Electronic Colloquium on Computational Complexity, Report No. 48 (2026)

ECCC

Optimal Random Self-Reductions for All Linear Problems

Abstract

The *linear problem* specified by an $n \times n$ matrix M over a finite field is the problem of computing the product of M and a given vector x. We present optimal error-tolerant random self-reductions (also known as worst-case to average-case reductions) for all linear problems: Given a linear-size circuit that computes Mx on an ε-fraction of inputs x for a positive constant ε, we construct a randomized linear-size circuit that computes Mx for all inputs x with high probability. This resolves the open problem posed by Asadi, Golovnev, Gur, Shinkar, and Subramanian (SODA'24), who presented *quantum* $n^{1.5}$-time random self-reductions for all linear problems. Somewhat surprisingly, we also demonstrate the quantum advantage of their quantum reduction over classical *uniform* algorithms, by proving that any classical subquadratic-time random self-reduction requires the advice complexity of $\Omega(\log(1/\varepsilon) \cdot \log n)$, as long as the field size is at most $1/\varepsilon$. We complement this advice complexity lower bound by presenting (1) a random self-reduction with the optimal advice complexity of $O(\log(1/\varepsilon) \cdot \log n)$ and (2) a uniform random self-reduction over a large finite field.

Contents

ISSN 1433-8092

1 Introduction

A matrix $M \in \mathbb{F}^{m \times n}$ over a finite field $\mathbb{F}$ induces a natural computational problem $\mathcal{L}_M$, which is called the *linear problem* specified by M: Given a vector $x \in \mathbb{F}^n$ as input, the task of $\mathcal{L}_M$ is to compute the product $Mx \in \mathbb{F}^m$. Linear problems are ubiquitous in theoretical computer science and arise as basic subroutines in many algorithms. Representative examples include: (1) encoding a message using a linear error-correcting code, which is equivalent to solving $\mathcal{L}_G$, where G is the generator matrix of the error-correcting code; (2) multipoint evaluation of a univariate polynomial on a fixed set of points, which is equivalent to $\mathcal{L}_V$ for a Vandermonde matrix V; and (3) the discrete Fourier transform, for which fast Fourier transforms solve $\mathcal{L}_M$ for the DFT matrix M.

Although a naïve algorithm computes any linear problem with $O(mn)$ field operations, many structured linear problems admit nearly linear-time algorithms (see, e.g., [Spi96; BM74; CT65]). Understanding which matrices admit such fast algorithms — and which provably do not — has been one of the central questions in theoretical computer science. For example, Valiant's program [Val77] seeks an explicit linear problem $\mathcal{L}_M$ that cannot be computed by linear-size arithmetic circuits of logarithmic depth; proving such lower bounds would follow from constructing a rigid matrix M (see [Ram20] for a survey on matrix rigidity).

Asadi, Golovnev, Gur, Shinkar, and Subramanian [AGGSS24] asked one fundamental property of linear problems: Do they admit error-tolerant random self-reducibility? A *random self-reduction* for a function f is a transformation that converts an average-case algorithm C that computes $f(x)$ on an ε-fraction of inputs x into a worst-case algorithm C' that computes $f(x)$ on all inputs x. If the parameter $\varepsilon > 0$ is close to 0, the reduction is called *error-tolerant*. Such a reduction is also called a *worst-case to average-case reduction from f to f* because it reduces the worst-case problem of computing $f(x)$ for all inputs x to the average-case problem of computing $f(x)$ for a small fraction of inputs x. Asadi, Golovnev, Gur, Shinkar, and Subramanian developed a *quantum* error-tolerant $n^{1.5}$-time random self-reduction for all linear problems, leaving open whether a *classical* error-tolerant worst-case to average-case reduction exists. To quote [AGGSS24, Open Problem 1.1]:

> "Are there efficient transformations of classical algorithms (or circuits) for general linear problems, which are only correct on 1% of their inputs, into similarly efficient worst-case classical algorithms (or circuits)?"

1.1 Our Results

In this paper, we provide complete answers to their open questions — in fact, both positive and negative answers for the non-uniform and uniform settings, respectively. In the non-uniform setting, every linear problem admits a random self-reduction that can be implemented by a linear-size circuit.

Theorem 1.1. *Let $\mathbb{F}$ be a finite field of size p and $M \in \mathbb{F}^{m\times n}$ be a matrix. Let $\varepsilon > 0$ be an arbitrary parameter. Suppose that there exists a circuit C of size s such that*

$$\Pr_{x\sim\mathbb{F}^n}[C(x) = Mx] \geq \varepsilon,$$

where the probability is over a uniformly random vector $x \in \mathbb{F}^n$. Then, there exists a randomized circuit C' of size $O\Big((s + (n + m) \cdot \log^{1+o(1)} p) \cdot \log(1/\varepsilon)/\varepsilon\Big)$ such that for every $x \in \mathbb{F}^n$,

$$\Pr_{C'}\big[C'(x) = Mx\big] \geq \frac{2}{3},$$

where the probability is over the internal randomness of C'.

Note that the size of C' is $O(s + n + m)$ for constants ε and p. The factor $\log^{1+o(1)} p$ in the running time is due to the time complexity of arithmetic operations over $\mathbb{F}$ (see, e.g., [GG13]). The constant $\frac{2}{3}$ can be amplified to $1 - \delta$ for every $\delta > 0$ by the standard technique of repetition of $O(\log(1/\delta))$ times and majority vote.

Theorem 1.1 improves the previous result of [AGGSS24] in three ways. First, our reduction is classical, at the cost of using non-uniform algorithms (i.e., circuits). Second, it improves the previous running time from $O(n^{1.5})$ to the optimal linear "time" of $O(n)$. Third, the proof is unexpectedly simple; see Section 2 for details.

As discussed in [AGGSS24], Theorem 1.1 is the first classical error-tolerant worst-case to average-case reduction for a large class of problems. Previously, such reductions are given only for a small number of specific problems that are of interest (e.g., [BRSV17; GR18; DLW20; BBB19; HLS22; AGGS22; NRW23; HS23; HS24; GSS24; HS25]). Although there are general frameworks for constructing worst-case to average-case reductions based on additive combinatorics [AGGS22] and hardness amplification [HS23], applying these frameworks to specific problems tends to be highly non-trivial. Our reduction gives, for the first time, such (non-quantum) reductions for a large class of problems, i.e., the class of all the linear problems, which includes as special cases the problems of computing discrete Fourier transformations, polynomial evaluation, and the Walsh–Hadamard transform. As a concrete example, we have the following:

Corollary 1.2. *Let $\mathbb{F}$ be a finite field of size $O(1)$. Let H_N be the $N \times N$ Hadamard matrix over $\mathbb{F}$, i.e,. $H_N := \Big((-1)^{\sum_{i=1}^n x_i y_i}\Big)_{(x,y)\in\{0,1\}^n\times\{0,1\}^n}$, where $N = 2^n$. The following are equivalent for every constant $\varepsilon > 0$.*

1. *For all large N, there exists a randomized circuit of size $O(N)$ that computes $H_N \cdot x$ for every input x with high probability.*
2. *For all large N, there exists a circuit of size $O(N)$ that computes $H_N \cdot x$ for an ε-fraction of inputs x.*

One may interpret this as an approach towards constructing a linear-size circuit for computing the Walsh–Hadamard transform based on an average-case circuit that succeeds only a small fraction of inputs. Alternatively, it strengthens any worst-case lower bound to an average-case lower bound; indeed, it is believed that the Walsh–Hadamard transform cannot be computed by $O(N)$-size circuits of depth $O(\log N)$ [AW17].

The only deficiency of Theorem 1.1 is that the reduction is non-uniform, whereas the quantum reduction of [AGGSS24] is uniform in the following sense. The reduction is granted oracle access to each entry of $M \in \mathbb{F}^{m\times n}$ (that is, the oracle answers the (i,j)-th entry of M given (i,j) as input) in addition to an oracle $\mathcal{O}$ such that $\Pr_{x\sim\mathbb{F}^n}[\mathcal{O}(x) = Mx] \geq \varepsilon$, and computes Mx with high probability for every input x. Following [AGGSS24], we define a classical uniform random self-reduction as follows.

Definition 1.3. *For a matrix $M \in \mathbb{F}^{m\times n}$, we say that a randomized oracle algorithm $A^{\mathcal{O},M}$ is an* ε-error-tolerant random self-reduction *for $\mathcal{L}_M$ if for every oracle $\mathcal{O}$ that satisfies*

$$\Pr_{x\sim\mathbb{F}^n}[\mathcal{O}(x) = Mx] \geq \varepsilon,$$

it holds that for every input $x \in \mathbb{F}^n$,

$$\Pr_A\big[A^{\mathcal{O},M}(x) = Mx\big] \geq \frac{2}{3},$$

where the probability is over the internal randomness of A, and the oracle M in $A^{\mathcal{O},M}$ answers the (i,j)-th entry of M for a given query (i,j).

The oracle access to each entry of M is necessary for an error-tolerant random self-reduction to exist, because the oracle $\mathcal{O}$ alone does not uniquely determine the matrix M if $\varepsilon \leq \frac{1}{2}$[1]. Somewhat surprisingly, we prove that no classical uniform sub-quadratic-time reduction exists, thereby negatively answering [AGGSS24, Open Problem 1.1] in the uniform setting.

Theorem 1.4. *For every constant $\gamma > 0$, for all large $n \in \mathbb{N}$ and for every $\varepsilon > 0$, every $a \in \mathbb{N}$, and every finite field $\mathbb{F}$ of size p, if there exists an ε-error-tolerant random self-reduction $A^{\mathcal{O},M}(\text{-};\alpha(\mathcal{O},M))$ for $\mathcal{L}_M$ that runs in time $n^{2-\gamma}$ and takes some advice string $\alpha(\mathcal{O},M) \in \{0,1\}^a$ for every matrix $M \in \mathbb{F}^{n\times n}$, then $a \geq \min\big\{\frac{\gamma}{5}\cdot\lfloor\log_p(1/\varepsilon)\rfloor\cdot\log_2 n,\ \sqrt{n}\big\}$.*

In particular, this theorem shows that if $p \leq 1/\varepsilon$ (i.e., $\lfloor\log_p(1/\varepsilon)\rfloor \geq 1$), the advice length a must be strictly positive, which indicates the non-existence of a uniform random self-reduction. This demonstrates the quantum advantage of the quantum random self-reduction of [AGGSS24] over uniform classical algorithms.

We complement our lower bound on the advice complexity by presenting two random self-reductions. First, we present a random self-reduction with the advice complexity of $O(\log(1/\varepsilon)\cdot\log n)$, which is optimal up to a constant factor when the field size p is constant.

Theorem 1.5. *There exists an ε-error-tolerant random self-reduction $A^{\mathcal{O},M}(\text{-};\alpha(\mathcal{O},M))$ for $\mathcal{L}_M$ that takes an advice string $\alpha(\mathcal{O},M)$ of length $O(\log(1/\varepsilon)\cdot\log n)$ for every matrix $M \in \mathbb{F}^{m\times n}$ over a finite field $\mathbb{F}$ of size p, runs in time $(n+m)\cdot(p/\varepsilon)^{O(\log(1/\varepsilon))}$, makes at most $(p/\varepsilon)^{O(\log(1/\varepsilon))}$ queries to $\mathcal{O}$, and reads $(p/\varepsilon)^{O(\log(1/\varepsilon))}$ rows and columns of M (in particular, it makes $(n+m)\cdot(p/\varepsilon)^{O(\log(1/\varepsilon))}$ queries to each entry of M).*

The advice complexity of Theorem 1.5 can be improved to $O(\log_p(1/\varepsilon)\cdot\log_2 n)$ in the special case where the set of inputs on which the oracle $\mathcal{O}$ correctly computes $\mathcal{L}_M$ contains a linear subspace of

[1]For example, if we take two distinct matrices $M_1, M_2 \in \mathbb{F}^{m\times n}$ and define the oracle $\mathcal{O}$ that outputs M_1x for half of the inputs $x \in \mathbb{F}^n$ and M_2x for the other half, then oracle access to $\mathcal{O}$ alone does not allow us to distinguish between M_1 and M_2.

high dimension (see Proposition 5.7). Note that this advice complexity is optimal up to a constant factor even for an arbitrary choice of p.

Second, we present a uniform random self-reduction when $p \geq 1/\varepsilon + 1 + o(1)$. This is tight up to the additive term 1+o(1) because of Theorem 1.4.

Theorem 1.6. *There exists a randomized oracle algorithm $A^{\mathcal{O},M}$ such that for every $\varepsilon > 0$ and every $\delta > 0$, for every matrix $M \in \mathbb{F}^{m\times n}$ over a finite field $\mathbb{F}$ of size at least $\frac{1}{\varepsilon} + 1 + \delta$, the algorithm $A^{\mathcal{O},M}$ is an ε-error-tolerant random self-reduction $A^{\mathcal{O},M}$ for $\mathcal{L}_M$, runs in time $O((m+n)\log^{1+o(1)} p/(\varepsilon^5\delta^2))$, makes $O(1/(\varepsilon^5\delta^2))$ queries to $\mathcal{O}$, and reads $O(1/(\varepsilon^5\delta^2))$ rows of M (in particular, it makes $O(n/(\varepsilon^5\delta^2))$ queries to each entry of M).*

One can apply Theorem 1.5 to any explicit matrix whose entry can be efficiently computed and obtain the equivalence between worst- and average-case algorithms with $O(\log n)$ bits of advice. For example, using the fact that each row and column of the Hadamard matrix can be computed in linear time, we obtain the following.

Corollary 1.7. *Let $\mathbb{F}$ be a finite field of size $O(1)$ and H_N be the $N \times N$ Hadamard matrix over $\mathbb{F}$. The following are equivalent for every constant $\varepsilon > 0$.*

1. *There exists a randomized linear-time algorithm for $\mathcal{L}_{H_N}$ with advice complexity $O(\log N)$.*
2. *There exists a randomized average-case linear-time algorithm with advice complexity $O(\log N)$ that computes $\mathcal{L}_{H_N}$ on an ε-fraction of inputs.*

2 Proof Overview

Notation For $n \in \mathbb{N}$, we denote by $[n] := \{1, \ldots, n\}$. For a finite set S, by $x \sim S$ we denote that x is chosen uniformly at random from S. For $k \in \mathbb{N}$, we denote by $\log^k n = (\log n)^k$ the k-th power of $\log n$. Unless otherwise specified, the base of the logarithm is 2. Unless stated otherwise, we ignore the cost of field operations over $\mathbb{F}_p$. That is, we treat each arithmetic operation in $\mathbb{F}_p$ as taking unit time (or constant circuit size). If one wishes to account for the bit-level complexity, all bounds can be multiplied by an additional factor of $\log^{1+o(1)} p$.

2.1 Nonuniform Reduction

The proof of [AGGSS24] is mathematically deep, leveraging additive combinatorics [AGGS22] and quantum computation. In contrast, our proof is *unexpectedly* simple. Indeed, Asadi, Golovnev, Gur, Shinkar, and Subramanian [AGGSS24] wrote:

> "Hirahara and Shimizu [HS23] recently provided a general paradigm that simplified, optimised, and unified known worst-case to average-case reductions for key problems of interest. Unfortunately, their approach inherently cannot capture the class of linear problems, as it does not admit the direct-product structure that is required for their techniques."

Surprisingly to the authors of [AGGSS24] (and the authors of the present paper), our key insight is that a certain "direct-product" structure is hidden inside the class of linear problems! In fact, our final algorithm does not rely on this insight, so readers interested only in the proof may skip the next paragraph.

