






Testing Isomorphism of Boolean Functions over Finite Abelian Groups

Swarnalipa Datta  




Indian Statistical Institute, Kolkata, India

Arijit Ghosh   

Indian Statistical Institute, Kolkata, India

Chandrima Kayal   

Université Paris Cité, CNRS, IRIF, France

Manaswi Paraashar   

University of Copenhagen, Denmark

Manmatha Roy   

Indian Statistical Institute, Kolkata, India

Abstract

Let f and g be Boolean functions over a finite Abelian group \mathcal{G} , where g is fully known and f is accessible via queries; that is, given any $x \in \mathcal{G}$, we can obtain the value $f(x)$. We study the problem of tolerant isomorphism testing: given parameters $\epsilon \geq 0$ and $\tau > 0$, the goal is to determine, using as few queries as possible, whether there exists an automorphism σ of \mathcal{G} such that the fractional Hamming distance between $f \circ \sigma$ and g is at most ϵ , or whether for every automorphism σ , the distance is at least $\epsilon + \tau$.

We design an efficient tolerant property testing algorithm for this problem over finite Abelian groups with constant exponent. The exponent of a finite group refers to the largest order of any element in the group. The query complexity of our algorithm is polynomial in s and $1/\tau$, where s bounds the spectral norm of the function g , and τ is the tolerance parameter. In addition, we present an improved algorithm in the case where g is Fourier sparse, meaning that its Fourier expansion contains only a small number of nonzero coefficients.

Our approach draws on key ideas from Abelian group theory and Fourier analysis, including the annihilator of a subgroup, Pontryagin duality, and a pseudo inner product defined over finite Abelian groups. We believe that these techniques will be useful more broadly in the design of property testing algorithms.

2012 ACM Subject Classification Theory of computation \rightarrow Boolean function learning

Keywords and phrases Analysis of Boolean functions, Abelian groups, Automorphism group, Function isomorphism, Spectral norm

Digital Object Identifier 10.4230/LIPIcs.APPROX/RANDOM.2025.66

Category RANDOM

Related Version *Full Version*: <https://arxiv.org/abs/2507.07654> [20]

Funding *Arijit Ghosh*: A. G. is partially supported by the Science & Engineering Research Board of the DST, India, through the MATRICS grant MTR/2023/001527.

Chandrima Kayal: C.K. is supported by French PEPR integrated project EPiQ (ANR-22-PETQ-0007).

Manaswi Paraashar: M.P. is supported by ERC grant (QInteract, Grant Agreement No 101078107).



© Swarnalipa Datta, Arijit Ghosh, Chandrima Kayal, Manaswi Paraashar, and Manmatha Roy; licensed under Creative Commons License CC-BY 4.0

Approximation, Randomization, and Combinatorial Optimization. Algorithms and Techniques (APPROX/RANDOM 2025).

Editors: Alina Ene and Eshan Chattopadhyay; Article No. 66; pp. 66:1–66:22



Leibniz International Proceedings in Informatics

Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

1 Introduction

Let \mathcal{G} be a finite Abelian group, and let $\widehat{\mathcal{G}}$ be its dual group, consisting of all characters, that is, homomorphisms from \mathcal{G} to $\mathbb{C}^\times := \{x \in \mathbb{C} \mid |x| = 1\}$. The Fourier coefficient for a function $f : \mathcal{G} \rightarrow \mathbb{C}$ corresponding to a character $\chi \in \widehat{\mathcal{G}}$ is denoted by $\widehat{f}(\chi)$ ¹, and the *spectral norm* of f is defined as

$$\|\widehat{f}\|_1 := \sum_{\chi \in \widehat{\mathcal{G}}} |\widehat{f}(\chi)|.$$

For more details refer to Section 2. The spectral norm is a key parameter in the analysis of Boolean functions, with wide-ranging applications in property testing, learning theory, cryptography, pseudorandomness, and quantum computing [43, 38, 46, 28, 44, 39, 13, 42, 27, 41, 45, 4]. Moreover, it plays an important role in understanding the coset complexity of subsets of Abelian groups [17, 31, 32, 15]. We will show that, for the isomorphism testing of Boolean functions over finite Abelian groups, the query complexity is essentially controlled by the spectral norm.

The area of property testing is concerned with determining whether a given object satisfies a fixed property or is “far” from satisfying it, where “far” is typically defined using a suitable distance measure. Over nearly three decades of research, *symmetry* has played a crucial role in property testing, appearing in various forms.

To illustrate the setup and the role of symmetry, we consider property testing of Boolean functions, which are functions of the form $f : \{-1, 1\}^n \rightarrow \{-1, 1\}$. The *fractional Hamming distance* between two Boolean functions f and g on n -bit inputs is defined as

$$\delta(f, g) = \Pr_{x \in \{-1, 1\}^n} [f(x) \neq g(x)],$$

that is, the fraction of inputs on which they differ.

Let S_n be the group of all permutations of $[n] := \{1, \dots, n\}$. For $\sigma \in S_n$, we define the permuted function $f \circ \sigma$ by

$$(f \circ \sigma)(x_1, \dots, x_n) := f(x_{\sigma(1)}, \dots, x_{\sigma(n)})$$

for all $(x_1, \dots, x_n) \in \{-1, 1\}^n$. A fundamental question in this setting is the following.

► **Question 1** (Function Isomorphism Testing over $\{-1, 1\}^n$). *Given a known Boolean function f and query access to an unknown Boolean function g , determine if there exists a permutation $\sigma \in \mathcal{G}$ such that $f \circ \sigma = g$, or for all $\sigma \in \mathcal{G}$, is the Hamming distance between $f \circ \sigma$ and g at least ϵ , under the promise that one of these is the case?*

The above question was first investigated by Fischer et al. [26] and was followed by a long line of work [1, 8, 11, 10, 2, 9].

The domain of Boolean functions discussed so far is the set $\{-1, 1\}^n$. It is of great interest, as we discuss later, to study Boolean functions whose domain \mathcal{D} has some algebraic structure associated with it, like \mathbb{Z}_2^n . In these settings, it is natural to study the isomorphism problem for Boolean functions under the *group of symmetry* of \mathcal{D} that preserves its underlying algebraic structure.

¹ Note that Fourier coefficients are complex numbers.

An extremely important example is that of $\mathcal{D} = \mathbb{Z}_2^n$, that is, functions of the form $f : \mathbb{Z}_2^n \rightarrow \{-1, 1\}$ defined over the vector space \mathbb{Z}_2^n . These functions arise in numerous areas of computer science and mathematics, see [40] and the references therein for several examples. A natural group of symmetry for \mathbb{Z}_2^n is the group of all non-singular $n \times n$ matrices over \mathbb{Z}_2 (see [22, Chapter 3]). We now describe the question of *Linear Isomorphism Testing of Boolean Functions* over \mathbb{Z}_2^n . For $A \in \mathbb{Z}_2^{n \times n}$, let $f \circ A : \mathbb{Z}_2^n \rightarrow \{-1, 1\}$ be the function $f \circ A(x) = f(Ax)$ for all $x \in \mathbb{Z}_2^n$. The *Linear Isomorphism Distance* between two functions $f : \mathbb{Z}_2^n \rightarrow \{-1, 1\}$ and $g : \mathbb{Z}_2^n \rightarrow \{-1, 1\}$ is defined as

$$\text{dist}_{\mathbb{Z}_2^n}(f, g) = \min_{A \in \mathbb{Z}_2^{n \times n} : A \text{ is non-singular}} \delta(f \circ A, g). \quad (1)$$

Assume that f and g satisfy the promise that either $\text{dist}_{\mathbb{Z}_2^n}(f, g) = 0$ or $\text{dist}_{\mathbb{Z}_2^n}(f, g) \geq \epsilon$, the question of *Linear Isomorphism Testing* is that of deciding which is the case.

