The Oracle Complexity of Simplex-based Matrix Games: Linear Separability and Nash Equilibria

Guy Kornowski GUY.KORNOWSKI@WEIZMANN.AC.IL
Ohad Shamir OHAD.SHAMIR@WEIZMANN.AC.IL
Weizmann Institute of Science

Editors: Nika Haghtalab and Ankur Moitra

Abstract

We study the problem of solving matrix games of the form $\max_{\mathbf{w}\in\mathcal{W}} \min_{\mathbf{p}\in\Delta} \mathbf{p}^\top A\mathbf{w}$, where A is some matrix and Δ is the probability simplex. This problem encapsulates canonical tasks such as finding a linear separator and computing Nash equilibria in zero-sum games. However, perhaps surprisingly, its inherent complexity (as formalized in the standard framework of oracle complexity (Nemirovski and Yudin, 1983)) is not well-understood. In this work, we first identify different oracle models which are implicitly used by prior algorithms, amounting to multiplying the matrix A by a vector from either one or both sides. We then prove complexity lower bounds for algorithms under both access models, which in particular imply a separation between them. Specifically, we start by showing that algorithms for linear separability based on one-sided multiplications must require $\Omega(\gamma_A^{-2})$ iterations, where γ_A is the margin, as matched by the Perceptron algorithm. We then prove that accelerated algorithms for this task, which utilize multiplications from both sides, must require $\tilde{\Omega}(\gamma_A^{-2/3})$ iterations, establishing the first oracle complexity barrier for such algorithms. Finally, by adapting our lower bound to ℓ_1 geometry, we prove that computing an ϵ-approximate Nash equilibrium requires $\tilde{\Omega}(\epsilon^{-2/5})$ iterations, which is an exponential improvement over the previously best-known lower bound due to Hadiji et al. (2024).

Keywords: Matrix games, linear separability, Nash equilibrium, oracle complexity, lower bounds

1. Introduction

Given a matrix $A \in \mathbb{R}^{n\times d}$ and some domain $\mathcal{W} \subset \mathbb{R}^d$, we consider the optimization problem (also known as a matrix game)

$$\max_{\mathbf{w}\in\mathcal{W}} \min_{\mathbf{p}\in\Delta^{n-1}} \mathbf{p}^\top A\mathbf{w} \;=\; \max_{\mathbf{w}\in\mathcal{W}} \min_{l\in\{1,\dots,n\}} (A\mathbf{w})_l \,, \tag{1}$$

where $\Delta^{n-1} := \{\mathbf{p} \in \mathbb{R}^n : \min_i p_i \geq 0, \sum_{i=1}^n p_i = 1\}$ is the probability simplex. We denote the optimal value of this problem as γ_A. The problem of finding $\mathbf{w} \in \mathcal{W}$ that approximates the optimum of such problems is extensively studied throughout machine learning, statistics, optimization and economics, as several important problems take this form, depending on the choice of the set $\mathcal{W}$. We discuss two prominent cases:

1. **Linear separability:** When $\mathcal{W} = \{\mathbf{w} \in \mathbb{R}^d : \|\mathbf{w}\|_2 \leq 1\}$ is the unit Euclidean ball, (1) corresponds to the canonical problem of finding a linear separator (namely, a vector $\mathbf{w}$ such that $A_l\mathbf{w} > 0$ for all rows A_l of A), and the optimal value γ_A is known as the *margin*. This is a fundamental classification and linear programming problem which can be dated back to the work of McCulloch and Pitts (1943), and is solved for instance by the well-known Perceptron algorithm (Rosenblatt, 1958).

2. **Nash equilibria:** When $\mathcal{W} = \Delta^{d-1} \subset \mathbb{R}^d$ is the simplex, (1) corresponds to maximizing the utility of a player in a zero-sum bilinear game, and γ_A is known as the game's *value*. Due to the minimax theorem, by symmetrically solving this problem for each of the players (namely, $\mathbf{p}$ and $\mathbf{w}$), this objective is equivalent to the canonical problem of finding a Nash equilibrium, or saddle point, in zero-sum matrix games (Nash, 1950).

Perhaps surprisingly, although these are canonical problems with a long history (see Section 3), the inherent complexity of solving them is relatively little studied. The goal of this paper is to study this question through the standard framework of oracle complexity (Nemirovski and Yudin, 1983). Specifically, we are interested in the performance limits of iterative algorithms, where each iteration is based on a simple computation involving the matrix A, such as multiplying it by some vector or extracting a row of A. This interaction between the algorithm and the matrix can be modeled as accessing an oracle, which simulates this computation and provides the algorithm with the result. We can then ask and rigorously study how many such oracle queries/computations are required, so that an algorithm with no prior knowledge of A will solve the associated matrix problem up to a given level of precision (see Section 2 for more details). In this paper, we focus on the high-dimensional setting, where the size of the matrix A is essentially unrestricted, and we are interested in bounds which are independent of (or only weakly dependent on) the matrix size.

Considering linear separability, it is well known that for matrices with normalized rows, the Perceptron algorithm finds a linear separator using $O(\gamma_A^{-2})$ iterations (each involving a single matrix-vector multiplication and extraction of a single row of A), independently of n, d (Novikoff, 1962). Half a century later, Soheili and Pena (2012); Yu et al. (2014) used acceleration techniques due to Nesterov (2005a); Nemirovski (2004) respectively, and provided accelerated algorithms which find a separator using only $O(\sqrt{\log n} \cdot \gamma_A^{-1})$ iterations. However, a closer inspection of these methods reveal that they rely on a stronger oracle access to the matrix A, amounting to multiplying A by vectors *on both sides* (instead of just one-sided multiplications as required by the Perceptron algorithm). Thus, one may ask whether such two-sided operations are necessary for these accelerated results, and how close they are to being optimal.

As for approximating Nash equilibria, the best known algorithms return an ϵ-suboptimal solution[1] using $O(\sqrt{\log(n)\log(d)} \cdot \epsilon^{-1})$ two-sided matrix multiplication queries, a rate achieved by several different algorithms in the literature (Nemirovski, 2004; Nesterov, 2005b, 2007; Rakhlin and Sridharan, 2013). Corresponding lower bounds were missing, and only recently Hadiji et al. (2024) proved a lower bound of $\Omega(\log(1/n\epsilon))$ for sufficiently small $\epsilon = \mathrm{poly}(1/n)$ and $n = d$.

Our contributions. In this work, we study the oracle complexity of solving matrix games involving the simplex as in (1). As mentioned earlier, we focus on the high-dimensional regime, where the bounds should be independent (or at least not polynomial) in the matrix size n, d, and the matrix A satisfies suitable magnitude constraints. Our contributions can be summarized as follows.

- **One-sided vs. two-sided oracles (Section 3:)** We start by identifying and formalizing different oracle models for matrix games that are implicitly used by existing algorithms, as we will see that these oracles lead to different complexities. One oracle model corresponds to querying rows of A,

1. More precisely, these algorithms guarantee returning an ϵ-approximate saddle point, namely $(\hat{\mathbf{w}}, \hat{\mathbf{p}}) \in \Delta^{d-1} \times \Delta^{n-1}$ such that $\max_{\mathbf{w}\in\Delta^{d-1}} \hat{\mathbf{p}}^\top A\mathbf{w} - \min_{\mathbf{p}\in\Delta^{n-1}} \mathbf{p}^\top A\hat{\mathbf{w}} \leq \epsilon$. This implies ϵ-suboptimality, since $\max\{\max_{\mathbf{w}\in\Delta^{d-1}} \hat{\mathbf{p}}^\top A\mathbf{w} - \gamma_A,\ \gamma_A - \min_{\mathbf{p}\in\Delta^{n-1}} \mathbf{p}^\top A\hat{\mathbf{w}}\} \leq \epsilon$ where γ_A is the optimal value. Since the latter inequality also implies the former (with 2ϵ replacing ϵ), we see that the two notions are the same up to a factor of 2.

together with "one-sided" multiplication queries of the form $\mathbf{w} \mapsto A\mathbf{w}$. The second (and stronger) oracle model we will consider allows "two-sided" multiplications $(\mathbf{p}, \mathbf{w}) \mapsto (\mathbf{p}^\top A, A\mathbf{w})$.

- **Linear separability with a one-sided oracle (Theorem 6):** We first show that any deterministic algorithm that performs row queries and one-sided multiplication queries, must require $\Omega(\gamma_A^{-2})$ queries in the worst-case in order to find a linear separator. The claim is proved using a classic lower bound technique due to Nemirovski and Yudin (1983). In particular, this establishes the optimality of the Perceptron algorithm under this oracle model.

- **Linear separability with a two-sided oracle (Theorem 7):** We prove that any deterministic algorithm which performs two-sided multiplication queries, must require $\tilde{\Omega}(\gamma_A^{-2/3})$ queries in the worst-case in order to find a linear separator (where the $\tilde{\Omega}$ hides a logarithmic dependence on the matrix size). To the best of our knowledge, this is the first oracle complexity lower bound for linear separability, which applies even to accelerated algorithms. Compared to the lower bound for the one-sided oracle, the proof of the lower bound for the two-sided oracle is substantially more involved, and requires some new proof ideas.

- **Nash equilibria with a two-sided oracle (Theorem 15):** We prove that under the two-sided oracle model, any deterministic algorithm requires $\tilde{\Omega}(\epsilon^{-2/5})$ oracle calls in order to find an ϵ-suboptimal strategy. Hence, the same lower bound holds for finding an ϵ-approximate Nash equilibrium, which is an exponential improvement over the previously known lower bound by Hadiji et al. (2024). The proof is based on an adaptation of the linear separability lower bound technique to an ℓ_1 geometry. Along the way, we also prove an identical lower bound for ℓ_1/simplex matrix games, where $\mathcal{W}$ in Eq. (1) is the unit ℓ_1 ball.

We conclude with a discussion and directions for future work in Section 6.

2. Preliminaries

Notation. We use capital letters to denote matrices, and bold-face letters to denote vectors. Vectors are always in column form. Given an indexed vector $\mathbf{v}_t$, $v_{t,i}$ denotes its i-th entry. $\mathbf{e}_i$ denotes the i-th standard basis vector. $\mathbf{1}$ is the all-ones vector. Given a matrix A, A_l refers to its l-th row. $\|\cdot\|$ refers to the Euclidean norm $\|\cdot\|_2$, $\|\cdot\|_p$ refers to the ℓ_p norm, and $\log(\cdot)$ refers to the natural logarithm, unless specified otherwise. Finally, $[n]$ is shorthand for $\{1, \ldots, n\}$.

Oracle Complexity. As described in the introduction, we study the complexity of matrix games via the well-known optimization-theoretic frameork of oracle complexity (Nemirovski and Yudin, 1983). This framework focuses on iterative methods, where each iteration utilizes some restricted information about the relevant objective function. In our context of matrix games as in Eq. (1), we assume that the domain $\mathcal{W}$ is fixed and known, and that the matrix A is known to belong to some set $\mathscr{A}$: For linear separability, in light of existing methods and upper bounds, it is natural to consider all $n \times d$ matrices whose rows have Euclidean norm at most 1. Similarly, for Nash equilibrium, it is natural to consider all $n \times d$ matrices whose entries have values in $[-1, +1]$. Crucially, the algorithm has no additional prior knowledge of A. In order to solve the matrix problem, the algorithm has access to an oracle $\mathcal{O}(\cdot)$ which provides some limited information about A: For example, given the vectors $\mathbf{p} \in \mathbb{R}^n, \mathbf{w} \in \mathbb{R}^d$ chosen by the algorithm, the oracle returns $\mathbf{p}^\top A$ and $A\mathbf{w}$. The algorithm interacts with the oracle for a given number of iterations, after which it returns an output

as a function of all previous oracle responses. One can then ask how many iterations are required by some algorithm, for the output to satisfy a certain performance metric for all $A \in \mathscr{A}$ (as a function of the problem parameters $n, d, \mathscr{A}, \mathcal{W}$ and the type of oracle $\mathcal{O}(\cdot)$). This framework naturally models standard scalable approaches for solving matrix games, and allows one to prove both upper bounds and unconditional lower bounds for iterative algorithms, assuming they interact with the matrix A in a manner corresponding to the given oracle.

