



Algorithms for Playing Games with Limited Randomness

Shankar Kalyanaraman* Christopher Umans†

July 3, 2007

Abstract

We study multiplayer games in which the participants have access to only limited randomness. This constrains both the algorithms used to compute equilibria (they should use little or no randomness) as well as the mixed strategies that the participants are capable of playing (these should be *sparse*). We frame algorithmic questions that naturally arise in this setting, and resolve several of them.

We give *deterministic* algorithms that can be used to find sparse ϵ -equilibria in zero-sum and non-zero-sum games, and a *randomness-efficient* method for playing repeated zero-sum games. These results apply ideas from derandomization (expander walks, and δ -independent sample spaces) to the algorithms of Lipton, Markakis, and Mehta (EC'03), and the online algorithm of Freund and Schapire (GEB '99).

Subsequently, we consider a large class of games in which sparse equilibria are known to exist (and are therefore amenable to randomness-limited players), namely games of small rank. We give the first “fixed-parameter” algorithms for obtaining approximate equilibria in these games. For rank- k games, we give a deterministic algorithm to find $(k + 1)$ -sparse ϵ -equilibria in time polynomial in the input size n and some function $f(k, 1/\epsilon)$. In a similar setting Kannan and Theobald (SODA '07) gave an algorithm whose run-time is $n^{O(k)}$. Our algorithm works for a slightly different notion of a game’s “rank,” but is fixed parameter tractable in the above sense, and extends easily to the multi-player case.

1 Introduction

Ever since Nash’s work in 1951 [Nas51] and Morgenstern and von Neumann’s treatise [MvN44] that introduced the subject, game theory has been one of the cornerstones of economic analysis. In recent years, game theory has increasingly become embedded into theoretical computer science. In particular, the field of algorithmic game theory has seen prolific contributions to areas such as auctions, mechanism design and congestion games.

Traditionally, designing algorithms for such games has been largely driven by the need to find equilibria *time-efficiently*. In this context, Chen and Deng [CD06] built on prior work [DP05, CD05, DGP06, GP06] and showed that the problem of finding a Nash equilibrium for the two-player game is complete for the PPAD class. This suggests that a polynomial-time algorithm is unlikely to exist. With Teng, Chen and Deng [CDT06] showed further that finding $O(1/n^{\Theta(1)})$ -approximate Nash equilibria is also PPAD-complete.

In this paper, we study algorithms for finding equilibria and playing games *randomness-efficiently*. By “playing a game” we mean the actions a player must take to actually sample and play a strategy from a mixed strategy profile, which necessarily entails randomness. There is an immense body of literature in theoretical computer science devoted to addressing computational issues when limited or no randomness is available, and for the same reasons it makes sense to study games whose players have limited access to randomness. Moreover, as we will see below, limited randomness in games motivates some special classes of games that have arisen in other contexts thereby providing a viewpoint that we believe is helpful.

Limited randomness imposes two major constraints on players. Firstly, players with access to limited randomness are precluded from executing mixed strategies with large support since playing a mixed strategy with

*Computer Science, California Institute of Technology, 1200 E. California Blvd, Pasadena CA 91125. email: shankar@cs.caltech.edu. Supported by NSF Grant CCF-0346991.

†Computer Science, California Institute of Technology, 1200 E. California Blvd, Pasadena CA 91125. email: umans@cs.caltech.edu. Supported by NSF Grant CCF-0346991, BSF Grant 2004329, a Sloan Research Fellowship, and an Okawa Foundation research grant.

support size m requires at least $\log m$ bits of randomness (to sample a single strategy from the distribution). Secondly, they are limited to executing algorithms to help find optimal strategies or equilibria that use a small amount of randomness. We look at both such manifestations of limited randomness.

1.1 Sparse strategies in single-round games

We first look at single-round games. For the case of zero-sum games where both players have n available strategies, Lipton and Young [LY94] showed that a random sample of $O(\frac{\log n}{\epsilon^2})$ strategies was sufficient to approximate the value of the game by ϵ . Lipton, Markakis and Mehta [LMM03] extended this to ϵ -equilibria for two-player nonzero-sum games. Indeed, they gave a randomized procedure that produced small-support strategies for an ϵ -equilibrium when given a Nash equilibrium with possibly large support. In the following theorem, we derandomize this procedure by using random walks on expander graphs:

Theorem 1.1 *Let $G = (R, C, n)$ be a two-player game, and let (p^*, q^*) be a Nash equilibrium for G . For every $\epsilon > 0$, there is a deterministic procedure P running in time $\text{poly}(|G|)^{1/\epsilon^2}$ such that the pair $(P(G, p^*, 1), P(G, q^*, 2))$ is an $O(\frac{\log n}{\epsilon^2})$ -sparse 4ϵ -equilibrium for G .*

This can be viewed as a *deterministic* “sparsification” procedure for ϵ -equilibria in general two player games. In zero-sum games one can find optimal strategies efficiently, and as a result, we obtain a deterministic polynomial time algorithm to find sparse ϵ -equilibria for zero-sum games:

Corollary 1.2 *Let $G = (R, C, n)$ be a two-player zero-sum game. For every $\epsilon > 0$, there is a deterministic procedure running in time $\text{poly}(|G|)^{1/\epsilon^2}$ that outputs a pair (p^*, q^*) that an $O(\frac{\log n}{\epsilon^2})$ -sparse ϵ -equilibrium for G .*

We point out that Freund and Schapire [FS96] obtained a similar result using an adaptive multiplicative-weight algorithm (discussed below) with $\text{poly}(|G|, 1/\epsilon)$ running time. We obtain the corollary above using a very different proof technique that flows from well-known derandomization techniques.

1.2 Reusing randomness in multiple round games

In single-round games a randomness-limited player requires sparse strategies, but in multiple-round games, we would like to be able to “reuse” randomness across rounds. This is an orthogonal concern to that of reducing randomness within a single round.

Freund and Schapire [FS99] proposed an adaptive online algorithm for a T -round two-player zero-sum game with n strategies available to each. Executing the mixed strategies produced by their algorithm uses $\Omega(T \log n)$ bits of randomness over T rounds, in the worst case. By making use of almost-pairwise independence, we show how to reuse randomness across rounds: it is possible to make do with just $O(\log n + \log \log T + \log(1/\epsilon))$ bits and achieve close to the same quality of approximation as in [FS99].

Theorem 1.3 *Let M be the $n \times n$ -payoff matrix for a two-player zero-sum T -round game with entries in $\{0, 1\}$. For any $\epsilon < 1/2$ and constant δ , there exists an online randomized algorithm R using $O(\log n + \log \log T + \log(1/\epsilon))$ random bits with the following property: for any arbitrary sequence Q_1, \dots, Q_T of mixed strategies played (adaptively) by the column player over T rounds, R produces a sequence of strategies S_1, \dots, S_T such that with probability at least $1 - \delta$:*

$$\frac{1}{T} \sum_{i=1}^T M(S_i, Q_i) \leq \frac{1}{T} \min_P \sum_{i=1}^T M(P, Q_i) + O\left(\sqrt{\frac{\log n}{T}} + \epsilon\right)$$

1.3 Games with sparse equilibria

As we have discussed, players with limited access to randomness can only achieve equilibria that are sparse. We saw before that in the general setting, we are able to deterministically “sparsify” if we are willing to settle for ϵ -equilibria. The sparsity cannot in general be less than $\log n$, though, so we are motivated to consider broad classes of games in which even sparser equilibria are guaranteed to exist.

Perhaps the simplest example is a 2-player games in which one player has only k available strategies, while the other player has $n \gg k$ available strategies. The results in the work of Lipton et al. [LMM03] imply there is a Nash equilibrium in this game with support size $k + 1$. This is somewhat unsatisfying – it means that in a two-player game one player may need to choose from less-sparse strategies than his opponent (i.e. requiring slightly *more* randomness) to achieve equilibrium. Theorem 1.4 rectifies this asymmetry by showing that k -sparse strategies suffice for the opposing player.

Theorem 1.4 *Let $G = (R, C, k, n)$ be a two-player game. Given p^* for which there exists a q^* such that (p^*, q^*) is a Nash equilibrium, we can compute in deterministic polynomial time q' for which (p^*, q') is a Nash equilibrium and $|\text{supp}(q')| \leq k$.*

We also give a deterministic polynomial time algorithm to compute such a limited-support strategy for one player, given the k -sparse strategy of the other. We extend this further to the multiplayer case and show that for an ℓ -player game where players 1 through $\ell - 1$ have $k_1, \dots, k_{\ell-1}$ pure strategies, respectively, the ℓ -th player need only play a $(\prod k_i)$ -sparse strategy to achieve equilibrium:

Corollary 1.5 *Let $G = (T_1, T_2, \dots, T_\ell, k_1, k_2, \dots, k_{\ell-1}, n)$ be an ℓ -player game where $\prod_{i=1}^{\ell-1} k_i < n$. Given $G, p_1^*, \dots, p_{\ell-1}^*$ there exists a deterministic polynomial-time procedure to compute \hat{p}_ℓ such that $(p_1^*, p_2^*, \dots, p_{\ell-1}^*, \hat{p}_\ell)$ is a Nash equilibrium for G and $|\text{supp}(\hat{p}_\ell)| \leq k = \prod_{i=1}^{\ell-1} k_i$.*

These bounds are tight.

1.3.1 Games of small rank

Perhaps the most significant technical contribution in this paper pertains to a generalization of the “unbalanced” games that we saw above, namely, games of small rank. This is a broad class of games (encompassing some natural examples – see Section 6.3) for which sparse equilibria are known to exist.

For 2-player games with rank- k payoff matrices, Lipton et al. describe an enumeration algorithm that finds $(k + 1)$ -sparse strategies in $O(n^{k+1})$ time. We improve dramatically on this bound but we must relax the problem in two ways: first, we compute an ϵ -equilibrium rather than an exact one; second we require that the payoff matrices be presented as a low-rank decomposition, with entries up to precision B (limited precision makes sense since we are only considering ϵ -equilibria).

Theorem 1.6 *Let $G = (R, C, n)$ be a two player game such that R and C have rank at most k . Furthermore, let $R = R_1 R_2, C = C_1 C_2$ be a decomposition of R, C with R_1, R_2, C_1, C_2 containing integer entries in $[-B, B]$. Then, for every $\epsilon > 0$, there is a deterministic procedure P running in time $(4B^2 k / \epsilon)^{2k} \text{poly}(|G|)$ that returns a 4ϵ -Nash equilibrium (p, q) with $|\text{supp}(p)|, |\text{supp}(q)| \leq k + 1$.*

To the best of our knowledge, Theorem 1.6 provides the first efficient “fixed-parameter” algorithm to this problem in the sense that the running time is polynomial in the input size n and some function $f(k, 1/\epsilon, B)$. The closest parallel to our result is by Kannan and Theobald [KT07] who consider a somewhat different definition of “rank” for two-player games: in their definition, the *sum* of the payoff matrices is required to have small rank. In that case, they present an algorithm that finds an ϵ -equilibrium in a rank k 2-player game in $O(n^{2k+o(1)} B^2)$ time. Their algorithm relies on results of Vavasis for solving indefinite quadratic programs [Vav92] and does not seem to generalize to $\ell > 2$ players.