In this work, we study Boolean functions over a finite Abelian group \mathcal{G} . The natural symmetries in this setting arise from the automorphism group of \mathcal{G} , denoted $\text{Aut}(\mathcal{G})$ (see Definition 31). For functions $f, g : \mathcal{G} \rightarrow \{0, 1\}$, we define their *automorphism distance* as

$$\text{dist}_{\mathcal{G}}(f, g) = \min_{\sigma \in \text{Aut}(\mathcal{G})} \delta(f \circ \sigma, g),$$

where $f \circ \sigma(x) = f(\sigma(x))$ for all $x \in \mathcal{G}$. We investigate the following general question.

► **Question 2** (Testing Isomorphism of Boolean Functions over Finite Abelian Groups). *Let \mathcal{G} be a finite Abelian group, and let $f, g : \mathcal{G} \rightarrow \{-1, 1\}$ be Boolean functions such that f is known and given query access to g , determine if f and g are isomorphic under the automorphism group $\text{Aut}(\mathcal{G})$ of \mathcal{G} (that is, $\text{dist}_{\mathcal{G}}(f, g) = 0$) or is the automorphism distance between them at least ϵ (that is, $\text{dist}_{\mathcal{G}}(f, g) \geq \epsilon$), under the promise that one of these is the case?*

We now give an equivalent problem in the setting of finite Abelian groups.

► **Definition 3** (Subset Equivalence Problem). *Given two subsets $A, B \subseteq \mathcal{G}$ of a finite Abelian group \mathcal{G} , does there exist an automorphism $\sigma \in \text{Aut}(\mathcal{G})$ such that $A = \sigma(B)$?*

To formulate a testing version of this problem, we introduce the following definitions. For subsets $A, B \subseteq \mathcal{G}$, the *distance* between A and B is defined by

$$\text{dist}(A, B) := \frac{|A \setminus B| + |B \setminus A|}{|\mathcal{G}|}. \quad (2)$$

We define the *automorphism distance* between A and B as

$$\text{dist}_{\mathcal{G}}(A, B) := \min_{\sigma \in \text{Aut}(\mathcal{G})} \text{dist}(A, \sigma(B)). \quad (3)$$

We assume *membership query access* to a subset of \mathcal{G} , meaning that we can query whether any given element of \mathcal{G} belongs to the subset. The testing version of the Subset Equivalence Problem is formulated below:

► **Question 4** (Testing Automorphism Distance Between Subsets of Finite Abelian Groups). *Let $\epsilon > 0$ and \mathcal{G} be a finite Abelian group. Let $A \subseteq \mathcal{G}$ be a known subset, and suppose we have membership query access to an unknown subset $B \subseteq \mathcal{G}$. Under the promise that either $\text{dist}_{\mathcal{G}}(A, B) = 0$ or $\text{dist}_{\mathcal{G}}(A, B) \geq \epsilon$, the goal is to determine which of the two cases holds.*

Question 4 is equivalent to the problem of testing can be viewed as a special case of *Boolean function isomorphism testing* as follows. Define the *indicator function* $\mathbb{1}_A : \mathcal{G} \rightarrow \{-1, 1\}$ of the set A by

$$\mathbb{1}_A(x) = \begin{cases} -1 & \text{if } x \in A, \\ 1 & \text{otherwise.} \end{cases}$$

Similarly, define $\mathbb{1}_B$ for the set B . Then Question 4 reduces to testing whether the Boolean functions $\mathbb{1}_A$ and $\mathbb{1}_B$ are isomorphic under the action of $\text{Aut}(\mathcal{G})$.

Previous works on property testing of Boolean functions over \mathbb{Z}_2^n critically exploit the Fourier analysis framework, where the vector space structure of the domain is essential to the proofs. However, since we are working with arbitrary finite Abelian groups, these established approaches that rely on the vector space structure do not apply. As a result, we had to develop new ideas, drawing on key concepts from the theory of finite Abelian groups. We outline some of these ideas below:

- We have used a notion of “orthogonal” subgroup H^\perp for a subgroup H of a finite Abelian group \mathcal{G} which is not naturally defined in the context of groups. To address this, we used the concept of *annihilator* of a subgroup. This concept is described in more detail in Section 3.
- For any group automorphism $A : \mathcal{G} \rightarrow \mathcal{G}$ and any function $f : \mathcal{G} \rightarrow \{-1, +1\}$, we needed a natural way to associate the Fourier coefficients $\{\widehat{f \circ A}(\chi) : \chi \in \widehat{\mathcal{G}}\}$ for the function $f \circ A$ with the Fourier coefficients $\{\widehat{f}(\chi) : \chi \in \widehat{\mathcal{G}}\}$ for f . We use ‘**Pontryagin duality**’ to deal with such scenario, which is of independent interest. More details can be found in Lemma 34 and Lemma 38 from Section 2.
- The vector space \mathbb{Z}_2^n can be naturally partitioned into cosets of a subspace. This property is essential in approaches like “coset hashing”, which is used in algorithms for property testing of Boolean functions over such domains (see [25, 30, 47]). However, for a finite Abelian group \mathcal{G} , this partitioning is not generally possible, as \mathcal{G} may not form a vector space. To address this, we instead use normal subgroups and partition \mathcal{G} using the cosets of these normal subgroups. For details, see Section 2.
- We use a new notion of inner product, called **pseudo-inner product** (see Definition 22) to analyze our algorithms.
- We also introduce a notion of independence for finite Abelian groups, which we call **pseudo-independence** (see Definition 32), to analyze our algorithms.
- Note that the characters of \mathbb{Z}_2^n are the 2nd roots of unity, meaning they take the values -1 or $+1$ at each point in \mathbb{Z}_2^n . This property is fundamental to the analysis of Boolean functions over \mathbb{Z}_2^n , see, for example, Gopalan et al. [30]. Since \mathcal{G} is a finite Abelian group, it can be written as $\mathcal{G} = \mathbb{Z}_{p_1^{m_1}} \times \cdots \times \mathbb{Z}_{p_n^{m_n}}$, where p_i are primes for all $i \in [n]$, which may not necessarily be distinct. In the case of \mathcal{G} , the characters are complex numbers, as they correspond to the \mathcal{L} -th roots of unity at each point in \mathcal{G} , where \mathcal{L} is the exponent of \mathcal{G} . Observe that $\mathcal{L} = \text{LCM}\{p_1^{m_1}, \dots, p_n^{m_n}\}$. Consequently, there is no natural concept of sign, which we address using a completely different approach. For more details, see [20], which we believe offers independent interest in the study of Boolean functions on more general domains.

1.1 Related work

Fischer et al. [26] were the first to investigate the problem of function isomorphism testing, a line of research that has since been extended by several subsequent works [1, 8, 11]. The problem of characterizing the *testability*² of *linearly invariant* properties of functions has been a central focus in property testing. This question has seen significant progress through various works [33, 37, 7]. Also, Bhattacharyya et al. [6] provided a characterization of linearly invariant properties that are constant-query testable with one-sided error. Testing linear isomorphism of Boolean functions over \mathbb{Z}_2^n was studied by Wimmer and Yoshida [47], whose algorithm has polynomial query complexity in terms of the spectral norm of the functions. Grigorescu, Wimmer and Xie [35] studied the same problem in the communication setting. Linear isomorphism of Boolean functions has also been explored in the context of combinatorial circuit design [16, 5, 48], error-correcting codes [21, 24, 36], and cryptography [18, 14, 23].