Hidden XOR Structure. Our key observation is that the XOR structure is hidden in any linear problem. For simplicity, consider the binary field $\mathbb{F} = \mathrm{GF}(2)$. Let f be the function that maps x to Mx for some matrix $M \in \mathbb{F}^{m \times n}$, and $\mathcal{O}$ be an oracle that computes $f(x)$ on an ε-fraction of inputs x. Consider the k-wise XOR function $f^{\oplus k}$ defined as $f^{\oplus k}(x_1, \dots, x_k) \coloneqq f(x_1) \oplus \cdots \oplus f(x_k)$. Then, $\mathcal{O}$ can also be used to compute $f^{\oplus k}$ on an ε-fraction of inputs. The reason is that $f^{\oplus k}(x_1, \dots, x_k) = f(\sum_{i=1}^{k} x_i)$ by the linearity of f, and thus $C(\sum_{i=1}^{k} x_i) = f^{\oplus k}(x_1, \dots, x_k)$ for an ε-fraction of the k-tuple $(x_1, \dots, x_k)$. Yao's XOR lemma for multi-output functions [HS25] shows that such an average-case algorithm can be converted into a worst-case algorithm, provided that $\varepsilon > \frac{1}{2}$. To improve ε to an arbitrary small constant, we observe that the output of f can be efficiently verified using a non-uniform advice (in a way similar to Freivalds' algorithm for verifying matrix multiplication [Fre79]). In general, a direct product theorem can be proved for verifiable problems (see, e.g., [IJK09]); thus, we can apply the techniques of hardness amplification in the case of an arbitrary small $\varepsilon > 0$. Our proof was inspired by the worst-case to average-case reductions for matrix multiplication recently presented by Hirahara and Shimizu [HS25].

We present the proof based on the ideas above in Appendix A. However, the reduction can be further simplified, and the final algorithm does not rely on the k-wise direct product structure, which we present next.

Actual Proof Outline. Now we present an outline of our non-uniform reduction. The non-uniform reduction is a composition of two surprisingly simple reductions. Consider a matrix M, and let $\mathcal{O}$ be an oracle such that $\Pr_{x \sim \mathbb{F}^n}[\mathcal{O}(x) = Mx] \geq \varepsilon$.

The first reduction, denoted by $A_1^{\mathcal{O}}$, is a non-uniform randomized oracle algorithm that takes an instance x and a *Trevisan–Vadhan advice string*[2] α as input and outputs Mx with probability 0.9 for any $x \in \mathbb{F}^n$. Specifically, $A_1^{\mathcal{O}}(x)$ repeatedly samples $z \sim \mathbb{F}^n$ and computes $y = \mathcal{O}(x + z) - Mz$, where Mz is given as Trevisan–Vadhan advice. Since $x + z$ is distributed uniformly over $\mathbb{F}^n$, we have $y = Mx$ with probability ε in every trial. Moreover, we can verify whether $y = Mx$ or not with high probability in time $O(m)$ by repeatedly sampling a random vector $r \sim \mathbb{F}^m$ and checking whether $r^\top y = r^\top Mx$, where $r^\top M$ is also given as Trevisan–Vadhan advice. By the standard averaging argument, we can fix the random seed of $A_1^{\mathcal{O}}$ (and thus the Trevisan–Vadhan advice) to obtain a deterministic circuit $C_1^{\mathcal{O}}$ that outputs Mx for a 0.9-fraction of inputs $x \in \mathbb{F}^n$.

The second reduction takes $x \in \mathbb{F}^n$ as input, samples $z' \sim \mathbb{F}^n$, and outputs $C_1^{\mathcal{O}}(x + z') - C_1^{\mathcal{O}}(z')$. Since each of $x + z'$ and z' is distributed uniformly over $\mathbb{F}^n$, by the union bound, we have $C_1^{\mathcal{O}}(x + z') - C_1^{\mathcal{O}}(z') = Mx$ with probability at least 0.8 over the internal randomness of the reduction. Details can be found in Section 3.

2.2 Lower Bound of the Advice Complexity

To prove the lower bound of the advice complexity (Theorem 1.4), we use the incompressibility method (see, e.g., [LV19, Chapter 6]). A high-level idea is that if there exists a too-good-to-be-true random self-reduction $A^{\mathcal{O},M}$ that probes a few entries of matrices M and computes $\mathcal{L}_M$ in the worst case, then one can compress a random matrix M better than its entropy, which contradicts the information-theoretic lower bound. A proof similar to ours was used by De, Trevisan, and Tulsiani [DTT10] to prove time-space tradeoffs for inverting one-way functions.

[2] A non-uniform advice α is called a *Trevisan–Vadhan advice* if it depends on both the instance size and the random seed [TV07]. See Definition 3.2.

Our goal is to prove that the advice complexity is $\Omega(\ell \log n)$, where we define

$$\ell := \min\left\{ \lfloor \log_p(1/\varepsilon) \rfloor, \sqrt{n/\log n} \right\}.$$

Let a be the length of an advice string, and assume towards a contradiction that $a \le o(\ell \log n)$. We choose a uniformly random ℓ-sparse matrix $M \in \{0,1\}^{n \times n}$, i.e., a random matrix which contains exactly ℓ ones. Note that the entropy of M is $\log_2 \binom{n^2}{\ell} \approx 2\ell \log_2 n$ (if $\ell \ll n$). Since M contains at most ℓ nonzero entries, we have $\Pr_x[Mx = 0] \ge p^{-\ell} \ge \varepsilon$ (Lemma 4.2). In particular, letting $\mathcal{O}$ be the oracle that always output all-zero vector, we have $\Pr_x[Mx = \mathcal{O}(x)] \ge \varepsilon$. By the property of the random self-reduction (Definition 1.3), $A^{\mathcal{O},M}(\mathbf{1}_n; \alpha(\mathcal{O}, M)) = M\mathbf{1}_n$ holds with probability $\frac{2}{3}$, where $\mathbf{1}_n$ denotes the all-one vector of length n, and $\alpha(\mathcal{O}, M)$ is the advice string of length a.

Using this random self-reduction $A^{\mathcal{O},M}$, we may compress M to a string of length less than $\log_2 \binom{n^2}{\ell}$ as follows. For each fixed advice string α, since the oracle algorithm $A^{\mathcal{O},M}(\text{-}; \alpha)$ probes at most $q \le n^{2-\gamma}$ queries to M, the algorithm can find at most $n^{-\gamma} \cdot \ell$ nonzero entries of M in expectation (Lemma 4.3). By the union bound over all $\alpha \in \{0,1\}^a$, assuming that $a \le o(\ell \log n)$, the oracle algorithm $A^{\mathcal{O},M}(\text{-}; \alpha)$ can find at most $\ell/4$ nonzero entries of M for every advice string α. In particular, $A^{\mathcal{O},M}(\mathbf{1}_n; \alpha)$ is equal to $A^{\mathcal{O},M'}(\mathbf{1}_n; \alpha)$, where M' is the $\ell/4$-sparse matrix which contains the nonzero entries found by A. Moreover, we have $A^{\mathcal{O},M}(\mathbf{1}_n; \alpha(\mathcal{O}, M)) = M\mathbf{1}_n$ with probability $\frac{2}{3}$. Since M' can be described by $\log_2 \binom{n^2}{\ell/4} \approx \frac{\ell}{2} \log_2 n$ bits, we can also describe $M\mathbf{1}_n$ using $\frac{\ell}{2} \log_2 n + a$ bits, where a is the length of the advice string $\alpha(\mathcal{O}, M)$. Since $\ell \le o(\sqrt{n})$, each row of M contains at most one nonzero entry with high probability (Lemma 4.4), which implies that $M\mathbf{1}_n$ tells us which rows of M contain nonzero entries. Then M can be descried by additional $\ell \cdot \log_2 n$ bits, which specify the column of the nonzero entry for each such row. In total, M can be described by $a + \frac{\ell}{2} \log_2 n + \ell \log_2 n \ll 2\ell \log_2 n$ bits, which is a contradiction.

Details can be found in Section 4.

2.3 Uniform Reduction

The proof of uniform reductions (Theorems 1.5 and 1.6) proceeds in two main steps. Let $\mathcal{O}$ be an oracle and $M \in \mathbb{F}^{m \times n}$ be a matrix such that $\Pr_{x \sim \mathbb{F}^n}[\mathcal{O}(x) = Mx] \ge \varepsilon$.

1. We first design a randomized algorithm A_1 that, given oracle access to $\mathcal{O}$ and M, computes Mx with probability $f(p, \varepsilon)$ for every x, where f is an appropriate function depending on $p = |\mathbb{F}|$ and $\varepsilon > 0$ (Lemmas 5.2 and 5.3).

2. We then construct another randomized oracle algorithm A_2 that, using A_1 as a subroutine, computes Mx with probability at least $2/3$ for every input x (Lemma 5.1).

Step 1 is proved separately for the cases where the field size $|\mathbb{F}|$ is large and where it is small, while Step 2 is proved by a common argument that applies to both cases. In what follows, we describe the proof sketch of each step. See Section 5 for the details.

Step 1 over Large Field (Section 5.2). The reduction of Step 1 for the large field case follows from the standard technique of polynomial interpolation [AGGS22; AGGSS24].

Let $x \in \mathbb{F}^n$ be the input. For $t \in \mathbb{F}$, let $\ell(t) = x + tr$, where $r \sim \mathbb{F}^n$ is a random vector. The oracle algorithm $A^{\mathcal{O}}$ chooses two random distinct points $t_1, t_2 \in \mathbb{F}$ uniformly at random and

compute $Mx = M\ell(0)$ by interpolating $\mathcal{O}(\ell(t_1))$ and $\mathcal{O}(\ell(t_2))$. Let X be the number of $t \in \mathbb{F} \setminus \{0\}$ such that $\mathcal{O}(\ell(t)) = M\ell(t)$. Since each $\ell(t)$ for $t \neq 0$ is distributed uniformly over $\mathbb{F}^n$, the probability that $\mathcal{O}(\ell(t)) = M\ell(t)$ is at least ε. Thus, by Markov's inequality, with probability $\Omega(\gamma\varepsilon)$, we have $X \geq (1-\gamma)\varepsilon p$ for any $\gamma > 0$. Conditioned on this event, if we pick up two distinct points $t_1 \neq t_2$ randomly, the probability that $\mathcal{O}(\ell(t_1)) = M\ell(t_1)$ and $\mathcal{O}(\ell(t_2)) = M\ell(t_2)$ is $\Omega((X/p)^2)$. If we set $\gamma = \delta\varepsilon$ and from $p \geq 1/\varepsilon + 1 + \delta$, this probability is at least $\Omega(\delta\varepsilon^3)$. Therefore, the algorithm $A^{\mathcal{O}}$ outputs Mx with probability $\Omega(\gamma\varepsilon) \cdot \Omega(\delta\varepsilon^3) = \Omega(\delta^2\varepsilon^5)$ over the choice of r, t_1, t_2.

Step 1 over Small Field (Section 5.3). The reduction of Lemma 5.3 consists of two parts. First, using the given oracle $\mathcal{O}$, we design an algorithm that computes Mx for all x belonging to some linear subspace $V \subseteq \mathbb{F}^n$ of dimension $\dim V = n - O(\log(1/\varepsilon))$ (Lemma 5.4). Next, using such an algorithm, we give an algorithm that computes Mx for all $x \in \mathbb{F}^n$ (Lemma 5.6). The $O(\log n)$ bits of advice is used in the second step.

For the first part, we apply the additive-combinatorics-based framework of Asadi, Golovnev, Gur, and Shinkar [AGGS22]. Let $S \subseteq \mathbb{F}^n$ denote the set of inputs on which the oracle $\mathcal{O}$ outputs the correct value, i.e., $S = \{x \in \mathbb{F}^n \mid \mathcal{O}(x) = Mx\}$. By the assumption on $\mathcal{O}$, we have $|S| \geq \varepsilon \cdot |\mathbb{F}|^n$. Then, by a probabilistic version of the Bogolyubov–Ruzsa lemma (Lemma 5.5) from [GSS24, Lemma 8], there exist an integer $t = O(\log(1/\varepsilon))$ and a subspace $V \subseteq \mathbb{F}^n$ of dimension $\dim(V) \geq n - O(\log(1/\varepsilon))$ such that for every $x \in V$,

$$\Pr\left[a_1, \ldots, a_t, b_1, \ldots, b_t \in S \,\middle|\, \sum_{i=1}^{t}(a_i - b_i) = x\right] \geq \varepsilon^{2t+1}.$$

where the probability is taken over $a_1, \ldots, a_t, b_1, \ldots, b_t \sim \mathbb{F}^n$.

Using this lemma, we can design a randomized algorithm that computes Mx for every $x \in V$ as follows: sample random vectors $a_1, \ldots, a_t, b_1, \ldots, b_t \in \mathbb{F}^n$ satisfying $a_1 + \cdots + a_t - b_1 - \cdots - b_t = x$, and output

$$\mathcal{O}(a_1) + \cdots + \mathcal{O}(a_t) - \mathcal{O}(b_1) - \cdots - \mathcal{O}(b_t).$$

By the guarantee of the lemma, this algorithm outputs Mx with probability $\varepsilon^{2t+1} = \varepsilon^{O(\log(1/\varepsilon))}$ for every $x \in V$.

For the second part, consider the subspace $V \subseteq \mathbb{F}^n$ obtained above, whose dimension is $\dim V = n - c$ for some $c = O(\log(1/\varepsilon))$. We choose standard basis vectors $e_{i_1}, \ldots, e_{i_c}$ such that $V \cup \{e_{i_1}, \ldots, e_{i_c}\}$ spans the entire space $\mathbb{F}^n$. The algorithm receives the indices $i_1, \ldots, i_c$ as an advice string of length $O(c \log n)$. Then every vector $x \in \mathbb{F}^n$ can be expressed as a linear combination of these basis vectors and a vector in V, namely

$$x = a_1 e_{i_1} + \cdots + a_c e_{i_c} + v,$$

where $v \in V$ and $a_1, \ldots, a_c \in \mathbb{F}$. The final algorithm simply guesses $a_1, \ldots, a_c$ randomly, which is correct with probability $|\mathbb{F}|^{-c}$, and then calculates Mx by adding $\sum_{k=1}^{c} a_k M e_{i_k} + Mv$, where Me_{i_k} can be calculated by querying the i_k-th column of M and Mv can be calculated by the algorithm of the first part.

Step 2. Amplifying the Success Probability (Section 5.1). Let A_1 be the randomized algorithm from Step 1, which computes Mx with success probability $f(p, \varepsilon)$. We run A_1 on the same

input x independently $\ell = O(1/f(p,\varepsilon))$ times, and collect the resulting outputs as $L = \{y_1, \dots, y_\ell\}$. With probability at least $2/3$, one of these vectors equals the correct vector Mx. Thus, our goal is to identify which element in L is equal to Mx.

Asadi, Golovnev, Gur, Shinkar, and Subramanian [AGGSS24] designed a quantum algorithm that verifies the correctness of a candidate solution in $O(n^{1.5})$ time. We instead take a different, simpler, and non-quantum approach. We arbitrarily select two distinct vectors $y, y' \in L$, and find an index i such that $y_i \neq y'_i$. We then query the i-th row of M through oracle access and compute the i-th coordinate of Mx. If y_i (resp. y'_i) differs from $(Mx)_i$, we remove y (resp. y') from the list L. We repeat this elimination process until the size of L is reduced to 1, and output the unique element that remains in L.

Since each iteration removes at least one of y or y', the total number of iterations is at most $|L| \leq \ell$. In particular, the number of oracle queries to M is bounded by $\ell \cdot n$. Moreover, as the correct vector Mx is never eliminated (whenever it is contained in L), the algorithm outputs Mx with probability at least $2/3$.

2.4 Related Work

Newton, Richelson, and Wilson [NRW23] presented a high-dimensional version of the theorem of Goldreich and Levin [GL89]. They gave an efficient (but not linear-time) algorithm that, given oracle $\mathcal{O}$ that agrees with an unknown matrix M on an ε-fraction of inputs (i.e., $\Pr_{x\sim\mathbb{F}^n}[\mathcal{O}(x) = Mx] \geq \varepsilon$), outputs a matrix M' that agrees with $\mathcal{O}$ with $\varepsilon^{\Omega(1)}$-fraction of inputs [NRW23, Theorem 2]. This differs from our setting in that we assume that a random self-reduction "knows" M (i.e., is given oracle access to each entry of M) and runs in *linear time*.

Asadi, Golovnev, Gur, and Shinkar [AGGS22] developed a framework for constructing error-tolerant worst-case to average-case reductions based on additive combinatorics, and presented a worst-case to average-case reduction for all linear problems in the computational model of *data structures*. Specifically, for each matrix $M \in \mathbb{F}^{m\times n}$, they considered the problem of preprocessing $x \in \mathbb{F}^n$ so that one can answer a query $i \in [m]$ with the i-th element of Mx by probing a few bits of the preprocessed data structure. This problem of constructing a data structure was shown to be equivalent to the average-case version in which x is chosen unifromly at random from $\mathbb{F}^n$.

