Abstract

In this paper, we establish a cubic Goldreich-Levin algorithm which makes polynomially-many queries to a function $f: \mathbb{F}_p^n \to \mathbb{C}$ and produces a decomposition of $f$ as a sum of cubic phases and a small error term. This is a natural higher order generalization of the classical Goldreich-Levin algorithm. The classical (linear) Goldreich-Levin algorithm has wide-ranging applications in learning theory, coding theory and the construction of pseudorandom generators in cryptography, as well as being closely related to Fourier analysis. Higher order Goldreich-Levin algorithms on the other hand involve central problems in higher order Fourier analysis, namely the inverse theory of the Gowers $U^k$ norms, which are well-studied in additive combinatorics. The only known result in this direction prior to this work is the quadratic Goldreich-Levin theorem, proved by Tulsiani and Wolf in 2011. The main step of their result involves an algorithmic version of the $U^3$ inverse theorem. More complications appear in the inverse theory of the $U^4$ and higher norms. Our cubic Goldreich-Levin algorithm is based on algorithmizing recent work by Gowers and Milićević who proved new quantitative bounds for the $U^4$ inverse theorem. In the process, we also solve a problem of local error correction for cubic Reed-Muller codes beyond the list decoding radius.
1 Introduction

Our work lies in the field of higher order Fourier analysis, which can be thought of as an extension of classical Fourier analysis to higher order characters. Due to its ability to detect linear patterns such as three term arithmetic progressions, classical Fourier analysis is a useful tool in the field of additive combinatorics. It was the key ingredient in the proof of Roth’s theorem, which states that every subset of positive upper density contains a three-term arithmetic progression. The same problem for longer $k$-term arithmetic progressions was solved by Szemerédi [15]. Known now as Szemerédi’s theorem, this is one of the milestones of additive combinatorics. Concretely, Szemerédi proved that every subset of the integers with positive upper density contains arbitrarily long arithmetic progressions. Szemerédi’s original proof used a regularity lemma, which gave poor quantitative bounds. In his seminal paper [8] that gave a new proof with reasonable quantitative bounds for Szemerédi’s theorem, Gowers introduced the concept of the Gowers $U^k$ norm and pioneered the technique of higher order Fourier analysis. This bridged one of the limitations of classical Fourier analysis; classical Fourier analysis is fundamentally unable to detect more complex patterns such as arithmetic progressions of lengths four and above. Gowers overcame this challenge by considering higher order phase functions, instead of just linear phase functions as in classical Fourier analysis. The work by Gowers laid the groundwork for the study of higher order Fourier analysis.

Parallel to these developments in the additive combinatorics world, classical Fourier analysis has also been a mainstay in the realm of computer science. The roots of such applications of classical Fourier analysis can be traced back to the field of property testing, which is the study of efficient algorithms that with high probability decide correctly whether the input is “close” to a certain property. Particularly, one of the first uses of classical Fourier analysis in property testing was to provide a simplified proof of correctness for the Blum-Luby-Rubinfeld (BLR) algorithm [4], which in constant queries detects if a function $f: \mathbb{F}_2^n \rightarrow \mathbb{F}_2$ is close to linear. Building off this linear testing regime, there is a very related question one could pose. Given a function $f$ that is close to linear, one could ask whether there exists an efficient algorithm that would identify all the candidate linear functions close to $f$. One application of the celebrated Goldreich-Levin algorithm [7] is to solve this problem. Beyond this context, the Goldreich-Levin algorithm also has wide-ranging applications in many areas of theoretical computer science, including in learning theory [14], coding theory [1] and the construction of pseudorandom generators in cryptography [13], the latter being the context that first motivated its study.

We can also describe the Goldreich-Levin algorithm through the lens of coding theory. Suppose we are given a function $f$ that is close to a linear function. A linear function can be interpreted as a Walsh-Hadamard code, so query access to $f$ corresponds to a corrupted codeword. It follows that the Goldreich-Levin algorithm also solves the list decoding of the Walsh-Hadamard code. In the context of coding theory, Reed-Muller codes are a generalization of Walsh-Hadamard codes from the linear setting to the setting of higher order polynomials. As one might expect, questions about proximity to higher degree polynomials naturally arise in the context of decoding Reed-Muller codes. It is therefore natural to study higher degree analogues of the BLR linearity test and Goldreich-Levin algorithm.

In the “99% regime”, where the main goal is to distinguish functions that are close to higher degree polynomials from those that are not, there is a natural generalization of the BLR linearity test via a local definition of polynomials by Alon et al. [2] known as the AKKLR test. The idea behind the AKKLR test is akin to taking a majority vote among higher-dimensional parallelepipeds, which is related to the local definition of polynomials via vanishing directional derivatives. Such a majority vote style of argument fails in the “1% regime”, where the goal is instead to detect any non-trivial agreement, which may potentially be very small, with a polynomial. This is a much more lofty goal than that in the 99% regime, because functions close to higher degree polynomials intuitively have a lot of structure we can exploit in majority vote arguments as compared to those studied in the 1% regime. We run into similar issues when trying to find analogues of higher order Goldreich-Levin. It was soon discovered that many of the obstacles facing such generalizations are similar to those in studying long arithmetic progressions, and that these analogues are closely related to difficult problems in higher order Fourier analysis, such as the inverse theorem for the $U^k$ norm.
In more detail, the Gowers $U^k$ norm for a function $f : \mathbb{F}_p^n \to \mathbb{C}$ is defined by
\[
\|f\|_{U^k}^2 = \mathbb{E}_{x,h_1,...,h_k \in \mathbb{F}_p} \partial_{h_1} \partial_{h_2} \cdots \partial_{h_k} f(x)
\]
where $\partial_h f(x) = f(x + h) - f(x)$ is the discrete (multiplicative) derivative. Since a degree $k - 1$ polynomial vanishes upon taking $k$ successive derivatives, it follows that $f(x) = \omega^{p(x)}$, where $p(x)$ is a degree $k - 1$ polynomial satisfies $\|f\|_{U^k} = 1$. A natural question to ask is if the converse is true; namely, if $f : \mathbb{F}_p^n \to \mathbb{C}$ where $\|f\|_{\infty} \leq 1$ is a function for which $\|f\|_{U^k} \geq \delta$ then does there exist a degree $k - 1$ polynomial $p(x)$ such that $|\langle f, \omega^p \rangle| \gg \delta$?

The first positive answer in this direction was given by Tao and Ziegler [19], who employed ergodic theoretic infinitary techniques to establish the following theorem.

**Theorem 1.1** ([19] Theorem 1.10), $U^k$ inverse theorem. For every $c > 0$, every positive integer $k$, and every prime $p \geq k$, there is a constant $c' > 0$ with the following property: for every function $f : \mathbb{F}_p^n \to \mathbb{C}$ with $\|f\|_{\infty} \leq 1$ and $\|f\|_{U^k} \geq c$, there is a polynomial $\pi : \mathbb{F}_p^n \to \mathbb{F}_p$ of degree at most $k - 1$ such that $\mathbb{E}_x f(x) \omega^{-\pi(x)} \geq c'$.

Roughly speaking, this theorem states that a function with large $U^k$ norm correlates with a quadratic phase function.

However, in many computer science applications such as those on communication complexity [21] and pseudorandom generators which fool low-degree polynomials [5], the existence of efficient algorithms often rely on good quantitative bounds from these inverse theorems. Ergodic methods by Tao and Ziegler unfortunately do not translate to reasonable quantitative bounds. The first work in this direction of getting good quantitative bounds was by Green and Tao [11] as well as Samorodnitsky [16] for the $U^3$ inverse theorem. Building off this work, Tulsiani and Wolf [20] established a quadratic Goldreich-Levin theorem by making algorithmic the $U^3$ inverse theorem.

The next in this line is work is by Gowers and Miličević [9], who proved the following bound for the $U^4$ inverse theorem.

**Theorem 1.2** ([9], quantitative $U^4$ inverse theorem). For every $c > 0$, and every prime $p \geq 5$, there is a constant $c' = O(\exp(\exp(\text{quasi-poly}(c^{-1}), p)))$ with the following property: for every function $f : \mathbb{F}_p^n \to \mathbb{C}$ with $\|f\|_{\infty} \leq 1$ and $\|f\|_{U^4} \geq c$, there is a polynomial $\pi : \mathbb{F}_p^n \to \mathbb{F}_p$ of degree at most 3 such that $\mathbb{E}_x f(x) \omega^{-\pi(x)} \geq c'$.

In this paper, we will establish a cubic Goldreich-Levin theorem by algorithmizing the $U^4$ inverse theorem.

**Theorem 1.3** (algorithmic $U^4$ inverse theorem). Given a prime $p \geq 5$ and $\delta, \epsilon > 0$, set $\eta^{-1} = \exp(\exp(\text{quasi-poly}(c^{-1})))$. For a bounded function $f : \mathbb{F}_p^n \to \mathbb{C}$ that satisfies $\|f\|_{U^4} \geq \epsilon$, there is an algorithm that makes $O(\text{poly}(n, \eta^{-1}, \log(\delta^{-1})))$ queries to $f$ and, with probability at least $1 - \delta$, outputs a cubic polynomial $P : \mathbb{F}_p^n \to \mathbb{F}_p$ such that $\mathbb{E}_x f(x) \omega^{-P(x)} > \eta$.

While in the quadratic Goldreich-Levin case, the main tool that was used was an algorithmic Balog-Szemerédi-Gowers Theorem, in order to establish the cubic Goldreich-Levin we will need a wider assortment of tools. We will give a more detailed discussion for why this is the case in the following section, but the main gist is that the quadratic Goldreich-Levin problem reduces to finding an affine function that overlaps greatly with a function defined via a suitable large Fourier spectrum. In the cubic case, however, we will need to find a bi-affine rather than affine function which agrees on a significant fraction of inputs. All of the standard additive combinatorics tools are in the univariate setting, and to handle the bivariate case we will need to piece together various theorems from the standard toolbox in intricate ways.

Another perspective on our algorithm is that it is effectively a self-correction procedure for Reed-Muller codes of order 3. In effect, our algorithm solves the problem of local error correction for cubic Reed-Muller codes beyond the list decoding radius.
**Theorem 1.4.** Given $f : \mathbb{F}_p^n \to \mathbb{F}_p$ such that there exists cubic polynomial $P$ satisfying $\text{dist}(f,P) \leq 1 - \frac{1}{p} - \epsilon$. There exist an algorithm \texttt{find-cubic} that in time $O(\text{poly}(n, \eta^{-1}))$ outputs with high probability a cubic polynomial $Q$ such that $\text{dist}(f,Q) \leq 1 - \frac{1}{p} - \exp(\text{exp quasi-poly}(\epsilon))$. Here $\text{dist}(P,Q)$ is the normalized Hamming distance.

One way to interpret this result is as follows. Recall that the Reed-Muller code of order $k$ over $\mathbb{F}_p^n$ is the set of polynomials of degree at most $k$ evaluated over $\mathbb{F}_p^n$. This means that we can think of $\hat{P}$ as a codeword and $f$ as a noisy version of the codeword that is obtained by “adding the noise” given by $f - P$. Up to the maximum distance in which decoding makes sense (being better than outputting a random constant codeword), we can decode in the sense of finding a codeword within a certain given distance even beyond the list decoding radius. This is a much more general extension of the $k = 3$ case in [3]. While in our case our algorithm can tolerate arbitrary noise since $f$ can in effect be any function, in [3] the version of the decoding problem that they solved only works when the noise is “structured”. Precisely, their algorithm is only able to tolerate noise in the form of higher degree polynomials.

By iteratively applying the algorithmic inverse theorem, we aim to turn such algorithmic inverse theorems into a Frieze-Kannan [8] style weak regularity decomposition for functions. Roughly speaking, we build on the dichotomy of structure and pseudorandomness for functions: we can write bounded theorems into a Frieze-Kannan [6] style weak regularity decomposition for functions. Roughly speaking, this end, Theorem 1.5 can be thought of as a cubic extension of the classical Goldreich-Levin algorithm.

**Theorem 1.5.** Let $\epsilon, \delta > 0$ and $B > 1$. Let \texttt{find-cubic} be an algorithm which given query access to a function $f : \mathbb{F}_p^n \to [-B,B]$ satisfying $\|f\|_{U^4} \geq \epsilon$ outputs with probability $1 - \delta$ a cubic polynomial phase function $\eta$ such that $|\langle f, \eta \rangle| \geq \eta$ for some $\eta = \eta(\epsilon,B)$. Then there is an algorithm $\texttt{U}_4$-weak-regularity such that given any function $g : \mathbb{F}_p^n \to [-1,1]$ outputs with probability at least $1 - 2\delta/\eta^2$ a decomposition

$$f = c_1\eta + \cdots + c_r\eta + g$$

satisfying $r \leq 2/\eta^2$ and $\|g\|_{U^4} \leq \epsilon$. The algorithm makes at most $r$ calls to \texttt{find-cubic}.

The Goldreich-Levin algorithm can be thought of as decomposing a function into linear phases. To this end, Theorem 1.5 can be thought of as a cubic extension of the classical Goldreich-Levin algorithm.

Lastly, we will give a modification of a part of the proof by Gowers and Miličević to obtain a better quantitative bound in the $U^4$ inverse theorem with one less exponent. More precisely, we prove the following theorem.

**Theorem 1.6.** For every $c > 0$, and every prime $p \geq 5$, there is a constant $c' = O(\exp(\text{quasi-poly}(c^{-1},p)))$ with the following property: for every function $f : \mathbb{F}_p^n \to \mathbb{C}$ with $\|f\|_{\infty} \leq 1$ and $\|f\|_{U^4} \geq c$, there is a polynomial $\pi : \mathbb{F}_p^n \to \mathbb{F}_p$ of degree at most 3 such that $\mathbb{E}_x f(x)\omega^{-\pi(x)} \geq c'$.

One of the exponents was introduced by Gowers and Miličević for technical Fourier analytic reasons, due to an application of principle of inclusion and exclusion to cut up some subspaces in order to bound the $L^1$ norm of a Fourier coefficient. By a more careful analysis of the subspaces motivated by the idea of high rank maps that Gowers and Miličević employ elsewhere in their proof, we can remove the exponential from this technical step.

**Outline.** In Section 2, we give an outline of the algorithm and discuss some of the difficulties faces in generalizing from the quadratic setting to the cubic one. In Section 3, we prove an algorithmic decomposition result which combined with our algorithmic $U^4$ inverse theorem, gives the cubic Goldreich-Levin algorithm. In Section 4, we collect some algorithmic primitives that we will be frequently using in our algorithmic $U^4$ inverse theorem, including a slightly generalized form of Goldreich-Levin as well as algorithmic versions of some additive combinatorics theorems. We first modularize the algorithmic $U^4$ inverse theorem and give self-contained proofs for each individual piece in Section 5, which we then combine together in Sub-section 5.1 to give a complete proof of our main theorem. In Section 6, we
discuss how to improve the quantitative bounds of Gowers and Milićević to remove an exponent in the bound for $\eta$.

Acknowledgements. This project was carried out as part of the 2021 Summer Program in Undergraduate Research+ (SPUR+) of the MIT Mathematics Department. We would like to thank our mentor Jonathan Tidor for suggesting this interesting problem, as well as for his dedicated mentorship and patient guidance. We would also like to thank Prof. Yufei Zhao for many insightful discussions and also for his constant support and encouragement. Furthermore, we would like to express our gratitude towards Prof. David Jerison and Prof. Ankur Moitra for organizing this program and for their invaluable advice during the weekly check-in meetings.

2 Ideas of proof

In this section, we aim to provide some intuition for the obstacles that one faces when generalizing from the quadratic Goldreich-Levin setting to the cubic Goldreich-Levin setting. We will also give a brief high-level overview of the flow of the algorithm and expound on some of the algorithmic sampling strategies in several key steps. Particularly, we draw parallels to the proof of the $U^4$ inverse theorem by Gowers and Milićević and explain the areas in which our algorithm differ by highlighting where we implicitly used their existential results to simplify our algorithm.

First, let us recall some notation and definitions.

Definition 2.1 (Discrete (multiplicative) derivative). Suppose that $f : \mathbb{F}_p^n \to \mathbb{C}$ is a function. Then for any $h \in \mathbb{F}_p^n$ we define the function $\partial_h f$ by

$$\partial_h f(x) := f(x + h)f(x).$$

By considering the function $f(x) = \omega^{\phi(x)}$, we can see where the name of the derivative comes from: we are taking a finite difference of the phase $\phi(x)$ and thereby “differentiating” it. We will also write $\partial_{h_1,h_2} f := \partial_{h_1}(\partial_{h_2} f)$.

Definition 2.2 ($U^k$ norm). Let $f : \mathbb{F}_p^n \to \mathbb{C}$ be a function. For each positive integer $k$, we define the $U^k$ norm of $f$ as

$$\|f\|_{U^k} := \mathbb{E}_{x,h_1,h_2,...,h_k \in \mathbb{F}_p^n} \partial_{x,h_1,h_2,...,h_k} f(x).$$

Definition 2.3 (Large Fourier spectrum). The $\gamma$-large Fourier spectrum of $f : \mathbb{F}_p^n \to \mathbb{C}$ is given by

$$\text{Spec}_\gamma(f) := \{ r \in \mathbb{F}_p^n : \hat{f}(r) \geq \gamma \}.$$

2.1 Overview of quadratic Goldreich-Levin

We begin by providing an overview of the proof of quadratic Goldreich-Levin by Tulsiani and Wolf. Here we recall the statement of their result.

Theorem 2.4. Given $\epsilon, \delta > 0$ there exists $\eta = \exp(-1/\epsilon^C)$ and a randomized algorithm Find-Quadratic running in time $O(n^4 \log n \cdot \text{poly}(1/\epsilon, 1/\eta, \log(1/\delta)))$ which given query access to $f : \mathbb{F}_p^n \to \{-1, 1\}$ either outputs a quadratic form $q$ or $\bot$. The algorithm has the following guarantee:

- If $\|f\|_{U^3} \geq \epsilon$ then with probability at least $1 - \delta$ it finds a quadratic form $q$ such that $\langle f, (-1)^q \rangle \geq \eta$.
- The probability that the algorithm outputs a quadratic form $q$ with $\langle f, (-1)^q \rangle \leq \eta/2$ is at most $\delta$.

In order to provide some intuition for the first step of their proof, we consider the case of $f(x) = \omega^g(x)$ where $g(x) = x^T M x$ is a quadratic form for some $M \in \text{Mat}_n(\mathbb{F}_2)$. Taking the derivative of a quadratic form reduces its degree by one to a linear term. Consequently, it turns out that $\|f\|_{U^3} = 1$ and by
definition we also have that $f$ evidently correlates well with a quadratic phase. Let us observe further that

$$\partial_h(-1)^{\phi(x)} = (-1)^{(x+h)^TM(x+h)}(-1)^{x^TMx}$$

which implies that $\partial_h(-1)^{\phi(x)}$ has precisely one Fourier coefficient concentrated at $(M + M^T)h$. The way to think about this intuitively is that we have differentiated a quadratic, so it is natural to expect that $\partial_h(-1)^{\phi(x)}$ has large correlation with a linear phase function.

This example suggests the following strategy: given $f$ with $\|f\|_{U^3} \geq \epsilon$ we extract the large Fourier spectrum of $\partial_h f$. In more detail, we may use the Goldreich-Levin algorithm to designate a random element of $\text{Spec}_\rho(\partial_h f)$ as $\phi(h)$. If we show that there is a suitable affine map $A(x) = Tx + b$ such that $A(h) = \phi(h)$ for lots of $h$ then we can reverse the argument in the above paragraph by “anti-differentiating” $A$ to recover the desired quadratic form.

There are many tools from additive combinatorics that allows us to find such forms of affine structure. It turns out we are able to use some Fourier analytic manipulation to get some form of “weak additive structure” on $\phi$ which can be boosted to the kind of affine structure that we want by invoking the following two theorems in succession.

**Theorem 2.5** (Modified Balog-Szemerédi-Gowers theorem). Let $A \subset \mathbb{F}_2^n$ be such that $\mathbb{P}_{a_1, a_2 \in A}[a_1 + a_2 \in A] \geq \rho$. Then there exists $A' \subset A$ with $|A'| \geq \rho|A|$ such that $|A + A'| \leq (2/\rho)^8|A|$.

**Theorem 2.6** (Freiman’s Theorem). Let $A \subset \mathbb{F}_2^n$ be such that $|A + A| \leq K|A|$. Then $A$ is contained in a subspace of size at most $2^{O(K^C)}|A|$.

By providing algorithmic versions of these theorems, Tulsiani and Wolf manage to implement something similar to the strategy we outlined earlier to get `find-quadratic`. One of the main contributions of the Tulsiani and Wolf paper is to therefore provide an algorithmic version of the Balog-Szemerédi-Gowers theorem. The proof of this theorem given by Gowers relies on a probabilistic method technique called dependent random choice, which conceivably can be modified to a randomized algorithm.

### 2.2 Generalizing to the cubic setting

One way to reframe the quadratic setting is to note the following recursive definition of the $U^k$ norms:

$$\|f\|_{U^k} = \mathbb{E}_h \|\partial_h f\|_{U^{k-1}}^k.$$  

 Particularly, we have that $\|f\|_{U^3}^8 = \mathbb{E}_h \|\partial_h f\|_{U^2}^4 = \mathbb{E}_h \|\partial_h f\|_{U^3}^4$ so taking one discrete derivative reduces the problem to the setting of classical Fourier analysis.

As we mentioned in the introduction, the cubic Goldreich-Levin setting ultimately boils down to giving an algorithmic version of the $U^3$ inverse theorem. If we try to do this same trick of taking one discrete derivative in the $U^4$ setting, we get $\|f\|_{U^4}^{16} = \mathbb{E}_h \|\partial_h f\|_{U^3}^4$ and combining with a Markov argument to remove the expectation we could use the $U^3$ inverse theorem to get quadratics that correlate with $\partial_h f$. This is the approach taken by Tao and Ziegler. However, much care needs to be taken in order to stitch together the quadratics corresponding to different $a$. Tao and Ziegler then translates the problem into the language of ergodic theory, and employ infinitary techniques.