1.2 Our results

We study Question 2 in the following setting: the function $g : \mathcal{G} \rightarrow \{-1, 1\}$ is known, while the function $f : \mathcal{G} \rightarrow \{-1, 1\}$ is unknown. We are promised that either $\text{dist}_{\mathcal{G}}(f, g) \leq \epsilon$ or $\text{dist}_{\mathcal{G}}(f, g) \geq \epsilon + \tau$. **Additionally, we assume that the exponent of the finite Abelian group \mathcal{G} is a constant. For the remainder of the paper, we denote the exponent of \mathcal{G} by $\mathcal{L}(\mathcal{G})$, or simply \mathcal{L} .** Observe that if $\mathcal{G} = \mathbb{Z}_{p_1^{m_1}} \times \cdots \times \mathbb{Z}_{p_n^{m_n}}$, then

$$\mathcal{L}(\mathcal{G}) = \text{LCM}\{p_1^{m_1}, \dots, p_n^{m_n}\}.$$

The goal is to design a randomized algorithm that, with probability at least $2/3$ over its internal randomness, determines which case holds while making as few queries as possible to the unknown function f . Each query allows the algorithm to obtain the value $f(x)$ for a chosen $x \in \mathcal{G}$.

Our results are developed in the context of *tolerant testing*, specifically in the (ϵ, τ) -tolerant testing model. This generalizes the setting of Question 2, which corresponds to the special case of $(0, \tau)$ -tolerant testing.

We give an efficient algorithm for the above problem. The algorithm's query complexity is polynomial in $(1/\tau)$ (independent of ϵ) and the spectral norm (see Definition 20) of the known function.

► **Theorem 5 (Main Result: (ϵ, τ) -Tolerant Isomorphism Testing of Boolean Functions over Finite Abelian Groups).** *Let \mathcal{G} be a finite Abelian group, with*

$$\mathcal{G} = \mathbb{Z}_{p_1^{m_1}} \times \cdots \times \mathbb{Z}_{p_n^{m_n}},$$

where each p_i is a prime (not necessarily distinct), and let $\epsilon \geq 0$ and $\tau \in (0, 1/2]$. Suppose we are given a known function $g : \mathcal{G} \rightarrow \{-1, 1\}$ satisfying $\|\hat{g}\|_1 \leq s$, and have query access to an unknown function $f : \mathcal{G} \rightarrow \{-1, 1\}$.

Assuming that the exponent \mathcal{L} of \mathcal{G} is constant, there exists a randomized algorithm that decides whether

$$\text{dist}_{\mathcal{G}}(f, g) \leq \epsilon \quad \text{or} \quad \text{dist}_{\mathcal{G}}(f, g) \geq \epsilon + \tau$$

with probability at least $2/3$, using $\tilde{O}\left(\frac{s^8}{\tau^8}\right)$ queries to f , where the $\tilde{O}(\cdot)$ notation hides factors that are polynomial in $\log s$ and $\log(1/\tau)$.

² See the definition of testability from Bhattacharyya et al. [6].

Note that Wimmer and Yoshida [47] proved $\Omega(\log \|\hat{g}\|_1)^3$ lower bound for the special case of $\mathcal{G} = \mathbb{Z}_2^n$. Recently, Datta et al. [19] have been able to show a lower bound of $\Omega(\|\hat{g}\|_1^2)$ for the above problem when $\mathcal{G} = \mathbb{Z}_2^n$.

The proof of the above theorem generalizes the work of Wimmer and Yoshida [47], who solved the $(\epsilon/3, 2\epsilon/3)$ -tolerant linear isomorphism problem for Boolean functions over \mathbb{Z}_2^n . Their testing algorithm and analysis crucially rely on the fact that \mathbb{Z}_2^n is a vector space over the field \mathbb{Z}_2 . However, finite Abelian groups do not necessarily have a vector space structure. Due to this, we had to introduce several new ideas and incorporate key concepts from the theory of finite Abelian groups. For more details, see Section 1.3 and Section 1.4.

Finite Abelian groups \mathcal{G} which are of the form $G_1 \times G_2 \times \cdots \times G_n$ have been used in various works, namely [34, 3]. We give the following two corollaries, which immediately follow from Theorem 5.

► **Corollary 6.** *Let \mathcal{G} be a finite Abelian group such that \mathcal{G} can be expressed as $G \times G \times \cdots \times G$ where the product takes place n -times and G itself is a finite Abelian group of constant size. Also, let $\epsilon, \tau \in (0, 1/2]$. Given a known function $g : \mathcal{G} \rightarrow \{-1, 1\}$ with $\|\hat{g}\|_1 \leq s$, and query access to a unknown function $f : \mathcal{G} \rightarrow \{-1, 1\}$, the query complexity of deciding whether $\text{dist}_{\mathcal{G}}(f, g) \leq \epsilon$ or $\text{dist}_{\mathcal{G}}(f, g) \geq \epsilon + \tau$ with probability at least $2/3$, is $\tilde{O}(\text{poly}(s, 1/\tau))$, where $\tilde{O}(\cdot)$ hides multiplicative factors that are polynomial in $\log s$ and $\log(1/\tau)$.*

More generally we get the following:

► **Corollary 7.** *Let \mathcal{G} be a finite Abelian group such that \mathcal{G} can be expressed as $G_1 \times G_2 \times \cdots \times G_n$ where $i \in [n]$ and $|G_i| = O(1)$ for all $i \in [n]$. Also let $\epsilon, \tau \in (0, 1/2]$. Given a known function $g : \mathcal{G} \rightarrow \{-1, 1\}$ with $\|\hat{g}\|_1 \leq s$, and query access to a unknown function $f : \mathcal{G} \rightarrow \{-1, 1\}$, the query complexity of deciding whether $\text{dist}_{\mathcal{G}}(f, g) \leq \epsilon$ or $\text{dist}_{\mathcal{G}}(f, g) \geq \epsilon + \tau$ with probability at least $2/3$, is $\tilde{O}(\text{poly}(s, 1/\tau))$, where $\tilde{O}(\cdot)$ hides factors that are polynomial in $\log s$ and $\log(1/\tau)$.*

The main technical result we require is the following algorithm to estimate the values of the large Fourier coefficients of $f : \mathcal{G} \rightarrow \{-1, 1\}$, given query access to f , without prior knowledge of the coefficients themselves. The set of large Fourier coefficients is defined using a threshold (see Theorem 8). Note that this set is unknown, as the function f is not known.

► **Theorem 8 (Generalized Implicit Sieve for Abelian groups).** *Let \mathcal{G} be a finite Abelian group with constant exponent. Also, let $\theta > 0$ be a threshold and $\mathcal{M} = \{x_i\}_{i=1}^{\tilde{m}} \subseteq \mathcal{G}$ be a set of independently and uniformly chosen \tilde{m} random points from \mathcal{G} . There exists an algorithm, *Generalized Implicit Sieve Algorithm* (see [20]), that takes θ and \mathcal{M} as input, makes $O\left(\frac{\ln(\tilde{m}/\theta^{16})}{\theta^8} + \tilde{m} \frac{\ln(\mathcal{N}\tilde{m})}{\theta^2} + \frac{\ln \mathcal{N}}{\theta^8}\right)$ queries to the truth table of a Boolean valued function $f : \mathcal{G} \rightarrow \{-1, 1\}$, and returns, with probability at least $\frac{94}{100}$, a labeled set of \tilde{m} examples, and the value of the function f at x_i for each $i \in [\tilde{m}]$, of the form $\{\chi_{\alpha_1}(x_i), \dots, \chi_{\alpha_{\mathcal{N}}}(x_i), f(x_i) \mid x_i \in \mathcal{M}\}$, where $\mathcal{S} = \{\alpha_1, \dots, \alpha_{\mathcal{N}}\}$ is some set that satisfies the following two properties:*

1. $\forall \alpha \in \mathcal{G}$ with $|\hat{f}(\chi_{\alpha})| \geq \theta$ then $\alpha \in \mathcal{S}$, and
2. $\forall \alpha \in \mathcal{S}$, $|\hat{f}(\chi_{\alpha})| \geq \theta/2$.