Our algorithm is (arguably) simpler, and moreover, it easily generalizes to $\ell > 2$ players, where small rank games still are guaranteed to have sparse equilibria. In the ℓ -player setting, we give an $O(((2B)^\ell k \ell / \epsilon)^{k \ell (\ell - 1)} \text{poly}(n^\ell))$ time deterministic procedure that computes such a sparse ϵ -equilibrium, when the payoff tensors are presented as a rank- k decomposition with entries up to precision B .

All of the algorithms for low-rank games rely on enumerating potential equilibria distributions in a basis dictated by the small rank decomposition. This seems like a technique that may be useful for algorithmic questions regarding low-rank games beyond those we have considered in this paper.

1.4 Outline

The rest of the paper is structured as follows. Section 2 goes over some preliminary definitions and lemmas. Section 3 presents algorithms for finding small-support strategies for single-round games, obtaining approximate Nash equilibria. In Section 4, we look at the multiple-round case and present a randomness-efficient variant of the adaptive online algorithm of Freund and Schapire. Games in which some players have very few strategies are discussed in Section 5 while our new algorithms for ϵ -equilibria in games with small rank are described in Section 6.

2 Preliminaries

Definition 2.1 For a finite strategy set S we define $\Delta(S)$ to be the set of probability distributions over S , i.e., all vectors $p = (p_s)_{s \in S}$ satisfying $\sum_{s \in S} p_s = 1$ with each $p_s \in [0, 1]$. A **mixed strategy** p is an element of $\Delta(S)$. The **support** of a mixed strategy $p \in \Delta(S)$ is the set $\text{supp}(p)$ given by $\text{supp}(p) = \{s \in S | p_s > 0\}$. A mixed strategy p is **k -sparse** if $|\text{supp}(p)| = k$.

In this paper, we will concern ourselves with games that can be specified by payoff matrices (or tensors) whose entries denote the payoff upon playing the corresponding strategy tuple. We will also assume, unless otherwise specified, that these entries are bounded and can be scaled to lie in $[-1, 1]$.

Definition 2.2 An **ℓ -player finite game** G is a tuple $(T_1, \dots, T_\ell, n_1, n_2, \dots, n_\ell)$ where T_i is the $(n_1 \times \dots \times n_\ell)$ ℓ -dimensional payoff tensor with $T_i(s_1, \dots, s_\ell)$ denoting the payoff to player i when the pure strategy ℓ -tuple (s_1, \dots, s_ℓ) is played in the game.

For ease of presentation, in the rest of this paper we will often restrict ourselves to ℓ -player games where $n_1 = n_2 = \dots = n_\ell = n$, which we denote by $G = (T_1, \dots, T_\ell, n)$. We often refer to players by their payoff tensors. For example, for the two-player game $G = (R, C, n)$ we will refer to the row player as R and the column player as C . All vectors are thought of as row vectors.

Definition 2.3 In an ℓ -player game $G = (T_1, T_2, \dots, T_\ell, n_1, n_2, \dots, n_\ell)$, we denote by $T_i(p_1, \dots, p_\ell)$ the payoff to the i -th player when the ℓ players play mixed strategies p_1, \dots, p_ℓ , i.e.,

$$T_i(p_1, \dots, p_\ell) = \sum_{i_1 \in [n_1], \dots, i_\ell \in [n_\ell]} p_{i_1} p_{i_2} \dots p_{i_\ell} T_i(i_1, i_2, \dots, i_\ell).$$

If we substitute some $a \in [n_j]$ for p_j we understand that to mean the distribution that places weight 1 on a and 0 everywhere else.

The following definition is standard:

Definition 2.4 Let $G = (T_1, \dots, T_\ell, n_1, \dots, n_\ell)$ be an ℓ -player. An ℓ -tuple (p_1^*, \dots, p_ℓ^*) with each $p_i^* \in \Delta([n_i])$ is an **ϵ -equilibrium** for G if: for every i and every $p \in \Delta([n_i])$,

$$T_i(p_1^*, \dots, p_{i-1}^*, p, p_{i+1}^*, \dots, p_\ell^*) \leq T_i(p_1^*, \dots, p_{i-1}^*, p_i^*, p_{i+1}^*, \dots, p_\ell^*) + \epsilon.$$

A **Nash equilibrium** is an ϵ -equilibrium for $\epsilon = 0$.

Let G be an ℓ -player game. It is well-known that given the supports of the ℓ different p_i^* in a Nash equilibrium, one can find the actual distributions by linear programming. We will use a similar fact repeatedly:

Lemma 2.5 Let $G = (T_1, T_2, \dots, T_\ell, n)$ be an ℓ -player game, and let $(p_1^*, p_2^*, \dots, p_\ell^*)$ be a Nash equilibrium. Given G and $p_1^*, p_2^*, \dots, p_{\ell-1}^*$ one can find a distribution q in deterministic polynomial time for which $(p_1^*, p_2^*, \dots, p_{\ell-1}^*, q)$ is also a Nash equilibrium.

Proof. Once we know the distributions $p_1^*, \dots, p_{\ell-1}^*$ in order to find a Nash equilibrium strategy q for player T_ℓ , we first determine the Nash equilibrium support for T_ℓ by considering the set

$$T = \{r \mid \forall r', T_\ell(p_1^*, \dots, p_{\ell-1}^*, r) \geq T_\ell(p_1^*, \dots, p_{\ell-1}^*, r')\}$$

We now find q satisfying the following linear program:

$$\begin{aligned} \sum_{i=1}^n q_i &= 1 \\ q_i &\geq 0; & i &= 1, \dots, n \\ q_i &= 0; & i &\notin T \\ T_\ell(p_1^*, \dots, p_{j-1}^*, i, p_{j+1}^*, \dots, p_{\ell-1}^*, q) &\geq T_\ell(p_1^*, \dots, p_{j-1}^*, i', p_{j+1}^*, \dots, p_{\ell-1}^*, q); & i \mid p_{j_i}^* &> 0; \\ & & j &= 1, \dots, \ell - 1; \\ & & i' &= 1, \dots, n \end{aligned}$$

From Nash's result [Nas51], we know that a solution to this linear program exists and this is the desired q . \square

3 Sparsifying Nash equilibria deterministically

In this section we give deterministic algorithms for “sparsifying” Nash equilibria (in the process turning them into ϵ -equilibria). In this way, a player with limited access to randomness, but who has access to an equilibrium mixed strategy, is able to produce a small strategy that can then be played.¹

3.1 Two-players

Lipton et al proved:

Theorem 3.1 (Lipton et al) *Let $G = (R, C, n)$ be a two-player game, and let (p^*, q^*) be a Nash equilibrium for G . There is a polynomial-time randomized procedure P such that with probability at least $1/2$, the pair $(P(G, p^*), P(G, q^*))$ is an $O(\log n/\epsilon^2)$ -sparse ϵ -equilibrium for G .*

The algorithm P is very simple: it amounts to sampling uniformly from the given equilibrium strategy. The analysis applies Chernoff bounds to show that the sampled strategies present the opposing players with approximately (within ϵ) the same weighted row- and column- sums, and hence comprise an ϵ -equilibrium. In our setting, since the players have limited randomness they cannot afford the above sampling (it requires at least as much randomness as simply playing (p^*, q^*)), so we derandomize the algorithm using an expander walk.

Theorem 1.1 (restated). *Let $G = (R, C, n)$ be a two-player game, and let (p^*, q^*) be a Nash equilibrium for G . For every $\epsilon > 0$, there is a deterministic procedure P running in time $\text{poly}(|G|)^{1/\epsilon^2}$ such that the pair $(P(G, p^*, 1), P(G, q^*, 2))$ is an $O(\log n/\epsilon^2)$ -sparse 4ϵ -equilibrium for G .*

Before proving the theorem we will give a convenient characterization of ϵ -equilibrium:

Lemma 3.2 *Let $G = (R, C, n)$ be a 2-player game. Define*

$$\begin{aligned} T_p &= \{i \mid (pC)_i \geq \max_r (pC)_r - \epsilon\} \\ S_q &= \{j \mid (Rq^T)_j \geq \max_t (Rq^T)_t - \epsilon\}. \end{aligned}$$

If $\text{supp}(p) \subseteq S_q$ and $\text{supp}(q) \subseteq T_p$, then (p, q) is an ϵ -equilibrium for G .

¹The question of *how* the player may obtain an equilibrium mixed strategy is a separate and well-studied topic, but not the focus of this work.

Proof. Consider an arbitrary $p' \in \Delta([n])$. Since $p'Rq^T$ is a convex combination of the $(Rq)_j$, it is at most $\max_j(Rq)_j$. And, since p is a convex combination of the $(Rq)_j$, with $\text{supp}(p) \subseteq S_q$, we have $pRq^T \geq \max_j(Rq)_j - \epsilon$. Thus $p'Rq^T \leq pRq^T + \epsilon$. Similarly, for an arbitrary $q' \in \Delta([n])$, we have $pCq'^T \leq \max_i(pC)_i$, and $pCq^T \geq \max_i(pC)_i - \epsilon$ since $\text{supp}(q) \subseteq T_p$. Thus $pCq'^T \leq pCq^T + \epsilon$. These two conditions guarantee that (p, q) is an ϵ -equilibrium. \square

We will use the Chernoff bound for random walks on an expander:

Theorem 3.3 (Gillman [Gil93]) *Let H be an expander graph with second largest eigenvalue λ and vertex set V , and let $f : V \rightarrow [-1, 1]$ be arbitrary with $E[f] = \mu$. Let X_1, X_2, \dots, X_t be the random variables induced by first picking X_1 uniformly in V and X_2, \dots, X_t by taking a random walk in H from X_1 . Then*

$$\Pr \left[\left| \frac{1}{t} \sum_i f(X_i) - \mu \right| > \delta \right] < e^{-O((1-\lambda)\delta^2 t)}.$$

Proof. [Proof of Theorem 1.1] When we are given G and p^* , we perform the following steps.