Remark 1 *In this paper, we focus on deterministic algorithms, whose oracle queries and output are deterministic function of the previous oracle responses. Since all state-of-the-art algorithms for the problems we consider are deterministic, this is without too much loss of generality. However, extending our lower bounds to randomized algorithms is certainly an interesting direction for future work (see Section 6 for more details).*

3. Oracle models

To motivate the oracles that we consider, let us begin by examining the classical Perceptron algorithm (Rosenblatt, 1958) for linear separability: Given the matrix A (and assuming that a linear separator exists), the algorithm iteratively searches for a row A_l of A such that $A_l \mathbf{w} < 0$ (where $\mathbf{w}$ is the current iterate), and then adds A_l to $\mathbf{w}$. This process is repeated until no such rows are found. It is well-known that this algorithm will terminate in at most $O(\gamma_A^{-2})$ iterations, resulting in a linear separator $\mathbf{w}$ such that $\min_{l\in[n]}(A\mathbf{w})_l > 0$. From an oracle complexity perspective, each iteration of the algorithm can be modeled via two operations on A: One is a right matrix-vector multiplication $\mathbf{w} \mapsto A\mathbf{w}$, and the second is the extraction of a row of A whose inner product with $\mathbf{w}$ is negative. We can formally model these operations via the following oracle:

Definition 2 (One-sided Oracle $\mathcal{O}_1^A$) *Given some $\mathbf{w} \in \mathbb{R}^d$ and index $l \in [n]$, the oracle $\mathcal{O}_1^A(l, \mathbf{w})$ returns $A\mathbf{w}$ and A_l.*

Thus, the Perceptron's convergence guarantee implies that $O(\gamma_A^{-2})$ queries to a one-sided oracle is sufficient for finding a linear separator. In fact, a more general result can be achieved by applying the well-known subgradient method on the equivalent convex problem $\min_{\mathbf{w}:\|\mathbf{w}\|\le 1} \max_{\mathbf{p}\in\Delta} \mathbf{p}^\top(-A)\mathbf{w}$. Specifically, by standard results, this method requires $O(1/\epsilon^2)$ subgradient computations in order to find a vector $\mathbf{w}$ whose value is ϵ-suboptimal (Nesterov, 2018). This holds for any ϵ, whereas the guarantee for the Perceptron is merely for the special case $\epsilon = \gamma_A$ (namely, we seek a solution whose value is > 0, with the optimal value being γ_A). From an oracle-complexity perspective, implementing such methods requires access to supergradients of the function $f(\mathbf{w}) = \min_{\mathbf{p}\in\Delta^{n-1}} \mathbf{p}^\top A\mathbf{w} = \min_{l\in[n]}(A\mathbf{w})_l$, which equal $A_{l_{\min}}$ where $l_{\min} \in \arg\min_{l\in[n]}(A\mathbf{w})_l$. Thus, we can model these methods as iteratively interacting with the following supergradient oracle:

Definition 3 (Supergradient Oracle $\mathcal{O}_\partial^A$) *Given some $\mathbf{w} \in \mathbb{R}^d$, the oracle $\mathcal{O}_\partial^A(\mathbf{w})$ returns $A_{l_{\min}}$ where $l_{\min} \in \arg\min_{l\in[n]}(A\mathbf{w})_l$.*

Note that this oracle is strictly weaker than a one-sided oracle (up to a factor of 2): On the one hand, we can simulate each call to a supergradient oracle by two calls to a one-sided oracle. On the other hand, a one-sided oracle allows us to extract any single row of A, and not just the one corresponding to the smallest entry in $A\mathbf{w}$. In what follows, we will prove lower bounds for any algorithm based

on a one-sided oracle, and thus the lower bounds automatically extend to any algorithm based on a supergradient oracle.

As we discussed earlier, works such as Soheili and Pena (2012) and Yu et al. (2014) show that the $O(\gamma_A^{-2})$ iteration bound of the Perceptron algorithm can actually be improved. In a nutshell, this is achieved by applying accelerated gradient methods on top of a smoothing of the objective function (using, for example, the log-sum-exp function instead of a hard max). These result in bounds of the form $O(\sqrt{\log(n)}/\gamma_A)$ for matrices with margin parameter γ_A, or more generally, $O(\sqrt{\log(n)}/\epsilon)$ iterations to get an ϵ-optimal solution. Accelerated methods to maximize the margin were also proposed in (Ji et al., 2021; Wang et al., 2023).

Why can these accelerated methods beat the Perceptron bound, from an oracle-complexity perspective? A close inspection of these methods reveal that they all actually require a stronger oracle than a supergradient (or even one-sided) oracle: They crucially require *two-sided* matrix-vector multiplications. In more detail, accelerated gradient methods can optimize convex functions with L-Lipschitz gradients to suboptimality ϵ with $O(\sqrt{L/\epsilon})$ gradient computations. Moreover, for any ϵ, the $\min$ (or $\max$) operator can be approximated to accuracy ϵ using a smooth function $\tilde{f}$ with $(\log(n)/\epsilon)$-Lipschitz gradients. Combining these two observations, it follows that one can optimize the original matrix problem to accuracy 2ϵ, using $O(\sqrt{(\log(n)/\epsilon)/\epsilon}) = O(\sqrt{\log(n)}/\epsilon)$ gradient computations of the function $\mathbf{w} \mapsto \tilde{f}(A\mathbf{w})$. The gradient of this function is given by $A^\top \tilde{f}'(A\mathbf{w})$, so its computation requires multiplying the matrix A from both the left and from the right. Thus, we are led to the following natural *two-sided* oracle model:

Definition 4 (Two-sided Oracle $\mathcal{O}_2^A$) *Given some $\mathbf{p} \in \mathbb{R}^n, \mathbf{w} \in \mathbb{R}^d$, the oracle $\mathcal{O}_2^A(\mathbf{p}, \mathbf{w})$ returns $A\mathbf{w}$ and $\mathbf{p}^\top A$.*

Clearly, a two-sided oracle is stronger than one-sided, since $\mathcal{O}_1^A(l, \mathbf{w})$ can be simulated by $\mathcal{O}_2^A(\mathbf{e}_l, \mathbf{w})$. As far as we know, all existing accelerated algorithms for linear separability can be implemented with such an oracle, so any lower bound for algorithms based on this oracle will apply to them. We also note that lower bounds with respect to two-sided oracles were studied in the context of solving linear equations (Nemirovsky, 1992), which does not directly relate to our results or proofs.

Although we have considered so far the linear separability problem, a two-sided oracle is also very natural to model algorithms for other matrix games. In particular, for computing a Nash equilibrium, a two-sided oracle corresponds precisely to a first-order (or gradient) oracle, which given $\mathbf{w}, \mathbf{p}$, returns the gradient of the function $(\mathbf{w}, \mathbf{p}) \mapsto \mathbf{p}^\top A\mathbf{w}$ (namely $\mathbf{p}^\top A$ and $A\mathbf{w}$). Under this well-studied setting, the best-known method dating back to Nemirovski (2004) requires $O(\sqrt{\log(n)\log(d)}/\epsilon)$ two-sided oracle calls in order to find an ϵ-optimal solution (see also Nesterov (2005b, 2007); Rakhlin and Sridharan (2013) and Hadiji et al. (2024) for a detailed discussion of other results). However, very little work appears to exist on lower bounds, with the notable exception of Hadiji et al. (2024), whose bound is $\Omega(\log(1/(n\epsilon)))$ – namely, logarithmic in $1/\epsilon$ – and when $n = d$.

Remark 5 *In all oracle definitions, we do not restrict the input $\mathbf{w}$ to lie in $\mathcal{W}$, nor $\mathbf{p}$ to lie in Δ^{n-1}. Thus, our lower bounds will apply equally to algorithms which can query outside these domains.*

Additional Related work

Dimension-dependent oracle complexity. Besides the accelerated algorithms for linear separability discussed earlier (with convergence rate $O(\log(n) \cdot \epsilon^{-1})$), there exists another family of

rescaling-based methods, which can achieve an even faster (logarithmic) dependence on ϵ^{-1}, but with iteration bounds scaling polynomially with the matrix dimensions (Dunagan and Vempala, 2004; Belloni et al., 2009; Pena and Soheili, 2016; Dadush et al., 2020). This is akin to the situation in convex optimization, where one can trade off between algorithms with dimension-independent, poly(ϵ^{-1})-dependent guarantees (using gradient or subgradient methods), and algorithms with poly-dimension-dependent, $\log(\epsilon^{-1})$-dependent guarantees (using methods such as interior points, ellipsoids, or center-of-gravity, see Nesterov, 2018). Similarly, several works developed algorithms for general matrix games that reduce the ϵ-dependence at the cost of polynomial dependencies on n, d, thus improving size-independent algorithms in certain parameter regimes (Carmon et al., 2019, 2024). Since our focus is on size-independent (or near independent) bounds, these family of methods are orthogonal to our work, although understanding the ultimate limits in those regimes is interesting as well.

Other oracle models. Going beyond matrix games, there exist quite a few oracle complexity lower bounds for more general minmax convex-concave optimization problems (e.g., Ibrahim et al., 2020; Ouyang and Xu, 2021), but the constructions do not apply to matrix games as we consider. Carmon et al. (2020b) considered the oracle complexity of optimizing the maximum of several linear functions using a ball oracle, which returns the optimum in a small ball around a given point. Although the structure of the objective function is closely related to ours, the oracle is different than the one we consider here (as far as we can surmise). Moreover, their lower bound construction requires non-homogeneous linear functions, which are not included in the matrix game settings that we consider. Clarkson et al. (2012) provide an $\Omega(\gamma_A^{-2})$ oracle complexity lower bound for linear separability, but for a weaker oracle which only returns individual matrix entries.

4. Linear separability: ℓ_2/simplex games

In this section, we assume that $\mathcal{W}$ is the unit Euclidean ball in $\mathbb{R}^d$, so the problem of interest is

$$\max_{\mathbf{w}\in\mathbb{R}^d:\|\mathbf{w}\|_2\leq 1} \min_{\mathbf{p}\in\Delta^{n-1}} \mathbf{p}^\top A\mathbf{w} \;=\; \max_{\mathbf{w}\in\mathbb{R}^d:\|\mathbf{w}\|_2\leq 1} \min_{l\in[n]} (A\mathbf{w})_l \;. \tag{2}$$

As previously discussed, this corresponds to the canonical problem of finding a max-margin linear separator for a dataset comprised of A's rows.

4.1. Oracle complexity with a one-sided oracle

We begin by formally showing that any algorithm using a one-sided oracle for T iterations cannot find a linear separator (a vector $\mathbf{w}$ such that $A\mathbf{w}$ has only positive entries), if the margin parameter is less than $\Omega(1/\sqrt{T})$. Since a supergradient oracle is weaker than a one-sided oracle, the same result automatically applies to any algorithm based on a supergradient oracle.

Theorem 6 *Suppose $d > 2T+1$. Then for any deterministic algorithm for solving Eq. (2), there exists a $(T+1)\times d$ matrix A satisfying*

$$\max_{l\in[T+1]} \|A_l\| = 1 \quad \textit{and} \quad \max_{\mathbf{w}:\|\mathbf{w}\|\leq 1} \min_{\mathbf{p}\in\Delta^{n-1}} \mathbf{p}^\top A\mathbf{w} \;\geq\; \frac{1}{\sqrt{T+1}} \;,$$

yet after T rounds of interaction with a one-sided oracle $\mathcal{O}_1^A$, the algorithm returns a vector $\mathbf{w}_{T+1}$ such that $\min_{\mathbf{p}\in\Delta^{n-1}} \mathbf{p}^\top A\mathbf{w}_{T+1} \leq 0$.

By fixing some $\gamma_A > 0$ and setting $T = \lceil \gamma_A^{-2} \rceil$, we can restate this result as follows: For any γ_A, and for any algorithm whose interaction with A is captured by a one-sided oracle, there exists a matrix A (with unit rows and margin parameter at least γ_A) such that the required number of iterations to find a linear separator is $\Omega(\gamma_A^{-2})$, as claimed in the introduction.

We emphasize that the proof of this particular theorem is a rather straightforward adaptation of existing techniques, and its main purpose is to complete the picture regarding the power of different oracles to solve this matrix game. Nevertheless, the proof provided below illustrates the iterative nature of such constructions, an idea which is also used in the proofs of our other results.

Proof [of Theorem 6] The proof is directly inspired by standard oracle complexity lower bounds for convex Lipschitz optimization due to Nemirovski and Yudin (1983), as well as the standard lower bound proof of the Perceptron algorithm (cf. Shalev-Shwartz and Ben-David, 2014, section 9.5, question 3). The main idea is to construct A's row as mutually orthogonal unit vectors, which are also orthogonal to the algorithm's queries $\mathbf{w}_t$. Therefore, the algorithm can recover the rows one at a time, by querying for a particular row. However, if the number of rows is larger than T, there will remain rows orthogonal to $\mathbf{w}_{T+1}$. Therefore, $\mathbf{w}_{T+1}$ will not be a linear separator for A. On the other hand, since A has $T+1$ orthogonal rows, it can be shown to be linearly separable with margin $1/\sqrt{T+1}$.

More formally, given an algorithm $\mathcal{A}$, consider the following iterative construction:

- Initialize $\mathcal{L}_0 = \emptyset$, and A_0 to be the the all-zeros $(T+1) \times d$ matrix.
- For $t = 1, 2, \ldots, T$:
  - Compute algorithm queries $\mathbf{w}_t, l_t$ (based on oracle outputs received so far).
  - Set $A_t := A_{t-1}$ and $\mathcal{L}_t := \mathcal{L}_{t-1}$.
  - If $l_t \notin \mathcal{L}_{t-1}$
    - Set the l_t-th row of A_t to be some arbitrary unit vector, orthogonal to $\mathbf{w}_1, \ldots, \mathbf{w}_t$ as well the rows of A_t indexed by $\mathcal{L}_t$.
    - Set $\mathcal{L}_t := \mathcal{L}_{t-1} \cup \{l_t\}$.
  - Feed $\mathcal{A}$ with $A_t \mathbf{w}_t$ and with the l_t-th row of A_t (as a response for its queries $\mathbf{w}_t, l_t$).
- Compute algorithm output $\mathbf{w}_{T+1}$ (based on oracle outputs so far).
- Set $A = A_T$, and set all rows $l \notin \mathcal{L}_T$ of A to be some mutually orthogonal unit vectors which are also orthogonal to $\mathbf{w}_1, \ldots, \mathbf{w}_{T+1}$.