The output of this algorithm can be seen as a $\tilde{m} \times (\mathcal{N} + 1)$ matrix Q with entries in powers of $\omega_{\mathcal{L}}$, where $\omega_{\mathcal{L}}$ is a primitive \mathcal{L}^{th} root of unity with \mathcal{L} being the exponent of \mathcal{G} .

³ Wimmer and Yoshida claimed an $\Omega(\|\hat{g}\|_1)$ lower bound for the case when $\epsilon = 0$ and $\tau > 0$. However, their proof applies only to the special case $\tau = \frac{1}{\|\hat{g}\|_1}$. Even if their argument can be extended to general values of τ , the lower bound established by Datta et al. [19] is quadratically stronger.

► **Remark 9** (Comments on the query complexity in Theorem 8). Note that the query complexity of the algorithm in Theorem 5 satisfies the following:

1. Query complexity is independent of $|\mathcal{G}|$, and \mathcal{L} is assumed to be constant, and
2. it is polynomial in \tilde{m} , $1/\theta$ and $\ln \mathcal{N}$.

Wimmer and Yoshida [47] proved a special case of the above theorem, which they referred to as the *Implicit Sieve Algorithm* for functions over \mathbb{Z}_2^n and they used for testing linear isomorphism testing. Theorem 8 generalizes the Implicit Sieve algorithm of Wimmer and Yoshida to finite Abelian groups. Note that the Implicit Sieve Algorithm of Wimmer and Yoshida [47] is itself a generalization of the celebrated and widely applicable Goldreich-Levin algorithm [29].

Using Theorem 5 we have the following result for the case of subset isomorphism problem.

► **Theorem 10.** *Let \mathcal{G} be a finite Abelian group with constant exponent, and $A, B \subset \mathcal{G}$. Also let $\epsilon, \tau \in (0, 1/2]$. Given the known set $A \subset \mathcal{G}$ with $\|\widehat{\mathbb{1}_A}\|_1 \leq s$, and membership query access to an unknown set $B \subset \mathcal{G}$, the query complexity of deciding whether $\text{dist}_{\mathcal{G}}(A, B) \leq \epsilon$ or $\text{dist}_{\mathcal{G}}(A, B) \geq \epsilon + \tau$ with probability at least $2/3$, is $\tilde{O}\left(\frac{s^8}{\tau^8}\right)$, where $\tilde{O}(\cdot)$ hides multiplicative factors polynomial in $\log s$ and $\log(1/\tau)$.*

The above result is part of a broader research program exploring the interplay between the spectral norm of indicator functions and the structure of subsets of finite Abelian groups. In additive combinatorics and additive number theory, the spectral norm serves as a key tool for measuring the non-uniformity or pseudorandomness of functions defined on finite Abelian groups or their vector spaces. Notable breakthroughs in this area include the quantitative version of Cohen’s Idempotent Theorem due to Green and Sanders [32], as well as Chang’s Lemma [12] and its various extensions.

We also prove that for *Fourier sparse* functions, testing can be performed more efficiently. The proof follows a similar approach to the theorem on tolerant isomorphism testing for Finite Abelian Groups (Theorem 5), with a key observation: when both functions are Fourier s -sparse, the number of nonzero Fourier coefficients is at most s , which significantly simplifies the learning process in the implicit sieve algorithm (Theorem 8). In particular, we do not need to estimate wt_4 (see [20]), which is required in the general case.

► **Theorem 11** ((ϵ, τ) -Tolerant Isomorphism Testing of Sparse Boolean Functions over Finite Abelian Groups). *Let $\epsilon \geq 0$ and $\tau \in (0, 1/2]$, and let \mathcal{G} be a finite Abelian group of constant exponent. Given a known function $g : \mathcal{G} \rightarrow \{-1, 1\}$ with sparsity s and query access to a unknown s -sparse Boolean valued function $f : \mathcal{G} \rightarrow \{-1, 1\}$, the query complexity of deciding whether $\text{dist}_{\mathcal{G}}(f, g) \leq \epsilon$ or $\text{dist}_{\mathcal{G}}(f, g) \geq \epsilon + \tau$ with probability at least $2/3$, is $\tilde{O}\left(\frac{s^4}{\tau^4}\right)$, where $\tilde{O}(\cdot)$ notation hides multiplicative factors polynomial in $\log s$ and $\log \log(1/\tau)$.*

1.3 Proof sketch

We start by giving the ideas behind and a sketch of the proof of our main result, Theorem 5. Throughout this section \mathcal{G} is assumed to be a finite Abelian group.

Proof sketch of Theorem 5 (Isomorphism Testing for finite Abelian groups). Given a known function $g : \mathcal{G} \rightarrow \{-1, 1\}$ and an unknown function $f : \mathcal{G} \rightarrow \{-1, 1\}$, where either

$$\text{dist}_{\mathcal{G}}(f, g) \leq \epsilon \quad \text{or} \quad \text{dist}_{\mathcal{G}}(f, g) \geq \epsilon + \tau.$$

The goal is to determine which case holds with probability at least $2/3$. The algorithm has query access to f , meaning it can evaluate $f(x)$ for any $x \in \mathcal{G}$.

- Our algorithm (see [20]) uses the Generalized Implicit Sieve Algorithm (see [20] and Theorem 8) as a subroutine. Given query access to f , \tilde{m} random elements $x_1, \dots, x_{\tilde{m}} \in \mathcal{G}$ and a parameter θ , the Generalized Implicit Sieve Algorithm returns an $\tilde{m} \times (\mathcal{N} + 1)$ matrix Q which satisfies the following with high probability (ignoring the last column of Q for simplicity):
 1. For some unknown set $\mathcal{S} = \{r_1, \dots, r_{\mathcal{N}}\}$, the (i, j) th entry of Q is $\chi_{r_j}(x_i)$. Thus the j th column of Q contains the evaluations of the characters corresponding to r_j on \tilde{m} random points in \mathcal{G} .
 2. If $r \in \mathcal{S}$ then $|\hat{f}(\chi_r)| \geq \theta/2$ and if r satisfies $|\hat{f}(\chi_r)| \geq \theta$, then $r \in \mathcal{S}$.
- Since

$$\hat{f}(\chi_{r_i}) = \mathbb{E}_{x \in \mathcal{G}}[f(x)\chi_{r_i}(x)]$$

for all $r_i \in \mathcal{S}$ (see Section 2), the output of the Generalized Implicit Sieve Algorithm can be used to estimate the Fourier coefficients $\hat{f}(\chi_{r_i})$, for $i \in [\mathcal{N}]$, with suitably chosen accuracy and high probability, provided that \tilde{m} is large enough. This follows from an application of the Hoeffding bound. However, note that the set \mathcal{S} remains unknown.

- We now describe how we use these estimates to construct a function $\tilde{f}: \mathcal{G} \rightarrow \mathbb{R}$ such that for some unknown group automorphism A , the correlation between $\tilde{f} \circ A$ and f is large. In order to do this, we need the idea of *pseudo-independence* over finite Abelian groups (see Definition 32).
- For any set of elements $r_1, \dots, r_k \in \mathcal{G}$, if there exists $\lambda_1, \dots, \lambda_k \in \mathbb{N} \cup \{0\}$, such that at least one λ_j is invertible in $\mathbb{Z}_{\mathcal{L}}$, where $\mathcal{L} = \text{LCM}\{p_1^{m_1}, \dots, p_n^{m_n}\}$, and $\sum_{i=1}^k \lambda_i r_i = 0$, then we call r_1, \dots, r_k to be *pseudo-dependent* (Note that any subset has a pseudo-independent spanning set, see Definition 32). If there does not exist any such λ_i 's, then r_1, \dots, r_k are *pseudo-independent*. The idea is, if there exists such $\lambda_1, \dots, \lambda_k \in \mathbb{Z}_{\mathcal{L}}$ with λ_j invertible in $\mathbb{Z}_{\mathcal{L}}$, then r_j can be written as a combination of the other elements. That is,

$$r_j = -\lambda_j^{-1} \left(\sum_{i \in [k], i \neq j} \lambda_i r_i \right), \quad (4)$$

where λ_j^{-1} is the inverse of λ_j in $\mathbb{Z}_{\mathcal{L}}$. Observe that the above definition generalized the notion of linear dependence and independence over vector spaces. Furthermore, whenever we define a group automorphism on the elements $r_1, \dots, r_{j-1}, r_{j+1}, \dots, r_k$, by the property of homomorphism it automatically gets defined on r_j . That is,

$$\psi(r_j) = -\lambda_j^{-1} \left(\sum_{i \in [k], i \neq j} \lambda_i \psi(r_i) \right).$$

Recall that λ_j is invertible in $\mathbb{Z}_{\mathcal{L}}$ and λ_j^{-1} is the inverse of λ_j in $\mathbb{Z}_{\mathcal{L}}$.