First, construct a multiset S of $[n]$ for which uniformly sampling from S approximates p^* to within ϵ/n . This can be done with $|S| \leq O(n/\epsilon)$. Denote by \tilde{p} the distribution induced by sampling uniformly from S . We identify S with the vertices of a constant-degree expander H , and we can sample $S' \subseteq S$ by taking a walk of length $t = O(\log n/\epsilon^2)$ steps in H . Note that this requires $O(\log |S| + O(t)) = O(\log n/\epsilon^2)$ random bits. Let p' be the probability distribution induced by sampling uniformly from S' . By Theorem 3.3 (and using the fact that C has entries in $[-1, 1]$), for each fixed i ,

$$\Pr[|(p'C)_i - (\tilde{p}C)_i| \geq \epsilon] \leq e^{-O(\epsilon^2 t)} < 1/n. \quad (1)$$

By a union bound $|(p'C)_i - (\tilde{p}C)_i| \leq \epsilon$ for all i with non-zero probability. This condition can be checked given G, p^* , and so we can derandomize the procedure completely by trying all choices of the random bits used in the expander walk.

When we are given G and q^* , we perform essentially the same procedure (with respect to R and q^*), and in the end we output a pair p', q' for which:

$$\begin{aligned} |(p'C)_i - (\tilde{p}C)_i| &\leq \epsilon \quad \forall i \\ |(Rq'^T)_j - (R\tilde{q}^T)_j| &\leq \epsilon \quad \forall j \end{aligned}$$

We claim that any such (p', q') is an 4ϵ -equilibrium, assuming (p^*, q^*) are an equilibrium. Using the fact that C, R have entries in $[-1, 1]$, and the fact that our multiset approximations to p^*, q^* have error at most ϵ/n in each coordinate, we obtain:

$$\begin{aligned} |(\tilde{p}C)_i - (p^*C)_i| &\leq \epsilon \quad \forall i \\ |(R\tilde{q}^T)_j - (Rq^{*T})_j| &\leq \epsilon \quad \forall j \end{aligned}$$

Define (as in Lemma 3.2):

$$\begin{aligned} T_{p'} &= \{i | (p'C)_i \geq \max_i (p'C)_i - 4\epsilon\} \\ S_{q'} &= \{j | (Rq'^T)_j \geq \max_j (Rq'^T)_j - 4\epsilon\}. \end{aligned}$$

Now, $w \in \text{supp}(p')$ implies $w \in \text{supp}(p^*)$ which implies $(Rq^{*T})_w = \max_j (Rq^{*T})_j$ (since (p^*, q^*) is a Nash equilibrium). From above we have that $\max_j (Rq'^T)_j \leq \max_j (Rq^{*T})_j + 2\epsilon$ and that $(Rq'^T)_w \geq (Rq^{*T})_w - 2\epsilon$. So $(Rq'^T)_w \geq \max_j (Rq'^T)_j - 4\epsilon$, and hence w is in $S_{q'}$. We conclude that $\text{supp}(p') \subseteq S_{q'}$. A symmetric argument shows that $\text{supp}(q') \subseteq T_{p'}$. Applying Lemma 3.2, we conclude that (p', q') is a 4ϵ -equilibrium as required. \square

Since an equilibrium can be found efficiently by Linear Programming in the two player zero-sum case, we obtain as a corollary:

Corollary 1.2 (restated). *Let $G = (R, C, n)$ be a two-player zero-sum game. For every $\epsilon > 0$, there is a deterministic procedure running in time $\text{poly}(|G|)^{1/\epsilon^2}$ that outputs a pair (p^*, q^*) that is an $O(\log n/\epsilon^2)$ -sparse ϵ -equilibrium for G .*

3.2 Three or more players

We extend the algorithm above to make it work for games involving three or more players.

Theorem 3.4 *Let $G = (T_1, T_2, \dots, T_\ell, n)$ be an ℓ -player game, and let $(p_1^*, p_2^*, \dots, p_\ell^*)$ be a Nash equilibrium for G . For every $\epsilon > 0$, there is a deterministic procedure P running in time $\text{poly}(|G|)^{1/\epsilon^2}$, such that the tuple $(P(G, p_1^*, 1), P(G, p_2^*, 2), \dots, P(G, p_\ell^*, \ell))$ is an $O((\ell \log n)/\epsilon^2)$ -sparse 4ϵ -equilibrium for G .*

Proof. The proof is almost identical to that of Theorem 1.1. When given $(G, p_1^*, 1)$ P samples $t = O((\ell - 1) \log n + \log \ell)/\epsilon^2$ strategies from player 1's multiset of strategies after identifying it with a constant-degree expander H and doing a t -step random walk on it. Let \tilde{p}_1 be the distribution obtained by sampling uniformly from the original multiset of strategies and \hat{p}_1 the distribution induced by sampling from the t strategies picked from the random walk. For some fixing of (i_2, \dots, i_ℓ) and j the Chernoff bound in (1) now becomes

$$\Pr[|T_j(\hat{p}_1, i_2, \dots, i_\ell) - T_j(\tilde{p}_1, i_2, \dots, i_\ell)| \geq \epsilon] \leq e^{-O(\epsilon^2 t)} < 1/(\ell n^{\ell-1})$$

By a union bound on all $\ell n^{\ell-1}$ possible fixings for (i_2, \dots, i_ℓ) and all j ,

$$|T_j(\hat{p}_1, i_2, \dots, i_\ell) - T_j(\tilde{p}_1, i_2, \dots, i_\ell)| < \epsilon$$

with positive probability. As before, we can derandomize the procedure completely by trying all choices of the random bits used in the expander walk.

Essentially the same procedure gives us $\hat{p}_2, \hat{p}_3, \dots, \hat{p}_\ell$. To show that $(\hat{p}_1, \dots, \hat{p}_\ell)$ constitute a 4ϵ -equilibrium, consider the set $\mathcal{S}_1 = \{i \mid T_1(i, \hat{p}_2, \dots, \hat{p}_\ell) \geq \max_i T_1(i, \hat{p}_2, \dots, \hat{p}_\ell) - 4\epsilon\}$. Define \mathcal{S}_j analogously with respect to T_j . It suffices to show that $\text{supp}(\hat{p}_j) \subseteq \mathcal{S}_j$ for all j . We sketch the argument for \hat{p}_1 ; symmetric arguments hold with respect to \hat{p}_j for all j .

By the same thread of reasoning as in the two-player case, for any $w \in \text{supp}(\hat{p}_1)$, $T_1(w, \hat{p}_2, \dots, \hat{p}_\ell) \geq T_1(w, p_2^*, \dots, p_\ell^*) - 2\epsilon$ and since $\text{supp}(\hat{p}_1) \subseteq \text{supp}(p_1^*)$,

$$T_1(w, p_2^*, \dots, p_\ell^*) = \max_i T_1(i, p_2^*, \dots, p_\ell^*) \geq \max_i T_1(i, \hat{p}_2, \dots, \hat{p}_\ell) - 2\epsilon.$$

Combining the two inequalities, we get $T_1(w, \hat{p}_2, \dots, \hat{p}_\ell) \geq \max_i T_1(i, \hat{p}_2, \dots, \hat{p}_\ell) - 4\epsilon$. \square

4 Limited randomness in repeated games

So far we have looked at optimizing the amount of randomness needed in single-round games where players execute their strategies only once. In this section, we investigate multiple-round games and in particular, the adaptive multiplicative weight algorithm of Freund and Schapire [FS99] for which we describe randomness-efficient modifications. In particular, we show that by using almost-pairwise independent random variables it is possible to achieve close to the same quality of approximation as in [FS99].

Note that we make crucial use of the full power of [FS99] – i.e., their guarantee still holds if the column player changes his play in response to the particular randomness-efficient sampling being employed by the row player:

Theorem 1.3 (restated). *Let M be the $n \times n$ -payoff matrix for a two-player zero-sum T -round game with entries in $\{0, 1\}$. For any $\epsilon < 1/2$ and constant δ , there exists an online randomized algorithm R using*

$O(\log n + \log \log T + \log(1/\epsilon))$ random bits with the following property: for any arbitrary sequence Q_1, \dots, Q_T of mixed strategies played (adaptively) by the column player over T rounds, R produces a sequence of strategies S_1, \dots, S_T such that with probability at least $1 - \delta$:

$$\frac{1}{T} \sum_{i=1}^T M(S_i, Q_i) \leq \frac{1}{T} \min_P \sum_{i=1}^T M(P, Q_i) + O\left(\sqrt{\frac{\log n}{T}} + \epsilon\right)$$

Proof. Our randomized online algorithm R is a modification of Freund and Schapire's multiplicative-weight adaptive algorithm [FS99]. For a game with payoff matrix M where both players have n strategies belonging to a strategy-set S , and for a sequence of mixed strategies (P_1, P_2, \dots, P_T) over T rounds for the first player described by

$$P_{i+1}(s) = \left(\frac{\beta^{M(s, Q_i)}}{\sum_s p_i(s) \beta^{M(s, Q_i)}} \right) p_i(s) \quad (2)$$

where $\beta = 1/(1 + \sqrt{2 \log n/T})$, the Freund-Schapire algorithm offers the following guarantee on the expected payoff over T rounds:

Lemma 4.1 ([FS99])

$$\frac{1}{T} \sum_{t=1}^T M(P_t, Q_t) \leq \min_P \frac{1}{T} \sum_{t=1}^T M(P, Q_t) + O\left(\sqrt{\frac{\log n}{T}}\right) \quad (3)$$

Running the Freund-Schapire algorithm requires $\Omega(T \log n)$ random bits in order to select a strategy from each distribution but we can optimize on this by using almost pairwise independent random variables.

As we did in the proof of Theorem 1.1 in Section 3.1, we can approximate any distribution P_t by a uniform distribution S_t drawn from a multiset of size $O(n/\epsilon)$ that approximates P_t to within ϵ/n and suffer at most $O(\epsilon)$ error. Therefore, under the uniform distribution over vertices $s \in S_i$ for all $i = 1, \dots, T$:

$$M(P_i, Q_i) - O(\epsilon) \leq E[M(S, Q_i)] \leq M(P_i, Q_i) + O(\epsilon)$$

Definition 4.2 [AGHP92] Let $Z_n \subseteq \{0, 1\}^n$ be a sample space and $X = x_1 \dots x_n$ be chosen uniformly from Z_n . Z_n is (ρ, k) -independent if for any positions $i_1 < i_2 < \dots < i_k$ and any k -bit string $t_1 \dots t_k$, we have

$$\left| \Pr_X[x_{i_1} x_{i_2} \dots x_{i_k} = t_1 \dots t_k] - 2^{-k} \right| \leq \rho$$

Alon et al [AGHP92] give efficient constructions of (ρ, k) -independent random variables over $\{0, 1\}^n$ that we can suitably adapt to obtain T $(\rho, 2)$ -independent random variables S_1, \dots, S_T over a larger alphabet of size $O(n/\epsilon)$ using $O(\log n + \log(1/\rho) + \log(1/\epsilon) + \log \log T)$ random bits.