To circumvent this, Gowers and Milčević instead opt to difference all the way down to $U^2$ via $\|f\|_{U^4}^{16} = \mathbb{E}_{a,b} \|\partial_{a,b} f\|_{U^2}^4$. This is because the $U^2$ setting is equivalent to working with classical Fourier analysis, and there are a lot more tools at our disposal. The trade-off here, however, is that we now end up having to work with the bivariate $\partial_{a,b} f$. Porting over an argument analogous to the previous section with $\phi$ picking out the large Fourier spectrum of $\partial_{a,b} f$, we would expect that the goal in this case is to find a bi-affine map $T$ such that $T(a,b) = \phi(a,b)$ for many $(a,b) \in (\mathbb{F}_2^n)^2$. There are few to no results in additive combinatorics that would allow us to directly work with bi-affine maps. We will need to therefore invoke the standard additive combinatorics toolbox in much more subtle ways in order to get bilinear extensions that can be used in our setting.
In slightly more detail, the algorithmic version of this first step is as follows. A Markov style argument using the identity at the beginning of this paragraph shows that since \( \| f \|_{L^4} \geq c \), there is a set \( A \subset (\mathbb{F}_p^n)^2 \) of density at least \( c^{16}/2 \) such that \( \| \partial_{a,b} f \|_{L^4} = \| \partial_{a,b} f \|_{L^2} \geq c^4/2 \) for all \( (a, b) \in A \). By Hölder’s, we have that \( \text{poly}(c) \leq \| \partial_{a,b} f \|_4^4 \leq \| \partial_{a,b} f \|_2^2 \| \partial_{a,b} f \|_{\infty}^2 \leq \| \partial_{a,b} f \|_{\infty}^2 \). Tracing through this argument carefully, we get a set \( A \) on which there exists a function \( \phi: A \to \mathbb{F}_p^n \) such that \( |\partial_{a,b} f(\phi(a, b))| \geq \text{poly}(c) \). Since we are working with the large Fourier spectrum, in order to identify \( \phi \) and give a membership tester for \( A \) we can effectively run Goldreich-Levin.

### 2.3 Proof overview

We now give a rough scheme of the proof, but because we are giving a high level overview it is more convenient for us to use language such as “1% structure” and “99% structure”. We describe something as being 1% if the density of the object in the appropriate ambient space is something like \( \epsilon > 0 \), while we call it 99% if its density is more like on the scale of \( 1 - \epsilon \).

Recall that the setting we are working with is as follows: we have membership tester for a set \( A \subset (\mathbb{F}_p^n)^2 \) as well as query access to a function \( \phi: A \to \mathbb{F}_p^n \). The goal is to find a bi-affine function \( T \) such that \( T(a, b) = \phi(a, b) \) holds for a large proportion of \( (a, b) \) in the domain \( A \).

There will be two concepts that arise in this section: one is the idea of additive structure satisfied by \( \phi \) on a set and another is the idea of additive structure on the domain itself. In the dream case \( \phi \) is close to bi-affine, so we would expect \( \phi \) to possess some form of additive structure. In additive combinatorics there is also the notion of a set possessing additive properties, usually in relation to its successive sumsets or difference sets containing linear structure. It turns out that having both types of additive structure will be crucial in the argument.

Actually instead of working with \( \phi \) it will turn out for technical reasons that it is more convenient for us to work with a certain convolution \( \psi \) of \( \phi \); morally we can think of \( \psi \) as a suitable weighted average of \( \phi \) across parallelograms. Intuitively, this form of averaging used to define \( \psi \) will allow us to do some form of majority vote over parallelograms to select a bi-affine map possessing large overlap with \( \phi \). Nevertheless, technicalities aside, \( \psi \) should possess similar additive properties as \( \phi \).

1. (1% \( \Rightarrow \) 99% structure for \( \phi \)) It turns out, it turns out that \( \phi \) has 1% additive structure on \( A \). In Gowers and Miličević’s proof, they pass to a subset \( A' \subset A \) to boost this 1% structure of \( \phi \), so that \( \phi \mid_{A'} \) has 99% structure. They do this via a “dependent random selection” probabilistic argument, where the rough idea is that we probabilistically select elements of \( A \) to include in \( A' \) via a certain distribution that biases our choices towards the inclusion of elements on which \( \phi \) respects additive structure. Because of the probabilistic nature of this proof of existence of \( A' \), it is conceivable that we can turn it into a probabilistic algorithm for testing membership in \( A' \); we can give a sampling randomized algorithm for testing membership in \( A' \) as long as we have a certifier which checks that the output set has the desired property of \( \phi \) possessing 99% structure on it. By some algebra, we can show that \( \psi \) also possesses a suitable version of 99% additive structure on \( A' \).

2. (Obtaining additive structure for the underlying set) As we have alluded to earlier, we would also like to pass from \( A' \) to a related set \( A'' \) which possess some additive structure, while maintaining the property that \( \psi \mid_{A''} \) still possesses 99% additive structure. The kind of set structure that is useful for us in this context turns out to be that of a high rank bilinear Bohr set, namely the level set of a bi-affine map \( \beta \). Roughly speaking, high rank bilinear Bohr sets are quasi-random in the sense that the number of solutions to linear equations on this Bohr set is approximately what we would expect for a random subset of \( (\mathbb{F}_p^n)^2 \).

This is helpful in our context because say if \( A'' \) was completely unstructured then despite knowing that \( \psi \mid_{A''} \) is additive we do not have enough control over whether we can suitably interpolate the values of \( \psi \) on \( A'' \) to obtain a bi-affine map \( T': A'' \to \mathbb{F}_p^n \). Therefore, having some structure on the underlying set \( A'' \) helps us to extract more information about \( \psi \).
To that end we will first need to identify the bi-affine map $\beta$, and then find an appropriate high rank level set. The latter is comparatively easier. The former can be done via a bilinear extension of the classical Bogolyubov theorem. The subtlety is that while classical Bogolyubov theorem is established by examining the large Fourier spectrum of an appropriate convolution and can thereforbe algorithmized easily by an application of Goldreich-Levin, the bilinear variant is much more involved. The bilinear variant requires careful successive applications of versions of Balog-Szemerédi-Gowers and Freiman’s theorems to find affine maps which cover a large Fourier spectrum, before stitching them together in an appropriate way. Since the versions of Balog-Szemerédi-Gowers and Freiman’s theorem that we require differ from that used in Tulsiani and Wolf, we develop these in detail in the section Algorithmic Tools.

(3) (99% structure $\Rightarrow$ 100% structure) At this stage we have restricted our attention to a set $A''$ that itself has a lot of structure and $\psi \mid_{A''}$ has 99% structure. By an intricate analysis using the quasi-random properties, namely that $A''$ possesses roughly an expected number of linear patterns with $\psi$ “respecting” these linear patterns, we can recover some bi-affine $T': A'' \to \mathbb{F}_p^n$ that agrees with $\psi$ via some form of majority vote over the linear patterns. With some manipulations, we can also show that this $T'$ agrees with $\phi$ on a significant fraction of $(\mathbb{F}_p^n)^2$ as well.

The next step is to extend the domain of $A''$ to $(\mathbb{F}_p^n)^2$. Gowers and Milićević build $T': (\mathbb{F}_p^n)^2 \to \mathbb{F}_p^n$ by showing that we can specify the values of $T$ on $(\mathbb{F}_p^n)^2 \setminus A''$ in a way that is consistent, by invoking the quasi-random properties of $A''$.

In this step our algorithmic version differs significantly in length from the proof of Gowers and Milićević. The nice part about algorithmizing this step is that knowing of the existence of this extends uniquely into $T$. That is, to translate into an algorithm, we sample enough linear patterns – which in this case turn out to basically be parallelograms – and then do a majority vote.

(4) (“Anti-differentiating” and symmetrization) At this point we have achieved the stated goal of recovering a bi-affine function $T$ such that $T(a, b) = \phi(a, b)$. Recall that $\phi(a, b)$ picked out the large Fourier spectrum of $\partial_{a,b}f$. We would therefore need to “anti-differentiate” $\phi(a, b)$ in order to recover information of $f$. For technical reasons, we also need $T$ to have some symmetry properties in order for this “anti-differentiating step” to work out. This symmetrization step involves dividing by 6, so in $\mathbb{F}_3$ and $\mathbb{F}_2$ some more care needs to be taken and there a couple more algorithmic linear algebraic steps. After implementing this “anti-differentiating step” we will have recovered the degree 3 term $\kappa(x)$ in our cubic phase that correlated with $f$.

To recover the lower degree terms, it can be shown that $\|f\omega^{-\kappa(x)}\|_{U^3}$ is large; by implementing the $U^3$ inverse theorem/quadratic Goldreich-Levin algorithm we can recover $q(x)$ such that $\omega^q(x)$ has large correlation with $f\omega^{-\kappa(x)}$. Putting this together, we get that $r(x) = \kappa(x) + q(x)$ is the desired cubic with large correlation with $f$.

Of note is that the quantitative bounds of find-cubic, namely the dependence of $\eta$ on $\epsilon$ in Theorem 1.3 is dependent on those obtained in the proof of the quantitative $U^4$ inverse theorem. Any improvement in the quantitative bounds in the $U^4$ inverse theorem would therefore have implications on find-cubic as well.

3 General decomposition theorem

Theorem 1.3, the algorithmic $U^4$ inverse theorem, is effectively a result of the form “if bounded $f$ has non-negligible $U^4$ norm then we can retrieve one of its large Fourier coefficient”. Oftentimes in additive combinatorics and also computer science, however, it is fruitful to study the set of all large Fourier coefficients rather than just one of the large Fourier coefficients. In the classical setting, we have the
Goldreich-Levin algorithm which achieves this goal. We will develop an analogue of this in the higher order Fourier analysis setting. This was also a problem studied by Tulsiani and Wolf in [20]. However, as we will see, their decomposition introduces an extra $L^1$ error term. By using the idea of averaging projections more carefully, we are able to remove this error term.

Tulsiani and Wolf proved the following general decomposition result [20, Theorem 3.1].

**Theorem 3.1.** Let $\mathcal{Q}$ be an arbitrary class of functions $\overline{f}: X \to [-1,1]$ that is also closed under negation. Let $\epsilon, \delta > 0$ and $B > 1$. Let $A$ be an algorithm which given oracle access to a function $f: X \to [-B,B]$ satisfying $\|f\|_S \geq \epsilon$ outputs with probability $1 - \delta$ a function $\overline{g} \in \mathcal{Q}$ such that $\langle f, \overline{g} \rangle \geq \eta$ for some $\eta = \eta(\epsilon, B)$. Then there exists an algorithm which given any function $g: X \to [-1,1]$ outputs with probability at least $1 - \delta/\eta^2$ a decomposition

$$g = c_1\overline{g_1} + \cdots + c_k\overline{g_k} + e + f$$

satisfying $k \leq 1/\eta^2$, $\|f\|_S \leq \epsilon$ and $\|e\|_1 \leq 1/2B$. Also, the algorithm makes at most $k$ calls to $A$.

A high level summary of their proof is as follows. At step $t$, we find some $\overline{g}_t$ which has good correlation with $f_t$ via $A$. Naïvely, via a Frank-Wolfe style argument, we would want to set $f_{t+1} = f_t - \langle f_t, \overline{g}_t \rangle \overline{g}_t$. The issue with this that is pointed out in [20] is that $\|f_t\|_\infty$ cannot be controlled, and it can be checked that $\langle f_t, \overline{g}_t \rangle$ degrades as $\|f_t\|_\infty$ increases. To that end we will need to truncate $f_t$ as we iterate so as to have a uniform $\ell_\infty$ bound. This truncation introduces an error term $e$, which they control for with its $\ell_1$ norm.

Our goal is to remove the $\ell_1$ error term to get an analogue of a kind of Frieze-Kannan weak regularity theorem [6] for functions. Our key insight is opting for using an averaging projections argument instead of going for the Frank-Wolfe style argument in [20]. To motivate our idea, we home in on the context of $U^3$ decomposition theorems for concreteness and ease of exposition. In other words, in the setting of Theorem 3.1, we take $\mathcal{Q}$ to be the class of quadratic phases and $\|\cdot\|_S$ to be the $U^3$ norm. First, we set up some notation. As in [10], recall that a quadratic factor is defined as follows.

**Definition 3.2.** Let $r_1, \ldots, r_d \in \mathbb{F}_p^n$ be vectors and let $M_1, \ldots, M_d \in \text{Mat}_n(\mathbb{F}_p)$ be symmetric matrices. Write $\mathcal{B}_1$ for the $\sigma$-algebra generated by the functions $r_j^T x$ and $\mathcal{B}_2$ be the $\sigma$-algebra generated by the functions $r_j^T x$ and $x^T M_j x$. We call $(\mathcal{B}_1, \mathcal{B}_2)$ a quadratic factor of complexity $(d_1, d_2)$.

One way to think about (weak) $U^3$ decomposition theorem is via the slogan that every bounded function is the sum of a “structured” function in the sense of being constant on the atoms of $\mathcal{B}_2$ of $(\mathcal{B}_1, \mathcal{B}_2)$ formed by projecting onto this quadratic factor and another “pseudorandom” function with small $U^3$ norm. The projection onto $\mathcal{B}_2$ effectively can be rewritten as a weighted sum of the quadratic phase functions corresponding to the quadratic forms defining it. We iteratively build up the quadratic factor: each time we identify a new quadratic via the algorithmic $U^3$ inverse theorem we add it to our quadratic factor. In this set-up, note that it is possible for the coefficients of the existing quadratic phases to change after projection upon the addition of the new quadratic factor. In [20], this is not accounted for; the coefficients are instead fixed and Tulsiani-Wolf studies $f - \sum_{i=1}^{k} c_i \omega^{\sigma_i}$. As such, they can control $\|\cdot\|_2$ but lose control of $\|\cdot\|_\infty$. By considering averaging projections, we can instead control for both of these norms at the same time, removing the need to do any form of truncation and therefore we will not introduce the error term $e$.

We begin by specifying the class of functions on which our decomposition theorem works, assuming the existence of an appropriate algorithmic inverse theorem. Let $X$ be a finite domain.

**Definition 3.3.** We say that a class $\mathcal{Q}$ of functions $\overline{f}: X \to [-1,1]$ is random samplable if for any $f: X \to [-1,1]$ and $g \in \mathcal{Q}$ there exists an algorithm that runs in time $O(\text{poly}(\epsilon, \log(\delta^{-1})))$ and gives query access to $h$ such that with probability at least $1 - \delta$ we have $\|\langle f, g \rangle - h\|_\infty \leq \epsilon$.

In other words, a random samplable $\mathcal{Q}$ allow us to approximate projections onto $\mathcal{Q}$ arbitrarily well. As an example, in the context of $U^k$ norms we have that the class of polynomial phases $\omega^{f(x)}$ where $\omega$ is a $p$th root of unity and $f(x)$ is a degree $k$ polynomial in $x$. 
Theorem 3.4. Let $\mathcal{Q}$ be a random samplable class of functions. Let $A$ be an algorithm which given query access to a function $f : X \rightarrow [-B, B]$ satisfying $\|f\|_{U^S} \geq \epsilon$ outputs with probability $1 - \delta$ a function $\overline{f} \in \mathcal{Q}$ such that $(f, \overline{f}) \geq \eta$ for some $\eta = \eta(\epsilon, B)$ and if $\|f\|_{U^S} < \epsilon$ outputs with probability $1 - \delta$ the symbol $\perp$. Then there exists an algorithm which given any function $f : X \rightarrow [-1, 1]$ outputs with probability at least $1 - 2\delta/\eta^2$ a decomposition

$$f = c_1\overline{f} + \cdots + c_r\overline{f} + g$$

satisfying $r \leq 10/9 \cdot \eta^{-2}$ and $\|g\|_S \leq \epsilon$. The algorithm makes at most $r$ calls to $A$.

Note here that $10/9$ is arbitrary and chosen for concreteness. We can replace $10/9$ by $1 + \epsilon$ for any $\epsilon > 0$.

Weak-regularity($f$):

- Initialize $g = f$, $f_{\text{struc}} = 0$ and $\mathcal{L} = \emptyset$. We use $\mathcal{L}$ to store the elements of $\mathcal{Q}$ that we identify.
- Run $A$ on $g$. If $\|g\|_{U^k} < \epsilon$, then return $f = g + f_{\text{struc}}$.
- Otherwise, suppose the output is $q$. Add $q$ to $\mathcal{L}$.
- Run Gram-Schmidt on $\mathcal{L}$ and let the output be $\mathcal{L'}$. Note that Gram-Schmidt also outputs the coefficients $\alpha_q = \mathcal{L}_b$ for $q \in \mathcal{L}, b \in \mathcal{L'}$. Now, since $\beta_b = \langle q, b \rangle$ for $b \in \mathcal{L'}$ is an inner product we can estimate it.
- Update $f_{\text{struc}} = \sum_{q \in \mathcal{L}} (\sum_{b \in \mathcal{L'}} \alpha_q \beta_b) \omega^a$ and $g = f - f_{\text{struc}}$. Now repeat from step 2.

Proof. The key of this argument is that we update $g$ at every iteration so there will be no cascading errors that cause us to lose control over $\|g\|_\infty$.

Consider the following modified form of Weak-regularity.

Weak-regularity'($f$):

- Initialize $g = f$, $f_{\text{struc}} = 0$ and $\mathcal{L} = \emptyset$. We use $\mathcal{L}$ to store the elements of $\mathcal{Q}$ that we identify.
- If $|\mathcal{L}| = \eta^{-2}$, return $f = g + f_{\text{struc}}$. Else, continue with Weak-regularity.

Note that it is equivalent to prove that upon termination of Weak-regularity', we have that $\|g\|_S \leq \epsilon$. Henceforth we will work exclusively with Weak-regularity' and drop the prime for notational convenience.

We apply approx-iprod$(\eta^3/50, \delta \eta^2/6)$ so that with probability at least $1 - \delta/(6r)$ we have that $\|\langle f, b \rangle - \beta_b\|_\infty \leq \eta/(50 \epsilon)$. In particular, since there are at most $r$ terms $\beta_b$ when we calculate $f_{\text{struc}}$, it follows that we have $\|f_{\text{struc}} - \mathbb{E}(f | B_2)\|_\infty \leq \eta/50$.

Let us begin by showing that $g$ is always bounded by $3$, which would then justify the application of the algorithmic inverse $U^k$ theorem $A$. Indeed, it suffices to note that $\|f - f_{\text{struc}}\|_\infty \leq \|f - \mathbb{E}(f | B_2)\|_\infty + \eta/50 \leq 3$.

We will use an energy increment method; specifically we will adapt the proof of Lemma 3.8 in [10] to get a “noisy” version.

Lemma 3.5. Let $\mathcal{B}$ be the $\sigma$-algebra corresponding to the elements of $\mathcal{L}$ at a certain stage of Weak-regularity and suppose

$$\|f - f_{\text{struc}}\|_S \geq \epsilon,$$

where $f_{\text{struc}}$ is as defined in Weak-regularity.

Then in the next stage Weak-regularity extends $\mathcal{L}$ by an element to $\mathcal{L}_1$ with corresponding $\sigma$-algebra $\mathcal{B}_1$ such that

$$\|\mathbb{E}(f | \mathcal{B}_1)\|_2^2 \geq \|\mathbb{E}(f | \mathcal{B})\|_2^2 + 9\eta/10.$$
Proof. Note that applying algorithm $A$ some $q \in Q$ such that $\eta \leq \langle \tilde{g}, q \rangle$. In particular, Weak-regularity forms $\mathcal{L}_1$ by adding $q$ to $\mathcal{L}$. Let the $\sigma$-algebra formed by $q$ be $\mathcal{B}_q$. Observing that

$$
\mathbb{E}_x \tilde{g}(x)q(x) = \mathbb{E}_x \mathbb{E}(\tilde{g} \mid \mathcal{B}_q)(x)q(x),
$$

it follows that $\|\mathbb{E}(\tilde{g} \mid \mathcal{B}_q)\|_1 \geq \eta$.

Let us note the following bound.

$$
\|g \mid B_1\|_2^2 - \|\tilde{g} \mid B_1\|_2^2 \leq \|\mathbb{E}(g - \tilde{g} \mid B_1)\|_2 (\|\mathbb{E}(g \mid B_1)\|_2 + \|\mathbb{E}(\tilde{g} \mid B_1)\|_2) \\
\leq 5 \|\mathbb{E}(g - \tilde{g} \mid B_1)\|_2 \\
\leq 5 \|g - \tilde{g}\|_2 \\
\leq 5 \|g - \tilde{g}\|_\infty \\
\leq \eta/10
$$

Now we are in a position to establish the energy increment, via Pythagoras’ Theorem. Note that Pythagoras’ tells us that

$$
\|\mathbb{E}(f \mid B_1)\|_2^2 = \|\mathbb{E}(f \mid B)\|_2^2 + \|\mathbb{E}(f \mid B_1) - \mathbb{E}(f \mid B)\|_2^2.
$$

This rearranges as

$$
\|\mathbb{E}(f \mid B_1)\|_2^2 - \|\mathbb{E}(f \mid B)\|_2^2 = \|\mathbb{E}(f \mid B_1) - \mathbb{E}(f \mid B)\|_2^2 \\
= \|\mathbb{E}(g \mid B_1)\|_2^2 \\
\geq \|\mathbb{E}(g \mid B_1)\|_2^2 - \|g \mid B_1\|_2^2 - \|\tilde{g} \mid B_1\|_2^2 \\
= 9\eta/10,
$$

as desired. \qed

The lemma effectively implies that the energy $\|\mathbb{E}(f \mid B)\|_2^2$ is a monovariant. If $\|f - f_{\text{struc}}\|_S \leq \epsilon$ then Weak-regularity would have terminated. Otherwise Lemma 9.3 allows us to extend $\mathcal{L}$ with a corresponding increment in energy by $9\eta/10$. Since $f$ is bounded, the energy $\|\mathbb{E}(f \mid B)\|_2^2$ lies in the interval $[0, 1]$. This means that the algorithm has to terminate in at most $10\eta/9$ steps, as desired.