We note that since $2T + 1 < d$, the dimensionality is sufficiently high to find orthogonal unit vectors as specified. Moreover, it is easy to verify that because of the orthogonality, it holds that $A_t \mathbf{w}_t = A \mathbf{w}_t$ for all $t \in [T]$: Namely, the responses given to the algorithm are consistent with the oracle responses on the matrix A. However, after T iterations, $|\mathcal{L}_T| \leq T$, yet there are $T+1$ rows. Therefore, at least one row of A will be chosen to be orthogonal to $\mathbf{w}_1, \ldots, \mathbf{w}_{T+1}$, and in particular to $\mathbf{w}_{T+1}$. Hence, $A\mathbf{w}_{T+1}$ contains a 0 entry, so $\min_{\mathbf{p} \in \Delta^{n-1}} \mathbf{p}^\top A \mathbf{w}_{T+1} \leq 0$.

On the other hand, the unit vector $\mathbf{w} := \frac{1}{\sqrt{T+1}} \sum_{l=1}^{T+1} A_l$ (where A_l is the l-th row of A, with the rows being mutually orthogonal unit vectors) satisfies $A\mathbf{w} = \frac{1}{\sqrt{T+1}} \cdot \mathbf{1}$, and therefore, $\max_{\mathbf{w}:\|\mathbf{w}\| \leq 1} \min_{\mathbf{p} \in \Delta^{n-1}} \mathbf{p}^\top A \mathbf{w} \geq \frac{1}{\sqrt{T+1}}$. ■

4.2. Oracle complexity with a two-sided oracle

We now turn to an oracle complexity lower bound for the much stronger two-sided oracle $\mathcal{O}_2^A$:

Theorem 7 *The following holds for some large enough universal constant $c > 0$: For any $T > c$, suppose d, n are sufficiently large so that $d > cT$ and $n > cT^2 \log(T)$. Then for any deterministic algorithm for solving Eq. (2), there exists an $n \times d$ matrix A satisfying*

$$\max_{l\in[n]} \|A_l\| \leq 1 \quad \textit{and} \quad \max_{\mathbf{w}:\|\mathbf{w}\|\leq 1} \min_{\mathbf{p}\in\Delta^{n-1}} \mathbf{p}^\top A\mathbf{w} \;\geq\; \frac{1}{cT\sqrt{T\log(n)}} \,,$$

yet after T rounds of interaction with the two-sided oracle $\mathcal{O}_2^A$, the algorithm returns a vector $\mathbf{w}_{T+1}$ such that $\min_{\mathbf{p}\in\Delta^{n-1}} \mathbf{p}^\top A\mathbf{w}_{T+1} \leq 0$.

As before, we can state this lower bounds in terms of a margin parameter γ_A: Given any small enough γ_A, we can choose T such that $1/(cT\sqrt{T\log(n)}) \geq \gamma_A$, and get that for any algorithm, there exists an $n \times d$ matrix A (for large enough n, d) with margin parameter at least γ_A and rows of norm at most 1, such that the required number of iterations T to find a linear separator is $\Omega(\log^{-1/3}(n) \cdot \gamma_A^{-2/3})$.

We now turn to discuss the proof of Theorem 7. We begin by noting that the proof technique of Theorem 6 is of no use here, as it is based on revealing the rows of A to the algorithm one at a time. This is not possible with a two-sided oracle, which allows us (for instance) to compute an arbitrary weighted combination of all rows of A using a single query. A second difficulty is that we consider a very simple class of homogeneous bilinear functions, which means that any lower bound necessarily has to be of this form (as opposed to more general min-max optimization problems, where for lower bounds we can use functions with a richer structure). To handle these difficulties, we introduce a different proof technique, which still forces the algorithm to discover information about A in an incremental manner, but in terms of certain vector outer products rather than rows. Specifically, we will utilize the following randomized construction of A:

Construction 8 *Given an iteration bound $T \geq 1$ such that $2T + 1 \leq \min\{n, d\}$, positive parameters α, β, and an algorithm $\mathcal{A}$ interacting with a two-sided oracle for T iterations, the matrix A is defined as*

$$A \;=\; \sum_{j=1}^{T} (\mathbf{v}_{j-1} - \mathbf{v}_j)\mathbf{u}_j^\top + \mathbf{v}_T\mathbf{u}_{T+1}^\top \,,$$

where $\forall j$, $\mathbf{v}_j \in \mathbb{R}^n$, $\mathbf{u}_j \in \mathbb{R}^d$, and these vectors are constructed iteratively as follows:

- *Let $\mathbf{v}_0 = \alpha\mathbf{1}$ (where $\mathbf{1}$ is the all-ones vector).*
- *For $t = 1, 2, \ldots, T + 1$:*
  - *Compute algorithm queries $\mathbf{p}_t, \mathbf{w}_t$ (based on oracle outputs received so far).*
  - *Let $\mathbf{u}_t$ be some unit vector in $\mathbb{R}^d$, which is orthogonal to $\mathbf{u}_1, \ldots, \mathbf{u}_{t-1}$ and to $\mathbf{w}_1, \ldots, \mathbf{w}_t$.*
  - *Let $\mathbf{v}_t = \beta(I - M_t M_t^\top)\boldsymbol{\xi}_t$, where (1) the columns of $M_t \in \mathbb{R}^{n\times s_t}$, $s_t \leq 2t$ are an orthogonal basis for $\mathbf{v}_0, \ldots, \mathbf{v}_{t-1}, \mathbf{p}_1, \ldots, \mathbf{p}_t$, and (2) $\boldsymbol{\xi}_t$ is an independent standard Gaussian random vector in $\mathbb{R}^n$.*

– *Let* $A_t = \sum_{j=1}^{t}(\mathbf{v}_{j-1} - \mathbf{v}_j)\mathbf{u}_j^\top$*, and feed* $\mathcal{A}$ *with* $\mathbf{p}_t^\top A_t$ *and* $A_t\mathbf{w}_t$ *(as a response to its queries* $\mathbf{p}_t, \mathbf{w}_t$*).*

Note that by construction, $\mathbf{v}_t$ is a standard Gaussian random vector in $\mathbb{R}^n$, scaled by β and projected on the subspace orthogonal to $\mathbf{v}_0, \ldots, \mathbf{v}_{t-1}, \mathbf{p}_1, \ldots, \mathbf{p}_t$ (hence, $\mathbf{v}_t$ is orthogonal to all these vectors). This is possible by the assumption $2T + 1 \leq n$, which implies that M_t is always a "thin" matrix representing the basis of a strict subspace of $\mathbb{R}^n$. Similarly, the assumption $2T + 1 \leq d$ implies that choosing $\mathbf{u}_t \in \mathbb{R}^d$ as orthogonal to $2t - 1$ given vectors (as described above) is indeed possible. The choice of a Gaussian distribution is crucially used in order to control the sign and magnitudes of various quantities associated with the matrix, as will be seen later in the proof.

For the construction to be useful, we need to ensure that the oracle responses as defined above (using intermediate matrices A_t) are all consistent with the same fixed matrix A in hindsight. This is formalized in the following lemma, whose proof easily follows from the orthogonality properties in the construction, as detailed in the appendix.

Lemma 9 (Responses simulate oracle on A) *For all* $t \in [T]$: $\mathbf{p}_t^\top A_t = \mathbf{p}_t^\top A$ *and* $A_t\mathbf{w}_t = A\mathbf{w}_t$*. Therefore, the sequences of vectors* $\{\mathbf{p}_t\}_{t=1}^{T}, \{\mathbf{w}_t\}_{t=1}^{T+1}$ *are the same as if the algorithm was fed with the oracle* $\mathcal{O}_2^A$ *for* T *iterations. Moreover,* $A\mathbf{w}_{T+1} = \sum_{j=1}^{T}(\mathbf{v}_{j-1} - \mathbf{v}_j)\mathbf{u}_j^\top\mathbf{w}_{T+1}$.

A second crucial requirement for the construction is that the resulting matrix A is linearly separable with some margin. This is formalized in the following lemma:

Lemma 10 (Separator exists) *For* A *as defined in Construction 8, it holds that*

$$\max_{\mathbf{w}:\|\mathbf{w}\|\leq 1} \min_{\mathbf{p}\in\Delta^{n-1}} \mathbf{p}^\top A\mathbf{w} \geq \frac{\alpha}{\sqrt{T+1}} .$$

The proof of Lemma 10 notes that $A\mathbf{w} = \frac{\alpha}{\sqrt{T+1}}\mathbf{1}$ for $\mathbf{w} := \frac{1}{\sqrt{T+1}}\sum_{t=1}^{T+1}\mathbf{u}_t$, which is a unit vector following the orthonormality properties in the construction, as detailed in the appendix.

A final consistency requirement for the construction is that we need to choose the α, β parameters appropriately, to satisfy the constraint that each row of A must have norm at most 1 (at least with high probability, which implies that a suitable choice of A exists). This can be ensured via the following lemma:

Lemma 11 (Rows of A are bounded) *Suppose* A *is constructed as in Construction 8. Then for any* $\delta \in (0, 1)$ *such that* $2\alpha^2 + 8T\log\left(\frac{2Tn}{\delta}\right)\beta^2 \leq 1$ *it holds with probability at least* $1 - \delta$ *that* $\max_{l\in[n]}\|A_l\| \leq 1$*, where* A_l *is the* l*-th row of* A*.*

The proof of the lemma appears in the appendix, and follows from Gaussian concentration properties. With these consistency components in place, the main task now is to show that the algorithm's output $\mathbf{w}_{T+1}$ cannot be a linear separator for A (or at least, with high probability over the randomized choice of A, which implies that a suitable A exists). This is formalized in the following proposition:

Proposition 12 (Algorithm returns a non-separator) *Suppose* A *is constructed as in Construction 8, and that* $\frac{4\alpha}{\beta} \leq \frac{1}{\sqrt{T}}$*. Then for any* $\delta \in (0, 1)$*, if* $T\sqrt{\frac{80\log(2T/\delta)}{n}} \leq \frac{1}{4}$*, then with probability at*

least $1 - \delta - \exp\left(T\log(2n) - \frac{n}{32}\right)$ *over the choice of* $\boldsymbol{\xi}_1, \ldots, \boldsymbol{\xi}_T$*, it holds that*

$$\sup_{\mathbf{w}\in\mathbb{R}^d} \min_{\mathbf{p}\in\Delta^{n-1}} \mathbf{p}^\top \sum_{j=1}^{T} (\mathbf{v}_{j-1} - \mathbf{v}_j)\mathbf{u}_j^\top \mathbf{w} \leq 0 \,,$$

and thus (by Lemma 9), $\min_{\mathbf{p}\in\Delta^{n-1}} \mathbf{p}^\top A\mathbf{w}_{T+1} \leq 0$.

The formal proof (which appears in the appendix) is rather involved. At a high level, we use Gaussian concentration properties and an ϵ-net argument, to show that with high probability over the randomized construction, it holds for any $\mathbf{w}$ that the vector $\sum_{j=1}^{T}(\mathbf{v}_{j-1} - \mathbf{v}_j)\mathbf{u}_j^\top \mathbf{w}$ contains an entry which is upper bounded by a certain expression, which we then prove to be non-positive. Therefore, $\min_{\mathbf{p}\in\Delta} \mathbf{p}^\top \sum_{j=1}^{T}(\mathbf{v}_{j-1} - \mathbf{v}_j)\mathbf{u}_j^\top \mathbf{w}$ is non-positive for all $\mathbf{w}$. Combining the proposition with the lemmas above, and choosing the α, β parameters appropriately, Theorem 7 follows, as detailed in the appendix.

Remark 13 (Number of oracle queries vs. number of matrix-vector multiplications) *As defined, the two-sided oracle allows two matrix-vector multiplications (one from each side of the matrix A) in each oracle call. Thus, up to a factor of* 2*, any complexity lower bound for a two-sided oracle automatically implies a lower bound on the required total number of matrix-vector multiplications (on either side of the matrix). However, we note that our proof technique can be potentially applied to get a slightly stronger result: Namely, a similar lower bound on the number of* alternations *between a right matrix-vector multiplication and a left matrix-vector multiplication, assuming the matrix size is large enough.*[2]

5. Nash equilibria: simplex/simplex games

We now turn to consider matrix games in which the domain $\mathcal{W} \subset \mathbb{R}^d$ is the probability simplex:

$$\max_{\mathbf{w}\in\Delta^{d-1}} \min_{\mathbf{p}\in\Delta^{n-1}} \mathbf{p}^\top A\mathbf{w} \,. \tag{3}$$

As previously discussed, the objective above corresponds to finding a Nash equilibrium in zero-sum bilinear games. In this section we will assume that each entry of A is in $[-1, +1]$, which is the standard normalization for this setting (cf. Nemirovski, 2004; Hadiji et al., 2024).