- To construct \tilde{f} we first identify a $R \subseteq \mathcal{S}$ (without knowing R) such that the elements in R are pseudo-independent and \mathcal{S} is a subset of the subgroup generated by R (which can be checked by a brute-force algorithm, see [20]). We prove that such an R can be found with high probability. This will allow us to relabel the elements in R as $e_1, \dots, e_{|R|} \in \hat{\mathcal{G}}$ and the elements in $\mathcal{S} \setminus R$ as a combination of $e_1, \dots, e_{|R|}$, using Equation (4), for some automorphism A of \mathcal{G} . Relabel the columns of Q by $\{S_1, \dots, S_{\mathcal{N}}\}$ where $S_1 = e_1, \dots, S_{|\tilde{B}|} = e_{|\tilde{B}|}$ such that for all $j \in \{|\tilde{B}| + 1, \dots, \mathcal{N}\}$, S_j can be written as

$$\lambda^{-1} \left(\sum_{i \in [|\tilde{B}|]} \lambda_i S_i \right),$$

where $\lambda, \lambda_1, \dots, \lambda_{|\tilde{B}|} \in \mathbb{N} \cup \{0\}$, and λ is invertible in $\mathbb{Z}_{\mathcal{L}}$. Note that the columns $S_1, \dots, S_{|\tilde{B}|}$ correspond to those in \tilde{B} .

- We will create \tilde{f} such that for all $j \in [\mathcal{N}]$, $\widehat{\tilde{f}}(\chi_{S_j}) = r_j$ and $\widehat{\tilde{f}}(\chi_S) = 0$ if $S \notin \{S_1, \dots, S_{\mathcal{N}}\}$ where

$$r_j = \frac{\sum_{i=1}^{\tilde{m}} f(x^{(i)}) Q[i, j]}{\tilde{m}}.$$

- Thus at this point we have constructed \tilde{f} that has high correlation with f under some unknown group automorphism A , with high probability. We show that to decide whether $\text{dist}_{\mathcal{G}}(f, g) \leq \epsilon$ or $\text{dist}_{\mathcal{G}}(f, g) \geq \epsilon + \tau$, it is sufficient to check the correlation between \tilde{f} and g under all possible group automorphisms on \mathcal{G} (see [20]) and this can be done without making queries to f .

We refer the reader to [20] for the details. ◀

As evident, the Generalized Implicit Sieve Algorithm (see [20]) plays an important role in the proof of our main result.

Proof idea of Theorem 8. (Generalized Implicit Sieve Algorithm). Before we outline the proof of Theorem 8, let us first explain the goal of the Generalized Implicit Sieve Algorithm (see [20]).

- The algorithm takes a Boolean valued function $f : \mathcal{G} \rightarrow \{-1, 1\}$ as input, where $\mathcal{G} = \mathbb{Z}_{p_1^{m_1}} \times \dots \times \mathbb{Z}_{p_n^{m_n}}$, and $p_i, i \in [n]$ are primes, and not necessarily distinct. It is also given a threshold θ and a set $\mathcal{M} = \{x_1, \dots, x_{\tilde{m}}\}$ of uniformly random points from \mathcal{G} , where \tilde{m} is sufficiently large. The algorithm considers a randomly permuted coset structure (H, \mathcal{C}) such that the subgroup H has codimension $\leq t$ in \mathcal{G} (see Definition 28), where $t \geq \log_{\mathcal{L}}\left(\frac{100^4 \tilde{m}^4}{\theta^{32}}\right)$, where $\mathcal{L} = \text{LCM}\{p_1^{m_1}, \dots, p_n^{m_n}\}$. Then wt_2 (see [20]) is estimated for each bucket with error $\pm \frac{\theta^2}{4}$ and confidence $1 - \frac{1}{100\mathcal{L}^t}$. If the estimated weight \widehat{wt}_2 is $\geq \frac{3\theta^2}{4}$, then the algorithm keeps the bucket. Otherwise, it discards the bucket. For each surviving bucket C , wt_4 (see [20]) is estimated with error $\pm \frac{\theta^4}{8} \widehat{wt}_2(C)$ and confidence $1 - \frac{1}{100\mathcal{L}^t}$. If the estimated weight \widehat{wt}_4 is $\geq \frac{3\theta^4}{4}$, then the algorithm keeps the bucket. Otherwise, it discards the bucket. Then it randomly picks $|\mathcal{M}| = \tilde{m}$ points $\{y_1, \dots, y_{\tilde{m}}\}$ from \mathcal{G} , and calculates $P_C f(y_i) \overline{P_C f(y_i - x_i)}$ for all $i \in [\tilde{m}]$. Note that P_C is the projection operator on functions $f : \mathcal{G} \rightarrow \mathbb{R}$, which is defined as follows ($C \subseteq \mathcal{G}$)

$$P_C f(x) := \sum_{\beta \in C} \widehat{f}(\beta) \chi_{\beta}(x).$$

The Generalized Implicit Sieve Algorithm makes $O\left(\frac{\ln \frac{\tilde{m}}{\theta^{16}}}{\theta^8} + \tilde{m} \frac{\ln(\mathcal{N}\tilde{m})}{\theta^2} + \frac{\ln \mathcal{N}}{\theta^8}\right)$ many queries.

We now outline the proof of Theorem 8.

- The first step in the proof of the theorem is to partition \mathcal{G} into cosets of a random subgroup. Since \mathcal{G} is Abelian, all its subgroups are normal. Define a random subgroup H by sampling β_1, \dots, β_k independently and uniformly from \mathcal{G} and using the pseudo-inner product operator on \mathcal{G} to obtain

$$H = \{\alpha \in \mathcal{G} : \alpha * \beta_i = 0 \ \forall i \in [k]\}.$$

By Definition 22 and Definition 24, H forms a subgroup of \mathcal{G} . Its cosets have the form

$$C(b) = \{\alpha \in \mathcal{G} : \alpha * \beta_i = b_i \ \forall i \in [k]\},$$

where $b = (b_1, \dots, b_k) \in \mathbb{Z}_{\mathcal{L}}^k$. We refer to the cosets of H as “buckets”.