The following lemma is key to the remainder of the proof and a proof of the lemma follows immediately afterwards.

Lemma 4.3 Let S_1, \dots, S_T be $(\rho, 2)$ -independent random variables. Then, for any δ :

$$\Pr_{S_1, \dots, S_T} \left[\frac{1}{T} \sum_{i=1}^T M(S_i, Q_i) \geq \frac{1}{T} E \left[\sum_{i=1}^T M(S_i, Q_i) \right] + \sqrt{\frac{1}{\delta} \left(\frac{1}{T} + \frac{2\rho n^2}{\epsilon^2} \right)} \right] \leq \delta$$

Setting $\rho = O(\delta \epsilon^6 / n^2)$ in Lemma 4.3 gives us:

$$\Pr_{S_1, \dots, S_T} \left[\frac{1}{T} \sum_{i=1}^T M(S_i, Q_i) \geq \frac{1}{T} E \left[\sum_{i=1}^T M(S_i, Q_i) \right] + \epsilon \right] \leq \delta \quad (4)$$

Therefore with probability at least $1 - \delta$ over the choice of randomness of S_1, \dots, S_T

$$\frac{1}{T} \sum_{i=1}^T M(S_i, Q_i) \leq \frac{1}{T} E \left[\sum_{i=1}^T M(S_i, Q_i) \right] + \epsilon \leq \frac{1}{T} \sum_{i=1}^T M(P_i, Q_i) + O(\epsilon)$$

Finally by application of Lemma 4.1 we have with probability at least $1 - \delta$

$$\frac{1}{T} \sum_{i=1}^T M(S_i, Q_i) \leq \frac{1}{T} \min_P \sum_{i=1}^T M(P, Q_i) + O\left(\sqrt{\frac{\log n}{T}}\right) + O(\epsilon) \quad (5)$$

Note that by our choice of ρ , we require $O(\log n + \log \log T + \log(1/\epsilon))$ random bits. This completes the proof of the theorem. \square

Proof. (Of Lemma 4.3) The proof is essentially a variation of the Chebyshev tail inequality for $(\delta, 2)$ independent random variables. Let $Z = \sum_i M(S_i, Q_i)$. Then,

$$\begin{aligned} \Pr \left[\frac{1}{T} Z \geq \frac{1}{T} EZ + \lambda \right] &\leq \Pr \left[\left| \frac{1}{T} Z - \frac{1}{T} EZ \right| \geq \lambda \right] \\ &= \Pr \left[\frac{1}{T^2} (Z - EZ)^2 \geq \lambda^2 \right] \\ &\leq \frac{1}{T^2} \frac{\sigma^2(Z)}{\lambda^2} \end{aligned} \quad (6)$$

We bound $\sigma^2(Z)$ as follows:

$$\begin{aligned} \sigma^2 &= E \left[\left(\sum_T M(S_i, Q_i) - E[M(S_i, Q_i)] \right)^2 \right] \\ &= \sum_T \sigma^2(M(S_i, Q_i)) + 2 \sum_{i,j \in [T]} (E[M(S_i, Q_i)M(S_j, Q_j)] - E[M(S_i, Q_i)] E[M(S_j, Q_j)]) \\ &= \sum_T \sigma^2(M(S_i, Q_i)) + \\ &\quad 2 \sum_{i,j \in [T]} \left(\sum_{s_i \in S_i, s_j \in S_j} M(s_i, Q_i)M(s_j, Q_j) \Pr[S_i = s_i, S_j = s_j] - E[M(s_i, Q_i)] E[M(s_j, Q_j)] \right) \\ &\leq \sum_T \sigma^2(M(S_i, Q_i)) + \\ &\quad 2 \sum_{i,j \in [T]} \left(\sum_{s_i \in S_i, s_j \in S_j} M(s_i, Q_i)M(s_j, Q_j) (\Pr[S_i = s_i] \Pr[S_j = s_j] + \delta) - E[M(S_i, Q_i)] E[M(S_j, Q_j)] \right) \\ &\quad \text{(By virtue of } (\delta, 2)\text{-independence of } S_i, S_j) \\ &\leq \sum_T \sigma^2(M(S_i, Q_i)) + 2 \sum_{i,j \in [T]} (\delta n^2 / \epsilon^2 + E[M(S_i, Q_i)] E[M(S_j, Q_j)] - E[M(S_i, Q_i)] E[M(S_j, Q_j)]) \\ &\quad \text{(There are at most } (n/\epsilon) \text{ possible strategies in the multisets } S_i, S_j) \\ &\leq T \max_j \sigma_j^2 + 2\delta n^2 T^2 / \epsilon^2 \end{aligned}$$

Let $\sigma_0 = \max_j \sigma_j$. Then,

$$\sigma^2(Z) \leq T \sigma_0^2 + 2\delta T^2 n^2 / \epsilon^2 \leq T + 2\delta T^2 n^2 / \epsilon^2 \quad (7)$$

Substituting in (6) and setting $\lambda = \sqrt{\frac{1}{\alpha} \left(\frac{1}{T} + \frac{2\delta n^2}{\epsilon^2} \right)}$ we get the desired inequality. \square

5 Unbalanced games

In this section we will look at what happens when one of the players (perhaps as a consequence of having limited randomness) is known to have very few – k – available strategies, while the other player has $n \gg k$ available strategies. In such a game does there exist a k -sparse strategy for the second player? We prove that this is indeed the case. The main technical tool we'll need is Carathéodory's Theorem. Since many of the results we obtain depend on the constructive feature of the theorem we record below such a proof.

Theorem 5.1 (Carathéodory's Theorem, constructive version) *Let v_1, v_2, \dots, v_n be vectors in a k -dimensional subspace of \mathbb{R}^m where $n \geq k + 1$, and suppose*

$$v = \sum_{i=1}^n \alpha_i v_i \quad \text{with} \quad \sum_i \alpha_i = 1 \quad \text{and} \quad \alpha_i \geq 0 \quad \text{for all } i \quad (8)$$

Then there exist $\alpha'_1, \dots, \alpha'_n$ for which $v = \sum_{i=1}^n \alpha'_i v_i$ with $\sum_i \alpha'_i = 1$ and $\alpha'_i \geq 0$ for all i , and $|\{i : \alpha'_i > 0\}| \leq k + 1$. Moreover the α'_i can be found in polynomial time, given the α_i and the v_i .

Proof. We write a linear combination involving the $n - 1 \geq k$ vectors $(v_1 - v_n), \dots, (v_{n-1} - v_n)$ as follows

$$\sum_{j=1}^{n-1} t \beta_j (v_j - v_n) = 0 \quad (9)$$

for some arbitrary t that we shall define later. Let $\beta_l = -\sum_{j=1}^{n-1} \beta_j$. Adding (8), (9) we get

$$\sum_{j=1}^n (\alpha_j + t \beta_j) v_j = v \quad (10)$$

Now, we set $t = \min_j -\alpha_j / \beta_j$, say without loss of generality $-\alpha_1 / \beta_1$. Rewriting (10),

$$\sum_{j=2}^n (\alpha_j - \alpha_1 \beta_j / \beta_1) v_j = v \quad (11)$$

Furthermore, $(\alpha_j - \alpha_1 \beta_j / \beta_1) > 0$ for all $j = 2, \dots, n$ and $\sum_{j=2}^n (\alpha_j - \alpha_1 \beta_j / \beta_1) = 1$. Hence, we have expressed v as a convex combination in $(n - 1)$ vectors from $\{v_1, \dots, v_n\}$. We proceed in this manner, until we are no longer able to write a linear combination in k vectors as was done in (9). This procedure takes $\text{poly}(n)$ time. \square

The main theorem in this section is below. We state it for two players for simplicity, and extend it to ℓ players later.

Theorem 1.4 (restated). *Let $G = (R, C, k, n)$ be a two-player game. Given p^* for which there exists a q^* such that (p^*, q^*) is a Nash equilibrium, we can compute in deterministic polynomial time q' for which (p^*, q') is a Nash equilibrium and $|\text{supp}(q')| \leq k$.*

Proof. We would like to apply Carathéodory's Theorem as was done to obtain a similar result in [LMM03] but we will need to modify our application since Carathéodory's Theorem applies to linear subspaces whereas the Nash equilibrium strategies define an affine subspace. By Lemma 2.5 given G, p^* we can construct a q such that (p^*, q) is also a Nash equilibrium. Consider the column vector u given by

$$u = Rq^T = \sum_{i=1}^n q_i m_i$$

where m_i is the i -th column vector in R . Since R is a $k \times n$ matrix, u is in the span of column vectors m_1, \dots, m_n that lie in a k -dimensional subspace of \mathbb{R}^k . Since (p^*, q) is a Nash-equilibrium, for all $i \in \text{supp}(q)$:

$$p^* m_i = \max_j p^* m_j = w^*$$

This is an additional linear constraint on the m_i s and hence m_1, \dots, m_n lie in a $(k-1)$ -dimensional affine subspace \mathcal{A} . Since u is a convex combination of the m_i 's,

$$p^*u = p^* \left(\sum_{i=1}^n q_i m_i \right) = \sum_{i=1}^n q_i (p^* m_i) = w^* \sum_{q_i > 0} q_i = w^*$$

and hence u also lies in \mathcal{A} . Define $m'_i = m_i - w^* 1_k$, where 1_k is the all-ones column vector. Then for all $i \in \text{supp}(q)$, $p^* m'_i = 0$. Therefore

$$u' = u - w^* 1_k = \sum_{q_i > 0} q_i m'_i$$

lies in a $(k-1)$ -dimensional subspace of \mathbb{R}^k . Applying Carathéodory's theorem 5.1, u' can thus be rewritten as a convex combination:

$$u' = \sum_i q'_i m'_i$$

where $|\text{supp}(q')| \leq k$. It follows that

$$u = u' + w^* 1_k = \sum_i q'_i m'_i + w^* 1_k = \sum_i q'_i (m'_i + w^* 1_k) = \sum_i q'_i m_i$$

We claim that (p^*, q') is the desired Nash equilibrium. This is true because $\text{supp}(q') \subseteq \text{supp}(q^*)$, and for each $j \in \text{supp}(p^*)$, $R(q')^T = Rq'^T$ and hence $\text{supp}(p^*) \subseteq \{j : (R(q')^T)_j = \max_s (R(q')^T)_s\}$. \square

The following theorem extends the result above to the general ℓ -player game where $\ell-1$ players play sparse strategies.