Following previous works on algorithms for this problem, we consider lower bounds for the oracle complexity of Eq. (3), using a two-sided oracle. We note that for this problem, a two-sided oracle is exactly equivalent to a first-order (or gradient) oracle for Eq. (3), which returns the gradient of $\mathbf{p}^\top A\mathbf{w}$ with respect to some given $\mathbf{p}, \mathbf{w}$ (namely, $A\mathbf{w}$ and $\mathbf{p}^\top A$). Thus, our lower bound automatically applies to solving Eq. (3) using a first-order oracle.

2. Specifically, consider a model where the algorithm performs a single matrix-vector multiplication in each iteration. Then we can modify Construction 8, so that if there is a sequence of right matrix-vector multiplications using $\mathbf{w}_{t_1}, \ldots, \mathbf{w}_{t_2}$, we can simply pick a single vector $\mathbf{u}_{t_1}$ which is orthogonal to all of them, rather than constructing a sequence of vectors $\mathbf{u}_{t_1}, \ldots, \mathbf{u}_{t_2}$. Similarly, if there is a sequence of left matrix-vector multiplications using $\mathbf{p}_{t_1}, \ldots, \mathbf{p}_{t_2}$, we can pick a single $\mathbf{v}_{t_1}$ which is orthogonal to all of them, rather than constructing a sequence of vectors $\mathbf{v}_{t_1}, \ldots, \mathbf{v}_{t_2}$. Thus, the total number of vectors $\mathbf{u}_1, \mathbf{v}_1, \mathbf{u}_2, \mathbf{v}_2, \ldots$, and hence ultimately the lower bound, does not scale with the total number of matrix-vector multiplications performed by the algorithm, but rather with the number of alternations between left and right matrix-vector multiplications.

We prove the lower bound in two stages: First, we show by reduction that it is sufficient to prove an oracle complexity lower bounds for the ℓ_1/simplex game

$$\max_{\mathbf{w}\in\mathbb{R}^d:\|\mathbf{w}\|_1\leq 1} \min_{\mathbf{p}\in\Delta^{n-1}} \mathbf{p}^\top A\mathbf{w} \;, \tag{4}$$

namely where the simplex domain is enlarged to the entire ℓ_1 ball. In the second stage, we prove such a lower bound for the ℓ_1/simplex game (which of course, may be of independent interest).

Following this plan, we start by proving that an algorithm for solving (3) (over the simplex) can be readily converted to an algorithm for solving (4) (over the ℓ_1 ball), with similar guarantees. Thus, to prove an oracle complexity lower bound for simplex/simplex games (3), it is sufficient to prove a lower bound for solving ℓ_1/simplex games (4).

Proposition 14 *Suppose there is an algorithm $\mathcal{A}$ that for any matrix $A \in [-1,+1]^{n\times 2d}$, after interacting with $\mathcal{O}_2^A$ for T iterations, returns a vector $\mathbf{w}_{T+1} \in \Delta^{2d-1}$ such that*

$$\left(\max_{\mathbf{w}\in\Delta^{2d-1}} \min_{\mathbf{p}\in\Delta^{n-1}} \mathbf{p}^\top A\mathbf{w}\right) - \left(\min_{\mathbf{p}\in\Delta^{n-1}} \mathbf{p}^\top A\mathbf{w}_{T+1}\right) \;\leq\; \epsilon(T,n,d) \;.$$

Then there is an algorithm such that for any $A \in [-1,+1]^{n\times d}$, after interacting with $\mathcal{O}_2^A$ for T iterations, it returns a vector $\mathbf{w}_{T+1} \in \mathbb{R}^d$, $\|\mathbf{w}_{T+1}\|_1 \leq 1$ such that

$$\left(\max_{\mathbf{w}\in\mathbb{R}^d:\|\mathbf{w}\|_1\leq 1} \min_{\mathbf{p}\in\Delta^{n-1}} \mathbf{p}^\top A\mathbf{w}\right) - \left(\min_{\mathbf{p}\in\Delta^{n-1}} \mathbf{p}^\top A\mathbf{w}_{T+1}\right) \;\leq\; \epsilon(T,n,d) \;.$$

The formal proof appears in the appendix. In a nutshell, the idea is that given a matrix $A \in [-1,+1]^{n\times d}$ and an algorithm for a simplex/simplex matrix game, we can feed the algorithm with the matrix $(A;-A) \in \mathbb{R}^{n\times 2d}$, and convert the algorithm's output $\mathbf{w}$ (a vector in the $(2d-1)$-simplex) to a d-dimensional vector $\mathbf{w}'$ in the ℓ_1 unit ball, so that $A\mathbf{w}' = (A;-A)\mathbf{w}$, leading to similar guarantees. We note that the reduction does require us to modify the matrix width d by a factor of 2, but this will not affect our lower bound by more than a small constant factor (as the bound we will show applies to any sufficiently large d). We also note that the computational complexity of the two algorithms in the reduction are essentially identical.

We now turn to consider an oracle complexity lower bound for ℓ_1/simplex games (4), using a two-sided oracle. Our main result is the following:

Theorem 15 *The following holds for some large enough universal constant $c > 0$: For any $T > c$, suppose d, n are sufficiently large so that $d > cT$ and $n > cT^2\log(T)$. Then for any deterministic algorithm for solving Eq. (4), there exists a matrix $A \in [-1,+1]^{n\times d}$ satisfying*

$$\max_{\mathbf{w}:\|\mathbf{w}\|_1\leq 1} \min_{\mathbf{p}\in\Delta^{n-1}} \mathbf{p}^\top A\mathbf{w} \;\geq\; \frac{1}{c\log(d)\sqrt{\log(n)}} \cdot \frac{1}{T^2\sqrt{T}} \;,$$

yet after T rounds of interaction with the two-sided oracle $\mathcal{O}_2^A$, the algorithm returns a vector $\mathbf{w}_{T+1}$ such that $\min_{\mathbf{p}\in\Delta^{n-1}} \mathbf{p}^\top A\mathbf{w}_{T+1} \leq 0$.

In particular, by fixing some small enough $\epsilon > 0$ and choosing T such that the lower bound in the theorem is at least ϵ, we can restate the bound as follows: If the number of iterations T is less than $\Omega(\log(d)^{-2/5}\log(n)^{-1/5}\epsilon^{-2/5})$, there exists a matrix A such that the optimal value is

at least ϵ, yet the algorithm's output has value less than 0, hence is suboptimal by at least ϵ. As discussed previously, by Proposition 14, this lower bound for ℓ_1/simplex automatically extends to simplex/simplex games, up to a small numerical constant which is absorbed in the $\Omega(\cdot)$ notation.

To prove the theorem, we utilize a construction very similar to the ℓ_2 case in the previous section, except that the vectors need to be scaled differently, to satisfy the different constraints on A and $\mathbf{w}$. Moreover, since now the constraint on A is that each individual entry is in $[-1,+1]$, we need to choose each $\mathbf{u}_t$ in Construction 8 more carefully, as follows:

Construction 16 *Suppose the matrix A is constructed as in Construction 8, where instead of $\mathbf{u}_t$ being an arbitrary unit vector (orthogonal to $\mathbf{u}_1,\ldots,\mathbf{u}_{t-1}$ and $\mathbf{w}_1,\ldots,\mathbf{w}_t$), it is chosen specifically as the unit vector $\frac{1}{\|(I-N_tN_t^\top)\boldsymbol{\xi}'_t\|}(I-N_tN_t^\top)\boldsymbol{\xi}'_t$, where the columns of $N_t\in\mathbb{R}^{d\times q_t}$, $q_t\leq 2t-1$ are an orthogonal basis for $\mathbf{u}_1,\ldots,\mathbf{v}_{t-1},\mathbf{w}_1,\ldots,\mathbf{w}_t$, and $\boldsymbol{\xi}'_t\sim\mathcal{N}(\mathbf{0},I_d)$ is an independent Gaussian.*

By construction, $\mathbf{u}_t$ still satisfies the desideratum that it is a unit vector orthogonal to $\mathbf{u}_1,\ldots,\mathbf{v}_{t-1}$ and $\mathbf{w}_1,\ldots,\mathbf{w}_t$. However, this particular construction gives us probabilistic control over the magnitude of its individual entries. Specifically, we have the following:

Lemma 17 (Good solution exists) *For A as defined in Construction 16, and for any $\delta\in(0,1)$, it holds with probability at least $1-2T\left(\delta+\exp(-d/48)\right)$ with respect to the construction that $\max_{\mathbf{w}:\|\mathbf{w}\|_1\leq 1}\min_{\mathbf{p}\in\Delta^{n-1}}\mathbf{p}^\top A\mathbf{w}\;\geq\;\frac{\alpha}{2T\sqrt{8d\log(2d/\delta)}}$.*

Lemma 18 (Entries of A are bounded) *Suppose A is constructed as in Construction 16. Then for any $\delta\in(0,1)$ such that $\sqrt{\frac{8\log(2d/\delta)}{d}}\cdot\left(\alpha+2T\beta\sqrt{2\log(2Tn/\delta)}\right)\leq 1$ it holds with probability at least $1-3T\left(\delta+\exp(-d/48)\right)$ that $A\in[-1,+1]^{n\times d}$.*

Note that the lemmas above are variants of Lemma 10 and Lemma 11 respectively. Combining them with Proposition 12 (which is unaffected by the additional constraint in Construction 16), and choosing α,β accordingly, the proof of Theorem 15 readily follows, as detailed in the appendix.

6. Discussion

In this paper, we studied the oracle complexity of solving matrix games based on the simplex. This well-studied problem class encapsulates tasks such as finding a linear separator and computing a Nash equilibrium in bilinear zero-sum games. By identifying distinct oracle models corresponding to either one-sided or two-sided multiplication by the matrix A, we were able to shed light on previous algorithmic approaches for this task, and to provide several new lower bounds.

For ℓ_2/simplex games which correspond to the linear separability task, we first showed that a one-sided oracle leads to complexity no better than $\Omega(\gamma_A^{-2})$, attained by the classical Perceptron algorithm. Interestingly, this implies a separation between our oracle models, since accelerated methods utilizing two-sided oracle access can achieve a better rate of $\tilde{O}(\gamma_A^{-1})$. However, we showed that even with such improved methods, the achievable rate can be no better than $\tilde{\Omega}(\gamma_A^{-2/3})$. To the best of our knowledge, this is the first oracle complexity lower bound for this setting. Nonetheless, it still leaves a gap of $\tilde{O}(\gamma_A^{-1/3})$ between the upper and lower bounds. Closing this gap, either by designing better algorithms or by improving the lower bound, is an interesting open problem.

As for simplex/simplex and ℓ_1/simplex games, we prove an $\tilde{\Omega}(\epsilon^{-2/5})$ iteration lower bound for computing an ϵ-suboptimal solution, for any algorithm using a two-sided oracle, implying the same lower bound for computing an ϵ-approximate Nash equilibrium. This is an exponential improvement over the previously known lower bound for this task due to Hadiji et al. (2024), leaving open a $\tilde{O}(\epsilon^{-3/5})$ gap compared to the best known upper bound of $\tilde{O}(\epsilon^{-1})$.

Another open problem is extending our lower bounds to randomized algorithms. Since our constructions required simulation of the algorithm's responses, it is not immediately clear how this can be achieved. We note that previous extensions of oracle complexity lower bounds to randomized algorithms (Nemirovski and Yudin, 1983; Carmon et al., 2020a; Arjevani et al., 2023) crucially relied on non-linear modifications of the "hard" target function, which is not possible when we restrict ourselves to bilinear functions.

Acknowledgments

This research is supported in part by European Research Council (ERC) grant 754705. GK is supported by an Azrieli Foundation graduate fellowship.

References

Yossi Arjevani, Yair Carmon, John C Duchi, Dylan J Foster, Nathan Srebro, and Blake Woodworth. Lower bounds for non-convex stochastic optimization. *Mathematical Programming*, 199(1):165–214, 2023.

Alexandre Belloni, Robert M Freund, and Santosh Vempala. An efficient rescaled perceptron algorithm for conic systems. *Mathematics of Operations Research*, 34(3):621–641, 2009.

Yair Carmon, Yujia Jin, Aaron Sidford, and Kevin Tian. Variance reduction for matrix games. *Advances in Neural Information Processing Systems*, 32, 2019.

Yair Carmon, John C Duchi, Oliver Hinder, and Aaron Sidford. Lower bounds for finding stationary points i. *Mathematical Programming*, 184(1):71–120, 2020a.

Yair Carmon, Arun Jambulapati, Qijia Jiang, Yujia Jin, Yin Tat Lee, Aaron Sidford, and Kevin Tian. Acceleration with a ball optimization oracle. *Advances in Neural Information Processing Systems*, 33:19052–19063, 2020b.