3 Nonuniform Reduction

Throughout this section, we use a Boolean circuit as the computational model, but it works for other computational models such as Turing machine.

Theorem 3.1. *Let $\varepsilon > 0$ be an arbitrary parameter. There exists a randomized oracle circuit $C^{\mathcal{O}}$ of size $O((n+m)\cdot\log(1/\varepsilon)/\varepsilon)$ such that, for any matrix $M \in \mathbb{F}^{m\times n}$ and any oracle $\mathcal{O}$ satisfying*

$$\Pr_{x\sim\mathbb{F}^n}[\mathcal{O}(x) = Mx] \geq \varepsilon,$$

it holds that

$$\forall x \in \mathbb{F}^n, \quad \Pr_{C^{\mathcal{O}}}[C^{\mathcal{O}}(x) = Mx] \geq \frac{2}{3},$$

where the probability is taken over the internal randomness of $C^{\mathcal{O}}$. Moreover, $C^{\mathcal{O}}$ makes at most $O(\log(1/\varepsilon)/\varepsilon)$ queries to $\mathcal{O}$.

Theorem 1.1 follows immediately from this by replacing $\mathcal{O}$ with an average-case circuit that computes the linear map $\mathcal{L}_M \colon x \mapsto Mx$. The proof consists of two parts. First, we present a hardness amplification technique that transforms the weak average-case solver $\mathcal{O}$ into a strong nonuniform average-case solver $\mathcal{O}'$ that runs in time $O(n \log(1/\varepsilon)/\varepsilon)$ and solves a $(1-\delta)$-fraction of instances for any fixed constant $\delta > 0$. Then, we transform the strong solver $\mathcal{O}'$ into a worst-case solver based on the well-known worst-case-to-average-case reduction of Blum, Luby, and Rubinfeld [BLR93].

3.1 Nonuniform Algorithm with Trevisan–Vadhan Advice

To begin with, we introduce the concept of Trevisan–Vadhan advice, which plays a key role in the proof of Theorem 3.1.

Recall that a nonuniform algorithm is an algorithm that takes both instance and *advice string* as input, where the advice string is a string that depends on the size of the instance. A *Trevisan–Vadhan advice* for a nonuniform algorithm A is an advice string α that depends on both the size of the instance and the random seed.

Definition 3.2 (Trevisan–Vadhan advice [TV07])**.** *For a function $\alpha \colon \mathbb{N} \times \{0,1\}^* \to \{0,1\}^*$ and a randomized algorithm A, let B be the probabilistic function defined as $B(x) \coloneqq A(x; r, \alpha(|x|, r))$ for a uniformly random r. We call B* a randomized algorithm with Trevisan–Vadhan advice α.

In the following, we show that any nonuniform randomized algorithm that is given Trevisan–Vadhan advice can be simulated by a circuit that has the same running time and the same success probability.

Lemma 3.3. *Let $f \colon \{0,1\}^* \to \{0,1\}^*$ be any function. For any nonuniform randomized algorithm $A(x; r, \alpha)$ that is given Trevisan–Vadhan advice $\alpha = \alpha(|x|, r)$ and runs in time T, there exists a circuit C of size $O(T)$ that satisfies*

$$\Pr_{x \sim \{0,1\}^n}[C(x) = f(x)] \geq \Pr_{x \sim \{0,1\}^n, r}[A(x; r, \alpha) = f(x)].$$

Proof. Suppose that the length of random seed r is at most m when the input length $|x|$ is n. By averaging, there exists a string $r^* \in \{0,1\}^m$ such that

$$\Pr_{x \sim \{0,1\}^n}[A(x; r^*, \alpha(|x|, r^*)) = f(x)] \geq \Pr_{\substack{x \in \{0,1\}^n \\ r \in \{0,1\}^m}}[A(x; r, \alpha) = f(x)].$$

Let $A'(x; \alpha'(|x|))$ be the deterministic nonuniform algorithm defined by $A'(x; \alpha') = A(x; r^*, \alpha(|x|, r^*))$, where the advice string is $\alpha'(|x|) = (r^*, \alpha(|x|, r^*))$. Then, A' has the same running time as A and satisfies

$$\Pr_{x \sim \{0,1\}^n}[A'(x; \alpha') = f(x)] \geq \Pr_{x \in \{0,1\}^n, r \in \{0,1\}^m}[A(x; r, \alpha) = f(x)].$$

Since A' is deterministic and runs in time T, there exists a circuit C of size $O(T)$ such that $C(x) = A'(x; \alpha'(|x|))$ for all $x \in \{0,1\}^n$. □

We present an efficient nonuniform verification algorithm based on the Freivalds' verification [Fre79] that takes Trevisan–Vadhan advice.

Lemma 3.4. *Let $\mathbb{F}$ be a finite field and $M \in \mathbb{F}^{m\times n}$ be a matrix. For every $\gamma \in (0,1)$, there exists a deterministic $O((n+m)\log(1/\gamma))$-time algorithm V that satisfies for every $x \in \mathbb{F}^n$ and $y \in \mathbb{F}^m$ that*

$$\Pr_{\substack{r_1,\ldots,r_\ell \sim \mathbb{F}^m \\ \forall i\in[\ell],\, \alpha_i = r_i^\top M}} [V(x,y,r_1,\ldots,r_\ell,\alpha_1,\ldots,\alpha_\ell) = 1] \begin{cases} = 1 & \text{if } Mx = y \\ \le \gamma & \text{if } Mx \neq y, \end{cases}$$

where $\ell = \lceil \log_2(1/\gamma) \rceil$.

Proof. The algorithm V outputs 1 if and only if $\alpha_i x = r_i^\top y$ for all $i \in [\ell]$. If $Mx = y$, then clearly V accepts with probability 1. On the other hand, if $Mx \neq y$, then for each $i \in [\ell]$, the algorithm V outputs 0 with probability $\Pr_{r_i}[r_i^\top Mx \neq r_i^\top y] \ge 1 - 1/|\mathbb{F}| \ge 1/2$. Since $r_1,\ldots,r_\ell$ are independent, we conclude that V outputs 1 with probability at most γ. □

Note that the algorithm V in Lemma 3.4 can be seen as a nonuniform randomized verification algorithm that takes Trevisan–Vadhan advice by viewing $r_1,\ldots,r_\ell$ as internal randomness of V and $\alpha_1,\ldots,\alpha_\ell$ as Trevisan–Vadhan advice.

3.2 Step 1. Hardness Amplification

We show how to transform a weak average-case solver $\mathcal{O}$ into a strong non-uniform average-case solver $\mathcal{O}'$.

Lemma 3.5. *Let $\varepsilon > 0$ be an arbitrary parameter and $\delta > 0$ be any constant. There exists an oracle circuit $C_0^{\mathcal{O}}$ of size $O((n+m)\log(1/\varepsilon)/\varepsilon)$ such that, for any matrix $M \in \mathbb{F}^{m\times n}$ and any oracle $\mathcal{O}$ satisfying*

$$\Pr_{x\sim\mathbb{F}^n}[\mathcal{O}(x) = Mx] \ge \varepsilon,$$

it holds that

$$\Pr_{x\sim\mathbb{F}^n}\left[C_0^{\mathcal{O}}(x) = Mx\right] \ge 1 - \delta.$$

Moreover, $C_0^{\mathcal{O}}$ makes at most $O(1/\varepsilon)$ queries to $\mathcal{O}$.

Proof. We present a nonuniform randomized oracle algorithm $A_{TV}^{\mathcal{O}}$ that takes Trevisan–Vadhan advice and runs in time $O((n+m)\log(1/\varepsilon)/\varepsilon)$. The desired circuit $C_0^{\mathcal{O}}$ is obtained by applying Lemma 3.3 to $A_{TV}^{\mathcal{O}}$.

Let V be the non-uniform verification algorithm given in Lemma 3.4 for M and $\gamma = \frac{\delta}{2T}$ and $k = \lceil \log_2(1/\gamma) \rceil$, where $T = O(1/\varepsilon)$. The description of the non-uniform algorithm $A_{TV}^{\mathcal{O}}$ is given in Algorithm 1.

Clearly, $A_{TV}^{\mathcal{O}}$ runs in time $O((m+n+s)Tk) = O((m+n+s)\log(1/\varepsilon)/\varepsilon)$ and receives $O((m+n+s)\log(1/\varepsilon)/\varepsilon)$ bits of Trevisan-Vadhan advice. We claim that for every $x \in \mathbb{F}^n$,

$$\Pr_{A_{TV}^{\mathcal{O}}}\left[A_{TV}^{\mathcal{O}}(x) = Mx\right] \ge 1 - \delta,$$

where the internal randomness of $A_{TV}^{\mathcal{O}}$ accounts for the choice of $\vec{r} = ((z_i)_{i\in[T]}, (r_{i,j})_{i\in[T],j\in[k]}$ specified in Algorithm 1. Since each $z_i \sim \mathbb{F}^n$ is chosen uniformly at random, the marginal distribution of $x + z_i$ is uniform over $\mathbb{F}^n$. Therefore, with probability at least $1 - \delta/2$, we obtain

Algorithm 1 Nonuniform algorithm $A^{\mathcal{O}}_{TV}(x; r, \alpha)$

Input: A vector $x \in \mathbb{F}^n$, random seed $\vec{r} = ((z_i)_{i\in[T]}, (r_{i,j})_{i\in[T],j\in[k]})$ for $z_i \sim \mathbb{F}^n$ and $r_{i,j} \sim \mathbb{F}^m$, and Trevisan–Vadhan advice $(Mz_i)_{i\in[T]}$ and $(r_{i,j}^\top M)_{i\in[T],j\in[k]}$.

1: **for** $i = 1$ to T **do**
2: Sample $z_i \sim \mathbb{F}^n$ uniformly at random and set $y = \mathcal{O}(x + z_i) - Mz_i$, where Mz_i is given as the Trevisan–Vadhan advice.
3: Let $\alpha_{i,j} = r_{i,j}^\top M$ be the Trevisan-Vadhan advice for $j \in [k]$.
4: **if** $V(x, y; r_{i,1}, \ldots, r_{i,k}, \alpha_{i,1}, \ldots, \alpha_{i,k})$ outputs 1 **then**
5: Output y
6: **end if**
7: **end for**
8: Output $\perp$ (meaning that $A^{\mathcal{O}}_{TV}$ fails to compute Mx)

$y = \mathcal{O}(x + z_i) - Mz_i = Mx$ in some i-th trial ($i \in [T]$). On the other hand, if $Mx \neq y$, then V outputs 0 with probability at least $1 - \frac{\delta}{2T}$ at Line 4 of each trial. By the union bound over T trials, with probability at least $1 - \delta/2$, the verifier V outputs 0 for any of T trials with $Mx \neq y$. Since the choice of z_i and $(r_{i,1}, \ldots, r_{i,k})$ are independent over $i \in [T]$, for any $x \in \mathbb{F}^n$, we have

$$\Pr_{A^{\mathcal{O}}_{TV}}\left[A^{\mathcal{O}}_{TV}(x) = Mx\right] \geq \left(1 - \frac{\delta}{2}\right)^2 \geq 1 - \delta.$$

From Lemma 3.3, we obtain an $O((n + m)\log(1/\varepsilon)/\varepsilon)$-time nonuniform randomized oracle algorithm $A^{\mathcal{O}}$ that satisfies

$$\Pr_{x\sim\mathbb{F}^n}\left[A^{\mathcal{O}}(x; \alpha) = Mx\right] \geq \Pr_{\substack{x\sim\mathbb{F}^n \\ A^{\mathcal{O}}_{TV}}}\left[A^{\mathcal{O}}_{TV}(x; \alpha) = Mx\right] \geq 1 - \delta.$$

This completes the proof. □

3.3 Step 2. Worst-Case-to-Average-Case Reduction

In the following, we transform the circuit $C^{\mathcal{O}}_0$ of Lemma 3.5 into the desired worst-case solver $C^{\mathcal{O}}$ using the idea from *uniform* worst-case-to-average-case reduction of Blum, Luby, and Rubinfeld [BLR93].

Lemma 3.6. *Let $\varepsilon > 0$ be an arbitrary parameter. There exists an $O(n + m)$-time randomized two-query oracle algorithm $A^{\mathcal{O}}$ such that, for any matrix $M \in \mathbb{F}^{m\times n}$ and any oracle $\mathcal{O}$ satisfying*

$$\Pr_{x\sim\mathbb{F}^n}[\mathcal{O}(x) = Mx] \geq 0.9,$$

it holds for every $x \in \mathbb{F}^n$ that

$$\Pr_{A^{\mathcal{O}}}\left[A^{\mathcal{O}}(x; \alpha) = Mx\right] \geq 0.8,$$

where the probability is taken over internal randomness of $A^{\mathcal{O}}$.

Proof. The algorithm $A^{\mathcal{O}}(x)$ on input x just samples $z \sim \mathbb{F}^n$ uniformly at random and then output $\mathcal{O}(z) + \mathcal{O}(x-z)$. Since the marginal distributions of z and $x-z$ are uniform over $\mathbb{F}^n$, by the union bound, with probability at least 0.8, we have $\mathcal{O}(z) = Mz$ and $\mathcal{O}(x-z) = M(x-z)$, in which case $\mathcal{O}(x) = Mx$. □

We are now ready to prove Theorem 3.1.

Proof of Theorem 3.1. By Lemma 3.5 for $\delta = 0.1$, there exists a circuit $C_0^{\mathcal{O}}$ of size $O((m+n)\log(1/\varepsilon)/\varepsilon)$ that satisfies

$$\Pr_{x \sim \mathbb{F}^n}\left[C_0^{\mathcal{O}}(x) = Mx\right] \geq 0.9.$$

Then, from Lemma 3.6 using $C_0^{\mathcal{O}}$ as the oracle, we obtain a randomized oracle circuit $C^{\mathcal{O}}$ of size $O((m+n)\log(1/\varepsilon)/\varepsilon)$ such that for every $x \in \mathbb{F}^n$,

$$\Pr_{C^{\mathcal{O}}}\left[C^{\mathcal{O}}(x;\alpha) = Mx\right] \geq 0.8,$$

where the probability is taken over the internal randomness of $C^{\mathcal{O}}$. Replacing $\mathcal{O}$ with the circuit $C^{\mathcal{O}}$, we obtain the desired circuit. □

4 Advice Lower Bound

In this section, we prove Theorem 1.4. In fact, we can prove the result even for the oracle $\mathcal{O}$ that always outputs all-zero vector for every input. Throughout this section, let $\mathcal{O}$ denote this oracle. For notational convenience, we suppress the oracle $\mathcal{O}$ in the superscript and simply write A^M in place of $A^{\mathcal{O},M}$. Using this notation, we state a version stronger than Theorem 1.4.

Theorem 4.1. *For every constant $\gamma > 0$, for all large $n \in \mathbb{N}$ and for every $\varepsilon > 0$, every $a \in \mathbb{N}$ and every finite field $\mathbb{F}$ of size p, if there exists an ε-error-tolerant random self-reduction $A^M(\text{-};\alpha(M))$ for $\mathcal{L}_M$ that runs in time $n^{2-\gamma}$ and takes some advice string $\alpha(M) \in \{0,1\}^a$ for every matrix $M \in \mathbb{F}^{n\times n}$, then $a \geq \min\left\{\frac{\gamma}{5} \cdot \lfloor \log_p(1/\varepsilon) \rfloor \cdot \log_2 n,\ \sqrt{n}\right\}$.*

Notation. For a vector $x = (x_1, \ldots, x_n) \in \mathbb{F}^n$ and $i \in [n]$, we write $x_{\leq i} = (x_1, \ldots, x_i)$. Let $\mathrm{wt}(x)$ denote the Hamming weight of $x \in \mathbb{F}^n$, i.e., the number of nonzero entries of x. Let $\binom{[n]}{\leq k} \subseteq \{0,1\}^n$ denote the set of all strings $x \in \{0,1\}^n$ such that $\mathrm{wt}(x) \leq k$. Let $\binom{n}{\leq k} = \sum_{0 \leq i \leq k} \binom{n}{i}$ be the size of $\binom{[n]}{\leq k}$. A *ℓ-sparse matrix* $M \in \{0,1\}^{n\times n} \subseteq \mathbb{F}^{n\times n}$ is a matrix that contains exactly ℓ ones.