66:10 Testing Isomorphism of Boolean Functions over Finite Abelian Groups

- We show that with high probability, all the big Fourier coefficients, that is, the coefficients with weight $\geq \frac{\theta^2}{4}$ belong to different buckets (see [20]).
- Moreover, we show that the weight of a bucket C_α is determined by the big coefficient $\chi_{\alpha(C)}$ contained in C_α with high probability, assuming that the big coefficients are in different buckets with high probability. We use the second moment method to show this. Let for each $\gamma \in \mathcal{G}$,

$$Y_{\gamma,\alpha} = \begin{cases} 1, & \text{if } \gamma \in C_\alpha \\ 0, & \text{otherwise.} \end{cases}$$

And,

$$Y_\alpha = \sum_{\gamma \in S} Y_{\gamma,\alpha}.$$

Also, let

$$X_{\gamma,C_\alpha} = \begin{cases} |\hat{f}(\chi_\gamma)|^2 & \text{if } \gamma \in C_\alpha \\ 0 & \text{otherwise.} \end{cases}$$

And,

$$X_{C_\alpha} = \sum_{\gamma \in S} X_{\gamma,C_\alpha}.$$

The $b = 0$ case (see [20]) is problematic, as the $Cov[Y_{\gamma_1,\alpha}Y_{\gamma_2,\alpha}] > 0$ in this case and hence finding an upper bound for $\Pr[|X_{C_\alpha} - \mathbb{E}[X_{C_\alpha}]| \geq \epsilon]$ becomes difficult using Chebyshev. We use **random shift** (see [20]) to avoid this, and study case by case to show that $Cov[Y_{\gamma_1,\alpha}Y_{\gamma_2,\alpha}] = 0$. This gives us an upper bound on the variance of X_{C_α} , which in turn helps us to show that the weight of a bucket C_α is determined by the big coefficient $\chi_{\alpha(C)}$ contained in C_α with high probability. This implies that with high probability, the Fourier coefficients other than the big coefficient $\chi_{\alpha(C)}$ does not contribute much to the bucket C_α .

- We then show that the algorithm does not discard any bucket C with $|\hat{f}(\chi_\gamma)| \geq \theta$, but it discards any bucket C with $|\hat{f}(\chi_\gamma)| < \frac{\theta^2}{4}$, for all $\gamma \in C$. Furthermore, we establish that $|\hat{f}(\chi_{\alpha(C)})| \geq \frac{\theta}{2}$ for every bucket $C = C_\alpha$ (where $\alpha(C)$ is the dominating element of C_α) among those that the algorithm has not discarded.
- Additionally, we show that the small weight coefficients do not contribute much to the weight of a bucket. That is, we show that $\Pr_x[|\sum_{r \in C_\alpha \setminus \{\chi_{\alpha(C)}\}} \hat{f}(\chi_r)\chi_r(x)| \geq \frac{\theta^2}{4}]$ is very small, where $\chi_{\alpha(C)}$ is the dominating Fourier coefficient of the bucket C_α .
- Finally, our target is to output a matrix Q (see [20]) with a small error, whose entries are characters evaluated at the points $x_i, i \in [\tilde{m}]$, where x_i are picked independently and uniformly at random. To show that, we pick y_i independently and uniformly at random, and then prove that the distance between $P_{C_\alpha}f(y_i)\overline{P_{C_\alpha}f(y_i - x_i)}$ and $|\hat{f}(\chi_{\alpha(C)})|^2\chi_{\alpha(C)}(x_i)$ is $\leq \frac{9\theta^4}{16}$ with high probability. Then we construct a matrix Q' (see [20]), and estimate the Fourier coefficients for all the large buckets. We then show that, up to some small error, the algorithm outputs the matrix Q (see [20]) whose entries are given by $\chi_{\alpha(C_i)}(x_j)$, $i \in [\mathcal{N}], j \in [\tilde{m}]$, where \mathcal{N} denotes the number of survived buckets. ◀

1.4 Challenges with finite Abelian Groups

We outline the main challenges encountered in proving our results for finite Abelian groups \mathcal{G} . Recall that \mathcal{G} can be written as $\mathcal{G} = \mathbb{Z}_{p_1^{m_1}} \times \cdots \times \mathbb{Z}_{p_n^{m_n}}$, where $p_i, i \in [n]$ are primes, and not necessarily distinct.

- While estimating wt_2 and wt_4 , it is required to find H^\perp for a subgroup H of \mathcal{G} . But due to the lack of vector space structure, H^\perp cannot be defined in terms of “basis”, which is possible when $\mathcal{G} = \mathbb{Z}_2^n$, as \mathbb{Z}_2^n is a vector space. This problem had to be taken care of in a different way. Given a subgroup H of \mathcal{G} , we define H^\perp by

$$H^\perp := \{x \in \mathcal{G} : x * y = 0 \ \forall y \in H\},$$

where $*$ is a pseudo inner product defined by

$$x * y := \left(\sum_{i=1}^T \frac{\mathcal{L}}{p_i^{m_i}} x^{(i)} \cdot y^{(i)} \right) \pmod{\mathcal{L}},$$

and $\mathcal{L} = LCM\{p_1^{m_1}, \dots, p_n^{m_n}\}$. It is to be noted that H^\perp is also a subgroup of \mathcal{G} .

This is extremely useful in proving the fact that, given a bucket C_α , $wt_2(C_\alpha)$ and $wt_4(C_\alpha)$ can be written as an average of some quantity, which in turn helps to reduce the query complexity of wt_2 and wt_4 estimation. More precisely, it follows that,

$$wt_2(C_\alpha) = \mathbb{E}_{x \in \mathcal{G}, z \in H^\perp} \left[f(x) f(x+z) \chi_r(z) \right]$$

and

$$wt_4(C_\alpha) = \mathbb{E}_{x, z_1, y_1 \in \mathcal{G}, z, y \in H^\perp} \left[f(z_1) f(x - z_1 - z) f(y_1) f(x - y_1 - y) \chi_r(z - y) \right]$$

respectively (see [20] for more details). Applying a concentration inequality the query complexity becomes dependent on θ and \tilde{m} only, where θ and \tilde{m} are as defined in Theorem 8, and it is not dependent on the order of the group \mathcal{G} .

- To prove our main theorem, we also need to show that, for any automorphism $A : \mathcal{G} \rightarrow \mathcal{G}$ and any function $f : \mathcal{G} \rightarrow \{-1, +1\}$, the Fourier coefficients $\widehat{f \circ A}(\chi_r)$, $r \in \mathcal{G}$ of $f \circ A$ are same as the Fourier coefficients of f upto some permutation. In case of \mathbb{Z}_2^n , it was quite easy, as \mathbb{Z}_2^n is a vector space. As an Abelian group is not a vector space in general, we needed to use Pontryagin duality and dual automorphisms. For more details, see Lemma 34 and Lemma 38 in Section 2.
- Due to the lack of vector space structure, we cannot partition \mathcal{G} into subspaces. Instead, we define a normal subgroup V_{0, r_1, \dots, r_k} (see Lemma 26), and partition \mathcal{G} , and hence $\widehat{\mathcal{G}}$ into its cosets. For details see Section 2.
- While testing isomorphism, the notion of linear independence is required. But when there is no vector space structure, linear independence does not make any sense. A different notion of “independence” needed to be defined in order to tackle this problem. For any set of elements $r_1, \dots, r_k \in \mathcal{G}$, if there exists $\lambda_1, \dots, \lambda_k \in \mathbb{N} \cup \{0\}$, with at least one λ_j an unit in $\mathbb{Z}_{\mathcal{L}}$ (that is, λ_j is invertible in $\mathbb{Z}_{\mathcal{L}}$), $\mathcal{L} = LCM\{p_1^{m_1}, \dots, p_n^{m_n}\}$, such that $\sum_{i=1}^k \lambda_i r_i = 0$, then we call r_1, \dots, r_k to be dependent. If there does not exist any such λ_i ’s, then r_1, \dots, r_k are independent. The idea is, if there exists such $\lambda_1, \dots, \lambda_k \in \mathbb{Z}_{\mathcal{L}}$ with λ_j an unit, then r_j can be written as a combination of the other elements. That is,

$$r_j = -\lambda_j^{-1} \left(\sum_{i \in [k], i \neq j} \lambda_i r_i \right).$$

So, whenever we define a group automorphism on the elements $r_1, \dots, r_{j-1}, r_{j+1}, \dots, r_k$, by the property of homomorphism it automatically gets defined on r_j . This plays an important role in the Algorithm (see [20] for more details).