Yair Carmon, Arun Jambulapati, Yujia Jin, and Aaron Sidford. A whole new ball game: A primal accelerated method for matrix games and minimizing the maximum of smooth functions. In *Proceedings of the 2024 Annual ACM-SIAM Symposium on Discrete Algorithms (SODA)*, pages 3685–3723. SIAM, 2024.

Kenneth L Clarkson, Elad Hazan, and David P Woodruff. Sublinear optimization for machine learning. *Journal of the ACM (JACM)*, 59(5):1–49, 2012.

Daniel Dadush, László A Végh, and Giacomo Zambelli. Rescaling algorithms for linear conic feasibility. *Mathematics of Operations Research*, 45(2):732–754, 2020.

John Dunagan and Santosh Vempala. A polynomial-time rescaling algorithm for solving linear programs. In *Proc. of the ACM Symposium on Theory of Computing (STOC)*, page 28. Citeseer, 2004.

Hédi Hadiji, Sarah Sachs, Tim van Erven, and Wouter M Koolen. Towards characterizing the first-order query complexity of learning (approximate) nash equilibria in zero-sum matrix games. *Advances in Neural Information Processing Systems*, 36, 2024.

Adam Ibrahim, Waıss Azizian, Gauthier Gidel, and Ioannis Mitliagkas. Linear lower bounds and conditioning of differentiable games. In *International conference on machine learning*, pages 4583–4593. PMLR, 2020.

Ziwei Ji, Nathan Srebro, and Matus Telgarsky. Fast margin maximization via dual acceleration. In *International Conference on Machine Learning*, pages 4860–4869. PMLR, 2021.

Warren S McCulloch and Walter Pitts. A logical calculus of the ideas immanent in nervous activity. *The bulletin of mathematical biophysics*, 5:115–133, 1943.

John F Nash. Equilibrium points in n-person games. *Proceedings of the national academy of sciences*, 36(1):48–49, 1950.

Arkadi Nemirovski. Prox-method with rate of convergence o (1/t) for variational inequalities with lipschitz continuous monotone operators and smooth convex-concave saddle point problems. *SIAM Journal on Optimization*, 15(1):229–251, 2004.

Arkadi Semenovich Nemirovski and David Borisovich Yudin. *Problem complexity and method efficiency in optimization.* Wiley, 1983.

Arkadi S Nemirovsky. Information-based complexity of linear operator equations. *Journal of Complexity*, 8(2):153–175, 1992.

Yurii Nesterov. Excessive gap technique in nonsmooth convex minimization. *SIAM Journal on Optimization*, 16(1):235–249, 2005a.

Yurii Nesterov. Smooth minimization of non-smooth functions. *Mathematical programming*, 103: 127–152, 2005b.

Yurii Nesterov. Dual extrapolation and its applications to solving variational inequalities and related problems. *Mathematical Programming*, 109(2):319–344, 2007.

Yurii Nesterov. *Lectures on convex optimization*, volume 137. Springer, 2018.

Albert BJ Novikoff. On convergence proofs on perceptrons. In *Proceedings of the Symposium on the Mathematical Theory of Automata*, volume 12-1, pages 615–622. New York, NY, 1962.

Yuyuan Ouyang and Yangyang Xu. Lower complexity bounds of first-order methods for convex-concave bilinear saddle-point problems. *Mathematical Programming*, 185(1):1–35, 2021.

Javier Pena and Negar Soheili. A deterministic rescaled perceptron algorithm. *Mathematical Programming*, 155:497–510, 2016.

Sasha Rakhlin and Karthik Sridharan. Optimization, learning, and games with predictable sequences. *Advances in Neural Information Processing Systems*, 26, 2013.

Frank Rosenblatt. The perceptron: a probabilistic model for information storage and organization in the brain. *Psychological review*, 65(6):386, 1958.

Shai Shalev-Shwartz and Shai Ben-David. *Understanding machine learning: From theory to algorithms*. Cambridge university press, 2014.

Negar Soheili and Javier Pena. A smooth perceptron algorithm. *SIAM Journal on Optimization*, 22 (2):728–737, 2012.

Roman Vershynin. Introduction to the non-asymptotic analysis of random matrices. *Compressed Sensing*, pages 210–268, 2012.

Guanghui Wang, Rafael Hanashiro, Etash Kumar Guha, and Jacob Abernethy. On accelerated perceptrons and beyond. In *The Eleventh International Conference on Learning Representations*, 2023.

Adams Wei Yu, Fatma Kilinç-Karzan, and Jaime Carbonell. Saddle points and accelerated perceptron algorithms. In *International Conference on Machine Learning*, pages 1827–1835. PMLR, 2014.

Appendix A. Missing proofs from Section 4.2

A.1. Proof of Lemma 9

By the orthogonality assumptions in the construction, for any t,

$$\mathbf{p}_t^\top A_t \;=\; \mathbf{p}_t^\top \left(\sum_{j=1}^{t}(\mathbf{v}_{j-1}-\mathbf{v}_j)\mathbf{u}_j^\top\right) \;=\; \mathbf{p}_t^\top \left(\sum_{j=1}^{T}(\mathbf{v}_{j-1}-\mathbf{v}_j)\mathbf{u}_j^\top + \mathbf{v}_T\mathbf{u}_{T+1}^\top\right) \;=\; \mathbf{p}_t^\top A\,.$$

Similarly,

$$A_t\mathbf{w}_t \;=\; \left(\sum_{j=1}^{t}(\mathbf{v}_{j-1}-\mathbf{v}_j)\mathbf{u}_j^\top\right)\mathbf{w}_t \;=\; \left(\sum_{j=1}^{T}(\mathbf{v}_{j-1}-\mathbf{v}_j)\mathbf{u}_j^\top + \mathbf{v}_T\mathbf{u}_{T+1}^\top\right)\mathbf{w}_t = A\mathbf{w}_t\,.$$

Since $\mathbf{p}_t, \mathbf{w}_t$ at iteration t are determined by the previous oracle calls, it follows by induction that the sequences of vectors $\{\mathbf{p}_t\}_{t=1}^{T}, \{\mathbf{w}_t\}_{t=1}^{T+1}$ are those produced by the algorithm given access to the oracle $\mathcal{O}_2^A$ for the matrix A. Finally, the expression for $A\mathbf{w}_{T+1}$ follows from the definition of A, and the fact that $\mathbf{u}_{T+1}$ is chosen to be orthogonal to $\mathbf{w}_{T+1}$.

A.2. Proof of Lemma 10

Consider $\mathbf{w} = \frac{1}{\sqrt{T+1}}\sum_{t=1}^{T+1}\mathbf{u}_t$, which is a unit vector since $\mathbf{u}_1,\ldots,\mathbf{u}_{T+1}$ are orthonormal. We have

$$\begin{aligned} A\mathbf{w} \;&=\; \left(\sum_{j=1}^{T}(\mathbf{v}_{j-1}-\mathbf{v}_j)\mathbf{u}_j^\top + \mathbf{v}_T\mathbf{u}_{T+1}^\top\right)\left(\frac{1}{\sqrt{T+1}}\sum_{t=1}^{T+1}\mathbf{u}_t\right) \\ &=\; \frac{1}{\sqrt{T+1}}\left(\sum_{j=1}^{T}(\mathbf{v}_{j-1}-\mathbf{v}_j)+\mathbf{v}_T\right) \;=\; \frac{1}{\sqrt{T+1}}\cdot\mathbf{v}_0 \;=\; \frac{\alpha}{\sqrt{T+1}}\cdot\mathbf{1}\,. \end{aligned}$$

Therefore, $\min_{\mathbf{p}\in\Delta^{n-1}}\mathbf{p}^\top A\mathbf{w} = \frac{\alpha}{\sqrt{T+1}}$ for this choice of unit vector $\mathbf{w}$, so the maximum over all unit vectors can therefore only be larger.

A.3. Proof of Lemma 11

Consider the vector $\mathbf{v}_t = \beta(I - M_tM_t^\top)\boldsymbol{\xi}_t$ in the construction, for some t. Recalling that the matrix M_t is a function of $\boldsymbol{\xi}_1,\ldots,\mathbf{x}_{t-1}$, which are independent of $\boldsymbol{\xi}_t$, we have by Lemma 22 that

$$\Pr\left(\|\mathbf{v}_t\|_\infty \geq z \;\middle|\; \boldsymbol{\xi}_1,\ldots,\boldsymbol{\xi}_{t-1}\right) \;\leq\; 2n\exp\left(-\frac{z^2}{2\beta^2}\right)$$

for all $z>0$. Since this holds for any realization of $\boldsymbol{\xi}_1,\ldots,\boldsymbol{\xi}_{t-1}$, we can apply a union bound over all t and get that

$$\Pr\left(\max_{t\in[T]}\|\mathbf{v}_t\|_\infty \geq z\right) \;\leq\; 2Tn\exp\left(-\frac{z^2}{2\beta^2}\right)\,.$$

Choosing z appropriately, this can equivalently be phrased as follows: For any $\delta \in (0,1)$, with probability at least $1-\delta$, $\max_{t\in[T],l\in[n]} |v_{t,l}| \leq \beta\sqrt{2\log(2Tn/\delta)}$. Under this event, and recalling that $v_{0,l} = \alpha$ by construction, it holds for any $l \in [n]$ that

$$\begin{aligned}
\|A_l\|^2 &= \left\| \sum_{j=1}^{T} (v_{j-1,l} - v_{j,l})\mathbf{u}_j^\top + v_{T,l}\mathbf{u}_{T+1}^\top \right\|^2 \\
&\overset{(1)}{=} \sum_{j=1}^{T} (v_{j-1,l} - v_{j,l})^2 + v_{T,l}^2 \overset{(2)}{=} \sum_{j=1}^{T} 2(v_{j-1,l}^2 + v_{j,l}^2) + v_{T,l}^2 \\
&\leq 2v_{0,l}^2 + 4\sum_{j=1}^{T} v_{j,l}^2 \leq 2\alpha^2 + 4T\beta^2 \cdot 2\log(2Tn/\delta) \,,
\end{aligned}$$

where (1) is because $\mathbf{u}_1, \ldots, \mathbf{u}_{T+1}$ are orthogonal unit vectors, and (2) is because $(x-y)^2 \leq 2x^2 + 2y^2$ for any $x, y \in \mathbb{R}$. By the assumption stated in the lemma, this expression is at most 1 as required.

A.4. Proof of Proposition 12

Fix some $\mathbf{w} \in \mathbb{R}^d$, and define the auxiliary vector $\mathbf{r} = (r_1, \ldots, r_T) \in \mathbb{R}^T$ as

$$\forall j \in [T-1] : \; r_j = \mathbf{u}_j^\top \mathbf{w} - \mathbf{u}_{j+1}^\top \mathbf{w} \;\;,\;\; r_T = \mathbf{u}_T^\top \mathbf{w} \,, \tag{5}$$

as well as the scalar $r_0 = \mathbf{u}_1^\top \mathbf{w}$. By construction, we have

$$\begin{aligned}
\sum_{j=1}^{T} (\mathbf{v}_{j-1} - \mathbf{v}_j)\mathbf{u}_j^\top \mathbf{w} &= \mathbf{v}_0(\mathbf{u}_1^\top \mathbf{w}) - \sum_{j=1}^{T-1} \mathbf{v}_j(\mathbf{u}_j^\top \mathbf{w} - \mathbf{u}_{j+1}^\top \mathbf{w}) - \mathbf{v}_T(\mathbf{u}_T^\top \mathbf{w}) \\
&= r_0\mathbf{v}_0 - \sum_{j=1}^{T-1} r_j\mathbf{v}_j - r_T\mathbf{v}_T = r_0\mathbf{v}_0 - \sum_{j=1}^{T} r_j\mathbf{v}_j \\
&= \alpha r_0 \mathbf{1} - \beta \sum_{j=1}^{T} r_j (I - M_j M_j^\top)\boldsymbol{\xi}_j \\
&= \alpha r_0 \mathbf{1} - \beta \left(\sum_{j=1}^{T} r_j \boldsymbol{\xi}_j - \sum_{j=1}^{T} r_j M_j M_j^\top \boldsymbol{\xi}_j \right) \,.
\end{aligned} \tag{6}$$

Our goal will be to show that with high probability over the random choice of the Gaussian random variables $\boldsymbol{\xi}_1, \ldots, \boldsymbol{\xi}_T$, *simultaneously* for any $\mathbf{r}$, the vector above contains a non-positive entry. Thus, with high probability, the expression $\sum_{j=1}^{T} (\mathbf{v}_{j-1} - \mathbf{v}_j)\mathbf{u}_j^\top \mathbf{w}$ will have a non-positive entry simultaneously for any $\mathbf{w}$, which implies the proposition. For that, we will analyze separately $\sum_{j=1}^{T} r_j \boldsymbol{\xi}_j$ and $\sum_{j=1}^{T} r_j M_j M_j^\top \boldsymbol{\xi}_j$, in the following two lemmas.