Corollary 1.5 (restated). *Let $G = (T_1, T_2, \dots, T_\ell, k_1, k_2, \dots, k_{\ell-1}, n)$ be an ℓ -player game where $\prod_{i=1}^{\ell-1} k_i < n$. Given $G, p_1^*, \dots, p_{\ell-1}^*$ there exists a deterministic polynomial-time procedure to compute \widehat{p}_ℓ such that $(p_1^*, p_2^*, \dots, p_{\ell-1}^*, \widehat{p}_\ell)$ is a Nash equilibrium for G and $|\text{supp}(\widehat{p}_\ell)| \leq k = \prod_{i=1}^{\ell-1} k_i$.*

Proof. The proof is similar to that of Theorem 1.4. Applying Lemma 2.5, we obtain p_ℓ such that $(p_1^*, \dots, p_{\ell-1}^*, p_\ell)$ is a Nash equilibrium for G . Consider the $k \times n$ matrix $T_{-\ell}$ formed by choosing each of $T_1, \dots, T_{\ell-1}$'s strategies. $T_{-\ell}$ is of rank at most $k < n$. As in proving Theorem 1.4, we observe that the column vector $u = T_{-\ell} p_\ell$ is in the span of column vectors $m_1^{-\ell}, \dots, m_n^{-\ell}$ that lie in a $(k-1)$ -dimensional affine subspace following the same line of argument with respect to $(p_1^*, \dots, p_{\ell-1}^*)$ imposing a constraint on the m_i 's. Translating back and forth between the linear and affine subspace and applying Carathéodory's Theorem we get $p'_\ell, \text{supp}(p'_\ell) \leq k$ the Nash equilibrium strategy for T_ℓ . The algorithm runs in time polynomial in n^ℓ . \square

In Theorem 5.2, we also show that these bounds are tight (for small values of k) in that we show examples of games for which there exist Nash equilibria where the ℓ -th player requires support $k = \prod_{i=1}^{\ell-1} k_i$.

Theorem 5.2 *For every $n, \ell \geq 2$, and $k_1, \dots, k_{\ell-1}$ such that $k = \prod_{i=1}^{\ell-1} k_i < n$, there exists an ℓ -player game $G = (T_1, T_2, \dots, T_\ell, k_1, k_2, \dots, k_{\ell-1}, n)$ for which the following hold:*

1. *There exists a Nash equilibrium $(p_1^*, p_2^*, \dots, p_\ell^*)$ with $|\text{supp}(p_i^*)| = k_i$ for $i = 1, 2, \dots, \ell-1$ and $|\text{supp}(p_\ell^*)| = k < n$. And, there is no p_ℓ with $|\text{supp}(p_\ell)| < \prod_{i=1}^{\ell-1} k_i = k$ such $(p_1^*, p_2^*, \dots, p_{\ell-1}^*, p_\ell)$ is a Nash equilibrium.*

Proof. (Sketch) For each T_i ; $i = 1, \dots, \ell$ we set values as follows. For some lexicographical ordering of $J = \{(v_1, \dots, v_{\ell-1}) \mid 1 \leq v_r \leq k_r; 1 \leq r \leq \ell-1\}$ given by $\alpha_1, \alpha_2, \dots, \alpha_k$ we set

$$T_i(\alpha_j, j) = \begin{cases} 1 & 1 \leq j \leq k; 1 \leq i \leq \ell, \\ 0 & \text{otherwise.} \end{cases} \quad (12)$$

Let $(p_1^*, p_2^*, \dots, p_\ell^*)$ be the mixed strategy ℓ -tuple where for each $i = 1, \dots, \ell-1$ p_i^* is the uniform distribution on the set of strategies $\{v_i \mid 1 \leq v_i \leq k_i\}$. Then, (p_1^*, \dots, p_ℓ^*) specifies a Nash equilibrium for G . This is easy

to see since the payoff to T_i upon playing p_i^* in response to $(p_1^*, \dots, p_{i-1}^*, p_{i+1}^*, \dots, p_\ell^*)$ is $1/k$ for $i = 1, \dots, \ell$ and 0 for any other strategy. Hence, T_i has no incentive to deviate unilaterally to any other strategy that will increase his payoff. Since this holds true for all $i = 1, \dots, \ell$ (p_1^*, \dots, p_ℓ^*) is a Nash equilibrium.

Next, suppose that T_ℓ decides to play a strategy with support $k' < k = \prod_{i=1}^{\ell-1} k_i$. Then, there is some $1 \leq j < k$ such that j is not included in T_ℓ 's support. Corresponding to j is some $(v_1^*, \dots, v_{\ell-1}^*)$ according to the lexicographical ordering described above. By leaving out v_i^* from its support and redistributing that weight onto other strategies, T_i has increased its payoff from $1/k$ to $1/k'$. Hence such a strategy cannot be an equilibrium. \square

6 Finding sparse ϵ -equilibria in low-rank games

We now consider games of rank k , which is a significant generalization of the “unbalanced” games in the previous section. Indeed, rank k games are perhaps the most general class of games for which sparse equilibria are guaranteed to exist. In this section we give algorithms to compute sparse ϵ -equilibria in this setting.

6.1 Two player case

Since we are computing ϵ -equilibria, we only expect the game specification to be given up to some fixed precision. We will be working with rank k matrices M expressed as $M_1 M_2$ (where M_1 is a $n \times k$ matrix and M_2 is a $k \times n$ matrix). Such a decomposition can be found efficiently via basic linear algebra. In the following theorem we take M_1 and M_2 , specified up to fixed precision, as our starting point.² As the example in §6.3 illustrates, such a decomposition is already available for many natural games. Our convention for expressing fixed precision entries will be to require them to be integers in the range $[-B, B]$ for a bound B .

Theorem 1.6 (restated). *Let $G = (R, C, n)$ be a two player game such that R and C have rank at most k . Furthermore, let $R = R_1 R_2$, $C = C_1 C_2$ be a decomposition of R, C with R_1, R_2, C_1, C_2 containing integer entries in $[-B, B]$. Then, for every $\epsilon > 0$, there is a deterministic procedure P running in time $(4B^2 k / \epsilon)^{2k} \text{poly}(|G|)$ that returns a 4ϵ -Nash equilibrium (p, q) with $|\text{supp}(p)|, |\text{supp}(q)| \leq k + 1$.*

Lipton et al. showed that there *exist* $(k + 1)$ -sparse Nash equilibria in this setting and this implies an enumeration algorithm to find an equilibrium in time approximately $n^{k+1} \text{poly}(|G|)$. Our algorithm shows that the problem is “fixed parameter tractable” [Ces05, DF99, DFS97] where ϵ , the rank k and precision B are the parameters.

Proof. Note that the payoff to the row-player when (p, q) is the strategy tuple for the game which is given by $p R q^T$ can now be written as $p R_1 R_2 q^T$ and likewise for the column player. The first step in our algorithm is to “guess” a collection of vectors to within $\delta = \epsilon / (2Bk)$ precision. We describe the “correct” guess relative to an (arbitrary) Nash equilibrium (p^*, q^*) for G . Let $p^{*'} = p^* C_1, q^{*'} = R_2 q^{*T}$. Note that from our choice of C_1, R_2 it holds that $p^{*'}, q^{*'}$ satisfy $-B \leq p_i^{*'}, q_i^{*'} \leq B; i = 1, \dots, k$. The algorithm is as follows:

1. Guess a \tilde{p}' such that for all $i = 1, \dots, k$ $|p_i^{*'} - \tilde{p}'_i| \leq \delta$. Similarly, guess \tilde{q}' such that for all $i = 1, \dots, k$ $|q_i^{*'} - \tilde{q}'_i| \leq \delta$.
2. Let $\alpha_s = (\tilde{p}' C_2)_s$ and $\beta_t = (R_1 \tilde{q}'^T)_t$. Set $S = \{s \mid \max_r \alpha_r - 2\epsilon \leq \alpha_s \leq \max_r \alpha_r\}$ and $T = \{t \mid \max_r \beta_r - 2\epsilon \leq \beta_t \leq \max_r \beta_r\}$.

²We note that computing M_1, M_2 of fixed precision such that $M_1 M_2$ approximates M is not necessarily always possible or straightforward. We state our theorem in this way to avoid these complications, a detailed discussion of which would be beyond the scope of this paper.

3. Find a feasible solution \bar{p} to the following linear program

$$|(\bar{p}C_1)_j - \bar{p}'_j| \leq \delta; \quad j = 1, \dots, k \quad (13)$$

$$\bar{p}_i \geq 0; \quad i = 1, \dots, n \quad (14)$$

$$\bar{p}_i = 0; \quad i \notin T \quad (15)$$

$$\sum_{i=1}^n \bar{p}_i = 1 \quad (16)$$

and a feasible solution \bar{q} to the analogous linear program in which the first set of constraints is

$$|(R_2\bar{q}^T)_j - \bar{q}'_j| \leq \delta; j = 1, \dots, k.$$

4. $v = \bar{p}C_1$ is a convex combination of the rows of C_1 , all of which lie in a k -dimensional subspace. From Carathéodory's Theorem (Theorem 5.1), we can find \hat{p} with $\text{supp}(\hat{p}) \subseteq \text{supp}(p)$ for which $|\text{supp}(\hat{p})| \leq k+1$ and $v = \hat{p}C_1$.

5. Similarly $u = R_2\bar{q}^T$ is a convex combination of the columns of R_2 , all of which lie in a k -dimensional subspace. Applying Carathéodory's Theorem again, we find \hat{q} with $\text{supp}(\hat{q}) \subseteq \text{supp}(q)$ for which $|\text{supp}(\hat{q})| \leq k+1$ and $u = R_2\hat{q}^T$.

6. Return \hat{p}, \hat{q} .

Correctness follows from the next two claims:

Claim 6.1 *A feasible solution to the linear programs formulated in step 3 of the algorithm exists.*

Proof. (Of Claim 6.1) We claim that if (p^*, q^*) is a Nash equilibrium for G then $\bar{p} = p^*$ is a feasible solution to the first LP and $\bar{q} = q^*$ is a feasible solution to the second LP.

Assume \bar{p}' and \bar{q}' are the correct guesses. We need to show that $\text{supp}(p^*) \subseteq T$ and $\text{supp}(q^*) \subseteq S$. Since \bar{p}' is a correct guess, $p^{*'} = p^*C_1$ satisfies $|p^{*'}_j - \bar{p}'_j| \leq \delta$. Therefore, p^* satisfies (13). Similarly for \bar{q}' we have $|q^{*'}_j - \bar{q}'_j| \leq \delta$ and so q^* satisfies (13) in the LP for q .