- We know that the characters of a Boolean function from \mathbb{Z}_2^n to $\{0, 1\}$ takes values -1 or $+1$ at each point of \mathbb{Z}_2^n . So, to determine the dominating character in a bucket at a point $x_i \in M$ with high probability, determining the sign of $P_{C_\alpha} f(y_i) P_{C_\alpha} f(y_i + x_i)$ is sufficient, where $\chi_{\alpha(C)}$ being the dominating character of the bucket C_α .

But in our case, the characters of a finite Abelian group \mathcal{G} are complex-valued at each point of \mathcal{G} . So sign does not make any sense here. The problem needed to be tackled differently. We first calculate $P_{C_\alpha} f(y_i) \overline{P_{C_\alpha} f(y_i - x_i)}$, and then we show that $P_{C_\alpha} f(y_i) \overline{P_{C_\alpha} f(y_i - x_i)} - |\widehat{f}(\chi_{\alpha(C)})|^2 \chi_{\alpha(C)}(x_i)$ is small with high probability, which helps us to understand the dominating character in the bucket C_α at the point $x_i \in M$. For details see [20].

2 Background on Finite Abelian Groups

Throughout this paper, \mathcal{G} denotes a finite Abelian group $\mathbb{Z}_{p_1^{m_1}} \times \dots \times \mathbb{Z}_{p_n^{m_n}}$, where p_i are primes for all $i \in [n]$ and may not be distinct, and $\omega_{\mathcal{L}}$ denotes a primitive \mathcal{L}^{th} root of unity, where $\mathcal{L} = LCM\{p_1^{m_1}, \dots, p_n^{m_n}\}$.

Let us define the characters of \mathcal{G} first.

► **Definition 12 (Character).** For $\mathcal{G} = \mathbb{Z}_{p_1^{m_1}} \times \dots \times \mathbb{Z}_{p_n^{m_n}}$, where $p_i, i \in [n]$ are primes, a character of the group \mathcal{G} is a homomorphism $\chi : \mathcal{G} \rightarrow \mathbb{C}^\times$ of \mathcal{G} , that is, χ satisfies the following: for all $x, y \in \mathcal{G}$ we have $\chi(x + y) = \chi(x)\chi(y)$.

Equivalently, a character of \mathcal{G} is of the form

$$\chi(x^{(1)}, \dots, x^{(n)}) = \chi_{r^{(1)}}(x^{(1)}) \dots \chi_{r^{(n)}}(x^{(n)}),$$

where $r^{(i)} \in \mathbb{Z}_{p_i^{m_i}}$ and $\chi_{r^{(i)}}$ is a character of $\mathbb{Z}_{p_i^{m_i}}$ and is defined by

$$\chi_{r^{(i)}}(x^{(i)}) = \omega_{p_i^{m_i}}^{r^{(i)} \cdot x^{(i)}}.$$

Thus for any $(r^{(1)}, \dots, r^{(n)}) \in \mathcal{G}$ we can define a corresponding character of \mathcal{G} as $\chi_{r^{(1)}, \dots, r^{(n)}}$ that on input $x = (x^{(1)}, \dots, x^{(n)})$ is defined as

$$\chi_{(r^{(1)}, \dots, r^{(n)})}(x^{(1)}, \dots, x^{(n)}) = \omega_{p_1^{m_1}}^{r^{(1)} \cdot x^{(1)}} \dots \omega_{p_n^{m_n}}^{r^{(n)} \cdot x^{(n)}} \quad (5)$$

$$= \omega_{\mathcal{L}}^{\sum_{i=1}^n r^{(i)} \cdot x^{(i)} \cdot \frac{n}{p_i^{m_i}}} \pmod{\mathcal{L}}, \quad (6)$$

where $\mathcal{L} = LCM\{p_1^{m_1}, \dots, p_n^{m_n}\}$.

Now let us look at some properties of characters.

► **Lemma 13.** Let χ be a character of \mathcal{G} . Then,

1. $\chi_0(x) = 1$ for all $x \in \mathcal{G}$.
2. $\chi(-x) = \chi(x)^{-1} = \overline{\chi(x)}$ for all $x \in \mathcal{G}$.
3. For any character χ of \mathcal{G} , where $\chi \neq \chi_0$, $\sum_{x \in \mathcal{G}} \chi(x) = 0$.
4. $|\chi(x)| = 1$ for all $x \in \mathcal{G}$.

Now let us define the dual group of \mathcal{G} .

► **Definition 14** (Dual group). *The set of characters of \mathcal{G} forms a group under the operation $(\chi\psi)(x) = \chi(x)\psi(x)$ and is denoted by $\widehat{\mathcal{G}}$, where χ and ψ are characters of \mathcal{G} . $\widehat{\mathcal{G}}$ is called the dual group of \mathcal{G} .*

The following theorem states that \mathcal{G} is isomorphic to its dual group.

► **Theorem 15.** $\widehat{\widehat{\mathcal{G}}} \cong \mathcal{G}$, that is \mathcal{G} and $\widehat{\mathcal{G}}$ are isomorphic to each other.

Let us look at the definition of Fourier transform for functions on \mathcal{G} .

► **Definition 16** (Fourier transform). *For any $\mathcal{G} = \mathbb{Z}_{p_1^{m_1}} \times \cdots \times \mathbb{Z}_{p_n^{m_n}}$, with $p_i, i \in [n]$ primes, and any function $f : \mathcal{G} \rightarrow \mathbb{C}$, the Fourier transform $\widehat{f} : \widehat{\mathcal{G}} \rightarrow \mathbb{C}$ is*

$$\widehat{f}(\chi_{r^{(1)}, \dots, r^{(n)}}) = \frac{1}{|\mathcal{G}|} \sum_{x \in \mathcal{G}} f(x) \omega_{p_1^{m_1}}^{-r^{(1)} \cdot x^{(1)}} \cdots \omega_{p_n^{m_n}}^{-r^{(n)} \cdot x^{(n)}},$$

where $x = (x^{(1)}, \dots, x^{(n)})$.

► **Remark 17.** The Fourier transform of a function $f : \mathcal{G} \rightarrow \mathbb{C}$ is defined by

$$\widehat{f}(\chi) = \frac{1}{|\mathcal{G}|} \sum_{x \in \mathcal{G}} f(x) \overline{\chi(x)},$$

where $\overline{\chi(x)}$ is the conjugate of $\chi(x)$. The Definition 16 follows from this, as $\chi = \chi_{r^{(1)}, \dots, r^{(n)}}$ for some $r^{(i)} \in \mathbb{Z}_{p_i^{m_i}}$, $i \in \{1, \dots, n\}$.

The following theorem states that any function from \mathcal{G} to \mathbb{C} can be written as a linear combination of characters of \mathcal{G} .

► **Theorem 18** (Fourier inversion formula). *Any function $f : \mathcal{G} \rightarrow \mathbb{C}$ can be uniquely written as a linear combination of characters of \mathcal{G} , that is,*

$$f(x) = \sum_{\chi_{r^{(1)}, \dots, r^{(n)}} \in \widehat{\mathcal{G}}} \widehat{f}(\chi_{r^{(1)}, \dots, r^{(n)}}) \chi_{r^{(1)}, \dots, r^{(n)}}(x), \quad (7)$$

where $x = (x^{(1)}, \dots, x^{(n)})$.

► **Theorem 19** (Parseval). *For any two functions $f, g : \mathcal{G} \rightarrow \mathbb{C}$,*

$$\mathbb{E}_{x \in \mathcal{G}}[f(x) \overline{g(x)}] = \sum_{\chi \in \widehat{\mathcal{G}}} \widehat{f}(\chi) \overline{\widehat{g}(\chi)}.$$

More specifically, if $f : \mathcal{G} \rightarrow \{-1, 1\}$ is a Boolean-valued function then

$$\sum_{\chi \in \widehat{\mathcal{G}}} |\widehat{f}(\chi)|^2 = 1.$$

Now let us define the Fourier sparsity s_f of a function f on \mathcal{G} .