Lemma 19 *The following holds with probability at least*

$$1 - \exp\left(T\log(2n) - \frac{n}{32} \right)$$

over the random choice of $\boldsymbol{\xi}_1, \ldots, \boldsymbol{\xi}_T$: For any $\mathbf{r} \in \mathbb{R}^T$, there exists a subset $\mathcal{L} \subseteq [n]$ such that

$$|\mathcal{L}| \geq \frac{n}{20} \quad \textit{and} \quad \min_{l \in \mathcal{L}} \left(\sum_{j=1}^{T} r_j \boldsymbol{\xi}_j \right)_l \geq \frac{1}{2} \|\mathbf{r}\| \,.$$

Proof Let $\Xi \in \mathbb{R}^{n \times T}$ be the matrix whose j-th column is $\boldsymbol{\xi}_j$, so that $\sum_{j=1}^{T} r_j \boldsymbol{\xi}_j = \Xi \mathbf{r}$. Note that Ξ is composed of $n \times T$ independent standard Gaussian entries.

Without loss of generality, it is enough to prove the bound for any unit $\mathbf{r}$: Namely that with high probability, for any unit $\mathbf{r}$,

$$(\Xi \mathbf{r})_l \;\geq\; \frac{1}{2} \quad \text{for at least } \frac{n}{20} \text{ indices } l \tag{7}$$

(the result for all $\mathbf{r} \in \mathbb{R}^T$ follows immediately by scaling $\mathbf{r}$). First, let us fix a unit $\mathbf{r}$ and some index $l \in [n]$, and note that $(\Xi \mathbf{r})_l$ has a standard univariate Gaussian distribution. A standard fact about this distribution is that the probability of getting more than 1 (namely, more than one standard deviation from the mean) is at least $0.1587... \geq 0.15$. Therefore, for any fixed $\mathbf{r}, l$,

$$\mathbb{E}\left[\mathbf{1}_{(\Xi \mathbf{r})_l \geq 1}\right] \;=\; \Pr((\Xi \mathbf{r})_l \geq 1) > 0.15 \,,$$

where $\mathbf{1}_E$ is the indicator function for the event E. Noting that $\left\{\mathbf{1}_{(\Xi \mathbf{r})_l \geq 1}\right\}_{l=1}^{n}$ are independent random variables (since the rows of Ξ are independent), it follows by a standard multiplicative Chernoff bound that

$$\begin{aligned} \Pr\left(\sum_{l=1}^{n} \mathbf{1}_{(\Xi \mathbf{r})_l \geq 1} \leq \frac{n}{20} \right) &= \Pr\left(\sum_{l=1}^{n} \mathbf{1}_{(\Xi \mathbf{r})_l \geq 1} \leq 0.15n \cdot \frac{1}{3} \right) \\ &\leq \exp\left(-\frac{2}{9} \cdot 0.15n \right) = \exp\left(-\frac{n}{30} \right) \,. \end{aligned}$$

This bound is true for any *fixed* unit $\mathbf{r}$. In order to extend the bound simultaneously for all unit $\mathbf{r}$, we will use a standard ϵ-net argument: Fix some $\epsilon > 0$, and let $\mathcal{N}_\epsilon$ be an ϵ-net of the unit Euclidean ball in $\mathbb{R}^T$, of size $\leq \left(\frac{2}{\epsilon} + 1\right)^T$ (namely, a collection of unit vectors so that each $\mathbf{r}$ in the unit ball is ϵ-close in Euclidean distance to one of the vectors). The existence of an ϵ-net of this size is shown, for example, in (Vershynin, 2012, Lemma 5.2). Using a union bound and the displayed equation above, it follows that with probability at least $1 - \left(\frac{2}{\epsilon} + 1\right)^T \exp\left(-\frac{n}{30}\right)$,

$$\forall \mathbf{r} \in \mathcal{N}_\epsilon \,, \;\; \sum_{l=1}^{n} \mathbf{1}_{(\Xi \mathbf{r})_l \geq 1} > \frac{n}{20} \,. \tag{8}$$

Assuming this event holds, let $\mathbf{s}$ be any other unit vector in $\mathbb{R}^T$, such that $\|\mathbf{r} - \mathbf{s}\| \leq \epsilon$ for some $\mathbf{r} \in \mathcal{N}_\epsilon$. Then letting Ξ_l be the l-th row of Ξ, we have

$$\max_{l \in [n]} |(\Xi \mathbf{r} - \Xi \mathbf{s})_l| \;=\; \max_{l \in [n]} |\Xi_l (\mathbf{r} - \mathbf{s})| \;\leq\; \max_{l \in [n]} \|\Xi_l\| \cdot \|\mathbf{r} - \mathbf{s}\| \;\leq\; \epsilon \cdot \max_{l \in [m]} \|\Xi_l\| \,. \tag{9}$$

Since Ξ_l is a standard Gaussian random vector in $\mathbb{R}^T$, it follows by a standard tail bound for the norm of such vectors that

$$\forall l \in [n] \Pr\left(\|\Xi_l\| > \frac{1}{2\epsilon} \right) \;\leq\; 2 \exp\left(-\frac{(1/2\epsilon)^2}{2T} \right) \;=\; 2 \exp\left(-\frac{1}{8\epsilon^2 T} \right) \,,$$

hence by a union bound,

$$\Pr\left(\max_{l\in[n]}\|\Xi_l\| > \frac{1}{2\epsilon}\right) \;\le\; 2n\exp\left(-\frac{1}{8\epsilon^2 T}\right) .$$

Combining this with Eq. (9), it follows that with probability at least $1-2n\exp\left(-\frac{1}{8\epsilon^2 T}\right)$, it holds simultaneously for any ϵ-close unit vectors $\mathbf{r},\mathbf{s}$ that

$$\max_{l\in[n]}|(\Xi\mathbf{r}-\Xi\mathbf{s})_l| \le \frac{1}{2} .$$

Combining the above with Eq. (8) using a union bound, it follows that with probability at least

$$1-\left(\frac{2}{\epsilon}+1\right)^T\exp\left(-\frac{n}{30}\right)-2n\exp\left(-\frac{1}{8\epsilon^2 T}\right) ,$$

it holds simultaneously for all unit vectors $\mathbf{s}$ in $\mathbb{R}^T$ that

$$\sum_{l=1}^{n}\mathbf{1}_{(\Xi\mathbf{s})_l\ge\frac{1}{2}} > \frac{n}{20} .$$

In particular, choosing $\epsilon=\frac{2}{n}$, we get that the probability of this event not occuring is at most

$$(n+1)^T\exp\left(-\frac{n}{30}\right)+2n\exp\left(-\frac{n^2}{32T}\right) .$$

Assuming $n\ge T$ (without loss of generality, since otherwise the probability lower bound in the lemma statement is less than 0), this can be loosely upper bounded by

$$(n+1)^T\exp\left(-\frac{n}{32}\right)+2n\exp\left(-\frac{n}{32}\right) \;=\; \left((n+1)^T+2n\right)\exp\left(-\frac{n}{32}\right) \;\le\; 2(2n)^T\exp\left(-\frac{n}{32}\right) ,$$

which equals $2\exp(T\log(2n)-\frac{n}{32})$. This establishes Eq. (7) for any unit $\mathbf{r}$ as required. ■

Lemma 20 *For any $\delta\in(0,1)$, it holds with probability at least $1-\delta$ over $\boldsymbol{\xi}_1,\ldots,\boldsymbol{\xi}_T$ that*

$$\forall\mathbf{r}\in\mathbb{R}^T,\quad \left\|\sum_{j=1}^{T} r_j M_j M_j^\top \boldsymbol{\xi}_j\right\|^2 \;\le\; 4\|\mathbf{r}\|^2 T^2\log\left(\frac{2T}{\delta}\right) .$$

Proof It is enough to prove that with probability at least $1-\delta$,

$$\forall\mathbf{r}\in\mathbb{R}^d:\|\mathbf{r}\|=1,\quad \left\|\sum_{j=1}^{T} r_j M_j M_j^\top \boldsymbol{\xi}_j\right\|^2 \;\le\; 4T^2\log\left(\frac{2T}{\delta}\right) \tag{10}$$

(the result for all $\mathbf{r}$ then follow by scaling).

By the triangle inequality and Cauchy-Schwartz, we have for any unit vector $\mathbf{r}$

$$\left\|\sum_{j=1}^{T} r_j M_j M_j^\top \boldsymbol{\xi}_j\right\|^2 \leq \left(\sum_{j=1}^{T} |r_j| \|M_j M_j^\top \boldsymbol{\xi}_j\|\right)^2 \leq \left(\sum_{j=1}^{T} r_j^2\right) \cdot \left(\sum_{j=1}^{T} \|M_j M_j^\top \boldsymbol{\xi}_j\|^2\right)$$
$$= \sum_{j=1}^{T} \|M_j M_j^\top \boldsymbol{\xi}_j\|^2 \,. \tag{11}$$

Since $\boldsymbol{\xi}_j$ is a standard Gaussian random vector (with zero mean and identity covariance), each $M_j M_j^\top \boldsymbol{\xi}_j$ (conditioned on $\boldsymbol{\xi}_1, \ldots, \boldsymbol{\xi}_{j-1}$ which in turn fixes M_j) is also Gaussian, with zero mean and covariance $(M_j M_j^\top)^2 = M_j M_j^\top$. Therefore, by a standard tail bound for Gaussian random vectors,

$$\Pr\left(\|M_j M_j^\top \boldsymbol{\xi}_j\| \geq z \mid \boldsymbol{\xi}_1, \ldots, \boldsymbol{\xi}_{j-1}\right) \leq 2\exp\left(-\frac{z^2}{2\mathrm{Tr}(M_j M_j^\top)}\right) = 2\exp\left(-\frac{z^2}{2\|M_j\|_F^2}\right)$$
$$\leq 2\exp\left(-\frac{z^2}{4j}\right) \,,$$

where $\|\cdot\|_F$ denotes Frobenius norm, and where the last step follows from the fact that M_j is a matrix composed of at most $2j$ orthonormal rows. Therefore, for any $\delta' \in (0,1)$, it holds with probability at least $1-\delta'$ over $\boldsymbol{\xi}_t$ that

$$\|M_j M_j^\top \boldsymbol{\xi}_j\| \leq \sqrt{4j \log(2/\delta')} \,.$$

Letting $\delta' = \delta/T$ and using a union bound over all $j \in [T]$, we get that with probability at least $1-\delta$, Eq. (11) is at most

$$\sum_{j=1}^{T} 4j \log(2T/\delta) \;\leq\; \sum_{j=1}^{T} 4T \log(2T/\delta) \;=\; 4T^2 \log(2T/\delta) \,,$$

which leads to Eq. (10) as required. ■

Combining Lemma 19 and Lemma 20 with a union bound, and applying them to Eq. (6), we get the following: With probability at least $1 - \delta - \exp\left(T\log(2n) - \frac{n}{32}\right)$ over $\boldsymbol{\xi}_1, \ldots, \boldsymbol{\xi}_T$, we have that for any $\mathbf{w}$, there is some subset $\mathcal{L} \subseteq [n]$ of size $|\mathcal{L}| \geq \frac{n}{20}$ for which the following inequalities

hold:

$$\begin{aligned}
\min_{l\in[n]}\left(\sum_{j=1}^{T}(\mathbf{v}_{j-1}-\mathbf{v}_j)\mathbf{u}_j^\top\mathbf{w}\right)_l &= \min_{l\in[n]}\left(\alpha r_0\mathbf{1}-\beta\left(\sum_{j=1}^{T}r_j\boldsymbol{\xi}_j-\sum_{j=1}^{T}r_jM_jM_j^\top\boldsymbol{\xi}_j\right)\right)_l \\
&\le \min_{l\in\mathcal{L}}\left(\alpha r_0-\beta\left(\sum_{j=1}^{T}r_j\boldsymbol{\xi}_j\right)_l+\beta\left(\sum_{j=1}^{T}r_jM_jM_j^\top\boldsymbol{\xi}_j\right)_l\right) \\
&\overset{(1)}{\le} \min_{l\in\mathcal{L}}\left(\alpha r_0-\frac{\beta}{2}\|\mathbf{r}\|+\beta\left(\sum_{j=1}^{T}r_jM_jM_j^\top\boldsymbol{\xi}_j\right)_l\right) \\
&\overset{(2)}{\le} \alpha r_0-\frac{\beta}{2}\|\mathbf{r}\|+\beta\sqrt{\frac{1}{|\mathcal{L}|}\sum_{l\in\mathcal{L}}\left(\sum_{j=1}^{T}r_jM_jM_j^\top\boldsymbol{\xi}_j\right)_l^2} \\
&\le \alpha r_0-\frac{\beta}{2}\|\mathbf{r}\|+\beta\sqrt{\frac{1}{|\mathcal{L}|}\sum_{l=1}^{n}\left(\sum_{j=1}^{T}r_jM_jM_j^\top\boldsymbol{\xi}_j\right)_l^2} \\
&\overset{(3)}{\le} \alpha r_0-\frac{\beta}{2}\|\mathbf{r}\|+\beta\sqrt{\frac{20}{n}\cdot 4\|\mathbf{r}\|^2T^2\log\left(\frac{2T}{\delta}\right)} \\
&= \alpha r_0-\beta\left(\frac{1}{2}-T\sqrt{\frac{80\log(2T/\delta)}{n}}\right)\|\mathbf{r}\| \\
&\overset{(4)}{\le} \alpha r_0-\frac{\beta}{4}\|\mathbf{r}\| \\
&\overset{(5)}{=} \frac{\beta}{4}\left(\frac{4\alpha}{\beta}\mathbf{u}_1^\top\mathbf{w}-\sqrt{\sum_{j=1}^{T-1}(\mathbf{u}_j^\top\mathbf{w}-\mathbf{u}_{j+1}^\top\mathbf{w})^2+(\mathbf{u}_T^\top\mathbf{w})^2}\right),
\end{aligned} \tag{12}$$

where (1) is by Lemma 19, (2) is by the fact that a minimum can be upper bounded by an average, (3) is by Lemma 20 and the fact that $|\mathcal{L}|\ge\frac{n}{20}$, (4) is by the assumption in the proposition statement, and (5) is by definition of the vector $\mathbf{r}$ and scalar r_0 in Eq. (5).