It remains to justify that the application of Gram-Schmidt, which itself runs in time $O(r^3)$; particularly, we need to demonstrate that we do not have redundant vectors with high probability. To that end, let the corresponding $\sigma$-algebra at the $r$th step of the algorithm be $\mathcal{B}_r$. Recall that algorithm $A$ gave $q \in Q$ such that $\eta \leq \langle \tilde{g}, q \rangle$. Write $g \coloneqq f - \mathbb{E}(f \mid B)$. By Hölder’s we also have that

$$
\|g - \tilde{g}\|_\infty \|q\|_1 \leq \eta/50.
$$

Consequently, it follows that $\langle g, q \rangle = \langle \tilde{g}, q \rangle + \langle g - \tilde{g}, q \rangle \leq 2\eta$. Expanding this further, we see that

$$
0 < 2\eta \leq \langle f - \mathbb{E}(f \mid B_{t-1}), q \rangle \\
\leq \langle f - \mathbb{E}(f \mid B_{t-1}), q - \mathbb{E}(q \mid B_{t-1}) \rangle \\
\leq \|f - \mathbb{E}(f \mid B_{t-1})\|_2 \|q - \mathbb{E}(q \mid B_{t-1})\|_2 \\
\leq 2 \|q - \mathbb{E}(q \mid B_{t-1})\|_2.
$$

This implies that $q$ is not measurable on $B_{t-1}$ and therefore in particular this means that $q$ is not a redundant vector which lies in the span of the existing vectors in $\mathcal{L}$, as desired. \qed

Note that we can sample to estimate the correlation between a function and a polynomial phase functions. It follows that we can take $Q$ to be the class of cubic phase functions. Recalling our algorithmic $U^4$ inverse theorem (Theorem 1.3), we obtain as a corollary the cubic Goldreich-Levin algorithm of Theorem 1.5.
Corollary 3.6 (Restatement of Theorem 1.5). Let $\epsilon, \delta > 0$ and $B > 1$. Let \textbf{find-cubic} be an algorithm which given query access to a function $f : \mathbb{F}_p^n \to [-B, B]$ satisfying $\|f\|_{U_4} \geq \epsilon$ outputs with probability $1 - \delta$ a cubic polynomial phase function $\bar{q}$ such that $|\langle f, \bar{q} \rangle| \geq \eta$ for some $\eta = \eta(\epsilon, B)$. Then there is an algorithm $U_4$-\textbf{weak-regularity} such that given any function $g : \mathbb{F}_p^n \to [-1, 1]$ outputs with probability at least $1 - 2\delta/\eta^2$ a decomposition

$$f = c_1 \bar{q} + \cdots + c_r \bar{q} + g$$

satisfying $r \leq 2/\eta^2$ and $\|g\|_{U_4} \leq \epsilon$. The algorithm makes at most $r$ calls to \textbf{find-cubic}.

4 Algorithmic tools

In this section, we enumerate some algorithmic primitives that we will be utilizing in later sections.

Chernoff tail bounds

Lemma 4.1. If $X$ is a random variable with $|X| \leq 1$ and $\mu_t = \frac{X_1 + \cdots + X_t}{t}$ where $X_i$ are samples, then

$$\mathbb{P}[|\mathbb{E}[X] - \mu_t| \geq \eta] \leq 2 \exp(-2\eta^2 t).$$

Classic Goldreich-Levin algorithm

Given query access to $f : \mathbb{F}_2^n \to \mathbb{F}_2$ and input $0 < \tau \leq 1$ there exists a poly$(n, 1/\tau)$-time algorithm \textbf{Goldreich-Levin}(f, $\tau$) that with high probability outputs a list $L = \{r_1, \ldots, r_k\}$ with the following guarantee:

- If $|\hat{f}(r)| \geq \tau$ then $r \in L$.
- For $r_i \in L$, we have $|\hat{f}(r_i)| \geq \tau/2$.

Noisy Goldreich-Levin

Let $f : \mathbb{F}_p^n \to \mathbb{C}$ be a 1-bounded function, that is $\|f\|_\infty \leq 1$. Given query access to a random function $f' : \mathbb{F}_p^n \to \mathbb{C}$ such that with probability $1 - \eta$ we have $|f'(x) - f(x)| \leq \omega$ then there is a randomized algorithm \textbf{fuzzy-GL} that makes $O(poly(n, 1/\tau, 1/\eta, 1/\omega, log(1/\delta)))$ queries to $f'$ and with probability $1 - \delta$ outputs a list $L = \{r_1, \ldots, r_k\}$ with the following guarantee:

- If $|\hat{f}(r)| \geq \tau$ then $r \in L$.
- For $r_i \in L$, we have $|\hat{f}(r_i)| \geq \frac{\tau}{2} - \frac{3}{2} \cdot (\eta + (1 - \eta) \cdot \omega)$.

Proof. The proof is standard. We will implement the divide-and-conquer strategy as in the classic Goldreich-Levin algorithm, with the key observation being that $\mathbb{F}_p^n$ has many subspaces.

The additional observation here is that even though we are working with a noisy query $f'$, where $f$ has a heavy Fourier coefficient $f'$ does as well. More precisely, write $f' = f + e$. Then we know that $\|e\|_\infty \leq 1$ and also that for at least $1 - \eta$ fraction of $r \in \mathbb{F}_p^n$ we have that $|e(r)| \leq \omega$. Let the set of $r$ satisfying the latter condition be $S$. Let us expand $\hat{f'} = \hat{f} + \hat{e}$ as follows

$$|\hat{f'}(x)| \geq |\hat{f}(x)| - |\mathbb{E}_r e(x)\omega^{-r}|$$

$$\geq |\hat{f}(x)| - \mathbb{E}_{r \in S}|e(r)| - \mathbb{E}_{r \notin S}|e(r)|$$

$$\geq |\hat{f}(x)| - \eta - (1 - \eta) \cdot \omega.$$

Particularly, if $|\hat{f}(x)| \geq \tau$ then $|\hat{f'}(x)| \geq \tau - \eta - (1 - \eta) \cdot \omega$. Similarly, if $|f(x)| < \tau$ then $|f'(x)| < \tau + \eta + (1 - \eta) \cdot \omega$.  

12
The high-level picture is that we iteratively split the coefficients \( A \) we are working with at the current stage of the algorithm into \( p \) buckets \( A = B_1 \sqcup B_2 \sqcup \cdots \sqcup B_p \). Define \( f_1^{(A)}(x) = \sum_{r \in B_1} \hat{f}(r) \omega^{r \cdot x} \). Recall by Parseval’s that we have the following expression
\[
\left\| f_1^{(A)} \right\|_2^2 = \sum_{r \in B_1} |\hat{f}(r)|^2.
\]
Because of our eventual choices of \( B_i \) as subspaces as well as the fact that the Fourier transform is defined as an expected value, we are able to sample and approximate \( \left\| f_1^{(A)} \right\|_2^2 \). Combining with our earlier observation, if our approximation of the norm is smaller than \( \frac{3}{4} \left( \tau - \eta - (1 - \eta) \cdot \omega \right)^2 \) then we will be able to conclude with high probability that there does not exist \( r \in B_i \) such that \( \hat{f}(r) \geq \tau \). Discard \( B_i \) which have small corresponding norms. This process allows us to refine our search and home in on the large Fourier coefficients. Now iterate the algorithm by partitioning up the remaining “alive” (i.e. buckets which have not been discarded) \( B_i \) until we get down to singleton sets.

In more detail, choose the buckets as follows. The buckets are indexed by two values \( a \in [n] \) and \( b \in \mathbb{F}_p^a \). For \( \vec{b} \in \mathbb{F}_p^a \) write \( \vec{b}_{a \cdot} \) for the vector obtained by truncating the first element of \( b \) and let the \( j \)th element of \( b \) be \( b_j \). Let the standard basis vectors for \( \mathbb{F}_p^a \) be \( e_1, \ldots, e_n \). The bucket \( B_{a,\vec{b}} \) is recursively defined as
\[
B_{a,\vec{b}} = \{(e_a)_{-1} \cap B_{(a-1),\vec{b}_{a\cdot-1}} + \vec{b}_1 e_a\}
\]
where if \( a = 0 \) the bucket corresponds to \( \mathbb{F}_p^a \). The initial buckets are \( B_{1,0}, \ldots, B_{1,p-1} \). The algorithm always splits a bucket \( B_{i,\vec{b}} \) into \( B_{i+1,(0,\vec{b})}, \ldots, B_{i+1,(p-1,\vec{b})} \).

Define \( f_{a,\vec{b}}(x) = \sum_{r \in B_{a,\vec{b}}} \hat{f}(r) \omega^{r \cdot x} \). Assume for the moment that we are able to estimate \( \left\| f_{a,\vec{b}} \right\|_2^2 \) to within an additive error of \( \pm \frac{1}{4} \left( \tau - \eta - (1 - \eta) \cdot \omega \right)^2 \). For all the “alive” buckets, approximate \( \left\| f_{a,\vec{b}} \right\|_2^2 \) and discard them if the value is smaller than \( \frac{3}{4} \left( \tau - \eta - (1 - \eta) \cdot \omega \right)^2 \). By our earlier computations:

- If there exists \( r \in B_{a,\vec{b}} \) such that \( \hat{f}(r) \geq \tau \) then it follows that \( \hat{f}(r) \geq \tau - \eta - (1 - \eta) \cdot \omega \) and so
  \[
  \left\| f_{a,\vec{b}} \right\|_2^2 \geq (\tau - \eta - (1 - \eta) \cdot \omega)^2.
  \]
  Given our precision, it follows we would not throw away any bucket that contains this \( r \).

- If there is \( r \) such that \( \hat{f}(r) < \frac{3}{4} \left( \tau - \eta - (1 - \eta) \cdot \omega \right) \) then it follows that \( \hat{f}(r) < \frac{1}{4} (\tau - \eta - (1 - \eta) \cdot \omega) \).

The singleton bucket \( B_{a,\vec{b}} \) that contains \( r \) has a corresponding
\[
\left\| f_{a,\vec{b}} \right\|_2^2 < \frac{1}{4} (\tau - \eta - (1 - \eta) \cdot \omega)^2.
\]
Given our precision this is strictly less than \( \frac{1}{4} (\tau - \eta - (1 - \eta) \cdot \omega)^2 \) if we had not already discarded the bucket corresponding to \( r \) we would have discarded it when we reduced down to singleton buckets.

Now, we expound on how we intend to estimate the 2-norm of \( f_{a,\vec{b}}(x) = \sum_{r \in B_{a,\vec{b}}} \hat{f}(r) \omega^{r \cdot x} = \sum_{r \in \mathbb{F}_p^a} 1_{B_{a,\vec{b}}}(x) \hat{f}(r) \omega^{r \cdot x} \). The first step is to write \( f_{a,\vec{b}} \) as a convolution. Consider \( u_{a,\vec{b}}(x) = \sum_{r \in B_{a,\vec{b}}} \omega^{r \cdot x} \). We will show that \( f_{a,\vec{b}} = f * u_{a,\vec{b}} \). Start by observing that
\[
\hat{u}_{a,\vec{b}}(y) = \mathbb{E}_x \sum_{r \in B_{a,\vec{b}}} \omega^{r \cdot x} \omega^{-y \cdot x} = \sum_{r \in B_{a,\vec{b}}} \mathbb{E}_x \omega^{(r-y) \cdot x} = 1_{B_{a,\vec{b}}}(y).
\]
Consequently, we have that
\[
g \ast \hat{u}_{a,\vec{b}}(r) = g(r) u_{a,\vec{b}}(r) = \hat{g}(r) 1_{B_{a,\vec{b}}}(y).
\]
This immediately implies that \( f_{a,\hat{b}} = f' \ast u_{a,\hat{b}} \).

Because convolution is defined as an expected value, we are in a slightly better shape to estimate \( f_{a,\bar{b}} \). To that end, we next describe how to calculate \( u_{a,\hat{b}} \). First, make the observation that
\[
B_{a,\hat{b}} = \langle e_1, \ldots, e_a \rangle^\perp + v_{\bar{b}} \text{ where } v_{\bar{b}} = \sum_{i=1}^a \bar{b}_i e_i.
\]
For simplicity write \( U_a = \langle e_1, \ldots, e_a \rangle \). Observe that
\[
\sum_{r \in U_a} \omega^{x-r} = 1_{U^\perp_a}(x)|U_a|.
\]
This in turn implies that
\[
u_{a,\hat{b}}(x) = \sum_{r \in U_a} \omega^{x-r+v_{\bar{b}}}
= \omega^{x-v_{\bar{b}}} \sum_{r \in U_a} \omega^{x-r}
= 1_{U^\perp_a}(x)|U_a|\omega^{x-v_{\bar{b}}}.
\]
Combining all the pieces that we have so far, and recalling that \( |U_a||U^\perp_a| = p^n \) we can write
\[
\left\| f_{a,\hat{b}} \right\|^2 = \mathbb{E}_{x \in F^n_p} [f' \ast u_{a,\hat{b}}(x)]^2
= \mathbb{E}_{x \in F^n_p} \left[ \mathbb{E}_{y \in F^n_p} [f'(x-y)u_{a,\hat{b}}(y)] \right]^2
= \mathbb{E}_{x \in F^n_p} \left[ \mathbb{E}_{y \in F^n_p} \left[ f'(x-y)|U^\perp_a| \omega^{y-v_{\bar{b}}} \right] \right]^2
= \mathbb{E}_{x \in F^n_p} \left[ \sum_{y \in U^\perp_a} f'(x-y) \omega^{y-v_{\bar{b}}} \right]^2
= \mathbb{E}_{x \in F^n_p} \left[ \mathbb{E}_{y \in U^\perp_a} [f'(x-y)\omega^{y-v_{\bar{b}}}] \right]^2.
\]
In this form, it becomes clear that we are able to sample to approximate \( \left\| f_{a,\hat{b}} \right\|^2 \). This allows us to perform run-time calculations using the Chernoff bound as described in Lemma 4.1. Since we have \( |f'(x-y)\omega^{y-v_{\bar{b}}}| \leq 1 \) by Lemma 4.1 for fixed \( x \) we can estimate \( \mathbb{E}_{y \in U^\perp_a} [f'(x-y)\omega^{y-v_{\bar{b}}}] \) to within an additive error of \( \frac{1}{\sqrt{n}} \cdot \tau - \eta - (1 - \eta) \cdot \omega \) with confidence \( 1 - \delta \) via at most \( O(\text{poly}(1/\tau, 1/\eta, 1/\omega) \cdot \log(1/\delta)) \) samples. Once more this time unfixing \( x \) by Lemma 4.1 it follows that we can estimate \( \left\| f_{a,\hat{b}} \right\|^2 \) to within an additive error of \( \frac{1}{2} \cdot (\tau - \eta - (1 - \eta) \cdot \omega)^2 \) with confidence \( 1 - \delta \) using \( O(\text{poly}(1/\tau, 1/\eta, 1/\omega) \cdot \log(1/\delta)) \) samples. Any “alive” bucket has 2-norm at least \( (\tau - \eta - (1 - \eta) \cdot \omega)^2 \) so by Parseval’s there can be at most \( (\tau - \eta - (1 - \eta) \cdot \omega)^{-2} \) “alive” buckets. Each bucket will be split at most \( n \) times, and finding the corresponding 2-norm for each bucket takes at most time \( O(\text{poly}(1/\tau, 1/\eta, 1/\omega) \cdot \log(1/\delta)) \) as we have already discussed. Combining all the estimates, it follows that the overall running time is \( O(\text{poly}(n, 1/\tau, 1/\eta, 1/\omega, \log(1/\delta))) \) as claimed.

**Classic Goldreich-Levin algorithm for \( F^n_p \)**

Given query access to \( f : F^n_p \rightarrow [-1, 1] \) and input \( 0 < \tau \leq 1 \) there exists a \( \text{poly}(n, 1/\tau) \)-time algorithm \( \text{Goldreich-Levin}(f, \tau) \) that with high probability outputs a list \( L = \{r_1, \ldots, r_k\} \) with the following guarantee:

- If \( |\hat{f}(r)| \geq \tau \) then \( r \in L \).
- For \( r_i \in L \), we have \( |\hat{f}(r_i)| \geq \tau/2 \).

**Proof.** This is a consequence of the proof for the noisy Goldreich-Levin.

**Algorithmic \( U^3 \) inverse theorem for \( F^n_2 \) (20, Theorem 4.1)**

Given \( \epsilon, \delta > 0 \) there exists \( \eta = \exp(-1/\epsilon^C) \) and a randomized algorithm \( \text{Find-Quadratic running in time } O(n^4 \log n \cdot \text{poly}(1/\epsilon, 1/\eta, \log(1/\delta)) \) which given query access to \( f : F^n_2 \rightarrow \{-1, 1\} \) either outputs a quadratic form or \( \perp \). The algorithm has the following guarantee:
• If \( \|f\|_{U^3} \geq \epsilon \) then with probability at least \( 1 - \delta \) it finds a quadratic form such that \( \langle f, (-1)^q \rangle \geq \eta \).

• The probability that the algorithm outputs a quadratic form \( q \) with \( \langle f, (-1)^q \rangle \leq \eta/2 \) is at most \( \delta \).

**Algorithmic \( U^3 \) inverse theorem for \( \mathbb{F}_p^n \)**

Using similar techniques as in [20], for any prime \( p \) we obtain an Algorithmic \( U^3 \) inverse theorem for \( \mathbb{F}_p^n \).

**Theorem 4.2** (Algorithmic \( U^3 \) inverse theorem). Given \( \epsilon, \delta > 0 \) there exists \( \eta = \exp(-1/\epsilon^C) \) and a randomized algorithm \textbf{Find-Quadratic} running in time \( O(n^4 \log n \cdot \text{poly}(1/\epsilon, 1/\eta, \log(1/\delta))) \) which given query access to \( f : \mathbb{F}_p^n \to \mathbb{C} \) that is \( 1 \)-bounded either outputs a quadratic form or \( \perp \). The algorithm has the following guarantee:

• If \( \|f\|_{U^3} \geq \epsilon \) then with probability at least \( 1 - \delta \) it finds a quadratic form such that \( \langle f, \omega^q \rangle \geq \eta \).

• The probability that the algorithm outputs a quadratic form \( q \) with \( \langle f, \omega^q \rangle \leq \eta/2 \) is at most \( \delta \).

**BSG-test**

The flow for \textbf{BSG-test} here mirrors closely that from [20], where the key idea is that the utilization of dependent random choice in the proof of Balog-Szemerédi-Gowers easily lends itself to a sampling argument.

We build a (random) bipartite graph \( G \) with vertices \( A \cup A \) (call one copy \( A^{(1)} \) and another \( A^{(2)} \)) and edge set \( E_\gamma \) for \( \gamma > 0 \) defined as

\[
E_\gamma := \{(a_1, a_2) : |\{(a, b) \in A \times A : a + b = a_1 + a_2\}| \geq (\rho/2 + \gamma) \cdot |A|\}.
\]

It can be shown that the edge density of \( G \) is at least \( \rho/2 - \gamma \).

**Claim 4.3.** If \( E(A, A) \geq \rho |A|^3 \) where \( E(A, A) \) is the additive energy of \( A \), then the density of \( (a_1, a_2) \in A \times A \) such that the number of \( \{(a, b) \in A \times A : a + b = a_1 + a_2\} \) is at least \( (\rho/2 + \gamma)|A| \) is at least \( \rho/2 - \gamma \).

**Proof.** For each \( x \in A + A \), let \( r_x \) be the number of \( (a, b) \in A \times A \) such that \( a + b = x \). Define a set \( S \) be the elements of \( A + A \) such that \( r_x \geq (\rho/2 + \gamma)|A| \). Then we have

\[
\rho |A|^3 \leq E(A, A) = \sum_{x \in S} r_x^2 + \sum_{x \notin S} r_x^2.
\]

Since \( r_x \leq (\rho/2 + \gamma)|A| \), we have that \( \sum_{x \in S} r_x^2 \leq (\rho/2 + \gamma)|A| \sum_{x \notin S} r_x \leq (\rho/2 + \gamma)|A|^3 \). Therefore,

\[
\sum_{x \in S} r_x^2 \geq \frac{\rho - \gamma}{2} |A|^3.
\]

For each \( x \in A + A \), \( r_x \leq |A| \), so

\[
\sum_{x \in S} r_x \geq \frac{1}{|A|} \sum_{x \in S} r_x^2 \geq \frac{\rho - \gamma}{2} |A|^2.
\]

Hence the density of \( (a_1, a_2) \in A \times A \) such that the number of \( \{(a, b) \in A \times A : a + b = a_1 + a_2\} \) is at least \( (\rho/2 + \gamma)|A| \) is at least \( \rho/2 - \gamma \). \( \square \)

We first establish a test for if an edge is present in \( G \).

\textbf{Edge-test}\( (a, b) \):\n
• Sample \( t \) elements of \( A \) say \( a_1, \ldots, a_t \).

• Answer 1 if for at least \( (\rho/2)t \) indices we have that \( a + b - a_i \in A \) and 0 otherwise.

As a direct consequence of Lemma 4.1 we have the following aq guarantee for \textbf{Edge-test}. 

15
Claim 4.4. Given \( \delta, \gamma > 0 \), the output of \( \text{Edge-test}(a, b) \) with \( t = O(\gamma^{-2} \rho^{-2} \cdot \log(\delta^{-1})) \) queries satisfies the following guarantee with probability at least \( 1 - \delta \):

- If \( \text{Edge-test}(a, b) \) outputs 1 then \( (a, b) \in E_{-\gamma} \).
- If \( \text{Edge-test}(a, b) \) outputs 0 then \( (a, b) \not\in E_{\gamma} \).

Let \( \eta = \rho/2 \). For a random element \( a \in A^{(2)} \) define the following sets:

- \( N_{\gamma}(a) := \{ b : (a, b) \in E_{\gamma} \} \), here implicitly \( N_{\gamma}(a) \subset A^{(1)} \).
- \( N_{\gamma}(b) := \{ c : (b, c) \in E_{-\gamma} \} \) for each \( b \in A^{(1)} \), here \( N_{\gamma}(b) \subset A^{(2)} \).
- \( M_{\gamma,\eta}(a) := \{ b \in N_{\gamma}(a) : \mathbb{P}_{c \in A^{(2)}}[c \in N_{\gamma}(b)] \geq \eta \} \).
- \( G_{\gamma,\gamma,\gamma,\eta_1,\eta_2,\eta_3,\eta_4}(a) := \{ b \in M_{\gamma,\eta_1}(a) : \mathbb{P}_{c \in M_{\gamma,\eta_2}}[d \in N_{\gamma_3}(b) \cap N_{\gamma_4}(c)] \leq \eta_3 \leq \eta_4 \} \).