Define $\ell := \min\left\{\lfloor \log_p(1/\varepsilon) \rfloor, \sqrt{n/\log n}\right\}$. We choose a uniformly random ℓ-sparse matrix $M \in \{0,1\}^{n\times n}$, i.e., a random matrix which contains exactly ℓ ones. Since Theorem 4.1 is trivial if $\lfloor \log_p(1/\varepsilon) \rfloor = 0$, we assume $\ell \geq 1$ in what follows.

4.1 Properties of Random Sparse Matrices

In this subsection, we prove properties of random sparse matrices that will be used in the proof of Theorem 4.1.

First, we observe that any sparse matrix has a certain agreement with the fixed oracle $\mathcal{O}$ that outputs the all-zero vector for all inputs.

Lemma 4.2. *Let $M \in \mathbb{F}^{n\times n}$ be a matrix that contains at most ℓ nonzero entries. Then,*

$$\Pr_{x\sim\mathbb{F}^n}[Mx = 0] \geq \frac{1}{p^\ell}.$$

Proof. Since M has at most ℓ nonzero entries, the rank of M is at most ℓ. By the rank-nullity theorem, we obtain $\dim(\ker(M)) \geq n - \ell$, which implies

$$\Pr_{x\sim\mathbb{F}^n}[Mx = 0] = \frac{p^{\dim(\ker(M))}}{p^n} \geq \frac{1}{p^\ell}. \qquad \square$$

By the choice of ℓ, we have $\log_p(1/\varepsilon) \geq \ell$. Lemma 4.2 shows that the oracle $\mathcal{O}$ that always outputs the all-zero vector agrees with Mx on an ε-fraction of inputs x. Thus, the random self-reduction $A^M(\text{-}; \alpha(M))$ correctly computes $\mathcal{L}_M$ in the worst case with probability $\frac{2}{3}$ for any ℓ-sparse matrix M.

We show that any algorithm A^M that makes q queries to a random sparse matrix M can find at most $O\big(q\ell/n^2\big)$ nonzero entries in M. In what follows, we assume without loss of generality that an oracle algorithm A^M does not make duplicated queries.

Lemma 4.3. *Let $q \in \mathbb{N}$ and $n \in \mathbb{N}$ be parameters such that $q \leq n^2/2$. Let M be a random ℓ-sparse matrix and A^M be an arbitrary* deterministic *adaptive oracle algorithm that makes at most q distinct queries to M. Let $W = (W_1, \ldots, W_q) \in \{0,1\}^q$ denote the bits which the oracle algorithm A^M receives from the oracle M in the order they are obtained. Then, for any $0 \leq \theta \leq \ell$, it holds that*

$$\Pr_M[\mathrm{wt}(W) \geq \theta] \leq \left(\frac{2eq\ell}{n^2\theta}\right)^\theta.$$

Proof. Let $(i_1, j_1), \ldots, (i_q, j_q) \in [n]^2$ denote the oracle queries of A^M. Note that these are random variables that depend on the answer sequence W from the oracle M; in particular, each query (i_k, j_k) depends on the previous $(k-1)$ answers $W_{\leq k-1}$ from the oracle.

For every $k \in [q]$ and every $w \in \{0,1\}^{k-1}$, consider the probability that $W_k = 1$ (equivalently, the (i_k, j_k)-th entry of M is 1), conditioned on the event that $W_{\leq k-1} = w$. Under this condition, (i_k, j_k) is fixed because (i_k, j_k) depends only on $W_{\leq k-1} = w$. Since there remain $\ell - \mathrm{wt}(w)$ ones among the unseen positions in $[n]^2 \setminus \{(i_1, j_1), \ldots, (i_{k-1}, j_{k-1})\}$, we have

$$\Pr_M[W_k = 1 \mid W_{\leq k-1} = w] = \frac{\ell - \mathrm{wt}(w)}{n^2 - (k-1)} \leq \frac{\ell}{n^2 - q} \leq \frac{2\ell}{n^2}.$$

Therefore, we obtain

$$\begin{aligned}
\Pr_M[\mathrm{wt}(W) \geq \theta] &= \Pr_M[\exists k_1 < \cdots < k_\theta,\, W_{k_1} = \cdots = W_{k_\theta} = 1] \\
&\leq \sum_{k_1<\cdots<k_\theta} \Pr_M[W_{k_1} = \cdots = W_{k_\theta} = 1] \\
&= \sum_{k_1<\cdots<k_\theta} \prod_{s=1}^{\theta} \Pr_M\big[W_{k_s} = 1 \mid W_{k_1} = \cdots = W_{k_{s-1}} = 1\big] \\
&\leq \sum_{k_1<\cdots<k_\theta} \prod_{s=1}^{\theta} \max_{w\in\{0,1\}^{k_s-1}} \Pr_M[W_{k_s} = 1 \mid W_{\leq k_s-1} = w] \\
&\leq \sum_{k_1<\cdots<k_\theta} \prod_{s=1}^{\theta} \frac{2\ell}{n^2} \\
&= \underbrace{\binom{q}{\theta}}_{\leq \left(\frac{eq}{\theta}\right)^\theta} \left(\frac{2\ell}{n^2}\right)^\theta \\
&\leq \left(\frac{2eq}{n^2}\cdot\frac{\ell}{\theta}\right)^\theta.
\end{aligned}$$

□

Next, we observe that a random ℓ-sparse matrix has at most one nonzero entry in each row with high probability.

Lemma 4.4. *Assume $\ell \leq \delta\sqrt{n}$. Let $M \in \{0,1\}^{n\times n}$ be a random ℓ-sparse matrix and $m_1, \ldots, m_n$ be the row vectors of M. Then, we have*

$$\Pr_M[\forall i \in [n],\, \mathrm{wt}(m_i) \leq 1] \geq 1 - O(\delta^2).$$

Proof. Fix $i \in [n]$. Let A^M be the algorithm that probes all the n entries in the i-th row of M, which makes exactly n queries. Applying Lemma 4.3 to A^M, we obtain

$$\Pr_M[\mathrm{wt}(m_i) \geq 2] \leq \left(\frac{2en\ell}{n^2\cdot 2}\right)^2 \leq O(\delta^2/n).$$

The claim follows from the union bound over $i \in [n]$. □

4.2 Description of M

We now present a short description of M using the random self-reduction A^M. More precisely, we show that a constant fraction of random ℓ-sparse matrices can be described succinctly. Let $\mathbf{1}_n$ denote the all-one vector of length n.

Lemma 4.5. *For all large n, assume that for every ℓ-sparse matrix $M \in \{0,1\}^{n\times n}$, there exists an advice string $\alpha(M) \in \{0,1\}^a$ such that*

$$\Pr_r\big[A^M(\mathbf{1}_n; r, \alpha(M)) = M\mathbf{1}_n\big] \geq \frac{2}{3},$$

where r denotes the internal randomness of A^M. Let $q \le n^{2-\gamma}$ be the number of oracle accesses made by A^M and let $\ell = o(\sqrt{n})$. Define $\eta := \frac{1}{4}$. Assume that $a \le \frac{\gamma}{5}\ell \log_2 n$. Then, there exist a set $G \subseteq \{0,1\}^{n\times n}$ with $|G| \ge \frac{1}{2}\cdot\binom{n^2}{\ell}$ and a function $\mathsf{Dec}\colon \binom{[q]}{\le \eta\ell} \times \{0,1\}^a \times [n]^\ell \to \{0,1\}^{n\times n}$ such that G is contained in the range of Dec, that is,

$$G \subseteq \left\{ \mathsf{Dec}(w, \alpha, j_1, \dots, j_\ell) \,\middle|\, w \in \binom{[q]}{\le \eta\ell}, \alpha \in \{0,1\}^a, j_1, \dots, j_\ell \in [n] \right\}$$

Proof. Fix an arbitrary $\alpha \in \{0,1\}^a$ and r. Applying Lemma 4.3 to $A^M(\mathbf{1}_n; r, \alpha)$ and $\theta := \eta\ell$, we have

$$\Pr_M\left[A^M(\mathbf{1}_n; r, \alpha) \text{ receives more than } \eta\ell \text{ ones from } M\right] \le \left(\frac{2eq\ell}{n^2\eta\ell}\right)^{\eta\ell} = 2^{O(\ell)} \cdot n^{-\gamma\eta\ell}.$$

By the union bound over all $\alpha \in \{0,1\}^a$, for every r, we have

$$\Pr_M\left[\exists \alpha \in \{0,1\}^a,\ A^M(\mathbf{1}_n; r, \alpha) \text{ receives more than } \eta\ell \text{ ones from } M\right] \le 2^a \cdot 2^{O(\ell)} \cdot n^{-\eta\gamma\ell} \le o(1),$$

where the last inequality holds because $a \le \frac{\gamma}{5}\ell\log_2 n \le \frac{\gamma}{4}\ell\log_2 n - O(\ell)$ for all large $n \in \mathbb{N}$. In particular, by taking the expectation over a random r, we obtain

$$\Pr_{M,r}\left[\forall \alpha \in \{0,1\}^a,\ A^M(\mathbf{1}_n; r, \alpha) \text{ receives at most } \eta\ell \text{ ones from } M\right] \ge 1 - o(1).$$

By the union bound and an averaging argument for r, there exists a fixed random seed r^* such that

$$\Pr_M\left[\begin{array}{l} A^M(\mathbf{1}_n; r^*, \alpha(M)) \text{ receives at most } \eta\ell \text{ ones from } M \text{ and} \\ A^M(\mathbf{1}_n; r^*, \alpha(M)) = M\mathbf{1}_n \end{array}\right] \ge \frac{2}{3} - o(1). \tag{1}$$

In the rest of the proof, we fix such r^*. Let $G \subseteq \{0,1\}^{n\times n}$ be the set of all matrices M such that

(i) $A^M(\mathbf{1}_n; r^*, \alpha(M)) = M\mathbf{1}_n$ and $A^M(\mathbf{1}_n; r^*, \alpha(M))$ receives at most $\eta\ell$ ones from M.

(ii) Each row of M contains at most one nonzero entry.

By Lemma 4.4 and Eq. (1), we have $\Pr_M[M \in G] \ge 2/3 - o(1) - o(1)$. In particular, for all sufficiently large n, we have $|G| \ge \frac{1}{2}\cdot\binom{n^2}{\ell}$.

Now we define the function $\mathsf{Dec}\colon \binom{[q]}{\le\eta\ell} \times \{0,1\}^a \times [n]^\ell \to \{0,1\}^{n\times n}$ as follows.

1. Given $w \in \binom{[q]}{\le\eta\ell}$, $\alpha \in \{0,1\}^a$, and $j_1, \dots, j_\ell \in [n]$, run the oracle algorithm $A^{(\cdot)}(\mathbf{1}_n; r^*, \alpha)$ by answering the i-th query with w_i, and let $y \in \mathbb{F}^n$ be the output.
2. Let $1 \le i_1 < \cdots < i_\ell \le n$ be the indices of non-zero row vectors in y (if the number of non-zero row vectors is not ℓ, then output the all-zero matrix).
3. Set $S = \{(i_1, j_1), \dots, (i_\ell, j_\ell)\}$ and output χ_S, where $\chi_S \in \{0,1\}^{n\times n}$ is the matrix such that the (i,j)-th entry of χ_S is 1 if and only if $(i,j) \in S$.

We claim that for every $M \in G$, there exist $w \in \binom{[q]}{\leq \eta\ell}$, $\alpha \in \{0,1\}^a$, and $j_1, \ldots, j_\ell \in [n]$ such that $M = \mathsf{Dec}(w, \alpha, j_1, \ldots, j_\ell)$. Let $\alpha := \alpha(M) \in \{0,1\}^a$, and let $w \in \{0,1\}^q$ be the sequence of answers that $A^M(\mathbf{1}_n; r^*, \alpha)$ receives from the oracle M. Note that $w \in \binom{[q]}{\leq \eta\ell}$ since $A^M(\mathbf{1}_n; r^*, \alpha)$ receives at most $\eta\ell$ ones from M. Thus, the output y defined in the first step of Dec is equal to $A^M(\mathbf{1}_n; r^*, \alpha)$. Moreover, since $A^M(\mathbf{1}_n; r^*, \alpha) = M\mathbf{1}_n$, we obtain $y = M\mathbf{1}_n$. Since each row of M contains at most one nonzero entry and M contains exactly ℓ ones, the set $\{i_1, \ldots, i_\ell\}$ of indices are equal to the set of all the indices i such that the i-th row of M contains a nonzero entry. Therefore, for every $k \in [\ell]$, there exists j_k such that the (i_k, j_k)-th entry of M is equal to 1. Since M contains exactly ℓ ones, we obtain $M = \chi_S = \mathsf{Dec}(w, \alpha, j_1, \ldots, j_\ell)$. □

Now we are ready to prove Theorem 4.1.

Proof of Theorem 4.1. Define $\ell := \min\left\{\lfloor \log_p(1/\varepsilon) \rfloor, \sqrt{n/\log n}\right\}$. Towards a contradiction, assume that $a \leq \frac{\gamma}{5}\ell \log n$. By Lemma 4.2, every ℓ-sparse matrix M satisfies the assumption of Definition 1.3, that is,

$$\Pr_{x \sim \mathbb{F}^n}[Mx = \mathcal{O}(x)] = \Pr_{x \sim \mathbb{F}^n}[Mx = 0] \geq \frac{1}{p^\ell} \geq \varepsilon.$$

The property of a random self-reduction implies that for every ℓ-sparse matrix M and every $x \in \mathbb{F}^n$,

$$\Pr_A[A^M(x, \alpha(M)) = Mx] \geq \frac{2}{3},$$

where the probability is over the internal randomness of the randomized oracle algorithm A. Applying Lemma 4.5 to A, we obtain a set G of compressible matrices such that

$$\frac{1}{2}\binom{n^2}{\ell} \leq |G| \leq \binom{q}{\leq \ell/4} \cdot 2^a \cdot n^\ell,$$

where the last inequality follows by counting the size of the range of Dec. Since $q \leq n^{2-\gamma}$, we have

$$\frac{1}{2}\left(\frac{n^2}{\ell}\right)^\ell \leq \frac{1}{2}\binom{n^2}{\ell} \leq \binom{q}{\leq \ell/4} \cdot 2^a \cdot n^\ell \leq n^{(2-\gamma)\ell/4+\ell} \cdot 2^a.$$

Taking its logarithm, we obtain

$$2\ell \log n - \ell \log \ell - 1 \leq ((2-\gamma)/4 + 1)\ell \log n + a.$$

Since $\log \ell \leq \frac{1}{2}\log n$, it follows that

$$a \geq \frac{\gamma}{4}\ell \log n - 1 > \frac{\gamma}{5}\ell \log n,$$

where the last inequality holds for all large n. This is a contradiction.

We conclude that $a > \frac{\gamma}{5}\ell \log n = \min\left\{\frac{\gamma}{5} \cdot \lfloor \log_p(1/\varepsilon) \rfloor \cdot \log_2 n, \frac{\gamma}{5}\sqrt{n \log n}\right\}$. The claim follows by observing that $\frac{\gamma}{5}\sqrt{n \log n} \geq \sqrt{n}$ for all sufficiently large n. □

5 Uniform Reduction

In this section, we prove Theorems 1.5 and 1.6. In Section 5.1, we prove a uniform reduction that amplifies the success probability of a worst-case solver for a linear problem. This uniform reduction will be used to prove both Theorems 1.5 and 1.6.

5.1 Uniform Worst-Case-to-Worst-Case Reduction

The common ingredient of the proofs of Theorems 1.5 and 1.6 is the following uniform worst-case-to-worst-case reduction that transforms a weak worst-case solver that solves a linear problem with a tiny success probability into a strong worst-case solver that solves the problem with high success probability.