We now wish to argue that the expression in Eq. (12) is necessarily non-positive, which would imply overall that

$$\min_{\mathbf{p}\in\Delta^{n-1}}\left(\sum_{j=1}^{T}(\mathbf{v}_{j-1}-\mathbf{v}_j)\mathbf{u}_j^\top\mathbf{w}\right) = \min_{l\in[n]}\left(\sum_{j=1}^{T}(\mathbf{v}_{j-1}-\mathbf{v}_j)\mathbf{u}_j^\top\mathbf{w}\right)_l \le 0\,.$$

Since $\mathbf{w}$ is arbitrary, and the probabilistic statements above hold with high probability simultaneously for any vectors $\mathbf{w},\mathbf{r}$, it follows that with this high probability,

$$\sup_{\mathbf{w}\in\mathbb{R}^d}\min_{\mathbf{p}\in\Delta^{n-1}}\left(\sum_{j=1}^{T}(\mathbf{v}_{j-1}-\mathbf{v}_j)\mathbf{u}_j^\top\mathbf{w}\right) = \min_{l\in[n]}\left(\sum_{j=1}^{T}(\mathbf{v}_{j-1}-\mathbf{v}_j)\mathbf{u}_j^\top\mathbf{w}\right)_l \le 0\,,$$

hence proving the proposition.

Indeed, the fact that Eq. (12) is non-positive follows from the proposition assumption that $\frac{4\alpha}{\beta} \leq \frac{1}{\sqrt{T}}$, and the following lemma:

Lemma 21 *For any integer $T > 1$ and $\delta \in \left(0, \frac{1}{\sqrt{T}}\right]$, it holds that*

$$\sup_{\mathbf{x} \in \mathbb{R}^T} \delta x_1 - \sqrt{\sum_{j=1}^{T-1} (x_{j+1} - x_j)^2 + x_T^2} \;\leq\; 0 \,.$$

Proof Let $M \in \mathbb{R}^{T \times T}$ be the symmetric matrix defined as

$$M \;:=\; \begin{pmatrix} 1 & -1 & 0 & 0 & 0 & \cdots & 0 \\ -1 & 2 & -1 & 0 & 0 & \cdots & 0 \\ 0 & -1 & 2 & -1 & 0 & \cdots & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & \cdots & \cdots & 0 & -1 & 2 & -1 \\ 0 & \cdots & \cdots & 0 & 0 & -1 & 2 \end{pmatrix} .$$

It is easily verified that the expression in the lemma statement equals $\delta \mathbf{e}_1^\top \mathbf{x} - \sqrt{\mathbf{x}^\top M \mathbf{x}}$, where $\mathbf{e}_1$ is the first standard basis vector. Moreover, M is positive semidefinite, as by definition $\mathbf{x}^\top M \mathbf{x} = \sum_{j=1}^{T-1} (x_{j+1} - x_j)^2 + x_T^2 \geq 0$ for all $\mathbf{x}$.

Suppose by contradiction that the lemma does not hold, namely there exists some $\mathbf{x} \in \mathbb{R}^T$ such that

$$\delta \mathbf{e}_1^\top \mathbf{x} - \sqrt{\mathbf{x}^\top M \mathbf{x}} \;>\; 0 \,. \tag{13}$$

Since $\sqrt{\mathbf{x}^\top M \mathbf{x}} \geq 0$, it follows that $\delta \mathbf{e}_1^\top \mathbf{x} > 0$, and therefore $\delta \mathbf{e}_1^\top \mathbf{x} + \sqrt{\mathbf{x}^\top M \mathbf{x}} > 0$. As a result,

$$\begin{aligned} 0 \;&<\; \left(\delta \mathbf{e}_1^\top \mathbf{x} - \sqrt{\mathbf{x}^\top M \mathbf{x}}\right) \left(\delta \mathbf{e}_1^\top \mathbf{x} + \sqrt{\mathbf{x}^\top M \mathbf{x}}\right) \\ &=\; \delta^2 (\mathbf{e}_1^\top \mathbf{x})^2 - \mathbf{x}^\top M \mathbf{x} \;=\; \mathbf{x}^\top \left(\delta^2 \mathbf{e}_1 \mathbf{e}_1^\top - M\right) \mathbf{x} \,. \end{aligned}$$

In other words, we get that $\mathbf{x}^\top (M - \delta^2 \mathbf{e}_1 \mathbf{e}_1^\top) \mathbf{x} < 0$, and therefore $M - \delta^2 \mathbf{e}_1 \mathbf{e}_1^\top$ is *not* a positive semidefinite matrix. However, we will now show that $M - \delta^2 \mathbf{e}_1 \mathbf{e}_1^\top$ is positive semidefinite, which leads to a contradiction, hence Eq. (13) cannot hold, and thus proving the lemma. Indeed, for any

vector $\mathbf{y}$, we have

$$\begin{aligned}
\mathbf{y}^\top (M - \delta^2 \mathbf{e}_1 \mathbf{e}_1^\top)\mathbf{y} = & -\delta^2 y_1^2 + \sum_{j=1}^{T-1} (y_j - y_{j+1})^2 + y_T^2 \\
\overset{(1)}{\geq} & -\delta^2 y_1^2 + \frac{1}{T}\left(\sum_{j=1}^{T-1} |y_j - y_{j+1}| + |y_T|\right)^2 \\
\overset{(2)}{\geq} & -\delta^2 y_1^2 + \frac{1}{T}\left(\sum_{j=1}^{T-1} (y_j - y_{j+1}) + y_T\right)^2 \\
= & -\delta^2 y_1^2 + \frac{1}{T} y_1^2 \;=\; \left(-\delta^2 + \frac{1}{T}\right) y_1^2 \\
\overset{(3)}{\geq} & \; 0 \,,
\end{aligned}$$

where (1) uses the fact that $\|\mathbf{z}\|_2 \geq \frac{1}{\sqrt{T}}\|\mathbf{z}\|_1$ for any $\mathbf{z} \in \mathbb{R}^T$, (2) uses the triangle inequality, and (3) is by the assumption that $\delta \in \left(0, \frac{1}{\sqrt{T-1}}\right)$. Therefore, $M - \delta^2 \mathbf{e}_1 \mathbf{e}_1^\top$ is a positive semidefinite matrix, which as explained above proves the lemma. ■

A.5. Proof of Theorem 7

Examining the conditions in Proposition 12, Lemma 10 and Lemma 11, we see that in order for all of them to be applicable, we must choose $\alpha \lesssim \frac{\beta}{\sqrt{T}}$ (Proposition 12), $\beta \lesssim \sqrt{\frac{1}{T\log(n)}}$ (Lemma 11), yet the margin guarantee is $\approx \frac{\alpha}{\sqrt{T}}$ (Lemma 10). Thus, to make the margin $\frac{\alpha}{\sqrt{T}}$ as large as possible, yet satisfying all of the constraints, we should choose $\beta \approx \sqrt{\frac{1}{T\log(n)}}$ and $\alpha \approx \frac{\beta}{\sqrt{T}} \approx \frac{1}{T\sqrt{\log(n)}}$, and get a margin guarantee of approximately $\frac{\alpha}{\sqrt{T}} \approx \frac{1}{T\sqrt{T\log(n)}}$.

A bit more formally, pick (say) $\delta = 1/10$ in both Proposition 12 and Lemma 11, and choose $\beta = \frac{1}{c\sqrt{T\log(n)}}$ for some sufficiently large universal constant c, as well as $\alpha = \frac{\beta}{4\sqrt{T}} = \frac{1}{4cT\sqrt{\log(n)}}$. It is easily verified that with this choice, the assumptions in Proposition 12, Lemma 10 and Lemma 11 are all satisfied, and (with a union bound), the resulting matrix A satisfies both $\max_{l\in[n]} \|A_l\| \leq 1$, $\min_{\mathbf{p}\in\Delta^{n-1}} \mathbf{p}^\top A \mathbf{w}_{T+1} \leq 0$ and $\max_{\mathbf{w}:\|\mathbf{w}\|\leq 1} \min_{\mathbf{p}\in\Delta^{n-1}} \mathbf{p}^\top A \mathbf{w} \geq \frac{\alpha}{\sqrt{T+1}} \geq \frac{1}{c'T\sqrt{T\log(n)}}$ (for some universal constant $c' > 0$) with some positive probability. Hence, by the probabilistic method, a suitable fixed matrix A satisfying all of the above must exist.

Appendix B. Missing proofs from Section 5

B.1. Proof of Proposition 14

Given any matrix $A \in [-1, +1]^{n\times d}$, define the operator

$$\psi(A) := (A; -A) \in \mathbb{R}^{n\times 2d} \,.$$

Somewhat abusing notation, we will also define ψ as a mapping from the unit ℓ_1 ball in $\mathbb{R}^d$ to Δ^{2d-1} as follows:

$$\psi(\mathbf{w}) = \sum_{j=1}^{d} |w_j| \left(\mathbf{1}_{w_j > 0} \cdot \mathbf{e}_j + \mathbf{1}_{w_j \leq 0} \cdot \mathbf{e}_{j+d} \right) \ \in \ \mathbb{R}^{2d} \,,$$

where $\mathbf{1}_E$ is the indicator function for the event E. In words, $\psi(\mathbf{w})$ is the $2d$-dimensional vector where the first d entries correspond to the positive entries in $\mathbf{w}$, and the last d entries correspond to the absolute values of the non-positive ones. We will also define the inverse operator ψ^{-1} from Δ^{2d-1} to the unit ℓ_1 ball in $\mathbb{R}^d$ as

$$\psi^{-1}(\mathbf{w}) := \sum_{j=1}^{d} (w_j - w_{d+j}) \mathbf{e}_j \in \mathbb{R}^d \,.$$

This is indeed a mapping to the unit ℓ_1 ball, since for any $\mathbf{w} \in \Delta^{2d-1}$,

$$\|\psi^{-1}(\mathbf{w})\|_1 \ = \ \sum_{j=1}^{d} |w_j - w_{d+j}| \ \leq \ \sum_{j=1}^{d} (|w_j| + |w_{d+j}|) \ = \ 1 \,.$$

Moreover, it is easily verified that $A\mathbf{w} = \psi(A)\psi(\mathbf{w})$ and $A\psi^{-1}(\mathbf{w}) = \psi(A)\mathbf{w}$.

Now, given the algorithm $\mathcal{A}$ for the simplex domain and a matrix $A \in \mathbb{R}^{n \times d}$, consider the following algorithm for the ℓ_1 domain: Run $\mathcal{A}$ on the matrix $\psi(A) \in [-1,+1]^{n \times 2d}$ for T iterations, resulting in the vector $\mathbf{w}_{T+1} \in \Delta^{2d-1}$, and return the vector $\psi^{-1}(\mathbf{w}_{T+1}) \in \mathbb{R}^d$ (which has ℓ_1 norm at most 1 by the above). Letting $\mathbf{w}^*$ be some optimal solution of $\max_{\mathbf{w} \in \mathbb{R}^d : \|\mathbf{w}\|_1 \leq 1} \min_{\mathbf{p} \in \Delta^{n-1}} \mathbf{p}^\top A \mathbf{w}$, we have

$$\begin{aligned}
&\left(\max_{\mathbf{w} \in \mathbb{R}^d : \|\mathbf{w}\|_1 \leq 1} \min_{\mathbf{p} \in \Delta^{n-1}} \mathbf{p}^\top A \mathbf{w} \right) - \left(\min_{\mathbf{p} \in \Delta^{n-1}} \mathbf{p}^\top A \psi^{-1}(\mathbf{w}_{T+1}) \right) \\
&= \ \left(\min_{\mathbf{p} \in \Delta^{n-1}} \mathbf{p}^\top A \mathbf{w}^* \right) - \left(\min_{\mathbf{p} \in \Delta^{n-1}} \mathbf{p}^\top A \psi^{-1}(\mathbf{w}_{T+1}) \right) \\
&= \ \left(\min_{\mathbf{p} \in \Delta^{n-1}} \mathbf{p}^\top \psi(A)\psi(\mathbf{w}^*) \right) - \left(\min_{\mathbf{p} \in \Delta^{n-1}} \mathbf{p}^\top \psi(A) \mathbf{w}_{T+1} \right) \\
&\leq \ \left(\max_{\mathbf{w} \in \Delta^{2d-1}} \min_{\mathbf{p} \in \Delta^{n-1}} \mathbf{p}^\top \psi(A) \mathbf{w} \right) - \left(\min_{\mathbf{p} \in \Delta^{n-1}} \mathbf{p}^\top \psi(A) \mathbf{w}_{T+1} \right) \ \leq \ \epsilon(T,n,d)
\end{aligned}$$

as required, where the last inequality is by the guarantee on the algorithm $\mathcal{A}$.