Tracing through the proof of Balog-Szemeredi-Gowers, we have the following.

Lemma 4.5. Let the graph with edge set \( E_\gamma \) have density at least \( \rho_\gamma \) and consider \( A' = G_{\gamma,\gamma,\gamma,\rho_\gamma,2^2\rho_\gamma,2^{10},\rho_\gamma,5}(u) \) for a uniformly random vertex \( u \in A^{(2)} \). Then with probability at least \( 3\rho_\gamma/4 \) the set \( A' \) satisfies both:

- \( |A'| \geq \rho_\gamma^2 |A|/16 \), and
- \( |A' + A'| \leq (1/\rho_\gamma)^{O(1)} |A| \).

What this means that with positive probability, by passing to \( G_{\gamma,\gamma,\gamma,\rho_\gamma,\rho,\rho,\rho}(u) \) for a randomly selected \( u \in A \) we would obtain a set with desired small doubling. This motivates us to give an approximate test to determine if \( b \in G_{\gamma,\gamma,\gamma,\rho_\gamma,\rho,\rho}(u) \).

\[ \text{BSG-test}(a, b, c_1, c_2, c_3, c_4, \gamma_1, \gamma_2, \gamma_3, \eta_1, \eta_2, \eta_3, \eta_4) : \]

- If \( \text{Edge-test}(a, b, c_1) = 0 \), return 0.
- Sample \( c_1, \ldots, c_m \) from \( A \). Compute \( T = m^{-1} \sum_{i=1}^m \text{Edge-test}(b, c_i, c_1) \) and if \( T \leq \eta_1 \) return 0.
- Sample \( a_1, \ldots, a_r \) from \( A \). For each \( i \in [r] \), only retain those \( i \) for which \( \text{Edge-test}(a_i, a_1, c_1) \) returns 1.
- Of the remaining samples \( a_1', \ldots, a_s' \), for each \( i \in [s] \) further sample \( b_1^{(i)}, \ldots, b_t^{(i)} \) as well as \( c_1^{(i)}, \ldots, c_u^{(i)} \) from \( A \).
- Compute:
  - \( X_{ij} = \text{Edge-test}(a_i, b_j^{(i)}, c_1^{(i)}) \)
  - \( Y_{ij} = \text{Edge-test}(a_i, c_j^{(i)}, c_1^{(i)}) \)
  - \( Z_{ij} = \text{Edge-test}(b_j^{(i)}, c_j^{(i)}, c_1^{(i)}) \)
- Let \( B_i \) be 1 if \( t^{-1} \sum_{j=1}^t X_{ij} \geq \eta_2/2 \) and 0 otherwise.
- Let \( C_i \) be 1 if \( u^{-1} \sum_{j=1}^u Y_{ij} Z_{ij} \leq \eta_3^3/20 \) and 0 otherwise.
- Answer 1 if \( s^{-1} \sum_{i=1}^s B_i C_i \leq \eta_4/5 \) and 0 otherwise.

Theorem 4.6. Let \( \rho, \delta > 0 \). Let \( A \) be a subset of a finite abelian group for which we have query access as well as the ability to sample a random element. Suppose \( E_+(A) \geq \rho |A|^{5} \). Then there exist sets \( A^{(1)} \subset A^{(2)} \) such that the output of \( \text{BSG-Test} \) satisfies the following with probability at least \( 1 - \delta \):

- \( \text{BSG-Test}(u, v, \rho, \delta) = 1 \) then \( v \in A^{(2)} \).
• BSG-Test \((u, v, \rho, \delta) = 0\) then \(v \notin A^{(1)}(u)\).

Moreover, if \(u\) is chosen uniformly random from \(u \in A\), then with probability at least \(\poly(\rho) \cdot |A|\) we have that:

- \(|A^{(1)}(u)| \geq \poly(\rho) \cdot |A|\),
- \(|A^{(2)}(u) + A^{(2)}(u)| \leq \poly(\rho^{-1}) \cdot |A|\).

Proof. First, by recalling \(E_{\gamma_1} \subset E_{\gamma_2}\) for \(\gamma_2 \leq \gamma_1\) we can check that

\[
A^{(1)}(u) := G_{\gamma_1, \gamma_2, \gamma_3, \gamma_4}(u) \subset \bigcap_{i=1}^{4} G_{\gamma_i}(u) =: A^{(2)}(u).
\]

Choose the parameters \(m, t, u\) such that the additive error in all the estimates in BSG-test is at most \(\rho^3/3200\) with probability at least \(1 - \delta\). Specifically, by Lemma 4.1, we take \(m, t, u\) to be poly\((\rho^{-1}, \log(\delta^{-1}))\). We can also take \(r = \poly(\rho^{-1}, \log(\delta^{-1}))\) with suitable hidden constants such that at least \(s = \poly(\rho^{-1}, \log(\delta^{-1}))\) samples are retained. This means we can take \(\eta_0' = \eta_2' = \eta_3' = \eta_4' = \rho^3/1600\). To choose \(\gamma\), divide the interval \([-\rho/2, \rho/2]\) into \(1600/\rho^2\) of length \(\rho^3/1600\) each. Choose \(\gamma\) such that the selected interval is given by \([\gamma - \gamma', \gamma + \gamma']\). Then set \(\gamma = \gamma_1\) and \(\gamma' = \gamma_i\) for \(i \in \{3\}\).

We also take \(\eta_1, \eta_2 = \rho/4, \eta_3, \eta_4 = \rho^3/160\) and \(\eta_4 = \rho/10\).

Tracing the proof of Balog-Szemerédi-Gowers as presented in [20, Proof of Theorem 7.86], we see that if \(|A^{(2)}(u)| \geq \rho^{O(1)}\eta\) then \(|A^{(2)}(u) + A^{(2)}(u)| \leq \rho^{-O(1)}\eta\). To that end it suffices to show that with high probability that \(|A^{(1)}(u)|\) is sufficiently large. Because \(A^{(1)}(u) \subset A^{(2)}(u)\) by our observation at the beginning of the proof, this would imply the desired small doubling of \(A^{(2)}(u)\).

To make the parallel with [20] more explicit, write

\[
B_{\gamma_1, \gamma_2, \gamma_3, \gamma_4}(u) := \{b \in M_{\gamma_1, \gamma_2, \gamma_3, \gamma_4}(u) : \mathbb{P}_{v \in M_{\gamma_2, \gamma_3, \gamma_4}(u)}[d \in N_{\gamma_2}(b) \cap N_{\gamma_3}(c)] \leq \eta_3\},
\]

where \(B_{\gamma_1, \gamma_2, \gamma_3, \gamma_4}(u) = M_{\gamma_1, \gamma_2, \gamma_3, \gamma_4}(u) \setminus M_{\gamma_1, \gamma_2, \gamma_3, \gamma_4}(u)\).

For simplicity of notation, write \(M(u) = M_{\gamma_1, \gamma_2, \gamma_3, \gamma_4}(u)\) and \(B(u) = M(u) \setminus A^{(1)}(u)\).

From the proof of Balog-Szemerédi-Gowers, we have that \(\mathbb{E}_u[M(u)] \geq \rho^2\cdot A/4\). To that end, once we obtain an upper bound on \(\mathbb{E}_u[B(u)]\) we will be able to obtain the desired lower bound on \(|A^{(1)}(u)|\).

To end off, we apply a modified form of [20] Claim 4.11. A pair \((v, v_1)\) is called bad if \(|N_{\gamma}(v) \cap N_{\gamma'}(v)| \leq 9\rho^3/1600\).

Claim 4.7. There exists a choice for the sub-interval \([\gamma_3 - \gamma_3', \gamma_1 + \gamma_1']\) of length \(\rho^3/1600\) in \([-\rho/2, \rho/2]\) such that

\[
\mathbb{E}[\#\{\text{bad pairs } (v, v_1) : v \in M_{\gamma_1, \gamma_2, \gamma_3, \gamma_4}(u) \land v_1 \in M_{\gamma_3, \gamma_4}(u)\}] \leq \rho^3/800 \cdot |A|^2.
\]

Before we begin the proof for this claim, using the observation that \(E_{\gamma} \subset E_{\gamma'}\) for \(\gamma' \leq \gamma\) we can conclude that \(M_{\gamma_1, \gamma_2, \gamma_3, \gamma_4}(u) \subset M_{\gamma_2, \gamma_3, \gamma_4}(u)\) for \(\gamma_2 \leq \gamma_1\) and \(\eta_2 \leq \eta_1\).

Proof. First, decompose

\[
\mathbb{E}_u[\#\{\text{bad pairs } (v, v_1) : v \in M_{\gamma_1, \gamma_2, \gamma_3, \gamma_4}(u) \land v_1 \in M_{\gamma_3, \gamma_4}(u)\}] = \mathbb{E}_u[\#\{\text{bad pairs } (v, v_1) : v \in M_{\gamma_1, \gamma_2, \gamma_3, \gamma_4}(u)\}] + \mathbb{E}_u[\#\{\text{bad pairs } (v, v_1) : v \in M_{\gamma_1, \gamma_2, \gamma_3, \gamma_4}(u) \land v_1 \in M_{\gamma_3, \gamma_4}(u)\}] - \mathbb{E}_u[M_{\gamma_1, \gamma_2, \gamma_3, \gamma_4}(u)].
\]

Of the \(|A^4|\) choices for \((v, v_1)\), if we get a bad pair then each \(u\) is in \(N_{\gamma}(v) \cap N_{\gamma'}(v)\) with probability at most \(\rho^3/1600\). Since \(M_{\gamma_1, \gamma_2, \gamma_3, \gamma_4}(u) \subset M_{\gamma_1}(u)\), it follows that the first summand is at most \(\rho^3/1600 \cdot |A|^2\).

We bound the second summand by

\[
|A| \cdot \mathbb{E}_u[M_{\gamma_1, \gamma_2, \gamma_3, \gamma_4}(u) \setminus M_{\gamma_1, \gamma_2, \gamma_3, \gamma_4}(u)] \leq |A| \cdot \left(\mathbb{E}_u[M_{\gamma_1}(u)] - \frac{\rho^3}{1600} \mathbb{E}_u[M_{\gamma_1}(u)]\right).
\]
Let $f(\gamma_3, \gamma_1) = \mathbb{E}_u |N_{-\gamma_3}| - \xi \mathbb{E}_u |N_{\gamma_1}|$. When $\gamma_3' = \gamma_1' = \rho/2$ we have that $f(\gamma_3, \gamma_1) \leq |A|$ and when $-\gamma_3' = \gamma_1' = \rho/2$ we have that $f(\gamma_3, \gamma_1) = 0$. Since $f(\gamma_3, \gamma_1)$ is monotonically increasing in the first variable and monotonically decreasing in the second variable, it follows that there must be an interval of length $\rho^2/1600$ say $[-\gamma_3', \gamma_1']$ such that $f(\gamma_3, \gamma_1) \leq \rho^2/1600 \cdot |A|$. It suffices to take $[\gamma_3 - \gamma_3', \gamma_1 + \gamma_1']$ to be the endpoints of this interval.

Lastly, note that
\[
\#\{\text{bad pairs } (v, v_1) : v \in M_{\gamma_3, \gamma_1}, v_1 \in M_{-\gamma_3, -\gamma_1}(u)\} \geq |B(u) \cdot \rho|A|/20.
\]
so that the above claim rewrites as $\mathbb{E}_u |B(u)| \leq (\rho^2/40)|A|$. Since there are 1600/$\rho^2$ choices for the interval that we picked, this happens with probability $\rho^2/1600 = \text{poly}(\rho)$. For this choice of parameters, we have that $\mathbb{E}_u |A^{(1)}(u)| \geq (\rho^2/4 - \rho^2/40) \cdot |A| = \text{poly}(\rho) \cdot |A|$, as desired.

\[\square\]

**Find-affine-map**

**Theorem 4.8.** Let $\rho, \delta > 0$. Let $A$ be a subset of $\mathbb{F}_p^n$ for which we have query access as well as the ability to sample a random element via $\text{sampler-A}$. Let $\phi : A \to \mathbb{F}_p^n$ be a function such that there exist $|A|^6$ quadruples $x \cdot y = z \cdot w$ such that $\phi(x) \cdot \phi(y) = \phi(z) \cdot \phi(w)$. Then there exists an algorithm $\text{find-affine-map}$ which makes $O(\text{poly}(n, \rho^{-1}, \log((\delta^{-1}))$ queries to $A$, $\text{sampler-A}$ and $\phi$ such that with probability $1 - \delta$ outputs an affine map $T$ that agrees with $\phi$ on at least a quasi-$\text{poly}(\rho)$ fraction of $A$.

This section builds off [20, Lemma 4.12].

**Proof.** The first step is to pass from the 1% additive structure on $A$ to 100% structure by passing to a suitable subset of $A$ à la Balog-Szemeredi-Gowers. Build the set $A_\phi$, which is the graph of $\phi$; specifically, $A_\phi = \{(x, \phi(x)) : x \in A\} \subset \mathbb{F}_p^{2n}$. Using our sampler $\text{sampler-A}$ for $A$, we are able to obtain a sampler for $A_\phi$ as follows.

\[\text{sampler-graph-A}(t):\]

- Run $\text{sampler-A}(t)$ and let its output be $a_1, \ldots, a_t$.
- Return $\{(a_i, \phi(a_i))\}$.

Set $\delta' = \delta/10$.

Sample a random element $u \in A_\phi$. Note that by Theorem 4.6 with probability $\text{poly}(\rho)$ this choice of $u$ would satisfy the guarantees of that theorem. If $u$ satisfies the guarantees of the theorem, then running the algorithm that we describe below will produce an affine map $T_u$ with the properties we desire. Observe that there is no immediate way for us to certify if a random $u$ satisfies a correct choice for Theorem 4.6 However, we are able to certify if the output $T_u$ satisfies the desired properties via sampling. To that end we delay the certification step to the end; in effect we run the algorithm assuming that $\text{BSG-test}$ is correct, and certify the map $T_u$ produced. Taking for granted that we have a certifier for $u$, if we sample $O(\text{poly}(\rho^{-1}, \log((\delta')^{-1}))$ values for $u$, then with probability $1 - \delta'$ we will obtain a desirable $T_u$. We now describe the certification test.

\[\text{certifier-u}(T_u, \phi):\]

- Sample $s$ values from $A_\phi$: $a_1, \ldots, a_s$.
- Return 1 if at least $\text{quasi-poly}(\rho) \cdot s$ of the $\{a_i\}$ satisfy $T_u(a_i) = \phi(a_i)$.

Via Lemma 4.1 if we sample $O(\text{poly}(\text{quasi-poly}(\rho), \log((\delta')^{-1}))$ then with probability $1 - \delta'$ if $T_u$ agrees with $\phi$ on at least a quasi-$\text{poly}(\rho)$ fraction of $A$ then $\text{certifier-u}$ returns 1.

Next, we describe how we will construct $T_u$. The rough idea is as follows. Freiman-Ruzsa guarantees that this set with small doubling that we have passed to is approximately a linear space. With some delicate pigeonholing, we will be able to extract an affine map from the projection onto the last $n$ coordinates.
Let $t = 4n^2 + \log(10\delta^{-1})$. Start by sampling $K = O(\text{poly}(\log(\delta^{-1}), \rho^{-1}))$ elements of $A_\phi$. Run BSG-test($u, \cdot$) on these sampled elements and only retaining those for which BSG-test returns 1. By Lemma 4.11 we can pick the hidden constants such that with probability at least $1 - \delta'$ the sampled points contain at least $t$ samples from $A_\phi(u)$.

In what follows, we will freely do change of basis arguments and the like, because the run-time for these linear algebraic algorithms are $O(n^3)$ and can be accommodated within our poly($n$) target on run-time.

In more detail, Freiman-Ruzsa tells us that $|\langle A_\phi^{(2)} \rangle| \leq \exp(\rho^{-C}|A|)$, so all the retained elements lies in a space of dimension at most $n + \log(\nu^{-1})$ where $\nu = \exp(\rho^{-C})$. Of these elements, choose a basis $v_1, \ldots, v_s$. Complete the basis to $v_1, \ldots, v_s$ so that the projection onto the first $n$ coordinates has full rank. By a change of basis, without loss of generality we assume that $v_i = (e_i, u_i)$ where $e_i$ are the standard basis vectors for $\mathbb{F}_p^n$. Let $M$ be the matrix with $v_1, \ldots, v_s$ as columns. Effectively, we can write

$$M = \begin{pmatrix} I & 0 \\ T & U \end{pmatrix}.\]$$

Here, $I$ is the $n \times n$ identity matrix. Via a lower bound on the dimension of the span of the retained points (Claim 4.14 of [20]) as well as the Freiman-Ruzsa upper bound from earlier, we obtain by pigeonhole lemma 4.1, we can pick the hidden constants such that with probability at least $\frac{1}{2}$

$$\mathbb{P}_{z \in \mathbb{F}_p^n}[\phi(z) = Tx + Uz] \geq \kappa,$$

where $\kappa = \Omega(\rho^{2} \cdot \nu) = \text{quasi-poly}(\rho^{-1})$.

We want to identify this “popular” value of $z$; once we have identified it we will take our affine map $T_0$ to be $T_0(x) = Tx + Uz$.

**popular-z($\phi, T, u$):**

- For each of $p^t$ possible values of $z \in \mathbb{F}_p^n$, sample $j = O(\kappa^{-1} \cdot k)$ uniformly random elements $b_1, \ldots, b_j$ from $\mathbb{F}_p^n$.
- If at least $k$ of the $\{b_i\}$ satisfy $\phi(b_i) = Tb_i + Uz$ then return the corresponding value of $z$.

As a direct consequence of Lemma 4.1 by taking $k = O(\log((\delta')^{-1}))$ we have the guarantee for popular-z that with probability $\delta'$ its output satisfies $\mathbb{P}_{z \in \mathbb{F}_p^n}[\phi(z) = Tx + Uz] \geq \kappa$.

Given the choice of the error parameter $\delta'$, we can use the union bound to combine the probability estimates and obtain the guarantees we desire.

\[\square\]

5 Finding correlated cubic phases

In this section, we give a complete proof to each theorem that will be needed to establish the algorithmic $U^4$ inverse theorem. Throughout the following two sections, a prime $p$ is fixed, and $G$ denotes $\mathbb{F}_p^n$.

An oracle is an imaginary machine that gives us a suitable computational ability within constant time. In most cases we will assume that we have basic oracles such as membership tester for a set and query access to a function. Specifically, we say that we have a membership tester for a set $A$ if there is an oracle which tells us whether an input $x$ is in $A$ or not. Also, we say that we have a query access to a function $f : X \to Y$ if there is an oracle which for an input $x \in X$ returns $f(x) \in Y$. The last type is an oracle access to probability distribution; for a function $f : X \to Y$ where $Y$ is the space of probability distribution on $Z$, then for each input $x \in X$, the oracle returns $z \in Z$ according to the probability distribution $f(x)$.

Throughout the algorithms, we consider each variable to be global, meaning that even if some sub-algorithms are terminated we can still access to variables that was computed already. We will only need poly($n$) large storage to do so.

Note that we say a function $f : G \to \mathbb{C}$ is bounded if $\|f\|_\infty \leq 1$.  

19
Before we begin proper, we introduce a primitive that allows us to sample from a set $A \subset (\mathbb{F}_p^n)^2$ or $A \subset \mathbb{F}_p^n$. Here we assume that we have query access to $A$ and the ambient group is $G$.

**Theorem 5.1.** Given a bounded $f : G \to \mathbb{C}$ and $\epsilon > 0$, define $A_1 \subset A_2 \subset G \times G$ so that $(a, b) \in A_1$ if $\| \partial_{a,b}f \|_\infty \geq 2\epsilon$ and $(a, b) \in A_2$ if $\| \partial_{a,b}f \|_\infty \geq \epsilon$.

There is an algorithm member-$A$ that makes $O(\text{poly}(n, 1/\epsilon, \log(1/\delta)))$ queries to $f$ and with probability $1 - \delta$ outputs 1 if $(a, b) \in A_1$ and with probability $1 - \delta$ outputs 0 if $(a, b) \notin A_2$. There is an algorithm query-$\phi$ that makes $O(\text{poly}(n, 1/\epsilon, \log(1/\delta)))$ queries to $f$ and with probability $1 - \delta$ outputs $\phi(a, b)$ such that $|\partial_{a,b}f(\phi(a,b))| \geq \epsilon$ if $(a, b) \in A_2$ and has no guarantees otherwise.

**Definition 5.2** (second-order vertical parallelogram). A **vertical parallelogram** is a set of 4 points $(x, y), (x, y+h), (x+w, y'), (x+w, y'+h) \in G \times G$ for some $x, y, y', h, w \in G$. A **second-order vertical parallelogram** is a quadruple $Q = (P_1, P_2, P_3, P_4)$ such that $((w(P_1), h(P_1)), (w(P_2), h(P_2)), (w(P_3), h(P_3)), (w(P_4), h(P_4)))$ form a vertical parallelogram where $w(P_i)$ and $h(P_i)$ denote the width and height of $P_i$, respectively.

**Definition 5.3** (4-arrangement, second-order 4-arrangement). A **4-arrangement** is a pair $(P_1, P_2)$ of vertical parallelograms of the same width and height. A **second-order 4-arrangement** is a pair $(Q_1, Q_2)$ of second-order vertical parallelograms of the same width and height.
Essentially, we define 4-arrangement and second-order 4-arrangement to figure out whether a map $\phi$ behaves like an affine map in each vertical parallelogram of fixed width and height. Therefore we need another concept that measures how much $\phi$ behaves well with the vertical parallelograms.

**Definition 5.4** ($\phi$ respects the (second-order) 4-arrangement). Given a map $\phi: G \times G \to G$ and a vertical parallelogram $P = ((x, y), (x, y + h), (x + w, y'), (x + w, y' + h))$, define $\phi(P)$ as $\phi(x, y) - \phi(x, y + h) - \phi(x + w, y') + \phi(x + w, y' + h)$. Then for a 4-arrangement $(P_1, P_2)$, $\phi$ respects $(P_1, P_2)$ if $\phi(P_1) = \phi(P_2)$. Similarly, for a second-order vertical parallelogram $Q = (P_1, P_2, P_3, P_4)$, define $\phi(Q)$ as $\phi(P_1) - \phi(P_2) - \phi(P_3) + \phi(P_4)$.

