i} = x], \end{aligned}$$

which reduces to

$$(1 - p_i) \cdot \sum_{x=0}^{n-1} u_2^i(x) \cdot \Pr[X_{-i} = x] + \epsilon < (1 - p_i) \cdot \sum_{x=0}^{n-1} u_1^i(x) \cdot \Pr[X_{-i} = x].$$

Applying the definition of expected utility we have

$$(1 - p_i) \cdot \mathbb{E}[u_2^i(X_{-i})] + \epsilon < (1 - p_i) \cdot \mathbb{E}[u_1^i(X_{-i})].$$

Therefore, we contradict the fact that s is an ϵ -equilibrium for G since i prefers deviating to strategy 1 also in G .

Case $\ell = 2$: We have that

$$p_i \cdot \mathbb{E}[\bar{u}_1^i(X_{-i})] + \frac{\epsilon}{2n} < p_i \cdot \mathbb{E}[\bar{u}_2^i(X_{-i})],$$

and, by applying the same arguments, we have

$$p_i \cdot \mathbb{E}[u_1^i(X_{-i})] + \epsilon < p_i \cdot \mathbb{E}[u_2^i(X_{-i})].$$

Therefore, we contradict the fact that s is an ϵ -equilibrium for G since i prefers deviating to strategy 2 also in G .

\Leftarrow : We now show that the converse is also true. The proof is very similar to the previous one. We assume for contradiction that s is not an ϵ -equilibrium in G . Then,

$$\exists i \in [n] : p_i \cdot \mathbb{E}[u_1^i(X_{-i})] + (1 - p_i) \cdot \mathbb{E}[u_2^i(X_{-i})] + \epsilon < \mathbb{E}[u_\ell^i(X_{-i})],$$

for some $\ell \in [2]$. We make, again, a case distinction on ℓ .

Case $\ell = 1$: We have that

$$(1 - p_i) \cdot \mathbb{E}[u_2^i(X_{-i})] + \epsilon < (1 - p_i) \cdot \mathbb{E}[u_1^i(X_{-i})],$$

which by definition is equivalent to

$$(1 - p_i) \cdot \sum_{x=0}^{n-1} (u_1^i(x) - u_2^i(x)) \cdot \Pr[X_{-i} = x] > \epsilon.$$

If we apply our transformation $u_1^i(x) - u_2^i(x) = 2n \cdot (\bar{u}_1^i(x) - \bar{u}_2^i(x))$, we have that

$$(1 - p_i) \cdot \sum_{x=0}^{n-1} (\bar{u}_1^i(x) - \bar{u}_2^i(x)) \cdot \Pr[X_{-i} = x] > \frac{\epsilon}{2n},$$

which means

$$(1 - p_i) \cdot \mathbb{E}[\bar{u}_1^i(X_{-i})] + \frac{\epsilon}{2n} < (1 - p_i) \cdot \mathbb{E}[\bar{u}_2^i(X_{-i})],$$

contradicting the assumption that s is an $\frac{\epsilon}{2n}$ -equilibrium in \bar{G} .

Case $\ell = 2$: Similarly, we have that

$$p_i \cdot \mathbb{E}[u_1^i(X_{-i})] + \epsilon < p_i \cdot \mathbb{E}[u_2^i(X_{-i})],$$

which, as in the above case, reduces to having

$$p_i \cdot \mathbb{E}[\bar{u}_1^i(X_{-i})] + \frac{\epsilon}{2n} < p_i \cdot \mathbb{E}[\bar{u}_2^i(X_{-i})],$$

contradicting the assumption that s is an $\frac{\epsilon}{2n}$ -equilibrium in \bar{G} .

□

Corollary 7.2. *If there is an FPTAS for computing ϵ -Nash equilibria of two-strategy self-anonymous games, then there is also an FPTAS for computing ϵ -Nash equilibria of two-strategy anonymous games.*

Proof. Pick any two-strategy anonymous game G , construct \bar{G} according to Theorem 7.2 (noting that the construction of \bar{G} is computable in polynomial time) and find a δ -equilibrium for $\delta = \frac{\epsilon}{2n}$. Since there is an FPTAS, we have a running time that is $\text{poly}(n, 1/\delta)$. For $\delta = \frac{\epsilon}{2n}$, the running time clearly is $\text{poly}(n, 1/\epsilon)$. Theorem 7.2 implies that a δ -equilibrium in \bar{G} is an ϵ -equilibrium in G . \square

We end this section with a brief discussion of why Theorem 7.2 (and hence Corollary 7.2) does not extend straightforwardly to more than two strategies. In the context of 3-strategy games, let $u_j^i(x, y, z)$ denote player i 's utility when amongst the $n - 1$ remaining players, x play 1, y play 2, and z play 3. If there were, say, 16 players, note that to be self-anonymous, the corresponding game \bar{G} would need to satisfy

$$\begin{aligned}\bar{u}_1^i(4, 6, 5) &= \bar{u}_2^i(5, 5, 5) = \bar{u}_3^i(5, 6, 4) \\ \bar{u}_1^i(5, 5, 5) &= \bar{u}_2^i(6, 4, 5) = \bar{u}_3^i(6, 5, 4) \\ \bar{u}_1^i(5, 6, 4) &= \bar{u}_2^i(6, 5, 4) = \bar{u}_3^i(6, 6, 3).\end{aligned}$$