Lemma 5.1. *Let $\varepsilon > 0$ be an arbitrary parameter. There exists an $O((n+m)/\varepsilon)$-time randomized oracle algorithm $A^{\mathcal{O},M}$ that, for any randomized oracle $\mathcal{O}$ and any matrix $M \in \mathbb{F}^{m\times n}$ satisfying*

$$\forall x \in \mathbb{F}^n, \quad \Pr_{\mathcal{O}}[\mathcal{O}(x) = Mx] \geq \varepsilon,$$

it holds that

$$\forall x \in \mathbb{F}^n, \quad \Pr_{A^{\mathcal{O},M}}[A^{\mathcal{O},M}(x) = Mx] \geq \frac{2}{3},$$

where the probability is taken over the internal randomness of $A^{\mathcal{O},M}$. Moreover, $A^{\mathcal{O},M}$ makes at most $O(1/\varepsilon)$ queries to $\mathcal{O}$ and reads $O(1/\varepsilon)$ rows of M.

Proof. The algorithm $A^{\mathcal{O},M}(x)$ runs $\mathcal{O}(x)$ for $T = O(1/\varepsilon)$ times and collects the outputs $L = \{y^{(1)}, \dots, y^{(T)}\} \subseteq \mathbb{F}^m$. Then, it removes a wrong vector $y^{(i)} \neq Mx$ from L one by one. To this end, it chooses two distinct vectors $y, z \in L$ ($y \neq z$). For an index $i \in [m]$ such that $y_i \neq z_i$, it computes $(Mx)_i$ by querying the i-th row of M. Since $y_i \neq z_i$, at least one of them is different from $(Mx)_i$, and we remove it from L. By repeating this procedure, we can remove all the wrong vectors from L. We present the description of $A^{\mathcal{O},M}$ in Algorithm 2.

Algorithm 2 Algorithm $A^{\mathcal{O},M}$

Input: Input $x \in \mathbb{F}^n$

1: Run $\mathcal{O}(x)$ for $T = O(1/\varepsilon)$ times and collect the outputs $L = \{y^{(1)}, \dots, y^{(T)}\} \subseteq \mathbb{F}^m$.
2: **for** every pair $y, z \in L$ with $y \neq z$ **do**
3: Choose $i \in [m]$ such that $y_i \neq z_i$.
4: Compute the i-th entry of Mx by querying M.
5: **if** $(Mx)_i \neq y_i$ **then**
6: Remove y from L.
7: **end if**
8: **if** $(Mx)_i \neq z_i$ **then**
9: Remove z from L.
10: **end if**
11: **end for**
12: **if** L contains a vector **then**
13: Output it.
14: **else**
15: Output an arbitrary vector.
16: **end if**

We prove the correctness of $A^{\mathcal{O},M}(x)$. By assumption of $\mathcal{O}$, the list L computed in Line 1 contains Mx with probability $1-(1-\varepsilon)^T \geq 2/3$. Henceforth, we assume that L contains $y = Mx$. The **for** loop of Line 2 never removes $y = Mx$ from L. On the other hand, any vector $z \in L$ with

$z \neq y = Mx$, there is an index $i \in [n]$ with $z_i \neq y_i = (Mx)_i$ and thus z will be removed from the list. Therefore, after the loop of Line 2, the list L contains only Mx and thus $A^{\mathcal{O},M}(x)$ outputs Mx with probability 2/3.

We bound the number of queries to $\mathcal{O}$ and M. During Lines 3–10, the number of vectors in L decreases by at least one. Therefore, the number of iterations in the **for** loop of Line 2 is at most T and thus $A^{\mathcal{O},M}$ makes at most T queries to $\mathcal{O}$ and reads at most T rows of M. This completes the proof. □

5.2 Uniform Reduction over Large Fields

We prove Theorem 1.6. From Lemma 5.1, it suffices to construct a worst-case solver that solves a linear problem for all inputs with a tiny success probability. We demonstrate such a worst-case to average-case reduction based on polynomial interpolation. Although the technique based on polynomial interpolation is standard and is previously known in previous works ([AGGS22; AGGSS24]), we state it explicitly for completeness.

Lemma 5.2. *Let $\delta, \varepsilon > 0$ be parameters and $\mathbb{F}$ be a finite field of size $p \geq \frac{1}{\varepsilon} + 1 + \delta$. Let $M \in \mathbb{F}^{m \times n}$ be a matrix over $\mathbb{F}$. Then, there exists a two-query randomized oracle algorithm $A^{\mathcal{O}}$ that is given oracle access to an oracle $\mathcal{O}$ such that*

$$\Pr_{x \sim \mathbb{F}^n}[\mathcal{O}(x) = Mx] \geq \varepsilon,$$

satisfies

$$\Pr_{A^{\mathcal{O}}}[A^{\mathcal{O}}(x) = Mx] \geq \frac{\delta^2 \varepsilon^5}{27}$$

for every $x \in \mathbb{F}^n$.

Proof. Let $x \in \mathbb{F}^n$ be the input. Take a random vector $r \sim \mathbb{F}^n$ and consider a random line $(\ell(t))_{t \in \mathbb{F}}$, where $\ell(t) = x + tr$. The algorithm $A^{\mathcal{O}}$ chooses two random distinct points $t_1, t_2 \in \mathbb{F}$ uniformly at random and compute $Mx = M \cdot \ell(0)$ by interpolating $\mathcal{O}(\ell(t_1))$ and $\mathcal{O}(\ell(t_2))$. See Algorithm 3 for the description.

Algorithm 3 Worst-case to average-case reduction $A^{\mathcal{O}}$ on input $x \in \mathbb{F}^n$

1: Sample $r \in \mathbb{F}^n$ uniformly at random
2: Let $\ell(t) = x + tr$ for $t \in \mathbb{F}$
3: Choose two random points $t_1, t_2 \sim \mathbb{F}$ conditioned on $t_1 \neq t_2$
4: Let $y_1 \leftarrow \mathcal{O}(\ell(t_1))$ and $y_2 \leftarrow \mathcal{O}(\ell(t_2))$
5: **for** $i \in [m]$ **do**
6: Compute a degree-one polynomial $P_i(t)$ that interpolates $(t_1, (y_1)_i)$ and $(t_2, (y_2)_i)$
7: **end for**
8: Output $(P_i(0))_{i \in [m]}$

Note that each entry of $M \cdot \ell(t)$ is a degree-one polynomial in t. Moreover, for every fixed $t \neq 0$, the marginal distribution of $M \cdot \ell(t)$ is uniform over $\mathbb{F}^m$. Let X be the number of $t \in \mathbb{F} \setminus \{0\}$ such

that $\mathcal{O}(\ell(t)) = M \cdot \ell(t)$. By assumption of $\mathcal{O}$, we have $\mathbb{E}_r[X] \geq (p-1)\varepsilon$. By Markov's inequality, for any $0 < \gamma \leq 1$, we have

$$\begin{aligned}
\Pr[X \leq (p-1)(1-\gamma)\varepsilon] &= \Pr[(p-1) - X \geq (p-1) - (p-1)(1-\gamma)\varepsilon] \\
&\leq \frac{(p-1) - \mathbb{E}[X]}{p - 1 - (p-1)(1-\gamma)\varepsilon} \\
&\leq \frac{(p-1)(1-\varepsilon)}{(p-1)(1-(1-\gamma)\varepsilon)} \\
&\leq 1 - \frac{\gamma\varepsilon}{1-(1-\gamma)\varepsilon} \\
&\leq 1 - \gamma\varepsilon.
\end{aligned}$$

Therefore, $\Pr[X \geq (p-1)(1-\gamma)\varepsilon] \geq \gamma\varepsilon$.

Set $\gamma = \delta\varepsilon/3$. If $X \geq (p-1)(1-\gamma)\varepsilon$, then the algorithm $A^{\mathcal{O}}(x)$ picks up t_1, t_2 such that $\mathcal{O}(\ell(t_1)) = M \cdot \ell(t_1)$ and $\mathcal{O}(\ell(t_2)) = M \cdot \ell(t_2)$ at Step 3 (and then outputs Mx) with probability at least

$$\begin{aligned}
\frac{X(X-1)}{(p-1)(p-2)} &\geq \frac{(p-1)(1-\gamma)\varepsilon \cdot ((p-1)(1-\gamma)\varepsilon - 1)}{(p-1)^2} \\
&= (1-\gamma)^2\varepsilon^2 - \frac{(1-\gamma)\varepsilon}{p-1} \\
&\geq (1-\gamma)^2\varepsilon^2 - \frac{1-\gamma}{1+\delta\varepsilon} \cdot \varepsilon^2 && \because p - 1 \geq \frac{1+\delta\varepsilon}{\varepsilon} \\
&= (1-\delta\varepsilon/3)^2 \cdot \varepsilon^2 \left(1 - \frac{1}{(1+\delta\varepsilon)(1-\delta\varepsilon/3)}\right) && \because \gamma = \delta\varepsilon/3 \\
&\geq (1-\delta\varepsilon/3)^2 \cdot \varepsilon^2 \left(1 - \frac{1}{1+\delta\varepsilon/3}\right) \\
&= (1-\delta\varepsilon/3)^2 \cdot \varepsilon^2 \cdot \frac{\delta\varepsilon}{3+\delta\varepsilon} \\
&\geq \frac{\varepsilon^3\delta}{9}.
\end{aligned}$$

Here, we used $\delta, \varepsilon \leq 1$. Therefore, for any $x \in \mathbb{F}^n$, the algorithm $A^{\mathcal{O}}(x)$ outputs Mx with probability at least $\gamma\varepsilon \cdot \frac{\varepsilon^3\delta}{9} = \frac{\delta^2\varepsilon^5}{27}$. □

The proof of Theorem 1.6 is now straightforward from Lemmas 5.1 and 5.2.

Proof of Theorem 1.6. From Lemma 5.2, we can construct a randomized oracle $\mathcal{O}'$ such that

$$\forall x \in \mathbb{F}^n, \quad \Pr_{\mathcal{O}'}[\mathcal{O}'(x) = Mx] \geq \frac{\delta^2\varepsilon^5}{27}$$

for every $x \in \mathbb{F}^n$. Then, from Lemma 5.1 using $\mathcal{O}'$ as oracle, the algorithm $A^{\mathcal{O}',M}$ satisfies

$$\forall x \in \mathbb{F}^n, \quad \Pr_{A^{\mathcal{O}',M}}[A^{\mathcal{O}',M}(x) = Mx] \geq \frac{2}{3}$$

for every $x \in \mathbb{F}^n$. Note that $A^{\mathcal{O}',M}$ makes at most $O(1/(\varepsilon^5\delta^2))$ queries to $\mathcal{O}'$ and reads $O(1/(\varepsilon^5\delta^2))$ rows of M. Moreover, the randomized oracle $\mathcal{O}'$ makes two queries to $\mathcal{O}$. Therefore, $A^{\mathcal{O}',M}$ makes at most $O(1/(\varepsilon^5\delta^2))$ queries to $\mathcal{O}$ and reads $O(1/(\varepsilon^5\delta^2))$ rows of M. This completes the proof. □

5.3 Reduction with Short Advice

We prove Theorem 1.5. As in the proof of Theorem 1.6, we present a worst-case-to-average-case reduction based on the additive-combinatorics framework developed in [AGGS22; AGGSS24].

Lemma 5.3. *Let $\varepsilon > 0$ be any parameter and $\mathbb{F}$ be a finite field of size $p = |\mathbb{F}|$. Let $M \in \mathbb{F}^{m\times n}$ be a matrix and $\mathcal{O}$ be an oracle satisfying*

$$\Pr_{x\sim\mathbb{F}^n}[\mathcal{O}(x) = Mx] \geq \varepsilon,$$

Then, there exists an $(n+m)\cdot\log(1/\varepsilon)$-time randomized oracle algorithm $A^{\mathcal{O},M}$ such that, $A^{\mathcal{O},M}$ makes $O(\log(1/\varepsilon))$ queries to $\mathcal{O}$ and reads $O(\log(1/\varepsilon))$ columns of M and is given a nonuniform advice $\alpha \in \{0,1\}^a$ for $a = O(\log(1/\varepsilon)\cdot\log n)$, and it holds that

$$\exists \alpha = \alpha(M,\mathcal{O}) \in \{0,1\}^a, \quad \forall x \in \mathbb{F}^n, \quad \Pr_A\big[A^{\mathcal{O},M}(x;\alpha) = Mx\big] \geq (\varepsilon/p)^{-O(\log(1/\varepsilon))}.$$

First, we present a uniform oracle algorithm that computes Mx for all $x \in V$ for some subspace $V \subseteq \mathbb{F}^n$ of dimension $n - O(\log(1/\varepsilon))$.

Lemma 5.4. *Let $\varepsilon > 0$ be any parameter and $\mathbb{F}$ be a finite field of size $p = |\mathbb{F}|$. There exists an $O(n+m)$-time randomized oracle algorithm $A^{\mathcal{O}}$ that satisfies the following: For any matrix $M \in \mathbb{F}^{m\times n}$ and any oracle $\mathcal{O}$ satisfying*

$$\Pr_{x\sim\mathbb{F}^n}[\mathcal{O}(x) = Mx] \geq \varepsilon,$$

there exists a subspace $V \subseteq \mathbb{F}^n$ of dimension $\dim V \geq n - O(\log(1/\varepsilon))$ such that

$$\forall x \in V, \quad \Pr_{A^{\mathcal{O}}}[A^{\mathcal{O}}(x) = Mx] \geq \varepsilon^{O(\log(1/\varepsilon))}.$$

Moreover, $A^{\mathcal{O}}$ makes $O(\log(1/\varepsilon))$ oracle queries to $\mathcal{O}$.

Note that $A^{\mathcal{O}}$ above does not read any entries of M. To prove Lemma 5.4, we invoke the following variant of the Bogolyubov–Ruzsa lemma [GSS24, Lemma 8]:

Lemma 5.5. *Let $\varepsilon > 0$ be any parameter and $\mathbb{F}$ be a finite field of size $p = |\mathbb{F}|$. Let $S \subseteq \mathbb{F}^n$ be a set of size $|S| = \varepsilon|\mathbb{F}|^n$. Let $t \geq \frac{\log(1/\varepsilon)}{2} + 1$ be a parameter. Then, there exists a subspace $V \subseteq \mathbb{F}^n$ of dimension $\dim V \geq n - O(\log(1/\varepsilon))$ such that*

$$\forall x \in V, \quad \Pr_{x_1,\dots,x_t,y_1,\dots,y_t\sim\mathbb{F}^n}[x_1,\dots,x_t,y_1,\dots,y_t \in S \,|\, v = x_1+\cdots+x_t-y_1-\cdots-y_t] \geq \varepsilon^{2t+1}.$$

The original lemma of [GSS24, Lemma 8] is stated only for the case where $p = 2$. A subsequent paper [SS25] claims that this lemma holds for general prime power p without a proof. For completeness, we provide a proof of Lemma 5.5 for general prime power p in Appendix B.

Proof of Lemma 5.4. We apply Lemma 5.5 to the set $S := \{x \in \mathbb{F}^n : \mathcal{O}(x) = Mx\}$ (if $|S| > \varepsilon|\mathbb{F}|^n$, take a subset of S of size $\varepsilon|\mathbb{F}|^n$). Let V be the subspace given by Lemma 5.5.

Let $t = \left\lceil \frac{\log(1/\varepsilon)}{2} \right\rceil + 1$. The algorithm $A^{\mathcal{O}}(x)$ on input $x \in \mathbb{F}^n$ chooses $a_1,\dots,a_t,b_1,\dots,b_t \sim \mathbb{F}^n$ uniformly at random conditioned on $a_1+\cdots+a_t-b_1-\cdots-b_t = x$. Then, outputs $\mathcal{O}(a_1)+\cdots+\mathcal{O}(a_t)-\mathcal{O}(b_1)-\cdots-\mathcal{O}(b_t)$.

By Lemma 5.5, with probability at least ε^{2t+1}, we have $a_1,\dots,a_t,b_1,\dots,b_t \in S$. If this occurs, we have $\mathcal{O}(a_i) = Ma_i$ and $\mathcal{O}(b_i) = Mb_i$ for all $i \in [t]$. Since $x = a_1+\cdots+a_t-b_1-\cdots-b_t$, the output satisfies $\mathcal{O}(a_1)+\cdots+\mathcal{O}(a_t)-\mathcal{O}(b_1)-\cdots-\mathcal{O}(b_t) = Mx$. □

Next, we present a non-uniform algorithm that computes Mx for all $x \in \mathbb{F}^n$ using the algorithm $A^{\mathcal{O}}$ of Lemma 5.4.