Then for a second-order 4-arrangement $(Q_1, Q_2)$, $\phi$ respects $(Q_1, Q_2)$ if $\phi(Q_1) = \phi(Q_2)$.

**Theorem 5.5.** Let $\eta, \delta, \epsilon > 0$. Given $A_1 \subset A_2 \subset G \times G$ and $\phi: G \times G \to G$ with the property that $\phi$ respects at least $\epsilon|G|^{32}$ second-order 4-arrangements in $A_1$.

Then $A_2$ has a subset $A'$ that contains at least $\operatorname{poly}(\eta^{-1}, \epsilon^{-1})|G|^{32}$ second-order 4-arrangements such that the proportion of its arrangements that are respected by $\phi$ is at least $1 - \eta$.

Let $\text{member-}A(u, \delta')$ be an algorithm that, with probability at least $1 - \delta'$, accepts if $u \in A_1$ and rejects if $u \notin A_2$. Suppose we also have query access to $\phi$.

Then the algorithm $\text{member-}A$-prime makes $O(\operatorname{poly}(\log(\delta^{-1}), \epsilon^{-1}, \eta^{-1}))$ queries to $\text{member-}A$ and $\phi$ and with probability $1 - \delta$ outputs 1 if $(a, b) \in A' \cap A_1$ and 0 if $(a, b) \notin A'$.

We effectively make algorithmic the dependent random selection process in [9, Lemma 3.11]. Note that specifying $A'$ tantamounts to selecting random elements $\{s_i\}_{i=1}^k$ with $s_i \in \mathbb{F}_p$, random $n \times n$ matrices $\{M_i\}_{i=1}^k$ with $M_i \in \text{Mat}_n(\mathbb{F}_p)$ and also $\{r(x,y)\}_{(x,y) \in \mathbb{F}_p^2 \times \mathbb{F}_p^2}$ with $r(x,y) \in [0,1]$. Specifically, we have that $(x, y) \in A'$ if $r(x,y) \leq 2^{-k} \prod_{i=1}^k \left(1 + \cos \left(\frac{2\pi}{p}(s_i, \phi(x, y)) + \langle x, M_i y \rangle\right)\right)$.

**Algorithm weighted-member-A-prime($\phi, x, y$):**

- **Input** membership test for $A$, query access to $\phi$, $(x, y) \in A$.
- **Output** the probability to choose $(x, y)$ as an element of $A'$
  - Sample $k$ random elements $s_1, \ldots, s_k$ as well as independent random $n \times n$ matrices $M_1, \ldots, M_k$ over $\mathbb{F}_p$. Return $2^{-k} \prod_{i=1}^k \left(1 + \cos \left(\frac{2\pi}{p}(s_i, \phi(x, y)) + \langle x, M_i y \rangle\right)\right)$.

Observe that in $\text{weighted-member-A-prime}$, we effectively have an output of a weighted set. To remove this source of randomness, we introduce a certifier for weighted sets $A'$; we can then repeat the selection procedure until we pass the certifier. Note that every 32-tuple from $G^{32}$ corresponds to a second-order 4-arrangement $Q$. Write $r_Q = \{r(x,y) : (x, y) \in Q\}$.

**Algorithm certifier-A-prime($\phi$):**

- **Input** query access to $\phi: G \times G \to G$.
- **Output** verification whether it has suitable guarantees, $R$
  - Sample $ar$ random 32-tuples from $G^{32}$.
  - Sample $32ar$ random reals from $[0,1]$ for the $r_Q$ for each of the second-order 4-arrangements $Q$ corresponding to each 32-tuple.
  - Only retain those tuples for which we have $r(x,y) \leq \text{weighted-member-A-prime}(x, y)$ for all $(x, y) \in Q$. If less than $r$ tuples remain, return 0.
  - Otherwise, let these corresponding second-order 4-arrangements be $Q_1, \ldots, Q_r$ and write $R = \bigcup_r r_Q$. Note that each tuple represents a second-order 4-arrangement and therefore can be thought of as two second-order vertical parallelograms, so we can write $Q_i = (P^{(i)}_1, P^{(i)}_2)$.
  - For each $i \in [r]$, compute $\ell_i = \phi(P^{(i)}_1) - \phi(P^{(i)}_2)$ and let $R$ be the number of $i$ such that $\ell_i = 0$. If $R/r \geq 1 - \rho$, return 1 and store $R$. Otherwise, return 0.

Putting everything together, we get a desired membership tester for $A'$. 
Note that this membership tester is dynamic, since we update $\mathcal{R}$ as we call member-A-prime on the fly.

Proof. Let $\delta' = \delta/2$. We will first prove that by picking the right parameters, we can ensure that with probability $1 - \delta'$ the output of certifier-A-prime has the following property: if certifier-A-prime outputs 1 and stores corresponding $\mathcal{R}$, then for any possible extension of $\mathcal{R}$ to $\{r_{(x,y)}\}_{(x,y) \in G \times G}$ obtained by drawing additional random reals from $[0, 1]$ when necessary, the set $A'$ corresponding to this choices of $\{r_{(x,y)}\}_{(x,y) \in G \times G}$, $\{s_i\}_{i=1}^k$ and $\{M_i\}_{i=1}^k$ has the property that:

(a) $A'$ contains at least $\text{poly}(\eta^{-1}, \epsilon^{-1})|G|^{32}$ second-order 4-arrangements, and

(b) $\phi$ respects at least a $(1 - \eta)$-fraction of these second-order 4-arrangements.

Take $\alpha = \text{poly}(\eta^{-1}, \epsilon^{-1}, \log((\delta')^{-1}))$, then by Lemma 4.1 if of the $\alpha r$ random tuples we sample, we have retained $r$ of them then with probability $1 - \delta'/2$ we have that $A'$ satisfies property (a).

Next, take $r = \text{poly}(\log((\delta')^{-1}), \eta^{-1})$. Note that at this stage in the algorithm, we may assume that any 32-tuple we work with corresponds to a second-order 4-arrangement with all its constituent elements lying in $A'$. By adjusting the constants, we can guarantee the existence of a set $A''$ that contains $\text{poly}(\eta, \epsilon)|G|^{32}$ second-order 4-arrangements with $\phi$ respecting at least a $(1 - 3\eta/4)$-fraction of them. Set $\rho = 3\eta/4$. By Lemma 4.1, with probability $1 - \delta'/2$ we can estimate $|R/r - (1 - 3\rho/4)|$ up to an additive error of $\rho/4$. This implies that with probability $1 - \delta'$ if certifier-A-prime returns 1 then the proportion of 4-arrangements that $\phi$ respects in $A'$ is at least $1 - \eta$.

Using the union bound, the upshot is that with probability $1 - \delta'$ if certifier-A-prime returns 1 then we have the guarantees of (a) and (b); here we know that both can be satisfied simultaneously because of the proof of existence in [9, Lemma 3.11]. This also ensures that if member-A-prime does not return $\perp$ then it has the guarantees we desire.

Lastly, we need to check that with high probability member-A-prime does not return $\perp$. To that end we need to calculate the probability that certifier-A-prime returns 1. For a random choice of random elements $\{s_i\}_{i=1}^k$ with $s_i \in \mathbb{F}_p$, random $n \times n$ matrices $\{M_i\}_{i=1}^k$ with $M_i \in \mathbb{M}_{n \times n}(\mathbb{F}_p)$ and also $\{r_{(x,y)}\}_{(x,y) \in G \times G}$ with $r_{(x,y)} \in [0,1]$, let $X$ be the random variable denoting the number of second-order 4-arrangements that are respected by $\phi$ and let $Y$ be the number of second-order 4-arrangements that are not. We claim that via Lemma 4.1 by taking $s = \text{poly}(\log((\delta')^{-1}), \epsilon^{-1}, \eta^{-1})$ we will be able to ensure that with probability $1 - \delta'$ member-A-prime will not return $\perp$. Equivalently, we will prove that $\mathbb{P}[X - \eta Y \geq 0] \geq \text{poly}(\eta, \epsilon)$. To that end recall that $\mathbb{E}[X - \eta Y] \geq \text{poly}(\eta, \epsilon)|G|^{32}$. We also that $X - \eta Y$ is bounded above by the number of second order arrangements respected by $\phi$ ins $A \supset A'$ which is in turn at most $\epsilon|G|^{32}$. In other words,

$$\mathbb{P}[X - \eta Y \geq 0] \geq \text{poly}(\eta, \epsilon)|G|^{32}$$

which is equivalent to the desired claim.

With our choice of $\delta'$, combining the above via the union bound gives the run-time as well as output guarantees we desire.
Theorem 5.6. Given subsets $A_1 \subset A_2 \subset G \times G$ where $A_1$ has density $\alpha$ such that $|\partial_{a,b}f(\phi(a,b))| \geq 2\sqrt{\alpha}$ for $(a,b) \in A_1$ and $|\partial_{a,b}f(\phi(a,b))| \geq \sqrt{\alpha}$ for $(a,b) \in A_2$, then $A_2$ has a subset $\tilde{A}$ of density $\Omega(\text{poly}(n, \alpha, \log(\delta^{-1})))$ such that for each $b$ we have that $\phi|_{\tilde{A}}$ is a Freiman homomorphism.

There is an algorithm $\text{member-A-tilde}$ that makes $O(\text{poly}(\alpha^{-1}, \log(\delta^{-1})))$ queries to $\text{member-A}$ and $\phi$ and with probability at least $1 - \delta$ outputs 1 if $(a,b) \in \tilde{A} \cap A_2$ and 0 if $(a,b) \notin \tilde{A}$.

\begin{algorithm}
\textbf{member-A-tilde}(A, \phi, a, b):
\textbf{Input} membership tests for $A_1, A_2$, query access to $\phi$: $G \times G \to G$, $(a,b) \in G \times G$
\textbf{Output} 1 if $(a,b) \in \tilde{A}$ and 0 otherwise with high probability
\begin{itemize}
  \item Using query access to $\phi$, execute $\text{find-affine-map}(A, \phi)$ with output $T_a$.
  \item If $\phi(a,b) \neq T_a$, output 0. Else, output 1.
\end{itemize}
\end{algorithm}

Proof. Note that for each $b$
\[
\mathbb{E}_a 1_{A_\bullet}(a)|\partial_{a,b}f(\phi(a,b))|^2 \geq \alpha d(b),
\]
where $d(b)$ is the density of $A_\bullet = \{a \in G : (a,b) \in A_2\} \subset G$. Then by Lemma 3.1 (or Proposition 6.1 in Gowers), there are at least $\alpha^4 d(b)^4 p^{3n}$ quadruples $(x,y,z,w) \in A_\bullet^4$ such that $x + y = z + w$ and $\phi(x,b) + \phi(y,b) = \phi(z,b) + \phi(w,b)$. Therefore by $\text{find-affine-map}$, there is an affine map $T_b$ which agrees with $\phi(\cdot, b)$ on at least $\text{poly}(\alpha d(b))$ fraction of $A_\bullet$. Such subset of $A_\bullet$ that agrees with $\phi(\cdot, b)$ is the set $\tilde{A}$. By Jensen's inequality and from $(1/p^n) \sum_{b \in \mathbb{F}_p} d(b) = \alpha$, we have that the density of $\tilde{A}$ is at least quasi-poly($\alpha$). Also, since for each $b$ the map $\phi$ agrees with an affine map on $\tilde{A}$, they are Freiman homomorphisms as well.

For $(a,b) \in G \times G$, if $(a,b) \notin A_2$, output 0. Otherwise, using $\text{find-affine-map}$, output an affine map $T_b$. If $\phi(a,b) = T_b(a)$, output 1, otherwise 0.

The overall algorithm fails when $\text{find-affine-map}$ fails, so the algorithm succeeds with probability at least $1 - \delta$. □

Theorem 5.7. Given $f: G \times G \to \mathbb{C}$ be a bounded function. Let $\text{approx-f}(\epsilon, \delta, x)$ be an oracle such that for every $x \in G \times G$ we have with probability at least $1 - \delta$ that $|f(x) - \text{approx-f}(\epsilon, \delta, x)| \leq \epsilon$.

Given oracle access to $\text{approx-f}$ there exists an algorithm $\text{bohr-aff-map}$ that makes $O(\text{quasi-poly}(\xi^{-1}, \gamma^{-1}), \text{poly}(n, \log(\delta^{-1})))$ queries to $\text{approx-f}$ and with probability $1 - \delta$ outputs bi-affine maps $T_1, \ldots, T_m$ such that for all but at most $\xi |G|$ points $(h,u) \in \{(h,u) : |f_{\bullet h}(u)|^2 \geq 2\}$ we have $T_i h = u$, where $m$ has quasipolynomial dependence on $\gamma, \xi$.

The existence of the bi-affine maps $T_1, \ldots, T_m$ with such a property follows from [9] Lemma 4.10. We next describe an algorithm to identify them.
Proof. We can think of \texttt{bogo-aff-map} as operating in two stages. The first stage which samples \( r \) elements and then runs \texttt{fuzzy-GL}(f_{\bullet}, \gamma) \) is effectively a certifier stage; we verify if the linear maps in \( \mathcal{L} \) already has the covering property we desire and terminate the algorithm if it does. Otherwise, \( \mathcal{L} \) does not cover the large Fourier spectrum and we can invoke the discussion in [9, Section 4.6] to proceed to the second stage where we generate an additional linear map to add to \( \mathcal{L} \).

For simplicity of notation write \( \Sigma_\gamma = \{(h,u) : |f_{\bullet}^h(u)| \geq \gamma \} \). Let \( \delta = O(\delta / \text{quasi-poly}(\xi^{-1}, \delta^{-1})) \) and observe that \( \log \delta = O(\text{poly}(\log(\delta^{-1}), \xi^{-1})) \). Take \( r = O(\text{poly}(\xi^{-1}, \log(\delta^{-1}))) \), \( \nu = \omega = \gamma / 10 \). Note that if we consider the corresponding \((h,u)\) from the output of \texttt{fuzzy-GL}(f_{\bullet}, \gamma) \) we obtain a set \( \Sigma \) slightly larger than \( \Sigma_\gamma \). Particularly, we have \( \Sigma_\gamma \subset \Sigma \subset \Sigma_{\gamma/5} \). We will show that the parameters we pick ensure that with probability at least \( 1 - \delta \) we have \( \bigcup_h \{ (h,u) : u \in \mathcal{L}(h) \} \) covers an at least \( 1 - \xi \) fraction of \( \Sigma \), which by our earlier observation will imply that the same is true for \( \Sigma_\gamma \). This would then show that when the algorithm terminates it would have the desired guarantees. Indeed, by Lemma [4.1] with probability at least \( 1 - \delta \) if we have less than \( \xi \) elements remaining in \( K_1 \) after the pruning in the first stage then we have at least \( \xi |G| \) of \( h \) that has a large Fourier coefficient not covered by the existing affine maps.

Before we move on to the rest of the proof, note by Parseval’s (as in the proof of [9, Lemma 4.10]) since \( \Sigma \subset \Sigma_{\gamma/5} \) this algorithm should terminate after at most \( \text{quasi-poly}(\xi^{-1}, \gamma^{-1}) \) iterations.

Next, we study the second stage. We can ensure that \texttt{find-affine-map} succeeds with probability at least \( 1 - \delta \). However, we also need to ensure that the application of \texttt{find-affine-map} is valid. To that end we need to check that we can give a polynomial time algorithm for each of the following tasks:

- Check for membership in \( Q \).
- Sample a random element from \( Q \).
- Query access to \( \sigma \).

\texttt{member-Q(h)}:

- Run \texttt{fuzzy-GL}(f_{\bullet}, \gamma). If the resulting list \( L_h^h \) is empty, return 0.
- Otherwise, for each \( \ell \in \mathcal{L} \) if \( \ell(h) \in L_h^h \) remove the corresponding value. At the end of this process, if \( L_h^h \) is empty, return 0. Otherwise, return 1.
query-sigma(h):
- Run member-Q on h. If the output is 0, return ⊥.
- Otherwise, run fuzzy-GL(f_h, γ). For each ℓ ∈ L if ℓ(h) ∈ L_h remove the corresponding value. Return an arbitrary element from the resulting list.

sampler-Q:
- Sample r elements from G and run member-Q on each of them, returning the first element on which member-Q outputs 1.

In each of these possibilities we can pick the parameters such that with probability 1 − 1/δ they achieve the goal of the algorithm. Given the guarantees for fuzzy-GL this is obviously true for member-Q and query-sigma. For sampler-Q, take r = poly(log(δ −1), log(1 − ξ), ξ). Since |Q| has density at least ξ, each of the r elements does not lie in Q with probability 1 − ξ. With our chosen parameters, it follows that the probability that at least one of the elements that we sample lies in Q is indeed 1 − 1/δ.

Remark: For each ℓ ∈ L = quasi-poly(ξ −1, γ −1) each of the sub-routines above runs in polynomial time.

We briefly recall the argument in [9, Theorem 4.15], which establishes the existence of such a β, to motivate our algorithm. The goal here is to find a L_2 approximation proj_β F of F. Given the maps {T_i}, a natural choice of such an approximation would be F'(x, y) = \sum_{i=1}^m \bar{F}_y(T_i y)\omega^{(x, T_i y)}. Because of the presence of redundant maps, namely T_i h = T_j h for some i ≠ j, we lose L_2 control easily and need to do one further truncation. We pick out distinct Fourier coefficients via

\[ u_i(y) = \begin{cases} 0 & T_j y = T_i y \text{ for some } j < i, \\ \hat{v}_y(T_i y) & \text{otherwise} \end{cases} \]

defined for each i, and then do one more round of approximation by picking out the large Fourier coefficients of u_i. Particularly, suppose the list of large Fourier coefficients for u_i is given by K_i = \{v_{i1}, \ldots, v_{ik_i}\} then if we consider β_{ij}(x, y) = x.T_iy + v_{ij}y the bi-affine map we desire is given by
\[ \beta(x, y) = (\beta_{11}(x, y), \ldots, \beta_{1k_1}(x, y), \ldots, \beta_m(x, y), \ldots, \beta_{m_1}(x, y), \ldots, \beta_{m_k}(x, y)). \]

In the following sub-routines, whenever we need to query \( f \) we will use the oracle access to \texttt{approx-f} to estimate \( f \).

<table>
<thead>
<tr>
<th><strong>box(φ, w, h):</strong></th>
<th><strong>Input</strong></th>
<th>query access to ( φ : G \times G \rightarrow G ), ( w, h \in G )</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Output</strong></td>
<td>estimate of ( |φ(w, h)| )</td>
<td></td>
</tr>
</tbody>
</table>
|                  | • Sample \( 3s \) values \( \{x_i\}_{i=1}^s \), \( \{y_i\}_{i=1}^s \), \( \{y'_i\}_{i=1}^s \) and output \[
\frac{1}{s} \sum_{i=1}^s φ(x_i, y_i)φ(x_i + h)φ(x_i + w, y_i)φ(x_i + w, y'_i + h). \]

<table>
<thead>
<tr>
<th><strong>bogo-u(ℒ, f, i, y):</strong></th>
<th><strong>Input</strong></th>
<th>query accesses to affine maps ( T_1, \ldots, T_m ) in ( ℒ ), query access to ( f ), integer ( 1 \leq i \leq m, y \in G )</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Output</strong></td>
<td>estimate of ( u_i(y) )</td>
<td></td>
</tr>
<tr>
<td></td>
<td>• Iterate through ( j = 1, \ldots, i-1 ) and if ( T_i y = T_j y ) then return 0.</td>
<td></td>
</tr>
</tbody>
</table>
|                  | • Otherwise, using \texttt{box(f, w, h)} to get a query access to \( \|f\| \), sample \( r \) values \( \{x_i\}_{i=1}^r \) from \( \mathbb{F}_p^m \) and return \[
\frac{1}{r} \sum_{i=1}^r \|f(x_i, y)\|^{\omega^{-1}T_i} y. \]

<table>
<thead>
<tr>
<th><strong>bohr-aff-map(f):</strong></th>
<th><strong>Input</strong></th>
<th>query access to ( f : G \times G \rightarrow G )</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Output</strong></td>
<td>explicit expression of the bi-affine map ( β )</td>
<td></td>
</tr>
<tr>
<td></td>
<td>• Run \texttt{boho-aff-map(f)} and let its output be ( ℒ = {T_1, \ldots, T_m} ).</td>
<td></td>
</tr>
<tr>
<td></td>
<td>• Using \texttt{bogo-u} to get a query access to each ( u_i ), run \texttt{fuzzy-GL}(( u_i, \zeta )) and let the output be ( L_i = {v_{i1}, \ldots, v_{ik_i}} ).</td>
<td></td>
</tr>
</tbody>
</table>
|                  | • For each \( i = 1, \ldots, m \) and \( j = 1, \ldots, k_i \), let \( β_{ij}(x, y) = x T_i y + v_{ij} y \) and return \[
β(x, y) = (β_{11}(x, y), \ldots, \beta_{1k_1}(x, y), \ldots, β_{m1}(x, y), \ldots, β_{mk}(x, y)). \]

**Proof.** Let \( \tilde{δ} = O(δ/m) \) where \( m = \text{quasi-poly}(ξ^{-1}) \) and also let \( ζ = ξ^2/(m^2 2^{m}) \). Suppose we are able to obtain query access to some \( u'_i \) such that with probability at least \( 1 - ζ/10 \) we have \( \|u_i - u'_i\|_∞ ≤ ζ/10 \), then for each \( i \) we can ensure with probability at least \( 1 - \tilde{δ} \) that the output of \texttt{fuzzy-GL}(\( u_i, ζ \)) is a list \( L_i \) with the property that \( \text{Spec}_ζ(f) ⊂ L_i ⊂ \text{Spec}_ζ/f(f) \). In particular, if we write \( u_i(y) = \sum_{v \in L_i} \hat{u}_i(v) ω^v \) then by Hölder’s and \cite{4} Corollary 4.13 that \( \|u_i - u'_i\|_2 ≤ O(ζ) \) with probability at least \( 1 - \delta \). In particular, if we set \( H(x, y) = \sum_{i=1}^m w_i(y)ω^{xT_i y} \) and \( H'(x, y) = \sum_{v \in \{T_1 y, \ldots, T_m y\}} (\|f\|_y) ω^{x v} \) then \( \|H - H'\|_2 ≤ ζ/2 \) with probability at least \( 1 - δ \) by the union bound. Combining with \cite{4} Lemma 4.14) which states that \( \|H' - F\|_2 ≤ ζ/2 \), by the triangle inequality we have that with probability at least \( 1 - δ \), we have that \( \|F - H\|_2 ≤ ξ \).