The proof of Theorem 7.2 envisages that $\bar{u}_1^i(5, 6, 4) - \bar{u}_3^i(5, 6, 4)$ should be a rescaled version of $u_1^i(5, 6, 4) - u_3^i(5, 6, 4)$, and similarly for $\bar{u}_1^i(5, 5, 5) - \bar{u}_2^i(5, 5, 5)$ and $\bar{u}_2^i(6, 5, 4) - \bar{u}_3^i(6, 5, 4)$. That is, the pairwise differences between the three values given by the above displayed formulae, are required to be proportionate to pairwise differences between three pairs of payoffs in G , which themselves may take any values in $[0, 1]$. Hence we get a conflict between the constraints that result from a natural approach to generalising the result. It may be that (for $k > 2$) there exists a more complicated reduction from an FPTAS for k -strategy games to an FPTAS for k -strategy self-anonymous games, perhaps by increasing the number of players.

8. Conclusions and Further Work

Our interest in the query complexity of anonymous games has resulted in an algorithm that has an improved runtime-efficiency guarantee, although limited to when the number of strategies k is equal to 2. Algorithm 2 (Theorem 4.1) finds an ϵ -NE faster than the PTAS of [20], for any $\epsilon \geq 1/\sqrt[4]{n}$. In

particular, for $\epsilon = 1/\sqrt[4]{n}$, their algorithm runs in subexponential time, while ours is just $\tilde{O}(n^{3/2})$; however, our ϵ -NE is not well-supported.

There are ways to potentially strengthen the results presented here, an obvious question being whether our main upper bound on query complexity can be quantitatively improved. (We also considered complementary lower bounds, but have not found any that come very close to the upper bound of Theorem 4.1.) Also, the ϵ -NE found by Algorithm 2 are not well-supported since all players are forced to randomize: if, say, a player is always paid 1 for playing strategy 1 and 0 for playing strategy 2, she must still allocate positive probability to strategy 2. This requirement to randomise is also a feature of the more recent algorithm of [13], extending our approach to more than 2 strategies. That leaves us with the question of whether there is a query-efficient algorithm that finds an ϵ -WSNE.

Acknowledgments

The first author was supported by EPSRC under grant EP/K01000X/1. A preliminary version of this paper appeared in the proceedings of the 11th conference on Web and Internet Economics (WINE). We thank the journal referees for their careful reading and improvements to the presentation of this paper. We thank Ilias Diakonikolas and Felix Fischer for advice on related work.

- [1] Althöfer, I., Klaus-Uwe, K., 1993. On the deterministic complexity of searching local maxima. *Discrete Applied Mathematics* 43 (2), 111–113.
- [2] Azrieli, Y., Shmaya, E., 2013. Lipschitz games. *Mathematics of Operations Research* 38 (2), 350–357.
- [3] Babichenko, Y., 2013. Best-reply dynamics in large binary-choice anonymous games. *Games and Economic Behavior* 81 (1), 130–144.
- [4] Babichenko, Y., 2014. Query complexity of approximate Nash equilibria. In: *Proceedings of the 46th Annual ACM Symposium on Theory of Computing. STOC '14*. ACM, USA, pp. 535–544.
- [5] Babichenko, Y., Rubinstein, A., 2016. Communication complexity of approximate Nash equilibria. CoRR arXiv:1608.06580.
- [6] Blonski, M., 1999. Anonymous games with binary actions. *Games and Economic Behaviour* 28 (2), 171–180.

- [7] Blonski, M., 2000. Characterization of pure strategy equilibria in finite anonymous games. *Journal of Mathematical Economics* 34 (2), 225–233.
- [8] Brandt, F., Fischer, F., Holzer, M., 2009. Symmetries and the complexity of pure Nash equilibrium. *Journal of Computer and System Sciences* 75, 163–177.
- [9] Chen, X., Cheng, Y., Tang, B., 2015. Well-supported versus approximate Nash equilibria: Query complexity of large games. CoRR arXiv:1511.00785.
- [10] Chen, X., Deng, X., Teng, S., 2009. Settling the complexity of computing two-player Nash equilibria. *Journal of the ACM* 56 (3), 1–57.
- [11] Chen, X., Durfee, D., Orfanou, A., 2015. On the Complexity of Nash Equilibria in Anonymous Games. In: *Proceedings of the Forty-seventh Annual ACM on Symposium on Theory of Computing. STOC '15*. ACM, pp. 381–390.
- [12] Cheng, S.-F., Reeves, D. M., Vorobeychik, Y., Wellman, M. P., 2004. Notes on equilibria in symmetric games. In: *Proceedings of the 6th International Workshop On Game Theoretic And Decision Theoretic Agents (GTDT)*.
- [13] Cheng, Y., Diakonikolas, I., Stewart, A., 2017. Playing anonymous games using simple strategies. In: *Proceedings of the Twenty-Eighth Annual ACM-SIAM Symposium on Discrete Algorithms. SODA '17*. Society for Industrial and Applied Mathematics, pp. 616–631.
- [14] Daskalakis, C., 2008. An efficient PTAS for two-strategy anonymous games. In: *Papadimitriou, C. H., Zhang, S. (Eds.), Internet and Network Economics. Vol. 5385*. Springer, pp. 186–197.
- [15] Daskalakis, C., De, A., Kamath, G., Tzamos, C., 2016. A size-free CLT for Poisson multinomials and its applications. In: *Proceedings of the Forty-eighth Annual ACM on Symposium on Theory of Computing. ACM*, pp. 1074–1086.
- [16] Daskalakis, C., Goldberg, P. W., Papadimitriou, C. H., 2009. The complexity of computing a Nash equilibrium. *SIAM Journal on Computing* 39 (1), 195–259.