Lemma 5.6. *Let $\delta > 0$ by any parameter and $\mathbb{F}$ be a finite field. There exists a non-uniform randomized oracle algorithm $B^{\mathcal{O}',M}$ that satisfies the following: For any matrix $M \in \mathbb{F}^{m \times n}$, a randomized oracle $\mathcal{O}'$ and a subspace $V \subseteq \mathbb{F}^n$ of dimension $\dim V \geq n - c$ for some parameter $c > 0$,*

$$\forall x \in V, \quad \Pr_{\mathcal{O}'}[\mathcal{O}'(x) = Mx] \geq \delta,$$

there exists a nonuniform advice $\alpha = \alpha(V) \in \{0,1\}^{O(c \log n)}$ such that

$$\forall x \in \mathbb{F}^n, \quad \Pr_{B^{\mathcal{O}',M}}[B^{\mathcal{O}',M}(x;\alpha) = Mx] \geq \delta \cdot |\mathbb{F}|^{-c}.$$

Moreover, $B^{\mathcal{O}',M}$ makes one query to $\mathcal{O}'$ and reads at most c columns of M.

Proof. Let $V \subseteq \mathbb{F}^n$ be a subspace of $\dim V = n - c$. Since the codimension of V is c, there exists a set of c vectors whose addition to V spans $\mathbb{F}^n$. In particular, we can choose these c vectors from the set of standard basis vectors $\{e_1, \ldots, e_n\}$[3] (recall that a standard basis vector e_i is the vector whose i-th entry is 1 and all other entries are zero). Let $e_{i_1}, \ldots, e_{i_c}$ be such vectors.

The nonuniform advice α is the tuple of the indices $(i_1, \ldots, i_c)$, which can be represented as a string of $c\lceil \log_2 n \rceil$ bits. The algorithm $B^{\mathcal{O}',M}(x;\alpha)$ is described in Algorithm 4.

Algorithm 4 Algorithm $B^{\mathcal{O}',M}(x;\alpha)$

Input: $x \in \mathbb{F}^n$ and $\alpha = (i_1, \ldots, i_c)$

1: Sample $a_1, \ldots, a_c \sim \mathbb{F}$ independently and uniformly at random
2: Set $v \leftarrow x - a_1 e_{i_1} - \cdots - a_c e_{i_c}$
3: **for** $j = 1$ to c **do**
4: Compute Me_{i_j} by querying the i_j-th column of M
5: **end for**
6: **Output** $\mathcal{O}'(v) + a_1 Me_{i_1} + \cdots + a_c Me_{i_c}$

Clearly, $B^{\mathcal{O}',M}(x;\alpha)$ makes one query to $\mathcal{O}'$ and reads at most c columns of M. Moreover, $B^{\mathcal{O}',M}(x;\alpha)$ runs in time $O(c(n+m))$.

We bound the success probability of $B^{\mathcal{O}',M}(x;\alpha)$. Note that any $x \in \mathbb{F}^n$ can be represented as

$$x = v + b_1 e_{i_1} + \cdots + b_c e_{i_c}$$

for some $v \in V$ and $b = (b_1, \ldots, b_c) \in \mathbb{F}^c$. Therefore, with probability $|\mathbb{F}|^{-c}$ over the choice of $a = (a_1, \ldots, a_c) \sim \mathbb{F}^c$, we have $a = b$. Thus, at Line 2, it holds that

$$\Pr_{a_1,\ldots,a_c \sim \mathbb{F}}[v \in V] \geq |\mathbb{F}|^{-c}.$$

Suppose $v \in V$. Then, by assumption of $\mathcal{O}'$, we have

$$\Pr_{\mathcal{O}'}[\mathcal{O}'(v) = Mv] \geq \delta.$$

[3]To see this, note that $e_i \notin V$ for some i (otherwise, V spans the entire space $\mathbb{F}^n$). Add e_i to V and repeat this argument for c times, which yields c standard basis vectors that are not belong to V.

Since $x = v + a_1 e_{i_1} + \cdots + a_c e_{i_c}$, we have

$$\begin{aligned}\Pr_{B^{\mathcal{O}',M}}[B^{\mathcal{O}',M}(x;\alpha) = Mx] &\geq \Pr[\mathcal{O}'(v) = Mv] \\ &\geq \Pr[\mathcal{O}'(x) = Mv | v \in V] \cdot \Pr[v \in V] \\ &\geq \delta \cdot |\mathbb{F}|^{-c}.\end{aligned}$$

This completes the proof. □

Now, we prove Lemma 5.3.

Proof of Lemma 5.3. We combine Lemmas 5.4 and 5.6. Given oracle access to $\mathcal{O}$, from Lemma 5.4, there exists an oracle algorithm $A_0^{\mathcal{O}}$ and a subspace $V \subseteq \mathbb{F}^n$ of dimension $\dim V \geq n - O(\log(1/\varepsilon))$ such that $A_0^{\mathcal{O}}(x) = Mx$ with probability $\varepsilon^{O(\log(1/\varepsilon))}$ for any $x \in V$. Using this algorithm $A_0^{\mathcal{O}}$ as oracle $\mathcal{O}'$, run the oracle algorithm $B^{\mathcal{O}',M}$ of Lemma 5.6. Note that the nonuniform advice α relies on V, which depends only on the original oracle $\mathcal{O}$.

For any $x \in \mathbb{F}^n$, it holds from Lemma 5.6 with $\delta = \varepsilon^{O(\log(1/\varepsilon))}$ and $c = O(\log(1/\varepsilon))$ that $\Pr[B^{\mathcal{O}',M}(x) = Mx] \geq (\varepsilon/|\mathbb{F}|)^{O(\log(1/\varepsilon))}$ for every $x \in \mathbb{F}^n$.

Since $A_0^{\mathcal{O}}$ makes $O(\log(1/\varepsilon))$ oracle queries to $\mathcal{O}$ and $B^{\mathcal{O}',M}$ makes one oracle queries to $\mathcal{O}'$, the total number of oracle queries to $\mathcal{O}$ is $O(\log(1/\varepsilon))$. Moreover, $B^{\mathcal{O}',M}$ reads at most $c = O(\log(1/\varepsilon))$ columns of M. □

Combining Lemmas 5.1 and 5.3, we obtain Theorem 1.5.

Proof of Theorem 1.5. From Lemma 5.3, we can construct a randomized nonuniform oracle $\mathcal{O}'$ such that

$$\exists \alpha = \alpha(M, \mathcal{O}) \in \{0,1\}^a, \quad \forall x \in \mathbb{F}^n, \quad \Pr_{\mathcal{O}'}\big[\mathcal{O}'(x;\alpha) = Mx\big] \geq \varepsilon' := (\varepsilon/p)^{O(\log(1/\varepsilon))}.$$

Moreover, $\mathcal{O}'$ makes at $O(\log(1/\varepsilon))$ queries to $\mathcal{O}$ and reads at most $O(\log(1/\varepsilon))$ columns of M.

Then, from Lemma 5.1 using $\mathcal{O}'$ as oracle, the algorithm $A^{\mathcal{O}',M}$ satisfies

$$\forall x \in \mathbb{F}^n, \quad \Pr_{A^{\mathcal{O}',M}}[A^{\mathcal{O}',M}(x) = Mx] \geq \frac{2}{3}$$

for every $x \in \mathbb{F}^n$. Moreover, $A^{\mathcal{O}',M}$ runs in time $O(n/\varepsilon') = n \cdot (p/\varepsilon)^{O(\log(1/\varepsilon))}$, makes at most $O(1/\varepsilon') = (p/\varepsilon)^{O(\log(1/\varepsilon))}$ queries to $\mathcal{O}'$, and reads $O(1/\varepsilon') = (p/\varepsilon)^{O(\log(1/\varepsilon))}$ rows of M.

The final algorithm runs $A^{\mathcal{O}',M}(x;\alpha)$ while simulating $\mathcal{O}'(\text{-};\alpha)$ using $\mathcal{O}$ and M whenever $\mathcal{O}'(\text{-};\alpha)$ is queried. Since the number of queries to $\mathcal{O}'$ is at most $O(1/\varepsilon')$, this algorithm makes at most $O(\log(1/\varepsilon)/\varepsilon')$ queries to $\mathcal{O}$ and reads at most $O(\log(1/\varepsilon)/\varepsilon')$ columns of M in total. Additionally, the final algorithm reads at most $O(1/\varepsilon')$ rows of M. □

Suppose that the oracle $\mathcal{O}$ solves the linear problem whenever the input x belongs to some unknown large subspace $V \subseteq \mathbb{F}^n$ (i.e., the set $\{x \in \mathbb{F}^n \colon \mathcal{O}(x) = Mx\}$ contains a subspace $V \subseteq \mathbb{F}^n$). In this case, using Lemmas 5.1 and 5.6, we can construct a non-uniform worst-case-to-worst-case reduction with an advice length that achieves the lower bound of Theorem 4.1 up to a constant factor.

Proposition 5.7. *Let $\mathbb{F}$ be a finite field of size $p = |\mathbb{F}|$. Let $\varepsilon > 0$ be any parameter such that $1/\varepsilon$ is a power of p. Let $\mathcal{O}$ be an oracle such that, for some subspace $V \subseteq \mathbb{F}^n$ of dimension $\dim V = n - \log_p(1/\varepsilon)$, it holds that*

$$\forall x \in V, \quad \mathcal{O}(x) = Mx.$$

(In particular, $\Pr_{x\sim\mathbb{F}^n}[\mathcal{O}(x) = Mx] \geq \varepsilon$.) Then, there exists a randomized $O(n/\varepsilon)$-time oracle algorithm $A^{\mathcal{O},M}$ such that

$$\exists \alpha = \alpha(M, \mathcal{O}) \in \{0,1\}^a, \quad \forall x \in \mathbb{F}^n, \quad \Pr_{A^{\mathcal{O},M}}\left[A^{\mathcal{O},M}(x;\alpha) = Mx\right] \geq \frac{2}{3},$$

where $a = O(\log_p(1/\varepsilon) \cdot \log n)$. Moreover, $A^{\mathcal{O},M}$ makes at most $O(1/\varepsilon)$ queries to $\mathcal{O}$ and reads at most $O(1/\varepsilon)$ columns of M.

Proof. From Lemma 5.6 for $\delta = 1$ and $c = \log_p(1/\varepsilon)$, we can construct a non-uniform randomized oracle algorithm $B^{\mathcal{O},M}$ such that

$$\exists \alpha = \alpha(V) \in \{0,1\}^{O(\log_p(1/\varepsilon)\cdot\log n)}, \quad \forall x \in V, \quad \Pr_{B^{\mathcal{O},M}}[B^{\mathcal{O},M}(x;\alpha) = Mx] \geq |\mathbb{F}|^{-\log_p(1/\varepsilon)} = \varepsilon.$$

Moreover, $B^{\mathcal{O},M}$ makes one query to $\mathcal{O}$ and reads at most $c = \log_p(1/\varepsilon)$ columns of M.

Then, the reduction from Lemma 5.1 using $B^{\mathcal{O},M}$ as oracle (we use the common advice string α for each execution of $B^{\mathcal{O},M}$) yields the algorithm $A^{\mathcal{O},M}$ that satisfies the claimed properties. Since $A^{\mathcal{O},M}$ executes $B^{\mathcal{O},M}$ for $O(1/\varepsilon)$ times, it makes at most $O(1/\varepsilon)$ queries to $\mathcal{O}$ and reads at most $O(1/\varepsilon)$ rows and at most $O(c/\varepsilon) = O(\log_p(1/\varepsilon)/\varepsilon)$ columns of M in total. □

Acknowledgements

We are grateful to Alexander Golovnev for bringing the open question of [AGGSS24] to our attention during the Dagstuhl seminar "Computational Complexity of Discrete Problems." We also thank the anonymous reviewers for their helpful comments. This work was supported by JSPS KAKENHI Grant Numbers 24K21317 (Shuichi Hirahara) and 23K16837, 23K18460 and 24K21317 (Nobutaka Shimizu), Japan.

References

[AGGS22] Vahid R Asadi, Alexander Golovnev, Tom Gur, and Igor Shinkar. "Worst-case to average-case reductions via additive combinatorics". In: *Proceedings of the Symposium on Theory of Computing (STOC)*. 2022, pp. 1566–1574. DOI: 10.1145/3519935.3520041. (Visited on 03/26/2023) (cit. on pp. 3, 5, 7–9, 19, 21).

[AGGSS24] Vahid R Asadi, Alexander Golovnev, Tom Gur, Igor Shinkar, and Sathyawageeswar Subramanian. "Quantum worst-case to average-case reductions for all linear problems". en. In: *Proceedings of the Symposium on Discrete Algorithms (SODA)*. Society for Industrial and Applied Mathematics, 2024, pp. 2535–2567. DOI: 10.1137/1.9781611977912.90 (cit. on pp. 2–5, 7, 9, 19, 21, 24).

[AW17] Josh Alman and R. Ryan Williams. "Probabilistic rank and matrix rigidity". In: *Proceedings of the Symposium on Theory of Computing (STOC)*. 2017, pp. 641–652. DOI: 10.1145/3055399.3055484 (cit. on p. 3).

[BBB19] Enric Boix-Adserà, Matthew S. Brennan, and Guy Bresler. "The Average-Case Complexity of Counting Cliques in Erdős-Rényi Hypergraphs". In: *Proceedings of the Symposium on Foundations of Computer Science (FOCS)*. 2019, pp. 1256–1280. DOI: 10.1109/FOCS.2019.00078 (cit. on p. 3).

[BLR93] Manuel Blum, Michael Luby, and Ronitt Rubinfeld. "Self-Testing/Correcting with Applications to Numerical Problems". In: *Journal of Computer and System Sciences* 47.3 (1993), pp. 549–595. DOI: 10.1016/0022-0000(93)90044-W (cit. on pp. 10, 12).

[BM74] Allan Borodin and R. Moenck. "Fast Modular Transforms". In: *Journal of Computer and System Sciences* 8.3 (1974), pp. 366–386. DOI: 10.1016/S0022-0000(74)80029-2 (cit. on p. 2).

[BRSV17] Marshall Ball, Alon Rosen, Manuel Sabin, and Prashant Nalini Vasudevan. "Average-case fine-grained hardness". In: *Proceedings of the Symposium on Theory of Computing (STOC)*. 2017, pp. 483–496. DOI: 10.1145/3055399.3055466 (cit. on p. 3).

[Cha02] Mei-Chu Chang. "A polynomial bound in Freiman's theorem". In: *Duke Mathematical Journal* 113 (3 2002). DOI: 10.1215/S0012-7094-02-11331-3 (cit. on p. 30).

[CT06] Thomas M Cover and Joy A Thomas. *Elements of Information Theory*. Wiley & Sons, Incorporated, John, 2006 (cit. on p. 31).

[CT65] James Cooley and John Tukey. "An algorithm for the machine calculation of complex Fourier series". In: *Mathematics of computation* 19.90 (1965), pp. 297–301 (cit. on p. 2).

[DLW20] Mina Dalirrooyfard, Andrea Lincoln, and Virginia Vassilevska Williams. "New Techniques for Proving Fine-Grained Average-Case Hardness". In: *Proceedings of the Symposium on Foundations of Computer Science (FOCS)*. 2020, pp. 774–785. DOI: 10.1109/FOCS46700.2020.00077 (cit. on p. 3).

[DTT10] Anindya De, Luca Trevisan, and Madhur Tulsiani. "Time Space Tradeoffs for Attacks against One-Way Functions and PRGs". In: *Advances in Cryptology - CRYPTO 2010, 30th Annual Cryptology Conference, Santa Barbara, CA, USA, August 15-19, 2010. Proceedings*. 2010, pp. 649–665. DOI: 10.1007/978-3-642-14623-7_35 (cit. on p. 6).