Consider $r \in \text{supp}(p^*)$ and $(R_1\bar{q}'^T)_r$. Since $|\bar{q}'_j - q^{*'}_j| \leq \epsilon/(2Bk)$ for all $j = 1, \dots, k$, we get

$$(R_1q^{*'}^T)_r - \epsilon \leq (R_1\bar{q}'^T)_r \leq (R_1q^{*'}^T)_r + \epsilon \quad (17)$$

Moreover p^* is a Nash equilibrium strategy and $r \in \text{supp}(p^*)$. Therefore,

$$\max_t (R_1\bar{q}'^T)_t - \epsilon \leq (R_1q^{*'}^T)_r = \max_t (R_1q^{*'}^T)_t \leq \max_t (R_1\bar{q}'^T)_t + \epsilon \quad (18)$$

Combining (17) and (18),

$$\max_t (R_1\bar{q}'^T)_t - 2\epsilon \leq (R_1\bar{q}'^T)_r \leq \max_t (R_1\bar{q}'^T)_t + 2\epsilon$$

and the right hand side can be simplified to yield:

$$\max_t (R_1\bar{q}'^T)_t - 2\epsilon \leq (R_1\bar{q}'^T)_r \leq \max_t (R_1\bar{q}'^T)_t.$$

Hence $r \in T$ and $\text{supp}(p^*) \subseteq T$. Similarly, we can show that $\text{supp}(q^*) \subseteq S$. \square

Claim 6.2 *(\hat{p}, \hat{q}) as returned by the algorithm is a 4ϵ -equilibrium.*

Proof. (Of Claim 6.2) We need to show that for any $p', q', (\hat{p}, \hat{q})$ satisfy

$$\begin{aligned}\hat{p}R\hat{q}^T &\geq p'R\hat{q}^T - 4\epsilon, \\ \hat{p}C\hat{q}^T &\geq \hat{p}Cq'^T - 4\epsilon\end{aligned}$$

Now,

$$\hat{p}C\hat{q}^T = ((\hat{p}C_1)C_2)\hat{q}^T$$

By our choice of \hat{p} in step 4. of the algorithm, $\hat{p}C_1 = \bar{p}C_1$. So,

$$\begin{aligned}\hat{p}C\hat{q}^T &= (\bar{p}C_1)C_2\hat{q}^T \\ &\geq (\bar{p}' - \delta 1_k)C_2\hat{q}^T \quad \text{from (13)} \\ &= \bar{p}'C_2\hat{q}^T - \delta 1_k C_2\hat{q}^T \\ &\geq \bar{p}'C_2\hat{q}^T - \epsilon\end{aligned}\tag{19}$$

Since $\text{supp}(\hat{q}) \subseteq \text{supp}(\bar{q})$ and $\text{supp}(\bar{q})$ contains only s for which $(\bar{p}'C_2)_s \geq \max_r(\bar{p}'C_2)_r - 2\epsilon$ we obtain

$$\begin{aligned}\bar{p}'C_2\hat{q}^T &\geq \max_r(\bar{p}'C_2)_r - 2\epsilon \\ &\geq \max_r((\bar{p}C_1 - \delta 1_k)C_2)_r - 2\epsilon \quad \text{from (13)} \\ &\geq \max_r(\hat{p}C_1C_2)_r - 3\epsilon \\ &\geq \hat{p}Cq'^T - 3\epsilon \quad \text{for any } q'\end{aligned}\tag{20}$$

Combining (19) and (20),

$$\hat{p}C\hat{q}^T \geq \hat{p}Cq'^T - 4\epsilon$$

Similarly, $\hat{p}R\hat{q}^T \geq p'R\hat{q}^T - 4\epsilon$ for any p' . \square

We analyze the run-time for the algorithm above in terms of the support parameter k and number of strategies n , and B . The first step of the algorithm where we “guess” \bar{p}' requires exhaustively going through all possible choices for each component of \bar{p}' in the interval $[-B, B]$ in steps of δ , and similarly for guessing \bar{q}' . This takes time $(2B/\delta)^{2k} = (4B^2k/\epsilon)^{2k}$. For each choice of (\bar{p}', \bar{q}') , we will need to solve the linear program above which takes $\text{poly}(|G|)$ time. The applications of Carathéodory’s Theorem also take $\text{poly}(|G|)$ and so the running time is as claimed and this completes the proof of the theorem. \square

6.2 Three or more players

In this section, we will look at obtaining approximate Nash equilibria for low-rank games with three or more players. This direction does not seem to have been studied before, and the previously known algorithms for low-rank games [KT07] don’t seem to extend to more than 2 players.

We begin by stating some definitions related to tensor rank.

Definition 6.3 *An arity- ℓ tensor T with dimension n has a p -decomposition if it can be expressed as: $T = \sum_{i=1}^p t_1^{(i)} \otimes t_2^{(i)} \otimes \dots \otimes t_\ell^{(i)}$ where each $t_j^{(i)}$ is an $n \times 1$ vector (i.e. it is the sum of p pure tensors).*

It will be convenient to aggregate the information in a p -decomposition of such a tensor T into a ℓ -tuple $(C_{i1}, C_{i2}, \dots, C_{i\ell})$ where each C_{ij} is an $n \times p$ matrix whose columns are given by $(t_{ij}^{(1)}, \dots, t_{ij}^{(p)})$. We will refer to the r -th row of a matrix C_{ij} as $C_{ij}(r)$. We will also define \odot to be the component-wise product of vectors:

Definition 6.4 *For two vectors $v, w \in \mathbb{R}^n$, define $(v \odot w)$ to be the vector $(v_1w_1, v_2w_2, \dots, v_nw_n) \in \mathbb{R}^n$.*

The following inequality will be useful in obtaining a result for the extension to three or more players.

Lemma 6.5 Let $x_1, x_2, \dots, x_\ell, \widetilde{x}_1, \widetilde{x}_2, \dots, \widetilde{x}_\ell \in \mathbb{R}^k$ be vectors satisfying $|x_i - \widetilde{x}_i| \leq \delta$ and furthermore, let $|\widetilde{x}_i| \leq B$ for all $i = 1, \dots, \ell$. Then,

$$\langle x_1, (\widetilde{x}_2 \odot \dots \odot \widetilde{x}_\ell) \rangle - \delta \ell k (B + \delta)^{\ell-1} \leq \langle x_1, (x_2 \odot x_3 \odot \dots \odot x_\ell) \rangle \leq \langle x_1, (\widetilde{x}_2 \odot \dots \odot \widetilde{x}_\ell) \rangle + \delta \ell k (B + \delta)^{\ell-1}$$

Proof.

$$\begin{aligned} \langle x_1, (x_2 \odot x_3 \odot \dots \odot x_\ell) \rangle &\geq \langle x_1, (\widetilde{x}_2 - \delta \mathbf{1}_k) \odot x_3 \odot \dots \odot x_\ell \rangle \\ &\geq \langle x_1, \widetilde{x}_2 \odot x_3 \odot \dots \odot x_\ell \rangle - \delta (B + \delta)^{\ell-2} \langle x_1, \mathbf{1}_k \rangle \\ &\geq \langle x_1, \widetilde{x}_2 \odot x_3 \odot \dots \odot x_\ell \rangle - \delta k (B + \delta)^{\ell-1} \\ &\vdots \\ &\geq \langle x_1, \widetilde{x}_2 \odot \dots \odot \widetilde{x}_\ell \rangle - (\ell - 1) \delta k (B + \delta)^{\ell-1} \\ &\geq \langle x_1, \widetilde{x}_2 \odot \dots \odot \widetilde{x}_\ell \rangle - \delta \ell k (B + \delta)^{\ell-1} \end{aligned}$$

The other side of the inequality can be similarly shown. \square

We begin by generalizing Lipton et al. [LMM03] to the multiplayer case. The following theorem shows that low rank games in the multiplayer case have sparse Nash equilibria; in the next section we give nontrivial algorithms to find approximate equilibria with this sparsity.

Corollary 6.6 Let $G = (T_1, \dots, T_\ell, n)$ be an ℓ -player game, and suppose T_i has rank k_i . Then there exists a Nash equilibrium (p_1^*, \dots, p_ℓ^*) with $|\text{supp}(p_i^*)| \leq 1 + \sum_{j=1}^\ell k_j$ for all i .

Proof. Let (q_1, \dots, q_ℓ) be a Nash equilibrium for G and let $(C_{i,1}, \dots, C_{i,\ell})$ be the tensor decomposition for T_i . Given q_1, \dots, q_ℓ , we can define vectors in \mathbb{R}^n :

$$\begin{aligned} w_1 &= (C_{1,1} (\quad \quad \quad q_2 C_{1,2} \odot \quad q_3 C_{1,3} \odot \quad \dots \quad \odot \quad q_\ell C_{1,\ell})^T) \\ w_2 &= (C_{2,2} (q_1 C_{2,1} \quad \quad \quad \odot \quad q_3 C_{2,3} \odot \quad \dots \quad \odot \quad q_\ell C_{2,\ell})^T) \\ w_3 &= (C_{3,3} (q_1 C_{3,1} \quad \odot \quad q_2 C_{3,2} \quad \quad \quad \odot \quad \dots \quad \odot \quad q_\ell C_{3,\ell})^T) \\ &\vdots \\ w_\ell &= (C_{\ell,\ell} (q_1 C_{\ell,1} \quad \odot \quad q_2 C_{\ell,2} \odot \quad \dots \quad \odot \quad q_{\ell-1} C_{\ell,\ell-1} \quad \quad \quad)^T) \end{aligned} \tag{21}$$

By the definition of a Nash equilibrium, the following conditions are satisfied:

$$\begin{aligned} \text{supp}(q_1) &\subseteq \{v : (w_1)_v = \max_u (w_1)_u\} \\ \text{supp}(q_2) &\subseteq \{v : (w_2)_v = \max_u (w_2)_u\} \\ &\vdots \\ \text{supp}(q_\ell) &\subseteq \{v : (w_\ell)_v = \max_u (w_\ell)_u\}. \end{aligned}$$

And indeed any $(\widehat{q}_1, \dots, \widehat{q}_\ell)$ satisfying these conditions (when the w_i are defined relative the the \widehat{q}_i) is a Nash equilibrium.