Furthermore, by Parseval’s and the fact that \( L ⊂ \text{Spec}_ζ/f(f) \) (analogous to the proof of \cite{4} Theorem 4.15), it follows that \( |L_i| = O(m^2 2^{2m}/ξ^2) \). In particular, if we can obtain query access to some \( u'_i \) as described via \texttt{bogo-u} then with probability \( 1 - δ \) we get a bi-affine map with the desired bound on its co-dimension and also \( \|F - \text{proj}_β F\|_2 ≤ ξ \). It suffices to check that we can indeed obtain such query access to \( u'_i \).
To that end, we consider the approximations given by bogo-u. It comes in three stages. First, we can approximate \( f \) by \( \text{approx-f}(\zeta/30, \zeta/30, \cdot) \) which has the property that \( \|f - \text{approx-f}(\zeta/30, \zeta/30, \cdot)\|_\infty \leq \zeta/30 \) with probability at least \( 1 - \zeta/30 \). Second, by taking \( s = O(\text{poly}(\zeta^{-1})) \) in box we can ensure that \( \|f - \text{box}(f, \cdot)\|_\infty \leq \zeta/30 \) with probability at least \( 1 - \zeta/30 \). Third, by taking \( r = O(\text{poly}(\zeta^{-1})) \) in bogo-u and assuming that we have (perfect) query access to \( f \), we can ensure that with probability at least \( 1 - \zeta/30 \)
\[
\|\mathbb{E}_x f(x, y)\omega^{x,T,y} \cdot 1_y : i = \min\{j : T,y = T,y\} - \text{bogo-u}(\cdot)\|_\infty \leq \zeta/30.
\]
Using the triangle inequality to combine the additive error estimates and also the union bound to combine the probability estimates, it follows that we can approximate each \( u_i \) via bogo-u up to an additive error of at most \( \zeta/10 \) with probability at least \( 1 - \zeta/10 \), as desired.

Lastly, we check the runtime guarantees of the algorithm. By Theorem \[5.7\] we have that \( \text{bogo-aff-map} \) makes \( O(\text{poly}(\zeta^{-1}) \cdot \text{poly}(n, \log(\delta^{-1})) \) queries to \( \text{approx-f} \). Note that the overall number of queries that the sub-routine bogo-u makes to \( \text{approx-f} \) is \( O(\text{poly}(\zeta^{-1})) \cdot m \). Next, the application of \( \text{fuzzy-GL} \) in \( \text{bohr-aff-map} \) makes \( O(\text{poly}(n, \zeta^{-1}, \log(\delta^{-1}))) \) queries to \( \text{bogo-u} \). In summation, it follows that we make \( O(\text{poly}(\zeta^{-1}) \cdot \text{poly}(n, \log(\delta^{-1}))) \) queries to \( \text{approx-f} \).

Now we introduce a crucial concept, a rank of bi-affine map. Later, we will get quasirandom property from a high rank bi-affine map. The dimension of a bi-affine map is the dimension of its range.

**Definition 5.9.** For a one-dimensional bi-affine map \( \beta : G \times G \to \mathbb{F}_p \), if we write it as \( \beta(x, y) = x.Ty + x.A + B.y + C \) for \( T \in \text{Mat}_n(\mathbb{F}_p) \) and column vectors \( A, B, C \), then the rank of \( \beta \) is defined to be the rank of \( T \). For a bi-affine map \( \beta : G \times G \to \mathbb{F}_p^k \), the rank of \( \beta \) is the smallest rank of one-dimensional bi-affine map \( (x, y) \mapsto u.\beta(x, y) \) for nonzero \( u \in \mathbb{F}_p^k \).

**Theorem 5.10.** Given an explicit representation of a bi-affine map \( \beta : G \times G \to \mathbb{F}_p^k \) and \( t \in \mathbb{Z}^+ \), then there exists an algorithm that runs in time \( O(kp^k + k^4) \) that outputs a basis for \( X_0 \) and \( Y_0 \) such that \( \dim X_0, \dim Y_0 \leq tk \) with the property that the corresponding Bohr decomposition has rank at least \( t \).

We first introduce a certifier of sorts for whether our decomposition has achieved the desired high rank condition. The algorithm \( \text{linear-translate} \) outputs a value of \( u \), if it exists, such that \( (x, y) \mapsto u.\beta(x, y) \) has rank at most \( t \).

**linear-translates(\beta):**

**Input** explicit expression of a bi-affine map \( \beta : G \times G \to \mathbb{F}_p^k \)

**Output** \( \bot \) if the rank of \( \beta \) is greater than \( t \), \( u \in \mathbb{F}_p^k \) if \( u.\beta(x, y) \) is of rank at most \( t \)

- For each of the \( p^k \) possibilities of \( u \in \mathbb{F}_p^k \) compute the rank of \( u.\beta(x, y) \) and output any choice of \( u \) for which this value is at most \( t \). If no such choice of \( u \) exists, output \( \bot \).

Using this certifier, we can then iteratively prune our space to identify the desired \( X_0 \) and \( Y_0 \).

**bohr-decomp(\beta, t):**

**Input** explicit expression of a bi-affine map \( \beta : G \times G \to \mathbb{F}_p^k \), a positive integer \( t \)

**Output** basis for \( X_0 \) and \( Y_0 \)

- If the output of \( \text{linear-translates}(\beta) \) is \( \bot \) then output \( X \) and \( Y \). Otherwise, if the output of \( \text{linear-translates}(\beta) \) is \( u \), then we can compute a basis \( b_1, \ldots, b_t \) for \( \langle u \rangle \).
- Write \( P = [b_1 \ldots b_t] \) we have that projection \( \pi_u \) to \( \langle u \rangle \) is given by \( P(P^T P)^{-1}P^T \). Replace \( \beta \) by \( \pi \circ \beta \) by composing the appropriate matrix and repeat from the first step.

**Proof.** Since \( \text{linear-translate} \) brute forces through all possibilities of \( u \in \mathbb{F}_p^k \), it runs in time \( p^k \).

Computing a basis for \( \langle u \rangle \subset \mathbb{F}_p^k \) takes time at most \( O(k^3) \). Therefore each iteration of the loop in \( \text{bohr-decomp} \) takes time \( O(p^k + k^3) \). By \[9\] Lemma 5.1, \( \text{bohr-decomp} \) terminates after at most \( k \) iterations for a total runtime of \( O(kp^k + k^4) \) and has the guarantees we desire. \( \square \)
From the theorem 5.10, we now find a Bohr decomposition of $\beta: G \times G \to G$ as follows: for each $(v, w, z) \in X_0 \times Y_0 \times G$, define $B_{v, w, z}$ be a level set $\{(x, y) \in G \times G : x|_{X_0} = v, y|_{Y_0} = w, \beta(x, y) = z\}$. In particular, we call such a bohr decomposition as a bilinear Bohr decomposition. We define the rank of a bilinear Bohr decomposition as the smallest rank of $B_{v, w, z}$ for each $(v, w, z) \in X_0 \times Y_0 \times G$.

In the next theorem, $A$ is the group algebra of $G$ and $\Sigma(A)$ is the subset of $A$ consisting of elements the sum of whose coefficients is 1. We define $\Sigma(A)$ because we want to see for a fixed pair of width and height $(w, h) \in G \times G$, to which element and to what extent $\phi(P)$ is concentrated. Interpreting this as probability distribution, in particular we want such vertical parallelograms are concentrated to one specific element, hence we want the probability distribution is highly close to a delta distribution. This naturally leads us to the following notion of $(1-\eta)$-bihomomorphism:

**Definition 5.11.** Given a non-negative function $\mu: G \times G \to \mathbb{R}$ and $\phi: G \times G \to A$, and a constant $0 \leq \eta \leq 1$, $\phi$ is a $(1-\eta)$-bihomomorphism with respect to $\mu$ if

$$\mathbb{E}_{w, h} \left\| P \in \mathcal{P}(w, h) \mu(P) \phi(P) \right\|^2 \geq (1-\eta) \mathbb{E}_{w, h} \left\| P \in \mathcal{P}(w, h) \mu(P) \right\|^2,$$

where $\mathcal{P}(w, h)$ is the set of vertical parallelograms whose width and height are $w$ and $h$, respectively. Furthermore, if $P$ is a vertical parallelogram whose vertices are $(x, y), (x, y + h), (x + w, y'), (x + w, y' + h)$, $\mu(P)$ and $\phi(p)$ are given as follows:

$$\mu(P) = \mu(x, y)\mu(x, y + h)\mu(x + w, y')\mu(x + w, y' + h),$$

$$\phi(P) = \phi(x, y)\phi(x, y + h)\phi(x + w, y')\phi(x + w, y' + h),$$

where for $x = \sum_{g \in G} c_g g \in A$, $x^* = \sum_g c_g(-g)$.

**Theorem 5.12.** Suppose there is a Bohr decomposition of a bi-affine map $\beta: G \times G \to F^p$ of rank $t$ and codimension $k$ with corresponding Bohr sets $\{B_{v, w, z}\}$. Let $\mu$ and $\xi$ be functions taking values on $[0, 1]$ that are constant on each $B_{v, w, z}$. Let $\phi: G \times G \to \Sigma(A)$ be a $(1-\eta)$-bihomomorphism with respect to $\mu$. Suppose also that $\mathbb{E}\xi = \zeta$. Suppose $0 < \gamma \leq \eta|\mu|^8/8$ and $p^{-1} \leq \eta p^{-\eta k}/8$. Then there exists $(v, w, z)$ such that $\phi$ is a $(1-4\eta)$-bihomomorphism with respect to $1_{B_{v, w, z}}$, the value of $\mu$ on $B_{v, w, z}$ is at least $\eta|\mu|^8$ and the value of $\xi$ on $B_{v, w, z}$ is at most $\gamma^{-1}\zeta$.

Suppose we have query access to the probability distribution $\phi: G \times G \to \Sigma(A)$. For any $\delta > 0$, suppose we have query access to $\mu_\delta': G \times G \to [-\varepsilon_1(\delta), 1 + \varepsilon_1(\delta)]$ which for each $(x, y) \in G \times G$ satisfies $|\mu(x, y) - \mu'(x, y)| \leq \varepsilon_1(\delta)$ with probability $1 - \delta$ and also query access to $\xi_\delta: G \times G \to [-\varepsilon_2(\delta), 1 + \varepsilon_2(\delta)]$ which for each $(x, y) \in G \times G$ satisfies $|\xi'(x, y) - \xi(x, y)| \leq \varepsilon_2(\delta)$ with probability $1 - \delta$. Suppose we have an explicit representation of $\beta$ as well as basis for the corresponding $X_0, Y_0$ in the Bohr decomposition. There is an algorithm high-rk-bohr-set running in time $O(\text{poly}(\log(\delta^{-1}), p^{-r-s}, p^{-k}, c_1^{-1}))$ that with probability $1 - \delta$ outputs $v, w, z$ corresponding to a Bohr set $B_{v, w, z}$ which satisfies the following properties:

- $\phi$ is a $(1 - 5\eta)$-bihomomorphism with respect to $1_{B_{v, w, z}}$,
- $\mu(x) \geq |\mu|^8/2 - \rho_1(\delta)$ for any $x \in B_{v, w, z}$ where $\rho_1(\delta) \to 0$ if $\delta \to 0$, and
- $\xi(x) \leq \gamma^{-1}\xi + \rho_2(\delta)$ for any $x \in B_{v, w, z}$, where $\rho_2(\delta) \to 0$ if $\delta \to 0$.

The existence part of this theorem follows from [9] Theorem 5.8. Algorithmically, we will go over all possible values of $v, w, z$ and run a certifier on each possibility.

Note that in order to compute $[\phi] = ([\phi], ||\phi||)$, we can use the fact that $[\phi]$ is the distribution obtained by evaluating $\phi$ at a randomly chosen vertical parallelogram.

box-dist(f):

- Sample 3 elements $x, y, y'$ uniformly at random from $G$ which corresponds to a random vertical parallelogram $P = \{(x, y), (x, y + h), (x + w, y'), (x + w, y' + h)\}$ and output $f(P)$ where

$$f(P) = f(x, y)f(x, y + h)f(x + w, y')f(x + w, y' + h)$$

28
Moreover, \( (f, g) = \mathbb{P}_{f_i \sim f, g_i \sim g} [f_i = g_i] \) which we can therefore approximate by sampling some \( \{f_i\} \) and \( \{g_i\} \) according to the distributions \( f \) and \( g \) respectively, and returning \( (\# \{ i : f_i = g_i \}) / (\# \text{ of samples}) \).

### inner-product-dist\((f, g)\):

**Input** oracle accesses to probability distributions \( f, g \)

**Output** estimate of \( \langle f, g \rangle \)

- Sample \( r \) elements according to the distribution \( f \), call them \( a_1, \ldots, a_r \). Similarly, sample \( r \) elements according to the distribution \( g \) and call them \( b_1, \ldots, b_r \).
- Output the fraction of \( i \) such that \( a_i = b_i \).

### sq-brac-dist\((f)\):

- Execute \( \text{inner-product-dist}(\text{box-dist}(f), \text{box-dist}(f)) \).

Next, we will introduce a primitive for estimating \( \| f \|, \| g \| \).

### sq-brac\((f, g)\):

- We approximate \( \| f \| \) by \( \tilde{f} \) and \( \| g \| \) by \( \tilde{g} \). Sample \( 3s_1 \) values \( \{x_i\}_{i=1}^{s_1}, \{y_i\}_{i=1}^{s_1}, \{y'_i\}_{i=1}^{s_1} \) and let

\[
\tilde{f}(w, h) = \frac{1}{r} \sum_{i=1}^{r} f(x_i, y_i) f(x_i, y_i + h) f(x_i + w, y'_i) f(x_i + w, y'_i + h)
\]

and similarly for \( \tilde{g}(w, h) \).
- Sample \( s_2 \) pairs \((w_i, h_i) \in G \times G\) and return \( \frac{1}{s_2} \sum_{i=1}^{s_2} \tilde{f}(w_i, h_i) \tilde{g}(w_i, h_i) \).

Because the number of Bohr sets is within a tolerable bound, we can enumerate all possibilities of the Bohr set and it suffices to output a Bohr set with each of the three properties we desire. We build such a certifier in the following algorithm.

### high-rk-bohr-set\((X_0, Y_0, \phi, A', \beta)\):

**Input** basis for \( X_0, Y_0 \), query access to \( \phi : G \times G \rightarrow G \), membership test for \( A' \), explicit expression of a bi-affine map \( \beta : G \times G \rightarrow \mathbb{F}_p \)

**Output** \((v, w, z) \in X_0 \times Y_0 \times G\) such that \( B_{v, w, z} \) has suitable properties.

Suppose \( \dim X_0 = r \) and \( \dim Y_0 = s \).

- For each of the \( p^k = O(\eta^{-1}) \) possible choices for each of \( v \in X_0 \) and \( w \in Y_0 \) as well as the \( p^k \) possible choices for \( z \) (for a total of \( p^{2k+k} = O(\eta^{-1}) \) choices for the triple \((v, w, z)\)), run each of the following tests.

  - **Test A:**
    - Execute \( \text{sq-brac-dist}(\phi) \). Return 1 if \( \ell > (3/2 - 5\eta)p^{-3r-5s}(p^{-7k} - 4p^{2k-1}) \).
  - **Test B:**
    - We execute \( \text{sampler}(B_{v, w, z}, 1, G^2) \) to select an element \( x \) from \( B_{v, w, z} \).
    - Estimate \([\mu]via \text{sq-brac}(\mu'_{\delta'}, \mu'_{\delta'})\) and let the output be \( R \). Return 1 if \( \mu'_{\delta'}(x) \geq R - \rho_1 / 2 \).
  - **Test C:**
    - For the value of \( x \) in Test B, return 1 if \( \xi'(x) \leq \gamma^{-1} \rho_2 / 2 \).
    - If the output for all the 3 tests above is 1, return the corresponding value of \( v, w, z \).
Proof. Let $\delta' = \delta/3$. By taking $r = O((\log(\delta^{-1}), p^{-r} - s, p^{-k}))$ in inner-product-dist, we are able to estimate $\ell$ to within an additive error of at most $\frac{1}{2}p^{3r - s}p^{-1}p^{-k}$ with confidence $1 - \delta'$. In particular this means that with probability at least $1 - \delta'$, we have $[\delta]^{\ell} \geq (1 - 5\eta)p^{3r - s}p^{-1}p^{-k}$. By [9, Lemma 5.6], we have that $[1_{B_{v,w,z}}] \leq p^{-3r - s}p^{-1}p^{-k}$. As a consequence of our choice of parameter $t$ it follows that $[\delta]^{\ell} \geq (1 - 5\eta)[1_{B_{v,w,z}}]^{\ell}$. That is, if Test A returns 1 then with probability $1 - \delta'$ we have that $\phi$ is a $(1 - 5\eta)$-bihomorphism with respect to $1_{B_{v,w,z}}$.

Let $\rho_1 = 3e(\delta')$. For simplicity of notation we will make the dependence on $\delta'$ implicit and write $\mu'_\rho = \mu'$ and $\epsilon(\delta') = \epsilon_1$. By taking $s_1 = O((\log(\delta^{-1}), \epsilon_1))$ in sq-brac we can estimate $[\mu']$ by $\mu_1$ to within an additive error of at most $\epsilon_1/4$ with confidence $1 - \delta'/4$. Further, by taking $s_2 = O((\log(\delta^{-1}), \epsilon_1))$ in inner-product we can estimate $[\mu]$ to within an additive error of at most $\epsilon_1/4$ with confidence $1 - \delta'/4$, so overall we will able to estimate $[\mu']$ to within an additive error of $\epsilon_1/2$ with confidence $1 - \delta'/2$. By our assumptions on $\rho'$ approximating $\rho$, this means that with probability $1 - \delta'$ if Test B returns 1, then $\mu(x) = [\mu] \rho_1$ for any $x \in B_{v,w,z}$.

Lastly, let $\rho_2 = 2e(\delta')$. For simplicity of notation we will write $\xi_\rho = \xi$. Since $\xi$ approximates $\xi$ to an additive error of at most $e(\delta')$ with confidence $1 - \delta'$, it follows that if Test C returns 1 then with probability $1 - \delta'$ we have that $\xi(x) \leq (\gamma^{-1})^\xi + \rho_2$.

By the union bound, it follows that if high-rk-bohr-set returns 1 then with probability $1 - \delta$ it has the three simultaneous guarantees on $B_{v,w,z}$ that we desire. $\square$

When we introduced the notion $(1 - \eta)$-bihomomorphism, we said that we want the probability distributions we care to be very close to delta distributions. Therefore, we need a definition of distance between two probability distributions.

**Definition 5.13.** For $\phi, \psi \in \Sigma(A)$, the distance between $\phi$ and $\psi$, $d(\phi, \psi)$, is $1 - \langle \phi, \psi \rangle$.

This quantity is deserved to be called a distance because it satisfies the triangle inequality.

**Theorem 5.14.** Let $\psi : G \times G \to \Sigma(A)$ be a $(1 - \eta)$-bihomomorphism on a high-rank bilinear Bohr set $B$ defined by a bi-affine map $\beta$ with codimension $k$ and rank $t$, and write $B'' = \{(w, h) : \beta(w, h) = 0\}$. Define $\psi'(a, b) = \langle 1_{B''}(a, b) \rangle$. Then there exists $\tilde{B} \subset B''$ of density $1 - \rho$ such that $d_{\psi}(w, h), \tilde{\psi}(w, h)) \leq 64\eta p^{-3k}$. Here we take $\rho = 16\eta p^{-16k}$.

Given query access to the probability distribution $\psi$ and an explicit description for $\beta$, there exists an algorithm query-tilde-psi that makes $O((\eta p^{-k}) \log(\delta^{-1}))$ queries to $\psi$ and with probability $1 - \delta$ outputs $\psi(a, b)$ for $(a, b) \in B$ and has no guarantees otherwise.

Since the $\| \cdot \|$ operator can be interpreted appropriately as forming a probability distribution by sampling a random vertical parallelogram, we can sample from $\psi'$.

**psi-prime($\phi, B, w, h$):**

- Sample $x, y, y'$ from $G$ uniformly at random and repeat until $(x, y), (x, y + h), (x + w, y), (x + w, y' + h) \in B$.
- Return $\psi(\phi, x, y) \psi(\phi, x, y + h) \psi(\phi, x + w, y') \psi(\phi, x + w, y' + h)$.

In order to identify $\psi$, we will effectively be doing a majority vote.

**query-tilde-psi($\phi, B''$, w, h):**

- If $(w, h) \notin B'$ return $\perp$. Else, execute $\psi-prime(\phi, B'', w, h)$ for $r$ times and return the most popular value among these $r$ values.

Proof. Take $r = O((1/8 - 64\eta p^{-3k}) \log(\delta^{-1}))$ in query-tilde-psi. By assumption, if $(w, h) \in \tilde{B}$, we have that $\mathbb{P}[\psi(w, h) = \tilde{\psi}(w, h)] \geq 1 - 64\eta p^{-3k}$. By Lemma 5.1, if we let the number of samples for which
\[\psi(w, h) = \widetilde{\psi}(w, h)\] be \(R\) then it follows that
\[
P \left[ R > \frac{p}{2} \right] \geq 1 - e^{-\frac{1}{2(1 - 64np^{-3k})} \cdot \left( \frac{4}{3} - 64np^{-3k} \right)^2} \geq 1 - \delta.
\]

This implies that the majority vote output of \texttt{query-tilde-psi} is with probability \(1 - \delta\) the value \(\widetilde{\psi}(w, h)\) for \((w, h) \in B\) as desired.