[Fre79] Rūsiņš Freivalds. "Fast probabilistic algorithms". In: *Proceedings of Mathematical Foundations of Computer Science (MFCS)*. 1979, pp. 57–69. DOI: 10.1007/3-540-09526-8_5 (cit. on pp. 6, 10).

[GG13] Joachim von zur Gathen and Jürgen Gerhard. *Modern Computer Algebra*. Cambridge University Press, 2013. DOI: 10.1017/cbo9781139856065 (cit. on p. 3).

[GL89] Oded Goldreich and Leonid A. Levin. "A Hard-Core Predicate for all One-Way Functions". In: *Proceedings of the Symposium on Theory of Computing (STOC)*. 1989, pp. 25–32. DOI: 10.1145/73007.73010 (cit. on p. 9).

[GR18] Oded Goldreich and Guy N. Rothblum. "Counting t-Cliques: Worst-Case to Average-Case Reductions and Direct Interactive Proof Systems". In: *Proceedings of the Symposium on Foundations of Computer Science (FOCS)*. 2018, pp. 77–88. DOI: 10.1109/FOCS.2018.00017 (cit. on p. 3).

[GSS24] Ashish Gola, Igor Shinkar, and Harsimran Singh. "Matrix Multiplication Reductions". In: *Proceedings of the Approximation, Randomization, and Combinatorial Optimization (APPROX/RANDOM)*. 2024, 34:1–34:15. DOI: 10.4230/LIPICS.APPROX/RANDOM.2024.34 (cit. on pp. 3, 8, 21, 29).

[HL20] Lianna Hambardzumyan and Yaqiao Li. "Chang's lemma via Pinsker's inequality". en. In: *Discrete mathematics* 343 (1 2020), p. 111496. DOI: 10.1016/j.disc.2019.04.015 (cit. on p. 30).

[HLS22] Monika Henzinger, Andrea Lincoln, and Barna Saha. "The Complexity of Average-Case Dynamic Subgraph Counting". In: *Proceedings of the Symposium on Discrete Algorithms (SODA)*. 2022, pp. 459–498. DOI: 10.1137/1.9781611977073.23 (cit. on p. 3).

[HS23] Shuichi Hirahara and Nobutaka Shimizu. "Hardness Self-Amplification: Simplified, Optimized, and Unified". In: *Proceedings of the Symposium on Theory of Computing (STOC)*. STOC 2023. 2023, pp. 70–83. DOI: 10.1145/3564246.3585189. (Visited on 05/26/2023) (cit. on pp. 3, 5, 27).

[HS24] Shuichi Hirahara and Nobutaka Shimizu. "Planted Clique Conjectures Are Equivalent". In: *Proceedings of the Symposium on Theory of Computing (STOC)*. 2024, pp. 358–366. DOI: 10.1145/3618260.3649751 (cit. on pp. 3, 27).

[HS25] Shuichi Hirahara and Nobutaka Shimizu. "Error-correction of matrix multiplication algorithms". en. In: *Proceedings of the Symposium on Theory of Computing (STOC)*. STOC '25: 57th Annual ACM Symposium on Theory of Computing. 2025, pp. 785–794. DOI: 10.1145/3717823.3718244. (Visited on 10/31/2025) (cit. on pp. 3, 6).

[IJK09] Russell Impagliazzo, Ragesh Jaiswal, and Valentine Kabanets. "Chernoff-Type Direct Product Theorems". In: *Journal of Cryptology* 22.1 (2009), pp. 75–92. DOI: 10.1007/s00145-008-9029-7 (cit. on p. 6).

[IMR14] Russell Impagliazzo, Cristopher Moore, and Alexander Russell. "An entropic proof of Chang's inequality". In: *SIAM Journal on Discrete Mathematics* 28 (1 2014), pp. 173–176. DOI: 10.1137/120877982 (cit. on p. 30).

[LV19] Ming Li and Paul M. B. Vitányi. *An Introduction to Kolmogorov Complexity and Its Applications, 4th Edition*. Texts in Computer Science. Springer, 2019. DOI: 10.1007/978-3-030-11298-1 (cit. on p. 6).

[NRW23] Parker Newton, Silas Richelson, and Chase Wilson. "A High Dimensional Goldreich-Levin Theorem". In: *Proceedings of the Symposium on Theory of Computing (STOC)*. 2023, pp. 1463–1474. DOI: 10.1145/3564246.3585224 (cit. on pp. 3, 9).

[Ram20] C. Ramya. "Recent Progress on Matrix Rigidity - A Survey". In: *CoRR* abs/2009.09460 (2020). arXiv: 2009.09460 (cit. on p. 2).

[Spi96] Daniel A. Spielman. "Linear-time encodable and decodable error-correcting codes". In: *IEEE Transactions on Information Theory* 42.6 (1996), pp. 1723–1731. DOI: 10.1109/18.556668 (cit. on p. 2).

[SS25] Igor Shinkar and Harsimran Singh. "A simplified reduction for error correcting matrix multiplication algorithms". In: *Proceedings of the Approximation, Randomization, and Combinatorial Optimization (APPROX/RANDOM)*. 2025, 29:1–29:15. DOI: 10.4230/LIPICS.APPROX/RANDOM.2025.29 (cit. on pp. 21, 30).

[TV07] Luca Trevisan and Salil Vadhan. "Pseudorandomness and Average-Case Complexity Via Uniform Reductions". In: *Computational Complexity* 16 (4 2007), pp. 331–364. DOI: 10.1007/s00037-007-0233-x (cit. on pp. 6, 10).

[Val77] Leslie G. Valiant. "Graph-Theoretic Arguments in Low-Level Complexity". In: *Mathematical Foundations of Computer Science 1977, 6th Symposium, Tatranska Lomnica, Czechoslovakia, September 5-9, 1977, Proceedings*. 1977, pp. 162–176. DOI: 10.1007/3-540-08353-7_135 (cit. on p. 2).

A Nonuniform Reduction based on XOR Lemma

We present another proof of Theorem 1.1 based on the XOR lemma. To prove Theorem 1.1, we need an auxiliary result.

Lemma A.1 (Direct Product Lemma; [HS23], [HS24])**.** *Let D be a set. For all sufficiently small $\varepsilon > 0$, for every $K \geq O(\log(1/\varepsilon)/(\delta\varepsilon)^2)$), for every function $S\colon D^K \to [0,1]$, it holds that*

$$\Pr_{x\sim D}\left[\left|\mathop{\mathbb{E}}_{y\sim\Gamma(x)}[S(y)] - \mathop{\mathbb{E}}_{y\sim D^K}[S(y)]\right| \leq \varepsilon\right] \geq 1-\delta.$$

Here, $\Gamma(x)$ is the distribution of $y' \in D^K$ defined by the following sampling procedure: Sample $y \sim D^K, k \sim [K]$, replace the k-th element of y with x to obtain y', and output y'.

We begin with showing how to transform the weak average-case solver C into a strong average-case solver C_1 of size $(s+n)\mathsf{poly}(1/\varepsilon)$ that solves 90% of instances.

Lemma A.2. *Let $\delta, \varepsilon > 0$ be parameters and C be a circuit of size s that satisfies*

$$\Pr_{x\sim\mathbb{F}^n}[C(x) = Mx] \geq \varepsilon.$$

Then, there exists a circuit C' of size $(s+n)\mathsf{poly}(1/\delta, 1/\varepsilon)$ such that

$$\Pr_{x\sim\mathbb{F}^n}\left[C'(x) = Mx\right] \geq 1-\delta.$$

Proof. We first present a nonuniform algorithm A' that is given Trevisan-Vadhan advice, runs in time $(n+s)\mathsf{poly}(1/\delta, 1/\varepsilon)$, and computes Mx with probability at least $1-\delta$. Then, we transform A' into the claimed circuit C' of size $(s+n)\mathsf{poly}(1/\delta, 1/\varepsilon)$.

Let V be the verification algorithm given in Lemma 3.4 for M and $\gamma = \frac{\delta}{2T}$ for some $T = O(\log(1/\delta)/\varepsilon)$. The description of the nonuniform algorithm A' is as follows:

Algorithm 5 Nonuniform algorithm A'

Input: $x \in \mathbb{F}^n$

1: **for** $T = O(\log(1/\delta)/\varepsilon)$ times **do**
2: Sample $x_1, \ldots, x_K \sim \mathbb{F}^n$ and $i \sim [K]$, where $K = O(\log(1/\varepsilon)/(\delta^2\varepsilon^2))$.
3: Set $\overline{x} = x_1 + \cdots + x_{i-1} + x + x_{i+1} + \cdots + x_K$
4: Compute $z := C(\overline{x}) - \sum_{j \neq i} Mx_j$, where Mx_j for $j \neq i$ are given as Trevisan-Vadhan advice.
5: **if** The verification algorithm V accepts (x, z) **then**
6: Output z
7: **end if**
8: **end for**

Clearly, A' runs in time $(n+s)\mathsf{poly}(1/\delta, 1/\varepsilon)$. We claim that A' given in Algorithm 5 satisfies

$$\Pr_{\substack{x \sim \mathbb{F}^n \\ A'}}\left[A'(x) = Mx\right] \geq 1 - \frac{\delta}{3},$$

where the internal randomness of A' accounts for the choice of $i \sim [K]$, x_j $(j \neq i)$, and the internal randomness of V. We apply Lemma A.1 for $D = \mathbb{F}^n$ and

$$S(x_1, \ldots, x_K) = \begin{cases} 1 & \text{if } C(x_1 + \cdots + x_K) = M(x_1 + \cdots + x_K) \\ 0 & \text{otherwise.} \end{cases}$$

Observe that, for any $x \in \mathbb{F}^n$, the tuple $(x_1, \ldots, x_{i-1}, x, x_{i+1}, \ldots, x_K)$ is the output of the sampling procedure $\Gamma(x)$ of Lemma A.1. From Lemma A.1, we have

$$\Pr_{x \sim \mathbb{F}^n}\left[\left|\mathop{\mathbb{E}}_{y \sim \Gamma(x)}[S(y)] - \mathop{\mathbb{E}}_{y \sim D^K}[S(y)]\right| \leq \frac{\varepsilon}{2}\right] \geq 1 - \frac{\delta}{3}.$$

Note that the marginal distribution of $x_1 + \cdots + x_K$ is uniform over $\mathbb{F}^n$ for $y = (x_1, \ldots, x_K) \sim D^K$, we have $\mathbb{E}_{y \sim D^K}[S(y)] \geq \varepsilon$. Therefore, for a $\left(1 - \frac{\delta}{3}\right)$-fraction of $x \in \mathbb{F}^n$, the circuit C outputs $M\overline{x}$ on input

$$\overline{x} = x_1 + \cdots + x_{i-1} + x + x_{i+1} + \cdots + x_K$$

with probability at least $\varepsilon/2$ over the choice of x_j $(j \neq i)$. Call such $x \in \mathbb{F}^n$ *good* and fix any good x. By our choice of T, with probability at least $1 - \frac{\delta}{3}$ (over the choice of x_j), there exists a trial such that C correctly computes $M\overline{x}$ on input $\overline{x}$. If C correctly computes $M\overline{x}$ on input $\overline{x}$, then V accepts with probability 1 at Line 5 and thus $A'(x)$ outputs Mx. Otherwise (i.e., $C(\overline{x}) \neq M\overline{x}$), the verification algorithm V rejects with probability at least $1 - 2^{-\ell} \geq 1 - \frac{\delta}{3T}$. By the union bound over T trials, with probability at least $1 - \frac{\delta}{3}$, the verification algorithm V rejects for any of the T trials with $C(\overline{x}) \neq M\overline{x}$. Since the random instance x, x_j, and the internal randomness of V are independent, we have

$$\begin{aligned}\Pr_{\substack{x \sim \mathbb{F}^n \\ A'}}\left[A'(x) = Mx\right] \geq\ & \underbrace{\left(1 - \frac{\delta}{3}\right)}_{\text{fraction of good } x} \cdot \underbrace{\left(1 - \frac{\delta}{3}\right)}_{\Pr_{x_j}[C(\overline{x}) = M\overline{x} \text{ at some trial}]} \cdot \underbrace{\left(1 - \frac{\delta}{3}\right)}_{\Pr_V[V \text{ correctly decides at all trials}]} \\ & \geq 1 - \delta.\end{aligned}$$

By derandomizing A' using Lemma 3.3, we obtain a deterministic circuit C' of size $(s+n)\mathsf{poly}(1/\delta, 1/\varepsilon)$ that satisfies

$$\Pr_{x\sim\mathbb{F}^n}\left[C'(x) = Mx\right] \geq \Pr_{\substack{x\sim\mathbb{F}^n \\ A'}}\left[A'(x) = Mx\right] \geq 1-\delta.$$

This completes the proof. □

We are now ready to prove Theorem 1.1.

Proof of Theorem 1.1. By Lemma A.2 for $\delta = 0.1$, there exists a circuit C_1 of size $(s+n)\mathsf{poly}(1/\varepsilon)$ that satisfies

$$\Pr_{x\sim\mathbb{F}^n}[C_1(x) = Mx] \geq 0.9.$$

By Lemma 3.6, there exists a circuit C' of size $(s+n)\mathsf{poly}(1/\varepsilon)$ that satisfies

$$^{\forall} x \in \mathbb{F}^n, \Pr_{C'}[C'(x) = Mx] \geq \frac{2}{3},$$

where the probability is taken over internal randomness of C'. This completes the proof. □

B Proof of a Variant of Bogolyubov–Ruzsa Lemma

In this section, we prove Lemma 5.5. The proof is obtained by modifying the original proof [GSS24, Lemma 8] so that it also holds over an arbitrary finite field $\mathbb{F}_p$ where p is a prime power.

Fourier Analysis. We use the Fourier analysis to analyze the functions on the vector space $\mathbb{F}_p^n$. For the set of all functions $\{f\colon \mathbb{F}_p^n \to \mathbb{C}\}$, we associate it with the inner product defined by $\langle f, g\rangle = \mathbb{E}_{x\sim\mathbb{F}_p^n}\left[f(x)\cdot\overline{g(x)}\right]$, which induces the ℓ^2 norm $\|f\|_2 = \sqrt{\langle f, f\rangle}$.

Write $p = b^m$ for some prime b and $m \in \mathbb{N}$. Let $\omega = \exp(2\pi i/b)$ be a primitive b-th root of unity. Define a trace map $\mathrm{tr}\colon \mathbb{F}_p \to \mathbb{F}_b$ as $\mathrm{tr}(x) = x + x^b + \cdots + x^{b^{m-1}}$. It is known that tr is $\mathbb{F}_b$-linear, i.e., $\mathrm{tr}(x+y) = \mathrm{tr}(x) + \mathrm{tr}(y)$ and $\mathrm{tr}(cx) = c\mathrm{tr}(x)$ for all $x, y \in \mathbb{F}_p$ and $c \in \mathbb{F}_b$. For $r \in \mathbb{F}_p^n$, we define the character $\chi_r : \mathbb{F}_p^n \to \mathbb{C}$ as $\chi_r(x) = \omega^{\mathrm{tr}(r^\top x)}$. The Fourier transform of a function $f : \mathbb{F}_p^n \to \mathbb{C}$ at $r \in \mathbb{F}_p^n$ is defined as $\widehat{f}(r) = \langle f, \chi_r\rangle = \mathbb{E}_{x\sim\mathbb{F}_p^n}\left[f(x)\cdot\overline{\chi_r(x)}\right]$, where $\mathbb{E}_{x\sim\mathbb{F}_p^n}$ denotes the expectation over x chosen uniformly at random from $\mathbb{F}_p^n$. We can recover $f(x)$ from $\widehat{f}(r)$ as

$$f(x) = \sum_{r\in\mathbb{F}_p^n} \widehat{f}(r)\cdot\chi_r(x).$$

The Parseval's identity states that the linear map $f \mapsto \widehat{f}$ preserves the ℓ^2 norm: $\|f\|_2 = \left\|\widehat{f}\right\|_2$.

The convolution of two functions $f, g : \mathbb{F}_p^n \to \mathbb{C}$ is defined as $(f*g)(x) = \mathbb{E}_{y\sim\mathbb{F}_p^n}[f(y)\cdot g(x-y)]$. It is known that the Fourier transform of the convolution of two functions is equivalent to the product of their Fourier transforms: For any $r \in \mathbb{F}_p^n$, it holds that $\widehat{f*g}(r) = \widehat{f}(r)\cdot\widehat{g}(r)$.