Set $s = \sum_{j=1}^\ell k_j$. Now, by Carathéodory's Theorem there exists $\widehat{q}_1 \in \mathbb{R}^n$ with $\text{supp}(\widehat{q}_1) \subseteq \text{supp}(q_1)$ and $|\text{supp}(\widehat{q}_1)| \leq s + 1$ for which:

$$\widehat{q}_1 (C_{2,1} | C_{3,1} | C_{4,1} | \dots | C_{\ell,1}) = q_1 (C_{2,1} | C_{3,1} | \dots | C_{\ell,1}),$$

since the right-hand-side is a convex combination of vectors in $\mathbb{R}^{s'}$ for $s' \leq s$. Similarly, there exists $\widehat{q}_2 \in \mathbb{R}^n$ with $\text{supp}(\widehat{q}_2) \subseteq \text{supp}(q_2)$ and $|\text{supp}(\widehat{q}_2)| \leq s + 1$ for which:

$$\widehat{q}_2 (C_{1,2} | C_{3,2} | C_{4,2} | \dots | C_{\ell,2}) = q_2 (C_{1,2} | C_{3,2} | C_{4,2} | \dots | C_{\ell,2}),$$

since the right-hand-side is a convex combination of vectors in $\mathbb{R}^{s'}$ for $s' \leq s$.

A symmetric argument gives us a sparse \widehat{q}_i from each q_i . Moreover, these \widehat{q}_i produce precisely the same vectors w_1, w_2, \dots, w_ℓ via (21). And, since $\text{supp}(\widehat{q}_i) \subseteq \text{supp}(q_i)$ for all i , the strategies \widehat{q}_i satisfy the above conditions for being a Nash equilibrium, assuming the original q_i did. \square

We now turn to algorithms for finding sparse approximate equilibria, with three or more players. We first consider the case when the tensor decomposition is known. This is not an unnatural assumption: in §6.3, we describe a class of games for which the tensor decomposition is naturally given by the description of the payoff functions.

Theorem 6.7 *Let $G = (T_1, \dots, T_\ell, n)$ be an ℓ -player game, and suppose we are given a k -decomposition of $T_i = (C_{i1}, \dots, C_{i\ell})$ where each of the C_{ij} is an $n \times k$ matrix with integer values in $[-B, B]$ for $i, j = 1, \dots, \ell$. Then for every $\epsilon > 0$, there is a deterministic procedure P running in time*

$$((2B)^\ell k\ell/\epsilon)^{k(\ell-1)^\ell} \text{poly}(|G|)$$

that returns a 4ϵ -Nash equilibrium $(p_1, p_2, \dots, p_\ell)$ with $|\text{supp}(p_i)| \leq 1 + \ell k$ for all i .

Proof. As in the two player case, our first step is to “guess” a collection of vectors to within $\delta = \frac{\epsilon}{k\ell(2B)^{\ell-1}}$ precision. We describe the “correct” guess relative to an (arbitrary) Nash equilibrium (p_1, \dots, p_ℓ) for G .

1. Let $(p'_{11}, \dots, p'_{1\ell}) = (p_1 C_{11}, p_2 C_{12}, \dots, p_\ell C_{1\ell})$ be an ℓ -tuple of $1 \times k$ vectors. Note that since all entries in $(C_{11}, C_{12}, \dots, C_{1\ell})$ lie in $[-B, B]$, entries in $(p'_{11}, \dots, p'_{1\ell})$ also lie in $[-B, B]$.

2. Guess an $(\ell - 1)$ -tuple $(\widetilde{p'_{12}}, \dots, \widetilde{p'_{1\ell}})$ such that

$$|(p'_{1i})_j - \widetilde{(p'_{1i})_j}| \leq \delta \quad i = 2, \dots, \ell; j = 1, \dots, k$$

3. For $C_{11}(r)$, the r -th row in C_{11} let

$$\begin{aligned} \alpha_r &= \left(C_{11} \left(\widetilde{p'_{12}} \odot \widetilde{p'_{13}} \odot \dots \odot \widetilde{p'_{1\ell}} \right)^T \right)_r; r = 1, \dots, n \\ \mathcal{S}_1 &= \left\{ r \mid \max_t \alpha_t - 2\epsilon \leq \alpha_r \leq \max_t \alpha_t \right\} \end{aligned}$$

4. By repeating steps 1 through 3 on T_2, \dots, T_ℓ we can similarly obtain $\mathcal{S}_2, \dots, \mathcal{S}_\ell$.

5. Find $(\bar{p}_1, \dots, \bar{p}_\ell)$ satisfying the following linear program

$$|(\bar{p}_j C_{mj})_r - (\widetilde{p'_{mj}})_r| \leq \delta \quad r = 1, \dots, k; j = 1, \dots, \ell; \quad (22)$$

$$(\bar{p}_j)_i \geq 0; \quad i = 1, \dots, n; j = 1, \dots, \ell \quad (23)$$

$$\sum_{i=1}^n (\bar{p}_j)_i = 1; \quad j = 1, \dots, \ell \quad (24)$$

$$(\bar{p}_j)_i = 0; \quad i \notin \mathcal{S}_j; j = 1, \dots, \ell \quad (25)$$

6. Let $D_i = (C_{1,i} \mid C_{2,i} \mid \dots \mid C_{i-1,i} \mid C_{i+1,i} \mid \dots \mid C_{\ell,i})$ for $i = 1, \dots, \ell$. $v = \bar{p}_i D_i$ is a convex combination of column vectors in D_i that are in an ℓk -dimensional subspace. Apply Carathéodory's Theorem (Theorem 5.1) to obtain \widehat{p}_i such that $v = \widehat{p}_i D_i$ for $i = 1, \dots, \ell$ and $\text{supp}(\widehat{p}_i) \leq \ell k + 1$.

7. Return $(\widehat{p}_1, \dots, \widehat{p}_\ell)$.

As before, in order to prove the correctness of the above algorithm we postulate the following claims:

Claim 6.8 *A feasible solution to the linear program formulated in step 5 of the algorithm exists.*

Proof. (Of Claim 6.8) We will show that $(\bar{p}_1 = p_1^*, \bar{p}_2, \dots, \bar{p}_\ell = p_\ell^*)$, the Nash equilibrium strategies for T_1, \dots, T_ℓ satisfy the linear program. Since $p_i^{*'} = p_i^* C_{11}$ satisfies $|p_i^{*'}[j] - \widetilde{p}_{mi}^*[j]| \leq \delta$ for $j = 1, \dots, k$ and for $m = [\ell] - \{i\}$, p_i^* satisfies (22). (23) and (24) follow from \bar{p}_i being a probability distribution. So, it suffices to show that p_1^*, \dots, p_ℓ^* satisfy (25), i.e. for any $r \in \text{supp}(p_i^*)$, $r \in \text{supp}(\mathcal{S}_i)$. We will show this to be true for $i = 1$; it symmetrically follows for $i = 2, \dots, \ell$. Note that $|\widetilde{p}_{1i}^*| \leq B$ and $|p_i^* C_{1i} - \widetilde{p}_{1i}^* C_{1i}| \leq \delta$. Then, applying Lemma 6.5 gives us

$$(C_{11}(p_2^* C_{12} \odot \dots \odot p_\ell^* C_{1\ell})^T)_r - \epsilon \leq (C_{11}(\widetilde{p}'_{12} \odot \dots \odot \widetilde{p}'_{1\ell})^T)_r \leq (C_{11}(p_2^* C_{12} \odot \dots \odot p_\ell^* C_{1\ell})^T)_r + \epsilon$$

and likewise,

$$\max_t (C_{11}(\widetilde{p}'_{12} \odot \dots \odot \widetilde{p}'_{1\ell})^T)_t - \epsilon \leq (C_{11}(p_2^* C_{12} \odot \dots \odot p_\ell^* C_{1\ell})^T)_r \leq \max_t (C_{11}(\widetilde{p}'_{12} \odot \dots \odot \widetilde{p}'_{1\ell})^T)_t + \epsilon$$

Combining these two equations,

$$\max_t (C_{11}(\widetilde{p}'_{12} \odot \dots \odot \widetilde{p}'_{1\ell})^T)_t - 2\epsilon \leq (C_{11}(\widetilde{p}'_{12} \odot \dots \odot \widetilde{p}'_{1\ell})^T)_r \leq \max_t (C_{11}(\widetilde{p}'_{12} \odot \dots \odot \widetilde{p}'_{1\ell})^T)_t$$

Hence $r \in \mathcal{S}_1$. \square

Claim 6.9 $(\widehat{p}_1, \dots, \widehat{p}_\ell)$ as returned by the algorithm is a 4ϵ -equilibrium.

Proof. (Of Claim 6.9) For the strategy tuple (p_1, \dots, p_ℓ) and given a decomposition of $T_i = (C_{i1}, \dots, C_{i\ell})$, the payoff to player i given by $T_i(p_1, \dots, p_\ell)$ may be expressed as

$$T_i(p_1, \dots, p_\ell) = p_i C_{ii} (p_1 C_{i1} \odot p_2 C_{i2} \odot \dots \odot p_{i-1} C_{i(i-1)} \odot p_{i+1} C_{i(i+1)} \odot \dots \odot p_\ell C_{i\ell})^T$$

Put this way, we need to show that for any (p'_1, \dots, p'_ℓ) , $\widehat{p}_1, \dots, \widehat{p}_\ell$ returned by the algorithm satisfy

$$\widehat{p}_i C_{ii} (\widehat{p}_1 C_{i1} \odot \dots \odot \widehat{p}_{i-1} C_{i(i-1)} \odot \widehat{p}_{i+1} C_{i(i+1)} \odot \dots \odot \widehat{p}_\ell C_{i\ell})^T \geq p'_i C_{ii} (\widehat{p}_1 C_{i1} \odot \dots \odot \widehat{p}_\ell C_{i\ell})^T - \epsilon$$

for $i = 1, \dots, \ell$. We will prove this for $i = 1$, the proof follows symmetrically for $i = 2, \dots, \ell$. First, we note that from the application of Carathéodory's theorem in step 6,

$$\widehat{p}_1 C_{11} (\widehat{p}_2 C_{12} \odot \dots \odot \widehat{p}_\ell C_{1\ell})^T = \widehat{p}_1 C_{11} (\bar{p}_2 C_{12} \odot \dots \odot \bar{p}_\ell C_{1\ell})^T \quad (26)$$

Then, applying Lemma 6.5

$$\widehat{p}_1 C_{11} (\bar{p}_2 C_{12} \odot \dots \odot \bar{p}_\ell C_{1\ell})^T \geq \widehat{p}_1 C_{11} (\widetilde{p}'_{12} \odot \dots \odot \widetilde{p}'_{1\ell})^T - \epsilon \quad (27)$$