\textbf{Theorem 5.15.} Let \(\epsilon, \delta > 0\) and let \(\beta'' : G \times G \to \mathbb{F}_p^k\) be a bi-linear map with rank at least 10\(k\). Define \(B'' = \{(x, y) \in G \times G : \beta''(x, y) = 0\}\). Given an explicit description of \(\beta''\) and query access to \(\psi : B'' \to G\) and the guarantee that \(\psi\) agrees with a bi-affine map \(T\) on an \((1 - \epsilon)\)-fraction of elements of \(B''\), the algorithm \texttt{bi-affine} makes \(O(\text{poly}(n, p^k, \log(\delta^{-1})))\) queries to \texttt{query-tilde-psi} and with probability \(1 - \delta\) outputs a bi-affine map that \(T\) that agrees with \(\widetilde{\psi}\) on an \((1 - \epsilon)\)-fraction of elements of \(B''\).

The quasirandomness from the high rank Bohr set condition allow us to bound the probability that a random vertical parallelogram has all its vertices in a high rank Bohr set \(B\).

\textbf{Lemma 5.16.} Suppose that \(B_{v, w, z}\) is a Bohr set of codimension \(k\) and rank \(t\), then the probability that a random vertical parallelogram has all its vertices in \(B\) is \(p^{-4k} \pm 4p^{k-t}\).

We effectively modify the proof of [9, Corollary 6.9]. To begin, we recap some notations from that paper.

\textbf{Definition 5.17.} For \(w \in G\), \(w\) is \(x\)-normal for a bi-affine map \(\beta : G \times G \to \mathbb{F}_p^k\) if the proportion of \(y\) such that \(\beta(w, y) = 0\) is exactly \(p^{-k}\). Similarly, \(h\) is \(y\)-normal for a bi-affine map \(\beta\) if the proportion of \(x\) such that \(\beta(x, h) = 0\) is exactly \(p^{-k}\).

We use this language to reformulate a special case of [9, Lemma 5.4] that we will be needing.

\textbf{Lemma 5.18.} Let \(\beta : G \times G \to \mathbb{F}_p^k\) be a bi-affine map of rank at least \(t\). For a uniformly random chosen \(h \in G\), we have that \(h\) is \(x\)-normal for \(\beta\) with probability at least \(1 - p^{k-t}\).

\textbf{Definition 5.19.} Let the vertical derivative \(\beta'\) of \(\beta\) be the function \(\beta'(x, h) = \beta(x, y + h) - \beta(x, y)\) and the mixed derivative \(\beta''\) of \(\beta\) be the function \(\beta''(w, h) = \beta'(x + w, h) - \beta'(x, h)\).

\textbf{Proof of Lemma 5.18.} First, if \(w\) is \(x\)-normal for \(\beta''\) then the probability that \(\beta''(w, h) = 0\) is \(p^{-k}\). Consequently, by Lemma 5.18, for a randomly chosen \((w, h)\) the probability that \(\beta''(w, h) = 0\) is \(p^{-k} \pm p^{k-t}\).

By Lemma 5.18 again, the probability that for a randomly chosen \(h\) we have that \(h\) is \(y\)-normal is at least \(1 - p^{k-t}\). Now, assume that we work with \(h\) which is \(y\)-normal. Note that \(\beta'(w, h) = 0\) rewrites as \(\beta'(x + w, h) - \beta'(x, h) = 0\) for every \(x\). Since \(h\) is \(y\)-normal, for a randomly chosen \(x\) the probability that \(\beta'(x, h) = 0\) is \(p^{-k}\). For such a choice of \(x\) we have that
\[
\beta'(x + w, h) = \beta'(x, h) = 0.
\]
(1)

Next, by Lemma 5.18 again the probability that both \(x\) and \(x + w\) are \(x\)-normal is at least \(1 - 2p^{k-t}\). If both \(x\) and \(x + w\) are \(x\)-normal then the probability that \(\beta(x, y) = z\) for a randomly chosen \(y\) is \(p^{-k}\) by Lemma 5.18. Similarly, the probability that \(\beta(x + w, y') = z\) for a randomly chosen \(y'\) is \(p^{-k}\). Since (1) rewrites as \(\beta(x, y) - \beta(x, y + h) = 0\) and \(\beta(x + w, y') - \beta(x + w, y' + h) = 0\) for all \(y, y'\), it follows that under the assumptions we have operating with, if \(y, y'\) satisfy \(\beta(x + w, y') = z\) and \(\beta(x, y) = z\) then we have that
\[
\beta(x, y) = \beta(x, y + h) = \beta(x + w, y') = \beta(x + w, y' + h) = z.
\]
Equivalently, the vertical parallelogram with vertices \((x, y), (x, y + h), (x + w, y'), (x + w, y' + h)\) lies in \(B_{v, w, z}\). Putting all the probability estimates together, we see that for randomly chosen \(x, y, y'\) this occurs with probability at least \(p^{-4k} \pm 4p^{k-t}\).\qed
In order to motivate the following argument, suppose for the moment that \( \tilde{\psi} : G \times G \to G \) and we were given the information that \( \tilde{\psi} \) agrees with \( T \) on an \((1 - \epsilon)\)-fraction of \( G \times G \), with the goal being to identify the bilinear part \( T_L \) of \( T \). Let the \( i,j \) entry of \( T_L \) be \( T_{ij} \) and let \( \{e_i\} \) be the standard basis vectors for \( G \). Call \((x,y)\) good if \( T(x,y) = \tilde{\psi}(x,y) \). Then note that by assumption, with probability \( 1 - 4\epsilon \) we have that all 4 points \((x + e_i, y + e_j), (x, y + e_j), (x + e_i, y), (x, y)\) are good, and we can recover

\[
T_{ij} = \tilde{\psi}(x + e_i, y + e_j) - \tilde{\psi}(x, y + e_j) - \tilde{\psi}(x + e_i, y) + \tilde{\psi}(x, y).
\]

What this suggests is that we can do a majority vote: sample a lot of points \((x,y)\) and return the most popular value of \( \tilde{\psi}(x + e_i, y + e_j) - \tilde{\psi}(x, y + e_j) - \tilde{\psi}(x + e_i, y) + \tilde{\psi}(x, y) \).

The issue with porting this scheme directly over into our setting is that while we could ensure that for any choice of \((x,y)\) that \((x + e_i, y + e_j), (x, y + e_j), (x + e_i, y), (x, y)\) lies in \( G \times G \), in our context the domain of \( \tilde{\psi} \) is \( B'' \) and this claim is no longer necessarily true. While \( B \) may not contain unit squares, we know via Lemma 5.16 that it (roughly) contains the expected number of random vertical parallelograms which allows us to run a modified argument of the above.

**bi-affine\((B, \psi)\):**

**Input** membership test for a Bohr set \( B \), query access to \( \tilde{\psi} : B'' \to G \)

**Output** explicit expression of a bi-affine map \( T_L \)

- Sample \( t \) elements from \( G \).
- For each choice of \( 1 \leq i, j \leq t \), do the following: sample \( 3r \) triples \((x, y, y')\) of \( G^3 \) via \( \text{sampler}(B, r, G^2) \). For each of these sampled points, test the membership of \((x + f_i, y + f_j), (x + y, f_j), (x + f_i, y'), (x, y)\) in \( B \). Only retain those triples for which all 4 elements lie in \( B \).
- Of the retained triples, compute \( \tilde{\psi}(x + f_i, y' + f_j) - \tilde{\psi}(x, y + f_j) - \tilde{\psi}(x + f_i, y') + \tilde{\psi}(x, y) \) and return the most popular value \( S_{ij} \).
- Note that \( T(f_i, f_j) \) is a linear combination of the entries \( T_{ij} \) of \( T_L \). With high probability, this system is overdetermined. Pass to a linearly independent subset of the \( \{f_i\} \) (via Gaussian elimination) and call these \( \ell_1, \ldots, \ell_n \). Suppose the most popular value (obtained from the previous step) corresponding to \( \ell_i \) and \( \ell_j \) is \( S'_{ij} \).
- Solve the linear system of \( \{S'_{ij} = T(f_i, f_j)\}_{i,j} \) for \( T_{ij} \) (via Gaussian elimination) and return \( T_L \).

**Proof of Theorem 5.16**. In **bi-affine** take \( t = n^2 + O(\log(\delta^{-1})) \) and let the sampled elements be \( \alpha_1, \ldots, \alpha_t \). Note that we can find a linearly independent subset of them in time \( O(t^3) = poly(n) \) via Gaussian elimination, so it suffices to check that with high probability \( \langle \alpha_1, \ldots, \alpha_t \rangle = G \). We will bound the probability that \( \langle \alpha_1, \ldots, \alpha_t \rangle \) span a subspace of dimension at most \( n - 1 \). The number of subspaces of \( G \) dimension \( n - 1 \) is given by \( O(\exp(n(n - 1))) \). The probability that a random \( \alpha_i \) lies in a specific \((n - 1)\)-dimensional subspace of \( G \) is \( p^{-1} \). Hence, we have that

\[
P[\dim\langle \alpha_1, \ldots, \alpha_t \rangle = n - 1] = p^{-t} \cdot \#\{\text{subspaces of dimension } n - 1\} = O(\exp(n(n - 1) - t))
\]

and for the choice of \( t \) this value is at most \( \delta/10 \). It follows that with probability \( 1 - \delta/10 \) we will be able to extract \( \ell_1, \ldots, \ell_n \) in the last step of **bi-affine**, where Gaussian elimination runs in time \( O(n^3) \). Recall that if \( \{x_i\} \) are linearly independent in \( X \) and \( \{y_j\} \) are linearly independent in \( Y \) then we have that \( \{x_i \otimes y_j\} \) are linearly independent in \( X \otimes Y \). By treating \( T_{ij} \) as \( e_i \otimes e_j \) it follows that by choosing linearly independent \( f_i \) that we get that \( T(f_i, f_j) \) are linearly independent in the variables \( T_{ij} \) and therefore we would be able to solve for \( T_{ij} \) via Gaussian elimination in the last step in time \( O(n^3) \).

Let \( \tilde{\delta} = \delta/(10t^3) \). In **bi-affine**, take \( r = O(p^{4k}(1/8 - \epsilon)^{-2} \log(\log(\delta^{-1}))) \). By Lemma 5.16, for each \( i, j \in [t] \) the probability that all of the elements \((x + f_i, y' + f_j), (x, y + f_j), (x + f_i, y'), (x, y)\), which
given that we sampled all of \( x, y, y', f_i, f_j \) randomly is therefore a random vertical parallelogram, lie in \( B \) is at least \( p^{-4\ell}/2 \). By our choice of parameters, we can ensure by Lemma 4.1 that for each choice of \( i, j \in [t] \), we have with probability at least \( 1 - \delta \) that at least \( O((1/8 - \epsilon)^{-2}\log(\delta^{-1})) \) samples are retained after testing for membership in \( B \).

Fix an \( i, j \), where our choice of \( \delta \) allows us to combine the estimates by a union bound at the end. Next, note that with probability \( 1 - 4\epsilon \) we have that \( T(f_i, f_j) = \psi(x + f_i, y' + f_j) - \psi(x, y + f_j) - \psi(x + f_i, y') + \tilde{\psi}(x, y) =: \rho_{ij}(x, y, y') \) since for each of the four elements \( (x + f_i, y' + f_j), (x, y + f_j), (x + f_i, y'), (x, y) \) we have with probability \( 1 - \epsilon \) that \( T(a, b) = \psi(a, b) \). Consider the step of majority vote in \( \text{bi-affine} \), where with high probability we are working with \( w = O((1/8 - \epsilon)^{-2}\log(\delta^{-1})) \) many samples. We have that \( 1[T(f_i, f_j) = \rho_{ij}(x, y, y')] \geq 1 - 4\epsilon \) so by Lemma 4.1 if we let \( W \) be the number of samples for which \( T(f_i, f_j) = \rho_{ij}(x, y, y') \) then

\[
P\left[ W > \frac{\mu}{2} \right] \geq 1 - e^{-\frac{\mu}{2(1-\epsilon)}} = 1 - \delta.
\]

In other words, for each \( i, j \in [t] \) with probability at least \( 1 - \tilde{\delta} \) we have that the most popular value \( S_{ij} \) coincides with \( T(f_i, f_j) \).

Using the union bound, we can combine our probability estimates to get the desired guarantees. It is also clear that each sub-step stays within acceptable run-time bounds.

**Theorem 5.20.** Let \( c, \delta > 0 \). Given a query access to a bounded \( f: G \to \mathbb{C} \) and an explicit description of a bi-affine map \( T: G \times G \to G \) such that \( \mathbb{E}_{a,b} |T(a,b)|^2 \geq c \), then there exists an algorithm \( \text{find-cubic} \) that makes \( O(\text{poly}(n, 1/c, \log(1/\delta))) \) queries to \( f \) and with probability \( 1 - \delta \) outputs a cubic \( \kappa \) with the guarantee that \( |\mathbb{E}_x f(x) \omega^{\kappa(x)}| \geq \text{quasi-poly}(c) \).

Let the bilinear part of \( T \) be \( T^L \) and and define \( \tau(x, y, z) = T^L(x, y) \cdot z \). Let \( \kappa(x) = \tau(x, x, x) \).

**find-cubic(T,f):**

**Input** explicit expression of a bi-affine map \( T \), query access to \( f: G \to \mathbb{C} \)

**Output** a cubic polynomial \( \kappa \)

- Using the formulas as described above, we can obtain \( \kappa(x) \). In turn this provides us with query access to \( g(x) \).
- Run \( \text{find-quadratic}(g) \) and let the output be \( q(x) \).
- Return \( \kappa + q \).

**Proof.** Given an explicit description to \( T^L \), we can get an explicit representation to the trilinear form \( \tau(a, b, c) = T^L(a, b) \cdot c \). In turn we are able to obtain an explicit description of \( \lambda(x) = \tau(x, x, x) \). We can combine Lemma 11.1 with the remarks at the end of section 10 of [9] to obtain that \( g(x) = f(x) \omega^{-\kappa(x)} \) has large \( U^3 \) norm: \( \|g\|_{U^3} = \Omega(\alpha) \). We finish by invoking Theorem 1.2. The algorithm guarantees that with probability at least \( 1 - \delta \) it outputs a quadratic form \( q \) with \( |\mathbb{E}_x f(x) \omega^{(\lambda+q)(x)}| = |\mathbb{E}_x g(x) \omega^{q(x)}| \geq \text{quasi-poly}(\alpha) \). Since \( \kappa = \lambda + q \) is a cubic, we indeed obtain the guarantees we claim.

**5.1 Putting everything together**

In this section, we will see how the theorems we have proven so far fit together.

**Proof of Theorem 5.3** Beginning with query access to \( f \) with \( \|f\|_{U^3} \geq \epsilon \), apply Theorem 5.1 and run the corresponding algorithm to get \( \text{member-A} \) such that with probability \( 1 - \delta/8 \) outputs 1 if \( (a, b) \in A_1 \) and 0 if \( (a, b) \notin A_2 \). We also have \( \text{query-phi} \) which in \( O(\text{poly}(n, \epsilon^{-1}, \log(\delta^{-1})) \) queries to \( f \) outputs \( \phi(a, b) \) with the desired properties with probability \( 1 - \delta/8 \).

Now, we want to pass from the implicit 1% structure on \( \phi \) to 99% structure for \( \phi \). Using \( \text{member-A} \) as well as \( \text{query-phi} \) as primitives, apply Theorem 5.6 to get a membership tester \( \text{member-A-tilde} \) for
a subset $\tilde{A} \subset A$ for which $\phi|_{\tilde{A}}(G \times \{b\})$ is a Freiman homomorphism. Since the density of $\tilde{A}$ is $\Omega(\text{poly}(\epsilon))$, it follows by [3, Lemma 3.7] that $\phi$ respects a $\Omega(\text{poly}(\epsilon))$ fraction of 4-arrangements in $A_1$, which in turn implies by [9, Corollary 3.9] that $\phi$ respects a $\Omega(\text{poly}(\epsilon))$ fraction of second-order 4-arrangements in $A_1$. Now, using member-$A$-tilde as a primitive in Theorem 5.15 we get a membership tester member-A-prime for a subset $A' \subset \tilde{A}$ such that $A'$ contains poly($\eta^{-1}$, $\eta^{-1}$)$|G|^32$ second-order 4-arrangements and $\phi$ respects a $1 - \eta$ fraction of these. We will set $\eta = 10^{-10}$. A back-of-the-envelope calculation will show that this choice of $\eta$ is sufficiently small for future use.

It is more convenient to now work with $\psi = \| \phi \|$. By [9, Lemma 4.1], we have that $\psi$ is a $(1 - \eta)$-bihomomorphism with respect to $\mu = \| 1_{A'} \|$. Before proceeding further, we describe how to:

- Obtain query access to $\psi$ given query access to $\phi$.
- Estimate $\mu$ given member-A-prime.

Note that $\psi$ can be interpreted as a probability distribution given by $\phi(P)$ for a uniformly random vertical parallelogram $P$ with width $w$ and height $h$. Here, if $P = ((x, y), (x, y + h), (x + w, y'), (x + w, y' + h))$, then $\phi(P) = \phi(x, y)\phi(x, y + h)^*\phi(x + w, y')^*\phi(x + w, y' + h)$. This allows us to have a query access to $\psi$.

\begin{verbatim}
psi(\phi, w, h):
Input query access to $\phi$: $G \times G \to A, w, h \in G$
Output estimate of $\psi(w, h)$
- Sample 3$r$ values $\{x_i\}_{i=1}^r$, $\{y_i\}_{i=1}^r$, $\{y'_i\}_{i=1}^r$ from $G$ such that $(x_i, y_i), (x_i, y_i + h), (x_i + w, y'_i), (x_i + w, y'_i + h) \in A'$. Then return
  \[
  \frac{1}{r} \sum_{i=1}^r \frac{\phi(x, y)\phi(x, y + h)^*\phi(x + w, y')^*\phi(x + w, y' + h)}.\]

The membership tester member-A-prime implies we have query access to $1_{A'}$.

Next, we recall a sub-routine that we first introduced in the previous section.

box(f, w, h):
- Sample 3$r$ values $\{x_i\}_{i=1}^r$, $\{y_i\}_{i=1}^r$, $\{y'_i\}_{i=1}^r$ and output
  \[
  \frac{1}{r} \sum_{i=1}^r f(x_i, y_i)f(x_i, y_i + h)f(x_i + w, y'_i)f(x_i + w, y'_i + h).\]

Assuming that the output of member-A-prime does satisfy the desired guarantees, by taking $r = O(\text{poly}(\tau^{-1}, \log(\xi^{-1})))$ it follows from Lemma 1.1 that box$(1_{A'}, w, h)$ effectively with probability at least $1 - \nu$ gives query access to $\mu'(\nu, \omega)$ such that $\|\mu - \mu'(\nu, \omega)\|_\infty \leq \omega$.

Next, we will obtain some structure on the underlying set first by applying bilinear Bogolyubov and then passing to a suitable high-rank bilinear Bohr set. Using box$(1_{A'}, w, h)$ as a primitive in Theorem 5.8 with $O(\exp(\text{quasi-poly}(\xi^{-1}) \cdot \text{poly}(n, \log(\delta^{-1})))$ queries to box$(1_{A'}, w, h)$ we have with probability $1 - \delta$ the output of a bi-affine map $\beta: G \times G \to \mathbb{R}^k$ with $\|F - \text{proj}_\beta F\|_2 \leq \xi$. In particular, note that Theorem 5.8 gives an explicit description of $\beta$. We now pass to a high rank bilinear Bohr set. Let $t = \lfloor 60k + 9 \log(\eta^{-1}) \rfloor$. Apply Theorem 5.10 using the explicit description of $\beta$ to get a basis for $X_0, Y_0$ with $\dim X_0, \dim Y_0 \leq tk$ such that the corresponding Bohr decomposition has rank at least $t$. To pass down to one specific bilinear Bohr set, we will apply Theorem 5.12. In the specific context of our application we are specifying to the following:

- $\phi := \psi/\|\psi\|_1$ where as we recall $\psi = \| \phi \|$.
- $\mu := \| 1_{A'} \|$.
- Write $B_{1_{A'}}((x, y))$ for the Bohr set that $(x, y)$ lies in; then $\xi(x, y) := \mathbb{E}_{(x, y) \in B_{1_{A'}}((x, y))} 1_{A'}(\tilde{x}, \tilde{y}) - \text{proj}_\beta \| 1_{A'}(\tilde{x}, \tilde{y}) \|^2$. 

34
We need to check that we have the primitives that Theorem 5.12 requires. First, we have sample access to the probability distribution via \( \psi \). By using box, it follows via Lemma 4.1 we can approximate \( \|1_{A'} \) to arbitrary additive precision with arbitrarily high probability. Lastly, since \( \xi \) is an expected value, we can approximate it as follows.

\[
\text{xi}(x,y):
\]
- We can iterate through all \( O(\eta^{-1}) \) choices for the triple \((v, w, z)\) to identify \( B := B_{v,w,z} \).
- Execute sampler\((B, t, G^2)\) and let its output be \((a_1, b_1), \ldots, (a_t, b_t)\).
- Return \( t^{-1} \sum_{i=1}^{t} \left( \|1_{A'}(a_i, b_i) - \text{proj}_\beta \|1_{A'}(a_i, b_i) \right)^2 \) where we use box to estimate \( \|1_{A'} \).

If we want to approximate \( \xi(x, y) \) within an additive error of \( \omega \) with confidence at least \( 1 - \nu \), we can pick \( t = O(\text{poly}(\omega, \log(\nu^{-1})) \) and estimate \( \|1_{A'} \) within an additive error of \( \omega/2 \) with confidence at least \( 1 - (\delta/(2t)) \). Particularly, within the bounds on our runtime, we can ensure that the outcomes \( B := B_{v,w,z} \) of high-rk-bohr-set in Theorem 5.12 has the guarantees that \( \mu(x) \geq |\mu|/4 \) and \( \xi(x) \leq 3/2\gamma^{-1}\xi \) for \( x \in B \).