B.2. Proof of Lemma 17

Consider $\mathbf{w} = \frac{1}{\sum_{t=1}^{T+1} \|\mathbf{u}_t\|_1} \sum_{t=1}^{T+1} \mathbf{u}_t$, which by construction satisfies $\|\mathbf{w}\|_1 \leq 1$. Since $\mathbf{u}_1, \ldots, \mathbf{u}_{T+1}$ are orthonormal, we have

$$\begin{aligned}
\left(\sum_{t=1}^{T+1} \|\mathbf{u}_t\|_1 \right) A\mathbf{w} \ &= \ \left(\sum_{j=1}^{T} (\mathbf{v}_{j-1} - \mathbf{v}_j) \mathbf{u}_j^\top + \mathbf{v}_T \mathbf{u}_{T+1}^\top \right) \left(\sum_{t=1}^{T+1} \mathbf{u}_t \right) \\
&= \ \sum_{j=1}^{T} (\mathbf{v}_{j-1} - \mathbf{v}_j) + \mathbf{v}_T \ = \ \mathbf{v}_0 \ = \ \alpha \cdot \mathbf{1} \,. \qquad (14)
\end{aligned}$$

Moreover, recalling the construction of $\mathbf{u}_t$ in Construction 16, we have by the second half of Lemma 22 that

$$\Pr\left(\|\mathbf{u}_t\|_\infty \geq \frac{z}{\sqrt{d}} \,\middle|\, \boldsymbol{\xi}'_1, \ldots, \boldsymbol{\xi}'_{t-1}\right) \;\leq\; 2d\exp\left(-\frac{z^2}{8}\right) + \exp\left(-\frac{d}{48}\right) .$$

This holds for any realization of $\boldsymbol{\xi}'_1, \ldots, \boldsymbol{\xi}'_{t-1}$, hence by a union bound

$$\Pr\left(\max_{t\in[T+1],i} |u_{t,i}| \geq \frac{z}{\sqrt{d}}\right) \;\leq\; (T+1)\left(2d\exp\left(-\frac{z^2}{8}\right) + \exp\left(-\frac{d}{48}\right)\right) .$$

Equivalently, by equating $2d\exp(-z^2/8)$ to δ, we have that for any $\delta \in (0,1)$, with probability at least $1-(T+1)\left(\delta+\exp(-d/48)\right)$,

$$\max_{t\in[T+1]} \|\mathbf{u}_t\|_\infty \leq \sqrt{\frac{8\log(2d/\delta)}{d}} , \tag{15}$$

in which case

$$\sum_{t=1}^{T+1} \|\mathbf{u}_t\|_1 \;\leq\; (T+1)d\sqrt{\frac{8\log(2d/\delta)}{d}} \;=\; (T+1)\sqrt{8d\log(2d/\delta)} .$$

Coarsely upper bounding $(T+1)$ by $2T$ and plugging into Eq. (14), it follows that with probability at least $1-(T+1)\left(\delta+\exp(-d/48)\right) \geq 1-2T(\delta+\exp(-d/48))$, each entry of $A\mathbf{w}$ is at least

$$\frac{\alpha}{2T\sqrt{8d\log(2d/\delta)}} .$$

This holds for the $\mathbf{w}$ we have chosen, so the maximum of $\min_{\mathbf{p}\in\Delta^{n-1}} \mathbf{p}^\top A\mathbf{w}$ over all vectors $\mathbf{w}$ in the unit ℓ_1 ball can only be larger, from which the lemma follows.

B.3. Proof of Lemma 18

Recalling the construction of $\mathbf{v}_t$ in Construction 8, and applying Lemma 22 combined with a union bound, it follows that for any $z > 0$,

$$\Pr\left(\max_{t\in[T]} \|\mathbf{v}_t\|_\infty > z\right) \;\leq\; 2Tn\exp\left(-\frac{z^2}{2\beta^2}\right) .$$

Equivalently, by choosing z appropriately, we have that for any $\delta \in (0,1)$, with probability at least $1-\delta$,

$$\max_{t\in[T]} \|\mathbf{v}_t\|_\infty \;\leq\; \beta\sqrt{2\log(2Tn/\delta)} .$$

Also, applying the same lemma with respect to $\mathbf{u}_t$ (as done in Eq. (15) above), we have that with probability at least $1-(T+1)\left(\delta+\exp(-d/48)\right)$,

$$\max_{t\in[T+1]} \|\mathbf{u}_t\|_\infty \leq \sqrt{\frac{8\log(2d/\delta)}{d}} .$$

Assuming these two events hold (with probability at least $1-\delta-(T+1)\left(\delta+\exp(-d/48)\right)$, by a union bound), and recalling that $v_{0,l}=\alpha$ for all l by construction, it holds for any $l\in[n], i\in[d]$ that

$$\begin{aligned}
|A_{l,i}| &= \left|\sum_{j=1}^{T}(v_{j-1,l}-v_{j,l})u_{j,i}+v_{T,l}u_{T+1,i}\right| \\
&\leq \sum_{j=1}^{T}|v_{j-1,l}-v_{j,l}|\cdot|u_{j,i}|+|v_{T,l}|\cdot|u_{T+1,i}| \\
&\leq \sqrt{\frac{8\log(2d/\delta)}{d}}\left(\sum_{j=1}^{T}|v_{j-1,l}-v_{j,l}|+|v_{T,l}|\right) \\
&\leq \sqrt{\frac{8\log(2d/\delta)}{d}}\left(\sum_{j=1}^{T}(|v_{j-1,l}+|v_{j,l}|)+|v_{T,l}|\right) \\
&\leq \sqrt{\frac{8\log(2d/\delta)}{d}}\left(\alpha+2T\beta\sqrt{2\log(2Tn/\delta)}\right) .
\end{aligned}$$

By the assumption stated in the lemma, this expression is at most 1, hence the entries of A are bounded in $[-1,+1]$ as required. All this holds with probability at least $1-\delta-(T+1)\left(\delta+\exp(-d/48)\right)$, which can be coarsely lower bounded by $1-3T\left(\delta+\exp(-d/48)\right)$.

B.4. Proof of Theorem 15

We construct the matrix A as in Construction 16. Examining the conditions in Proposition 12, and Lemma 17 and Lemma 18, and ignoring log factors momentarily, we see that in order for all of them to be applicable, we must choose $\alpha\lesssim\frac{\beta}{\sqrt{T}}$ (Proposition 12), satisfy $\beta\lesssim\frac{\sqrt{d}}{T}$ and $\alpha\lesssim\sqrt{d}$ (Lemma 18), yet the game value guarantee is $\approx\frac{\alpha}{T\sqrt{d}}$ (Lemma 17). Thus, to make the game value $\frac{\alpha}{T\sqrt{d}}$ as large as possible, yet satisfying all of the constraints, we should choose $\beta\approx\frac{\sqrt{d}}{T}$ and $\alpha\approx\frac{\beta}{\sqrt{T}}\approx\sqrt{\frac{d}{T^3}}$, and get a margin guarantee of approximately $\frac{\alpha}{T\sqrt{d}}\approx\frac{1}{T^2\sqrt{T}}$.

A bit more formally, pick (say) $\delta=1/(50T)$ in both Proposition 12, Lemma 17 and Lemma 18, and choose

$$\beta=\frac{\sqrt{d}}{cT\sqrt{\log(nT)\log(dT)}}$$

for some sufficiently large universal constant c, as well as

$$\alpha=\frac{\beta}{4\sqrt{T}}=\frac{\sqrt{d}}{4cT\sqrt{T\log(nT)\log(dT)}} .$$

It is easily verified that with this choice, as well as the theorem assumptions, the conditions in Proposition 12, Lemma 17 and Lemma 18 are all satisfied, and (with a union bound), the resulting matrix A satisfies both $A\in[-1,+1]^{n\times d}$ and $\min_{\mathbf{p}\in\Delta^{n-1}}\mathbf{p}^\top A\mathbf{w}_{T+1}\leq 0$, as well as (for some constant $c'>0$)

$$\max_{\mathbf{w}:\|\mathbf{w}\|\leq 1}\min_{\mathbf{p}\in\Delta^{n-1}}\mathbf{p}^\top A\mathbf{w}\geq\frac{\alpha}{c'T\sqrt{d\log(dT)}}=\frac{1}{4cc'\log(dT)\sqrt{\log(nT)}}\cdot\frac{1}{T^2\sqrt{T}} ,$$

all with some positive probability. Hence, by the probabilistic method, a suitable fixed matrix A satisfying all of the above must exist. Coarsely upper bounding $\log(dT)$ by $\log(d^2) = 2\log(d)$, and $\log(nT)$ by $\log(n^2) = 2\log(n)$, and plugging in the displayed equation above, the theorem follows.

Appendix C. Auxiliary lemma

Lemma 22 *Let $\boldsymbol{\xi} \sim \mathcal{N}(\mathbf{0}, I_q)$ be a standard Gaussian random variable, and define the vector $\mathbf{x} = \beta(I - MM^\top)\boldsymbol{\xi}$ where M is some matrix composed of orthonormal columns, and $\beta > 0$. Then for all $z > 0$,*

$$\Pr(\|\mathbf{x}\|_\infty \geq z) \;\leq\; 2q \exp\left(-\frac{z^2}{2\beta^2}\right) .$$

Moreover, if the number of columns in M is less than $q/2$, then it also holds that

$$\Pr\left(\frac{\|\mathbf{x}\|_\infty}{\|\mathbf{x}\|_2} \geq \frac{z}{\sqrt{q}}\right) \;\leq\; 2q \exp\left(-\frac{z^2}{8}\right) + \exp\left(-\frac{q}{48}\right) .$$

Proof $\mathbf{x}$ has a zero-mean Gaussian distribution, with covariance matrix $\beta^2(I - MM^\top)^2 = \beta^2(I - MM^\top)$. In particular, for any fixed index l, the coordinate x_l is a zero-mean univariate Gaussian with variance $\beta^2\mathbf{e}_l^\top(I - MM^\top)\mathbf{e}_l = \beta^2(1 - \|\mathbf{e}_l^\top M\|^2) \leq \beta^2$. Therefore, by a standard Gaussian tail bound, $\Pr(|x_l| \geq z) \leq 2\exp\left(-z^2/2\beta^2\right)$ for any $z > 0$. Applying a union bound over all indices $l \in [q]$, we get

$$\Pr(\|\mathbf{x}\|_\infty \geq z) \;\leq\; 2q \exp\left(-\frac{z^2}{2\beta^2}\right) , \tag{16}$$

from which the first inequality in the lemma follows.

As for the second inequality, note that since M has at most $q/2$ orthonormal columns, then $I - MM^\top$ is a projection matrix to a subspace of $\mathbb{R}^q$ of dimensionality at least $q/2$. Thus, $\|(I - MM^\top)\boldsymbol{\xi}\|$ has the same distribution as the norm of a standard Gaussian random variable on $\mathbb{R}^s$ where $s \geq q/2$. Using a standard tail lower bound for such Gaussian norms (see for example (Shalev-Shwartz and Ben-David, 2014, Lemma B.12)), it follows that

$$\begin{aligned}\Pr\left(\|(I - MM^\top)\boldsymbol{\xi}\|^2 \leq \frac{1}{2}\cdot\frac{q}{2}\right) &\leq \Pr\left(\|(I - MM^\top)\boldsymbol{\xi}\|^2 \leq \frac{1}{2}\cdot s\right) \leq \exp\left(-\frac{s}{24}\right) \\ &\leq \exp\left(-\frac{q}{48}\right) .\end{aligned}$$

Therefore, since $\mathbf{x} = \beta(I - MM^\top)\boldsymbol{\xi}$,

$$\Pr\left(\|\mathbf{x}\|_2 \leq \frac{\beta\sqrt{q}}{2}\right) \;=\; \Pr\left(\|(I - MM^\top)\boldsymbol{\xi}\|^2 \leq \frac{q}{4}\right) \;\leq\; \exp\left(-\frac{q}{48}\right) .$$

Combining this with Eq. (16) using a union bound, it follows that

$$\Pr\left(\frac{\|\mathbf{x}\|_\infty}{\|\mathbf{x}\|_2} \geq \frac{2z}{\beta\sqrt{q}}\right) \;\leq\; 2q \exp\left(-\frac{z^2}{2\beta^2}\right) + \exp\left(-\frac{q}{48}\right) .$$

Substituting z instead of $2z/\beta$ and simplifying a bit, the result follows. ■