By the same argument as before, since we picked only those r in $\text{supp}(\widehat{p}_1) \subseteq \mathcal{S}_1$ such that

$$(C_{11} (\widetilde{p}'_{12} \odot \dots \odot \widetilde{p}'_{1\ell})^T)_r \geq \max_r (C_{11} (\widetilde{p}'_{12} \odot \dots \odot \widetilde{p}'_{1\ell})^T)_r - 2\epsilon$$

we get

$$\widehat{p}_1 C_{11} (\widetilde{p}'_{12} \odot \dots \odot \widetilde{p}'_{1\ell})^T \geq \max_r C_{11}(r) (\widetilde{p}'_{12} \odot \dots \odot \widetilde{p}'_{1\ell})^T - 2\epsilon \quad (28)$$

Applying Lemma 6.5 and from Carathéodory's Theorem,

$$\begin{aligned} \max_r (C_{11} (\widetilde{p}'_{12} \odot \dots \odot \widetilde{p}'_{1\ell})^T)_r &\geq \max_r (C_{11} (\bar{p}_2 C_{12} \odot \dots \odot \bar{p}_\ell C_{1\ell})^T)_r - \epsilon \\ &= \max_r (C_{11} (\widehat{p}_2 C_{12} \odot \dots \odot \widehat{p}_\ell C_{1\ell})^T)_r - \epsilon \\ &\geq p'_1 C_{11} (\widehat{p}_2 C_{12} \odot \dots \odot \widehat{p}_\ell C_{1\ell})^T - \epsilon \quad \text{for any } p'_1 \end{aligned} \quad (29)$$

Combining (26)–(29),

$$\widehat{p}_1 C_{11} (\widehat{p}_2 C_{12} \odot \dots \odot \widehat{p}_\ell C_{1\ell})^T \geq p'_1 C_{11} (\widehat{p}_2 C_{12} \odot \dots \odot \widehat{p}_\ell C_{1\ell})^T - 4\epsilon \quad \text{for any } p'_1$$

\square

The total number of guesses made in step 2. of the procedure is $k(\ell - 1)\ell$ (k guesses for each component of the $(\ell - 1)$ different vectors in step 2, repeated ℓ times in step 4.). Each component is within $[-B, B]$ and is guessed to δ accuracy, so the guessing takes time

$$(2B/\delta)^{k(\ell-1)\ell} = (2B(k\ell(2B)^{\ell-1})/\epsilon)^{k(\ell-1)\ell} = ((2B)^\ell k\ell/\epsilon)^{k(\ell-1)\ell}$$

For each guess, we solve a linear program, taking time $\text{poly}(|G|)$, for a total running time as claimed. This completes the proof of the theorem. \square

6.3 An example of games with known low-rank tensor decomposition

Many natural games are specified implicitly (rather than by explicitly giving the tensors) by describing the payoff function, which itself is often quite simple. In such cases, the tensor ranks may be significantly smaller than n , and moreover, a low-rank decomposition into components with bounded entries can often be derived from the payoff functions.

One prominent example is simple ℓ -player congestion games as described in [FPT04, Pap05]. Such a game is based on a graph $\mathcal{G}(V, E)$ with n vertices and m edges. Each player's strategy set corresponds to a subset $S_p \subseteq 2^E$, the set of all subsets of edges. We define the payoff accruing to some strategy l -tuple (s_1, \dots, s_ℓ) as $U(s_1, \dots, s_\ell) = -\sum_e c_e(s_1, \dots, s_\ell)$ where $c_e(s_1, \dots, s_\ell) = |\{i | e \in s_i, 1 \leq i \leq \ell\}|$ is thought of as the *congestion* on paths s_1, \dots, s_ℓ . Let $G = (T_1, \dots, T_\ell, N = 2^m)$ be the game corresponding to the situation described above where for $i = 1, \dots, \ell$ and strategy tuple (s_1, \dots, s_ℓ) , $T_i(s_1, \dots, s_\ell) = -\sum_e c_e(s_1, \dots, s_\ell)$.

Theorem 6.10 *For $i = 1, \dots, \ell$ T_i as defined above is of rank at most ℓm . Furthermore, an explicit ℓm -decomposition $(C_{i1}, C_{i2}, \dots, C_{i\ell})$ for T_i exists where C_{ij} are $n \times k$ matrices with entries in $\{-1, 0, 1\}$.*

Proof. In order to give an ℓm -decomposition of T_1 say, we need to construct (C_1, \dots, C_ℓ) where $C_i, i = 1, \dots, \ell$ are $n \times (\ell m)$ matrices. Consider the tensors $\{T_{i,j}\}_{i=1, \dots, \ell, j=1, \dots, m}$ that are described as follows: fix some s_i to be the strategy for player i . For all $e_j \in E(\mathcal{G}), j = 1, \dots, m$ such that $e_j \in s_i$ let

$$T_{i,j}(s'_1, \dots, s'_{i-1}, s_i, s'_{i+1}, \dots, s'_\ell) = -1$$

where $s'_1, \dots, s'_{i-1}, s'_{i+1}, \dots, s'_\ell \in 2^E$. Then, $T_1 = \sum_{i=1}^\ell \sum_{j=1}^m T_{i,j}$. This follows easily since each tuple (s_1, \dots, s_ℓ) in some $T_{i,j}$ contributes -1 to $T_1(s_1, \dots, s_\ell)$ iff $e_j \in s_i$. Summing over all $T_{i,j}$ for $i = 1, \dots, \ell$, e_j contributes exactly $-c_{e_j}(s_1, \dots, s_\ell)$ to $T_1(s_1, \dots, s_\ell)$ from the definition of $c_{e_j}(s_1, \dots, s_\ell)$ above. Summing over all $j = 1, \dots, m$ we obtain the total contribution from all the edges. Next, we claim that each $T_{i,j}$ is a rank-1 tensor. Indeed, $T_{i,j} = v_{1,j} \otimes \dots \otimes v_{\ell,j}$ where $v_{k,j} = 1_N$ the all-ones $N \times 1$ column vector for $k = 1, \dots, i-1, i+1, \dots, \ell$ and $v_{i,j}$ is the $N \times 1$ column vector given by:

$$v_{i,j}[k] = \begin{cases} -1 & e_j \in s_k; \\ 0 & \text{otherwise.} \end{cases}$$

for $k = 1, \dots, N$. \square

7 Conclusions

There are many other interesting questions that are raised by viewing game theory through the lens of requiring players to be randomness-efficient. In this paper, we have framed some of the initial questions that arise and have provided answers to several of them. In particular, we have exploited the extensive body of work in derandomization to construct deterministic algorithms for finding sparse ϵ -equilibria (which can be played with limited randomness), and for playing repeated games while reusing randomness across rounds. The efficient fixed-parameter algorithms we describe for finding ϵ -equilibria in games of small rank significantly

improve over the standard enumeration algorithm, and to the best of our knowledge, they are the first such results for games of small rank.

The notion of *resource-limited* players has been an extremely useful one in game theory, and we think that it is an interesting and natural question in this context to consider the case in which the limited computational resource is *randomness*. These considerations expose a rich and largely untapped area straddling complexity theory and game theory.

References

- [AGHP92] N. Alon, O. Goldreich, J. Hastad, and R. Peralta. Simple constructions of almost k -wise independent random variables. *Random Structures and Algorithms*, (3):289–304, 1992.
- [CD05] X. Chen and X. Deng. 3-NASH is PPAD-complete. *Electronic Colloquium on Computational Complexity (ECCC)*, (134), 2005.
- [CD06] X. Chen and X. Deng. Settling the Complexity of Two-Player Nash Equilibrium. In *Foundations of Computer Science (FOCS 2006)*, pages 261–272. IEEE Computer Society, 2006.
- [CDT06] X. Chen, X. Deng, and S.-H. Teng. Computing Nash Equilibria: Approximation and Smoothed Complexity. In *Foundations of Computer Science (FOCS 2006)*, pages 603–612. IEEE Computer Society, 2006.
- [Ces05] M. Cesati. Compendium of parameterized problems. <http://bravo.ce.uniroma2.it/home/cesati/research/compendium/>, 2005.
- [DF99] R.G. Downey and M.R. Fellows. *Parameterized Complexity*. Springer-Verlag, 1999.
- [DFS97] R.G. Downey, M.R. Fellows, and U. Stege. Parameterized complexity: A framework for systematically confronting computational intractability. In *Proceedings of the First DIMATIA Symposium*, 1997.
- [DGP06] C. Daskalakis, P.W. Goldberg, and C.H. Papadimitriou. The complexity of computing a Nash equilibrium. In *Symposium on Theory of Computing (STOC 2006)*, pages 71–78, 2006.
- [DP05] C. Daskalakis and C.H. Papadimitriou. Three-player games are hard. *Electronic Colloquium on Computational Complexity (ECCC)*, (139), 2005.
- [FPT04] A. Fabrikant, C.H. Papadimitriou, and K. Talwar. The complexity of pure-strategy equilibria. In *Symposium on Theory of Computing (STOC 2004)*, pages 604–612, 2004.
- [FS96] Y. Freund and R. Schapire. Game theory, on-line prediction and boosting. In *COLT 1996*, pages 325–332, 1996.
- [FS99] Y. Freund and R. Schapire. Adaptive game playing using multiplicative weights. *Games and Economic Behavior*, 29:79–103, 1999.
- [Gil93] D. Gillman. A Chernoff bound for random walks on expander graphs. In *Foundations of Computer Science (FOCS 1993)*, pages 680–691. IEEE, 1993.
- [GP06] P.W. Goldberg and C.H. Papadimitriou. Reducibility among equilibrium problems. In *Symposium on Theory of Computing (STOC 2006)*, pages 61–70, 2006.
- [KT07] R. Kannan and T. Theobald. Games of fixed rank: A hierarchy of bimatrix games. In *ACM-SIAM Symposium on Discrete Algorithms*, 2007.
- [LMM03] R.J. Lipton, E. Markakis, and A. Mehta. Playing large games using simple strategies. In *EC '03: Proceedings of the 4th ACM conference on Electronic commerce*, pages 36–41, New York, NY, USA, 2003. ACM Press.

- [LY94] R.J. Lipton and N.E. Young. Simple strategies for large zero-sum games with applications to complexity theory. In *STOC '94: Proceedings of the twenty-sixth annual ACM symposium on Theory of computing*, pages 734–740, New York, NY, USA, 1994. ACM Press.
- [MvN44] O. Morgenstern and J. von Neumann. *Theory of Games and Economic Behavior*. Princeton University Press, 1944.
- [Nas51] J.F. Nash. Non-cooperative games. *Annals of Mathematics*, 54:286–295, 1951.
- [Pap05] C.H. Papadimitriou. Computing Correlated Equilibria in Multi-Player Games. In *Proceedings of the 37th Annual ACM Symposium on Theory of Computing*, pages 49–56, May 2005.
- [Vav92] S.A. Vavasis. Approximation algorithms for indefinite quadratic programming. *Math. Program.*, 57:279–311, 1992.