The next step is to run a majority vote style argument to pass from 99% structure to 100% structure. Apply Theorem 5.14 using \( \psi \) to get query access to \( \tilde{\psi} \) and also the explicit description for the Bohr set \( B \) from Theorem 5.12. The output of Theorem 5.14 gives us query access \( \text{query-tilde-psi} \) to \( \tilde{\psi} \) with the property that \( d(\psi(w, h), \tilde{\psi}(w, h)) \leq 64n\eta^{-3k} \). The existence of such a \( \tilde{\psi} \) follows from [9, Lemma 6.21]. We will first show that such a \( \tilde{\psi} \) is unique. We will prove that there cannot be \( \tilde{\psi}_1 \neq \tilde{\psi}_2 \) such that for both \( i \) we have that

\[
d(\psi(w, h), \delta_{\tilde{\psi}_i(w, h)}) \leq 64n\eta^{-3k}.
\]

Indeed, note by applying the triangle inequality for \( d \) on \( d(\psi(w, h), \delta_{\tilde{\psi}_1(w, h)}) \) we get that

\[
d(\psi(w, h), \psi(w, h)) \leq 2\epsilon.
\]

Here, \( \epsilon = 64n\eta^{-3k} \). This in turn implies by Hölder’s that

\[
1 - 2\epsilon \leq \|\psi(w, h)\|^2 \leq \|\psi(w, h)\|_1 \|\psi(w, h)\|_\infty = \|\psi(w, h)\|_\infty.
\]

So it follows that for \( \tilde{\psi}_2 \) we must have that \( \langle \delta_{\tilde{\psi}_2(w, h)}, \psi(w, h) \rangle \leq 2\epsilon \) and so \( d(\psi(w, h), \delta_{\tilde{\psi}_2(w, h)}) \geq 1 - 2\epsilon > \epsilon \) since \( \epsilon < 1/3 \).

Specifically, [9, Lemma 6.21] constructs such a \( \tilde{\psi} \) by attempting to approximate \( \psi \) with a (single-valued) function that is almost additive in each variable separately. The \( \tilde{\psi} \) constructed satisfies the following three simultaneous properties:

1. \( \tilde{\psi}(w_1, h) + \tilde{\psi}(w_2, h) = \tilde{\psi}(w_1 + w_2, h) \) for all triples \((w_1, w_2, h)\) outside a set of density at most \( \delta \).

2. \( \tilde{\psi}(w, h_1) + \tilde{\psi}(w, h_2) = \tilde{\psi}(w, h_1 + h_2) \) for all triples \((w, w_1, h)\) outside a set of density at most \( \delta \).

3. \( d(\psi(w, h), \delta_{\tilde{\psi}(w, h)}) \leq 64n\eta^{-3k} \) for all \((w, h)\) outside a set of density at most \( \delta \).

Here we retain the parameters as defined in Theorem 5.14. As we have show that such a \( \tilde{\psi} \), if it exists, is unique, it follows that the \( \tilde{\psi} \) given by \( \text{query-tilde-psi} \) also satisfies the properties (1) and (2). Note that there is still a portion of \( B'' \) on which \( \tilde{\psi} \) is not a fortiori additive in each of the variables.

In order to ensure that we get a function that is additive in both variables on the entire of \( B'' \), Gowers and Milčević takes this \( \tilde{\psi} \) of [9, Lemma 6.21] and massages it further in [9, Section 6.6]. They define a set \( B_4 \subseteq B'' \) in order to facilitate their argument of extending \( \tilde{\psi} \) to a function \( \psi_2 \) that is defined on all of \( B'' \) and additive in both variables. If we examine their argument in [9, Lemma 6.24] closely, they effectively obtain a set \( B_5 \subseteq B'' \) such that \( B_5 \) has density \( 1 - 16\delta^{3/4} \) in \( B'' \) such that \( \tilde{\psi}|_{B_5} \) can be extended uniquely
to a function $\bar{\psi}_2$ defined on all of $B''$ and still additive in each of the variables. Next, they apply Corollary 7.8 in order to extend $\bar{\psi}_2$ to a bi-affine map $\phi_1 : G^2 \to G$ that has agreement with $\psi$ on a exp-exp quasi-poly$(\epsilon^{-1})$ fraction of $G^2$. Given the unique existence of such a $\bar{\psi}_2$ from which a bi-affine map $\phi_1$ is constructed, we can find $\phi_1$ algorithmically via a majority vote. Indeed, note that $\phi_1$ and $\text{query-tilde-psi}$ agrees on $B_5$ which has density $1 - 16\delta^{1/4}$, so we can apply Theorem 5.15 to recover such a bi-affine map. Say the bi-affine map output of this application of Theorem 5.15 is $T$.

To finish up, we need to “symmetrize” and “anti-differentiate” to recover the correlating cubic. To that end, using the description of $T$ provided by Theorem 5.15 as input in Theorem 5.20, we can find a cubic $\kappa$ such that

$$|\mathbb{E}_x f(x) \omega^{-P(x)}| > \eta$$

where $\eta^{-1} = \exp\exp\text{quasi-poly}(\epsilon^{-1})$, as desired.

For the runtime guarantees, it suffices to observe that each individual step of the algorithm stays within $O(\text{poly}(n, \eta, \log(\delta^{-1})))$ queries to $f$.

Lastly, we prove the equivalence between the formulation of the cubic Goldreich-Levin as an algorithmic $U^4$ inverse theorem à la Theorem 1.3 and the formulation in terms of decoding Reed-Muller codes à la Theorem 1.4.

**Equivalence of Theorem 1.3 and Theorem 1.4** First, observe that it suffices to establish that $P[f(x) = P(x)] \geq (1/p) + \epsilon$ implies that $\|\omega^f\|_{U^4} \geq \epsilon$, at which point we can invoke the $U^4$ inverse theorem (Theorem 1.2) to get the existence of a polynomial $Q$ satisfying the conditions as stated.

Now, we note that

$$\left| \sum_{t \neq 0} \omega^{t(f(x) - P(x))} \right| = \begin{cases} p - 1 & \text{if } f(x) = P(x) \\ 1 & \text{otherwise} \end{cases}$$

where in the case $f(x) \neq P(x)$ we recall that $\sum_i \omega^i = 0$. This implies that

$$\left| \sum_{t \neq 0} \mathbb{E}_x \omega^{t(f(x) - P(x))} \right| \geq (p - 1) \cdot P[f(x) = P(x)] - (1 - P[f(x) = P(x)]) > pe.$$  

By pigeonhole principle, there exists some $t \neq 0$ such that

$$|\mathbb{E}_x \omega^{t(f(x) - P(x))}| \geq \frac{p}{p - 1} \epsilon \geq \epsilon.$$  

This in particular implies that

$$\epsilon \leq |\mathbb{E}_x \omega^{f - tP}| \leq \|\omega^{f - tP}\|_{U^4} = \|\omega^f\|_{U^4} = \|\omega^f\|_{U^4}$$

where for the second inequality we used Gowers-Cauchy-Schwarz, the penultimate equality follows from the discrete derivative definition of $U^4$ norms which causes $P$ to vanish and the final equality comes from symmetry of the $p$th roots of unity. This last inequality establishes the desired. □

6 Improving the quantitative bounds

One of the major open problems in the field of additive combinatorics is that of the polynomial Freiman-Ruzsa conjecture. Recall that the Freiman-Ruzsa theorem was one of the additive combinatorics tools that we used “under the hood” in order to establish our algorithmic $U^4$ inverse theorem. Roughly speaking, the Freiman-Ruzsa theorem is a structural theorem that gives a description of sets $A$ with small doubling, that is, satisfying $|A + A| \leq K|A|$.
Theorem 6.1 (Freiman-Ruzsa Theorem). Let $A \subset \mathbb{F}_p^n$ be such that $|A + A| \leq K|A|$. Then $A$ is contained in a subspace of size at most $\exp(O(K^C))|A|$.

However, as can be observed, we incur an exponential loss in $K$ as we pass to the conclusion in the structure theorem. We might want to avoid such an exponential loss, but it can be seen that in such a formulation the exponential loss is inevitable. As an example, we can take $A = V \cup \{p_1, \ldots, p_k\}$ such that span($p_1, \ldots, p_k$) $\cap V = \{0\}$. In this case, $A$ has small doubling but the smallest subspace containing $A$ has size on the order of $2^k|A|$. The polynomial Freiman-Ruzsa conjecture instead asks if we can get a polynomial dependence if we instead require the subspace to cover not the entire of $A$ but just a large fraction of $A$.

Theorem 6.2 (Polynomial Freiman-Ruzsa Conjecture). Let $A \subset \mathbb{F}_p^n$ be such that $|A + A| \leq K|A|$. Then there exists an affine subspace $V \subset \mathbb{F}_p^n$ such that $|V| \leq |A|$ and $|V \cap A| \geq (1/K)^{O(1)}|A|$.

The best bounds in this direction are due to Sanders [17], who obtained a quasi-polynomial dependence in the parameters.

Theorem 6.3 ([17]). Let $A \subset \mathbb{F}_p^n$ be such that $|A + A| \leq K|A|$. Then there exists an affine subspace $V \subset \mathbb{F}_p^n$ such that $|V| \leq |A|$ and $|V \cap A| \geq K^{-O(\log^3 K)}|A|$.

It was proven independently by Green and Tao [12] as well as Lovett [15] that polynomial bounds in the $U^3$ inverse theorem is equivalent to the polynomial Freiman-Ruzsa Conjecture. Motivated by this result, we conjecture that the same is true in the $U^4$ setting.

Conjecture 6.4. The polynomial $U^4$ inverse conjecture and polynomial Freiman-Ruzsa conjecture are equivalent.

To that end, it seems reasonable to try to improve the quantitative bounds of Gowers and Miličević, who get a doubly exponential of quasipolynomial dependence in $\eta$ to just a quasipolynomial dependence. This would then match the best known bound by Sanders in terms of Freiman-Ruzsa, and would be evidence for believing the truth of Conjecture 6.4. We make progress in this direction by removing one of the two exponents in the quantitative bounds of Gowers and Miličević.

One of the exponents arises due to a technical Fourier analytic argument in [9] Theorem 4.15, which we restate here for the purposes of having a comparison with our improved version later.

Theorem 6.5 ([9]). For every $\zeta > 0$ there exists a positive integer $k$ with the following property. Let $f : G^2 \to \mathbb{C}$ be any bounded function. Then there is a bi-affine map $\beta : G^2 \to \mathbb{F}_p^k$ such that for $F = \|f$ we have the approximation $\|F - \text{proj}_\beta F\|_2 \leq \zeta$. Moreover, $k$ can be taken to be $4m^3A^m/\zeta^2$ where $m = \exp(2^{O(m)}(\log(\zeta^{-1}) + \log p)^6)$.

We briefly describe where the exponents arise in this step. In order to approximate $F$, it is reasonable to just approximate it by its large Fourier spectrum; precisely, we may consider $F'(x,y) = \sum_{r : r \in \text{Spec}_\gamma(F_{x,y})} \wedge_r \omega^{x,r}$. In order to get the affine structure $\beta$, it follows that we would want to cover the large Fourier spectrum $\bigcup_{\gamma} \text{Spec}_\gamma(F_{x,y})$ by affine maps. Gowers and Miličević does this in [9] Lemma 4.10]. The issue approximating $F(x,y)$ with $F'(x,y) = \sum_{i=1}^m \wedge_{T_i y} \omega^{x,T_i y}$ is that it is possible that $T_i y = T_j y$. As we have discussed before, Gowers and Miličević considers $u_i$ where

$$u_i(y) = \begin{cases} 0, & T_j y = T_i y \text{ for some } j < i, \\ \wedge_{T_i y}, & \text{otherwise} \end{cases}$$

for each $i$ so that we can try to approximate $F$ using $G(x,y) = \sum_{i=1}^m u_i(y) \omega^{x,T_i y}$. In order to show that $G(x,y)$ is a good approximation, it turns out that we would need to get a handle on $\|\tilde{u}_i\|_1$, which is where an exponential arises.

Gowers and Miličević write $u_i = v_i(y)1(y = \min\{j : T_j y = T_i y\}$ where $v_i(y) = \mathbb{E}_x F(x,y) \omega^{x,T_i y}$. Since $\|v_i(y)\|_1 \leq 1$ by [9] Lemma 4.11], it suffices to estimate $\|1(y = \min\{j : T_j y = T_i y\})\|_1$. To that end, they
write \( 1(y = \min\{j : T_j y = T_i y\}\) as a \(\pm 1\) combination of indicators of subspaces, and the \(L^1\) norm of each of these subspaces would be bounded by 1 so that the desired \(L^1\) norm would be bounded by the number of such subspaces we introduce. However, they do this partitioning directly using principle of inclusion and exclusion, which produces \(2^n\) subspaces and causes an exponential to arise; precisely, they write
\[
1(y = \min\{j : T_j y = T_i y\}\) = \prod_{k < i}(1 - 1(y \in \{j : T_j y = T_i y\})).
\]

We aim to do a more careful analysis of \(1(y = \min\{j : T_j y = T_i y\}\). We can think of our end goal as effectively finding some affine maps such that for all \(i\) we have that \(1(y = \min\{j : T_j y = T_i y\}\) is constant on the level sets of these affine maps. The intuition for what we want to do is that we can “pre-partition” our ambient space in a way to make many \(T_i - T_j\) have “high rank”. This reduces the amount of “overlapping space” that we would need to do the PIE argument on, and would give better bounds.

As an illustration of this idea, suppose that \(\ker(T_1 - T_2) = \{x_1 = 0, x_2 = 0\}\). Our goal is to introduce some additional affine forms such that the indicator this subspace is constant on the corresponding level sets. Using Gowers and Milicić’s argument, we would write \(1(x_1 = 0, x_2 = 0) = p^{-2} \sum_{s,t \in \mathbb{F}_p} \omega^{sT_1 x + t x_2}\). We partition the space by adding in the \(p^2\) forms \(\{tx_1 + sx_2\}_{s,t \in \mathbb{F}_p}\). However, note that \(tx_1 + sx_2 \in \text{span}\{x_1, x_2\}\) so if we have “pre-partitioned” our space by introducing the forms \(\{x_1, x_2\}\) then \(\ker(T_1 - T_2)\) would automatically be measurable and we would have saved many redundant forms.

**Theorem 6.6.** For every \(\zeta > 0\) there exists a positive integer \(k\) with the following property. Let \(G = \mathbb{F}_p^n\) and let \(f : G \times G \to \mathbb{C}\) be any bounded function. Then there is a map \(\beta : G^2 \to \mathbb{F}_p^k\) where each coordinate \(\beta_i : G^2 \to \mathbb{F}_p\) is either bilinear in both variables or linear in one of the two variables, such that, writing \(F\) for the mixed convolution \(\bigotimes f\) and \(\text{proj}_j\) for the averaging projection on to the level sets of \(\beta\), we have the approximation \(\|F - \text{proj}_j F\|_2 \leq \zeta\). Moreover, \(k\) can be taken to be \(O(m^2(\log m + \log(\zeta^{-1}))\) where \(m = \exp(2^{O(\log(\zeta^{-1}) + \log p)})\).

**Proof.** Let \(\epsilon, \gamma, r > 0\) be constants to be chosen later. Define \(g = f \ast \hat{f}\) (so that \(F = g \iff g\)). By [Lemma 4.10](#), there exist affine maps \(T_1, \ldots, T_m : \mathbb{F}_p^n \to \mathbb{F}_p^n\) such that for all but at most \(\epsilon |G|\) values of \(h\), the \(\gamma\)-large spectrum of \(g_{\text{lin}}(h)\) (meaning here the set of \(u\) such that \(|\hat{g}_{\text{lin}}(u)|^2 \geq \gamma\)) is contained in the set \(\{T_1 h, \ldots, T_m h\}\) where \(m = \exp(2^{O(\log(\zeta^{-1}) + \log p)})\).

Let \(\mathcal{B} = (L_1, \ldots, L_i)\) be a list of linear forms such that if \(E[1(T_i y = T_j y)] \geq p^{-r}\), then \(1(T_i y = T_j y)\) is \(\mathcal{B}\)-measurable. Note that one can choose \(t \leq m^{2r}\). This is because for fixed \(i, j\), if we write \(U_{ij} = \ker(T_i - T_j)\) then \(\text{dim} \bigcup_{i,j} U_{ij} \leq r\) and we can add to \(\mathcal{B}\) a basis for \(U_{ij}^\perp\). Doing this for all possible pairs of \(i, j\), we see that \(t \leq m^{2r}\).

For \(1 \leq i \leq m\), define \(u_i(y) = 1(T_i y \notin \{T_1 y, \ldots, T_{i-1} y\}\).

**Claim 6.7.** \(\|u_i - \mathbb{E}(u_i | \mathcal{B})\|_2 \leq m \cdot p^{-r}\).

**Proof.** We write \(\mathcal{I} = \{(i, j) : \mathbb{P}(T_i = T_j) \geq p^{-r}\}\) to keep track of the pairs of affine maps which have a lot of overlap. Consider \(v_i(y) = 1(T_i y \neq T_j y, \forall j < i \text{ s.t. } (j, i) \in \mathcal{I})\) corresponding to the “small” overlaps. Then we have that
\[
\|v_i - \mathbb{E}(v_i | \mathcal{B})\|_2 \leq \|v_i - v_i'\|_2 \leq \mathbb{P}(T_i y = T_j y \text{ for some } j < i \text{ s.t. } (j, i) \notin \mathcal{I}) \leq m \cdot p^{-r}.
\]

Define \(H(x, y) = \sum_{i=1}^n \hat{F}_{\gamma y}(T_i y) u_i'(y) \omega^{x \cdot T_i y}\), where for each \(y\) we have that \(\hat{H}_{\gamma y}\) is the restriction of \(\hat{F}_{\gamma y}\) to the set \(\{T_1 y, \ldots, T_m y\}\). Because \(F\) is defined by a convolution, we have that \(\hat{F}_{\gamma y} = |\hat{g}_{\gamma y}|^2\). So if \(y\) is such that \(\text{Spec}_{\gamma}(g_{\gamma y}) \subset \{T_1 y, \ldots, T_m y\}\) then by Hölder’s we have that
\[
\|\hat{H}_{\gamma y} - \hat{F}_{\gamma y}\|_2 \leq \|\hat{H}_{\gamma y} - \hat{F}_{\gamma y}\|_1 \|\hat{H}_{\gamma y} - \hat{F}_{\gamma y}\|_\infty \leq \gamma.
\]

For the \(y\) such that \(\text{Spec}_{\gamma}(g_{\gamma y}) \not\subset \{T_1 y, \ldots, T_m y\}\) we will use the
naïve bound that \( \|H - \hat{F}\|_2^2 \leq 1 \). Taken together, since the density of the latter is at most \( \epsilon \), this implies that \( \|H - F\|_2^2 \leq \epsilon + \gamma \).

Let \( \mathcal{Y} \) be \( \mathcal{Y}_1 = (T_1, \ldots, T_m) \) refined by \( \mathcal{B} \). Furthermore, let \( \beta: G^2 \to \mathbb{R}_p^k \) be the natural map corresponding to \( \mathcal{Y} \) where we concatenate the maps coordinate-wise. Note that here we can take \( k \leq m^2r + m \).

For the next part of the argument, write

\[
H(x, y) = \sum_{i=1}^{n} \frac{F_{i,y}(T_i, y)}{\omega_{i,T,y}} u'_i (y) \omega^{x,T,y}.
\]

We have that (III) is \( \mathcal{Y}_1 \)-measurable and therefore \( \mathcal{Y}' \)-measurable. One way to think about Claim 6.7 is that (II) is “almost \( \mathcal{B} \)-measurable” and so “almost \( \mathcal{Y}' \)-measurable”. For (I), write \( w_i(y) = \hat{F}_{i,y}(T_i, y) \) for simplicity of notation and note that because \( F \) is 1-bounded, it follows that \( |w_i(y) - \mathbb{E}(w_i | \mathcal{B}')(y)| \leq 1 \) so that \( \|w_i - \mathbb{E}(w_i | \mathcal{B}')\|_2 \leq 1 \).

Using the fact that \( |(u'_i(y) - \mathbb{E}(u'_i | \mathcal{B}))(w_i(y) - \mathbb{E}(w_i | \mathcal{B}_2)(y)) \omega^{x,T,y}| \leq 1 \) and the Cauchy-Schwarz inequality, we can write

\[
\|H - \mathbb{E}(H | \mathcal{Y}')\|_2^2 = \mathbb{E}_{x,y} \left| \sum_{i=1}^{m} (u'_i(y) - \mathbb{E}(u'_i | \mathcal{Y}))(w_i(y) - \mathbb{E}(w_i | \mathcal{B}_2)(y)) \omega^{x,T,y} \right|^2
\]

\[
\leq \mathbb{E}_{x,y} \left| \sum_{i=1}^{m} (u'_i(y) - \mathbb{E}(u'_i | \mathcal{B}))(w_i(y) - \mathbb{E}(w_i | \mathcal{B}_2)(y)) \omega^{x,T,y} \right|^2
\]

\[
\leq \mathbb{E}_{y} \left| \sum_{i=1}^{m} (u'_i(y) - \mathbb{E}(u'_i | \mathcal{B}))(w_i(y) - \mathbb{E}(w_i | \mathcal{B}_2)(y)) \right|
\]

\[
\leq \sum_{i=1}^{m} \|u'_i - \mathbb{E}(u'_i | \mathcal{B})\|_2 \|w_i - \mathbb{E}(w_i | \mathcal{B}_2)\|_2
\]

\[
\leq m^{1/2} \cdot p^{-r/2}
\]

Combining the estimates above via the triangle inequality and taking \( \epsilon = \gamma = \xi^2/8 \) and \( r = 2(\log m + \log \xi - 1) \), it follows that

\[
\|F - \text{proj}_\beta F\|_2 \leq \|F - \mathbb{E}(H | \mathcal{Y}')\|_2 \leq \|F - H\|_2 + \|H - \mathbb{E}(H | \mathcal{Y}')\|_2 \leq \xi/2 + \xi/2 = \xi
\]

since the closest approximation to \( F \) in \( L^2 \) by a function that is \( \mathcal{Y}' \)-measurable is \( \text{proj}_\beta F \). Furthermore, by our choice of parameters, we have that for the corresponding \( \beta: G \to \mathbb{R}_p^k \) we can take \( k = O(m^2(\log m + \log(\xi^{-1}))) \), as desired.

\[\square\]

In comparison with Theorem 6.5 [9, Theorem 4.15], note the codimension of the bi-affine map \( \beta \) that we obtain is indeed smaller by one exponent.

**Proof of Theorem 7.6** In the proof of [9], whenever Theorem 6.5 [9, Theorem 4.15] is used, replace such a usage with Theorem 6.6 instead.  \[\square\]

**References**