For a set $S \subseteq \mathbb{F}_p^n$, we define the indicator function $\mathbf{1}_S : \mathbb{F}_p^n \to \{0,1\}$ as $\mathbf{1}_S(x) = 1$ if $x \in S$ and 0 otherwise. For a parameter $c > 0$, we define

$$\mathrm{Spec}_c(S) = \left\{r \in \mathbb{F}_p^n\colon \left|\widehat{\mathbf{1}_S}(r)\right| > c\right\}.$$

Lemma B.1 (Chang's inequality [Cha02])**.** *For a set $S \subseteq \mathbb{F}_p^n$ of size $|S| = \varepsilon \cdot p^n$ and any parameter $\rho > 0$, it holds that*

$$\dim(\mathrm{span}(\mathrm{Spec}_{\rho\varepsilon}(S))) \leq \frac{2\ln(1/\varepsilon)}{\rho^2}.$$

We note that the original lemma of Lemma B.1 from [Cha02] is stated for $S \subseteq \mathbb{Z}/n\mathbb{Z}$. Although the proof of the lemma for the case of $S \subseteq \mathbb{F}_2^n$ can be found in many places (e.g., [IMR14; HL20]), we could not find a proof for the case of $S \subseteq \mathbb{F}_p^n$ for a general prime power p. For completeness, we provide a proof of Lemma B.1 for this general case in Appendix B.1. The proof follows an elegant argument based on information theory, which can be found in [HL20; IMR14].

Proof of Lemma 5.5. The proof follows the original proof ([SS25, Lemma 8]) so that it also holds over $\mathbb{F}_p$ (the only difference is that we use the Fourier analysis over $\mathbb{F}_p^n$ instead of $\mathbb{F}_2^n$ and the use of $\mathbb{F}_p^n$-version of Chang's inequality).

Take an arbitrary set $S \subseteq \mathbb{F}_p^n$ of size $|S| = \varepsilon \cdot p^n$ and let $R := \mathrm{Spec}_{\varepsilon/2}(S) \setminus \{0\}$. The subspace V is $V := \mathrm{span}(R)^\perp = \{v \in \mathbb{F}_p^n \colon v^\top r = 0 \text{ for all } r \in R\}$. Note that $\dim V = n - \dim R \geq n - 8\ln(1/\varepsilon)$ from Lemma B.1. Fix an arbitrary $x \in V$. Observe that for any $v \in \mathbb{F}_p^n$, the probability

$$\Pr_{x_1,\dots,x_t,y_1,\dots,y_t \sim \mathbb{F}_p^n}[x_1,\dots,x_t,y_1,\dots,y_t \in S \,|\, v = x_1 + \cdots + x_t - y_1 - \cdots - y_t]$$

equals to the convolution

$$\underbrace{\mathbf{1}_S * \cdots * \mathbf{1}_S}_{t \text{ times}} * \underbrace{\mathbf{1}_{-S} * \cdots * \mathbf{1}_{-S}}_{t \text{ times}}(v), \tag{2}$$

where we write $-S = \{-s \colon s \in S\}$.

Now we bound Eq. (2). Observe that

$$\widehat{\mathbf{1}_{-S}}(r) = \mathop{\mathbb{E}}_{x \sim \mathbb{F}_p^n}\left[\mathbf{1}_S(-x) \cdot \omega^{-\mathrm{tr}(r^\top x)}\right] = \mathop{\mathbb{E}}_{y \sim \mathbb{F}_p^n}\left[\mathbf{1}_S(y) \cdot \omega^{\mathrm{tr}(r^\top y)}\right] = \overline{\mathop{\mathbb{E}}_{y \sim \mathbb{F}_p^n}\left[\mathbf{1}_S(y) \cdot \omega^{-\mathrm{tr}(r^\top y)}\right]} = \overline{\widehat{\mathbf{1}_S}(r)}.$$

Suppose $v \in V$. Since $r^\top v = 0$ for all $r \in R$, we have

$$\begin{aligned}
(2) &= \sum_{r \in \mathbb{F}_p^n} \widehat{\mathbf{1}_S}(r) \cdot \cdots \cdot \widehat{\mathbf{1}_S}(r) \cdot \widehat{\mathbf{1}_{-S}}(r) \cdot \cdots \cdot \widehat{\mathbf{1}_{-S}}(r) \cdot \chi_r(v) \\
&= \sum_{r \in \mathbb{F}_p^n} \left|\widehat{\mathbf{1}_S}(r)\right|^{2t} \cdot \chi_r(v) \\
&= \underbrace{\left|\widehat{\mathbf{1}_S}(0)\right|^{2t}}_{=\varepsilon^{2t}} \underbrace{\chi_0(v)}_{=1} + \sum_{r \in R} \left|\widehat{\mathbf{1}_S}(r)\right|^{2t} \cdot \underbrace{\chi_r(v)}_{=1} + \sum_{r \notin R \cup \{0\}} \left|\widehat{\mathbf{1}_S}(r)\right|^{2t} \cdot \chi_r(v) \\
&\geq \varepsilon^{2t} - \sum_{r \notin R \cup \{0\}} \left|\widehat{\mathbf{1}_S}(r)\right|^{2t} \cdot |\chi_r(v)| \\
&= \varepsilon^{2t} - \sum_{r \notin R \cup \{0\}} \left|\widehat{\mathbf{1}_S}(r)\right|^{2t}.
\end{aligned}$$

Here, note that $\widehat{\mathbf{1}_S}(0) = \mathbb{E}_{x \sim \mathbb{F}_p^n}[\mathbf{1}_S(x)] = \varepsilon$.

Now we bound $\sum_{r\notin R\cup\{0\}}\left|\widehat{\mathbf{1}_S}(r)\right|^{2t}$. By Parseval's identity for $\mathbf{1}_S$, we have $\varepsilon = \|\mathbf{1}_S\|_2^2 = \sum_{r\in\mathbb{F}_p^n}\left|\widehat{\mathbf{1}_S}(r)\right|^2$. Moreover, $\left|\widehat{\mathbf{1}_S}(r)\right| \le \varepsilon/2$ for all $r \notin R\cup\{0\}$ by the definition of R. Therefore, we have

$$\sum_{r\notin R\cup\{0\}}\left|\widehat{\mathbf{1}_S}(r)\right|^{2t} \le \max_{r\notin R\cup\{0\}}\left|\widehat{\mathbf{1}_S}(r)\right|^{2t-2}\cdot\sum_{r\neq 0}\left|\widehat{\mathbf{1}_S}(r)\right|^2 \le (\varepsilon/2)^{2t-2}\cdot(\varepsilon-\varepsilon^2).$$

Therefore, (2) is at least

$$\varepsilon^{2t} - (\varepsilon/2)^{2t-2}\cdot(\varepsilon-\varepsilon^2) = \varepsilon^{2t-2}\left(\varepsilon^2 - \frac{\varepsilon-\varepsilon^2}{2^{2t-2}}\right) \ge \varepsilon^{2t+1}$$

if $t \ge \frac{\log_2(1/\varepsilon)}{2} + 1$. This completes the proof. □

B.1 Proof of Chang's Inequality over $\mathbb{F}_p^n$

In this subsection, we prove Lemma B.1.

Notation of Information Theory. Let X be a random variable taking values in a finite set Ω. The entropy of X is defined as $H(X) = \sum_{x\in\Omega}\Pr[X=x]\ln\frac{1}{\Pr[X=x]}$ (note that we use the natural logarithm). Unless otherwise specified, we use the natural logarithm. Let $(X_1,\dots,X_n)$ be a random vector taking values in Ω^n. The subadditivity of entropy states that $H(X_1,\dots,X_n) \le H(X_1)+\cdots+H(X_n)$ (cf. [CT06, Theorem 2.6.6]).

Let $d_{\mathrm{TV}}(X,Y) = \frac{1}{2}\sum_{x\in\Omega}|\Pr[X=x]-\Pr[Y=x]|$ denote the total variation distance between X and Y. The following is a well-known fact about the total variation distance:

Fact B.2. *For any random variables X and Y taking values in a finite set Ω and any function $f\colon\Omega\to\mathbb{C}$ such that $\|f\|_\infty := \max_{x\in\Omega}|f(x)| \le 1$, it holds that*

$$\left|\mathop{\mathbb{E}}_{x\sim X}[f(x)] - \mathop{\mathbb{E}}_{y\sim Y}[f(y)]\right| \le 2d_{\mathrm{TV}}(X,Y).$$

Proof. This fact is well known if f is a real-valued function. We can prove it by using the triangle inequality:

$$\begin{aligned}\left|\mathop{\mathbb{E}}_{x\sim X}[f(x)] - \mathop{\mathbb{E}}_{y\sim Y}[f(y)]\right| &\le \sum_{x\in\Omega}|f(x)|\cdot|\Pr[X=x]-\Pr[Y=x]| \\ &\le \sum_{x\in\Omega}|\Pr[X=x]-\Pr[Y=x]| \\ &= 2d_{\mathrm{TV}}(X,Y).\end{aligned}$$

□

Let U be the uniform distribution over Ω. From Pinsker's inequality, we can relate the entropy of X and the total variation distance between X and the uniform distribution U as follows:

$$\ln|\Omega| - H(X) \ge 2d_{\mathrm{TV}}(X,U)^2. \tag{3}$$

Lemma B.3. *Let $S \subseteq \mathbb{F}_p^n$ be a set of size $|S| = \varepsilon \cdot p^n$. Let $e_1, \dots, e_n \in \mathbb{F}_p^n$ be the standard basis of $\mathbb{F}_p^n$. Then, it holds that*

$$\sum_{i \in [n]} \left| \widehat{\mathbf{1}_S}(e_i) \right|^2 \leq 2\varepsilon^2 \ln(1/\varepsilon).$$

Proof. Let $X = (X_1, \dots, X_n) \sim S$ be a random vector uniformly distributed over S and $U = (U_1, \dots, U_n) \sim \mathbb{F}_p^n$ be a random vector uniformly distributed over $\mathbb{F}_p^n$. For any $i \in [n]$ and $a \in \mathbb{F}_b$, note that $\Pr[\mathrm{tr}(X_i) = a] = \frac{|\{r \in S \colon \mathrm{tr}(r_i) = a\}|}{|S|} = \frac{|\{r \in S \colon \mathrm{tr}(r_i) = a\}|}{p^n} \cdot \frac{p^n}{|S|} = \varepsilon^{-1} \cdot \Pr[U \in S \text{ and } \mathrm{tr}(U_i) = a]$. Recall that $p = b^m$ for a prime b and $m \in \mathbb{N}$. Then, we have

$$\begin{aligned}
\widehat{\mathbf{1}_S}(e_i) &= \mathbb{E}\left[\mathbf{1}_S(U) \cdot \omega^{-\mathrm{tr}(U_i)}\right] \\
&= \sum_{a \in \mathbb{F}_b} \omega^{-a} \cdot \Pr[U \in S \text{ and } \mathrm{tr}(U_i) = a] \\
&= \varepsilon \cdot \sum_{a \in \mathbb{F}_b} \omega^{-a} \cdot \Pr[\mathrm{tr}(X_i) = a] \\
&= \varepsilon \cdot \left(\mathbb{E}\left[\omega^{-\mathrm{tr}(X_i)}\right] - \mathbb{E}\left[\omega^{-\mathrm{tr}(U_i)}\right] \right). && \because \mathbb{E}\left[\omega^{-\mathrm{tr}(U_i)}\right] = \frac{1}{b} \sum_{a \in \mathbb{F}_b} \omega^{-a} = 0
\end{aligned}$$

Since the function $f \colon x \mapsto \omega^{-x}$ satisfies $\|f\|_\infty = 1$, we can apply Fact B.2 to get

$$\begin{aligned}
\left| \widehat{\mathbf{1}_S}(e_i) \right|^2 &\leq 4\varepsilon^2 \cdot d_{\mathrm{TV}}(X_i, U_i)^2 \\
&\leq 2\varepsilon^2 \cdot (\ln p - H(X_i)). && \because \text{Eq. (3)}
\end{aligned}$$

Summing over $i \in [n]$, we obtain

$$\begin{aligned}
\sum_{i \in [n]} \left| \widehat{\mathbf{1}_S}(e_i) \right|^2 &\leq 2\varepsilon^2 \cdot \left(n \ln p - \sum_{i \in [n]} H(X_i) \right) \\
&\leq 2\varepsilon^2 \cdot (n \ln p - H(X)) && \because \text{subadditivity of entropy} \\
&= 2\varepsilon^2 \cdot (\ln p^n - \ln |A|) \\
&= 2\varepsilon^2 \cdot \ln(1/\varepsilon).
\end{aligned}$$

This completes the proof. □

We can replace the standard basis $e_1, \dots, e_n$ in Lemma B.1 with any basis of $\mathbb{F}_p^n$ as follows:

Corollary B.4. *Let $S \subseteq \mathbb{F}_p^n$ be a set of size $|S| = \varepsilon \cdot p^n$. Let $(b_1, \dots, b_n) \in \mathbb{F}_p^n$ be any basis of $\mathbb{F}_p^n$. Then, it holds that*

$$\sum_{i \in [n]} \left| \widehat{\mathbf{1}_S}(b_i) \right|^2 \leq 2\varepsilon^2 \ln(1/\varepsilon).$$

Proof. Since $(b_1, \dots, b_n)$ is a basis of $\mathbb{F}_p^n$, there exists a nonsingular matrix $B \in \mathbb{F}_p^{n \times n}$ such that $b_i^\top B = e_i^\top$ for all $i \in [n]$. Let $B \cdot S := \{Bs \colon s \in S\}$. Then, we obtain

$$\begin{aligned}
\widehat{\mathbf{1}_S}(b_i) &= \mathop{\mathbb{E}}_{r \sim \mathbb{F}_p^n} \left[\mathbf{1}_S(r) \cdot \omega^{-\mathrm{tr}(b_i^\top r)} \right] \\
&= \mathop{\mathbb{E}}_{r \sim \mathbb{F}_p^n} \left[\mathbf{1}_S(r) \cdot \omega^{-\mathrm{tr}(e_i^\top B r)} \right] \\
&= \mathop{\mathbb{E}}_{s \sim \mathbb{F}_p^n} \left[\mathbf{1}_S(B^{-1}s) \cdot \omega^{-\mathrm{tr}(e_i^\top s)} \right] && \text{set } s = Br \\
&= \mathop{\mathbb{E}}_{s \sim \mathbb{F}_p^n} \left[\mathbf{1}_{BS}(s) \cdot \omega^{-\mathrm{tr}(e_i^\top s)} \right] \\
&= \widehat{\mathbf{1}_{BS}}(e_i).
\end{aligned}$$

Since $|B \cdot S| = |S|$, we can apply Lemma B.1 for $B \cdot S$ to obtain the desired inequality. □

Now, we are ready to prove Lemma B.1.

Proof of Lemma B.1. Let $S \subseteq \mathbb{F}_p^n$ be a set of size $|S| = \varepsilon \cdot p^n$. Suppose for contradiction that $\mathrm{Spec}_{\rho\varepsilon}(S)$ contains more than $d := 2\ln(1/\varepsilon)/\rho^2$ linearly independent vectors $b_1, \dots, b_d \in \mathbb{F}_p^n$. Extend them to obtain a basis $(b_1, \dots, b_n)$ of $\mathbb{F}_p^n$ that contains those vectors. Since $b_1, \dots, b_d \in \mathrm{Spec}_{\rho\varepsilon}(S)$, we have

$$\sum_{i \in [d]} \left| \widehat{\mathbf{1}_S}(b_i) \right|^2 > d \cdot \rho^2 \varepsilon^2 = 2\varepsilon^2 \ln(1/\varepsilon).$$

On the other hand, by Corollary B.4, we have

$$\sum_{i \in [n]} \left| \widehat{\mathbf{1}_S}(b_i) \right|^2 \leq 2\varepsilon^2 \ln(1/\varepsilon).$$

This is a contradiction. □

ECCC ISSN 1433-8092
https://eccc.weizmann.ac.il