- [17] Daskalakis, C., Papadimitriou, C. H., 2007. Computing equilibria in anonymous games. In: In Proceedings of the Forty-eighth Annual IEEE Symposium on Foundations of Computer Science. FOCS '07. IEEE Computer Society, pp. 83–93.
- [18] Daskalakis, C., Papadimitriou, C. H., 2008. Discretized multinomial distributions and Nash equilibria in anonymous games. In: In Proceedings of the 49th Symposium on Foundations of Computer Science (FOCS). pp. 25–34.
- [19] Daskalakis, C., Papadimitriou, C. H., 2009. On oblivious PTAS's for Nash equilibrium. In: Proceedings of the 41st Annual ACM Symposium on Theory of Computing. STOC '09. ACM, USA, pp. 75–84.
- [20] Daskalakis, C., Papadimitriou, C. H., 2015. Approximate Nash equilibria in anonymous games. *Journal of Economic Theory* 156, 207–245.
- [21] Daskalakis, C., Papadimitriou, C. H., 2015. Sparse covers for sums of indicators. *Probability Theory and Related Fields* 162 (3), 679–705.
- [22] Diakonikolas, I., Kane, D. M., Stewart, A., 2015. Optimal learning via the Fourier transform for sums of independent integer random variables. CoRR arXiv:1505.00662.
- [23] Diakonikolas, I., Kane, D. M., Stewart, A., 2016. The Fourier transform of Poisson multinomial distributions and its algorithmic applications. In: Proceedings of the Forty-eighth Annual ACM Symposium on Theory of Computing. STOC '16. ACM, pp. 1060–1073.
- [24] Etessami, K., Yannakakis, M., 2010. On the complexity of Nash equilibria and other fixed points. *SIAM Journal on Computing* 39 (6), 2531–2597.
- [25] Fearnley, J., Gairing, M., Goldberg, P. W., Savani, R., 2015. Learning equilibria of games via payoff queries. *Journal of Machine Learning Research* 16, 1305–1344.
- [26] Fearnley, J., Savani, R., 2014. Finding approximate Nash equilibria of bimatrix games via payoff queries. In: Proceedings of the Fifteenth ACM Conference on Economics and Computation. EC '14. ACM, USA, pp. 657–674.

- [27] Goldberg, P. W., Marmolejo Cossío, F. J., Wu, Z. S., 2016. Logarithmic query complexity for approximate Nash computation in large games. In: Gairing, M., Savani, R. (Eds.), Proceedings of the 9th International Symposium on Algorithmic Game Theory. SAGT '16. Springer Berlin Heidelberg, pp. 3–14.
- [28] Goldberg, P. W., Roth, A., 2014. Bounds for the query complexity of approximate equilibria. In: Proceedings of the Fifteenth ACM Conference on Economics and Computation. EC '14. ACM, USA, pp. 639–656.
- [29] Hart, S., Nisan, N., 2013. The query complexity of correlated equilibria. In: Algorithmic Game Theory, 6th International Symposium, SAGT 2013. Springer, p. 268.
- [30] Hong, Y., 2013. On computing the distribution function for the Poisson binomial distribution. Computational Statistics & Data Analysis 59 (1), 41–51.
- [31] Kalai, E., 2004. Large robust games. Econometrica 72 (6), 1631–1665.
- [32] Kash, I. A., Friedman, E. J., Halpern, J. Y., 2009. Multiagent learning in large anonymous games. In: Eighth International Conference on Autonomous Agents and Multiagent Systems (AAMAS). pp. 765–772.
- [33] Lipton, R. J., Markakis, E., Mehta, A., 2003. Playing large games using simple strategies. In: Proceedings of the 4th ACM Conference on Electronic Commerce. EC '03. ACM, USA, pp. 36–41.
- [34] Rubinstein, A., 2016. Settling the complexity of computing approximate two-player Nash equilibria. In: In Proceedings of the 57th Symposium on Foundations of Computer Science (FOCS). pp. 258–265.
- [35] Zhang, S., 2009. Tight bounds for randomized and quantum local search. SIAM Journal on Computing 39 (3), 948–977.