► **Definition 20** (Fourier Spectral Norm). *The Fourier Spectral Norm or the Spectral norm of $f : \mathcal{G} \rightarrow \mathbb{C}$, denoted by $\|f\|_1$ is defined as the sum of the absolute value of its Fourier coefficients. That is,*

$$\|f\|_1 = \sum_{r \in \mathcal{G}} |\widehat{f}(\chi_r)|.$$

► **Definition 21** (Sparsity and Fourier Support).

- *Fourier support* $\text{supp}(\hat{f})$ of a function $f : \mathcal{G} \rightarrow \mathbb{C}$ denotes the set $\{\chi \mid \hat{f}(\chi) \neq 0\}$.
- *The Fourier sparsity* s_f of a function $f : \mathcal{G} \rightarrow \mathbb{C}$ is defined to be the number of non-zero Fourier coefficients in the Fourier expansion of f (Theorem 18). Alternately, Fourier sparsity can be defined as the size of the Fourier support. In this paper, by sparsity of a function, we mean the Fourier sparsity of the function. Moreover, by s -sparse function we mean functions with Fourier Sparsity s .

We know that since \mathbb{Z}_2^n is also a vector space over the field \mathbb{Z}_2 , it has a linear algebraic structure. The characters are in one-to-one correspondence with \mathbb{Z}_2 -linear forms. However, since a general Abelian group is not a vector space, these do not hold here. But one can define something along similar lines.

An element r of \mathcal{G} is of the form $(r^{(1)}, \dots, r^{(n)})$, where $r^{(i)}$ is an element of $\mathbb{Z}_{p_i^{m_i}}$.

► **Definition 22** (Pseudo inner product). For $x, r \in \mathcal{G}$ we denote by $*$ the following pseudo inner product.

$$r * x := \left(\sum_{i=1}^n \frac{\mathcal{L}}{p_i^{m_i}} r^{(i)} \cdot x^{(i)} \right) \pmod{\mathcal{L}},$$

where $\mathcal{L} = \text{LCM}\{p_1^{m_1}, \dots, p_n^{m_n}\}$.

► **Observation 23.** $r * x \neq 0 \Rightarrow \chi_r(x) \neq 1$.

Now, we partition \mathcal{G} with the help of a normal subgroup.

► **Definition 24.** Let $r_1, \dots, r_k \in \mathcal{G}$ such that $r_j = (r_j^{(1)}, \dots, r_j^{(n)})$, where $r_j^{(i)} \in \mathbb{Z}_{p_i^{m_i}}$ for all $i \in [n]$ and $j \in [k]$. Let $b = (b_1, \dots, b_k) \in \mathbb{Z}_{\mathcal{L}}^k$, where $\mathcal{L} = \text{LCM}\{p_1^{m_1}, \dots, p_n^{m_n}\}$. We define the set V_{b, r_1, \dots, r_k} by

$$V_{b, r_1, \dots, r_k} = \{x \in \mathcal{G} : r_j * x = b_j \pmod{\mathcal{L}} \forall j \in [k]\}. \quad (8)$$

That is,

$$V_{b, r_1, \dots, r_k} = \left\{ x \in \mathcal{G} : \left(\sum_{i=1}^T \frac{\mathcal{L}}{p_i^{m_i}} r_j^{(i)} \cdot x^{(i)} \right) = b_j \pmod{\mathcal{L}} \forall j \in [k] \right\}.$$

If $b_1 = b_2 = \dots = 0$ then we use V_{0, r_1, \dots, r_k} to denote V_{b, r_1, \dots, r_k} .

► **Remark 25.** When $k = 1$, then $b \in \mathbb{Z}_{\mathcal{L}}$, and we have only one element $r \in \mathcal{G}$, so this set becomes $V_{b, r} = \{r * x = b \pmod{\mathcal{L}}\}$.

► **Lemma 26.** V_{0, r_1, \dots, r_k} is a normal subgroup of \mathcal{G} and for any b, r_1, \dots, r_k either V_{b, r_1, \dots, r_k} is a coset of V_{0, r_1, \dots, r_k} or $V_{b, r_1, \dots, r_k} = \emptyset$.

Proof. Clearly, $0 \in V_{0, r_1, \dots, r_k}$, where 0 is the identity element of \mathcal{G} . Let $x, y \in V_{0, r_1, \dots, r_k}$. Then, since $r_j * (x - y) = \sum_{i=1}^T \frac{\mathcal{L}}{p_i^{m_i}} r_j^{(i)} \cdot x^{(i)} - \sum_{i=1}^T \frac{\mathcal{L}}{p_i^{m_i}} r_j^{(i)} \cdot y^{(i)} = 0 \pmod{\mathcal{L}}$ for all $j \in [k]$, so $x - y \in V_{0, r_1, \dots, r_k}$. So V_{0, r_1, \dots, r_k} is a subgroup of \mathcal{G} , and hence a normal subgroup of \mathcal{G} since \mathcal{G} is Abelian.

Since V_{0, r_1, \dots, r_k} is a normal subgroup of \mathcal{G} we can now consider its cosets. For each coset B of V_{0, r_1, \dots, r_k} , let $a(B)$ be a fixed coset representative (we choose a coset representative and fix it all through the proof). Then, if $y \in B$, then $y = a(B) + x$, where $x \in V_{0, r_1, \dots, r_k}$. Now, for each r_j ,

$$r_j * y = r_j * (a(B) + x) = r_j * a(B),$$

since $r_j * x = 0$ for all r_j , $j \in [k]$. So, $r_j * y$ is fixed for each coset B . If we assume $r_j * a(B) = b_j \pmod{\mathcal{L}}$ for each $j \in [k]$, then we have V_{b,r_1,\dots,r_k} as a coset of V_{0,r_1,\dots,r_k} , where $b = (b_1, \dots, b_k)$.

If for a $b = (b_1, \dots, b_k) \in \mathbb{Z}_{\mathcal{L}}^k$, there does not exist any $a(B)$ such that $r_j * a(B) = b_j \forall j \in [k]$, $V_{b,r_1,\dots,r_k} = \emptyset$. \blacktriangleleft

► **Remark 27.** Note that since \mathcal{G} is an Abelian group, any subgroup of \mathcal{G} is a normal subgroup of \mathcal{G} .

► **Definition 28 (Codimension).** Let V_{b,r_1,\dots,r_k} be as defined in Definition 24. Then the codimension of V_{b,r_1,\dots,r_k} is given by

$$\text{Codim}(V_{b,r_1,\dots,r_k}) = \min_{k'} \left\{ k : \exists r'_1, \dots, r'_{k'} \in \{r_1, \dots, r_k\} \text{ such that } \right. \\ \left. V_{b,r_1,\dots,r_k} = \{x \in \mathcal{G} : r'_j * x = b_j \pmod{\mathcal{L}} \forall j \in [k']\} \right\},$$

where $b = (b_1, \dots, b_k) \in \mathbb{Z}_{\mathcal{L}}^k$, and $\mathcal{L} = \text{LCM}\{p_1^{m_1}, \dots, p_n^{m_n}\}$.

► **Definition 29 (Random coset structure).** Let us consider $\beta_1, \dots, \beta_k \in \mathcal{G}$, which are chosen independently and uniformly at random from \mathcal{G} . Also let $H = \{\alpha \in \mathcal{G} : \alpha * \beta_i = 0 \forall i \in [k]\}$. So H is a subgroup, hence a normal subgroup of \mathcal{G} . Then, we define the sets $C(b)$ by $C(b) = \{\alpha \in \mathcal{G} : \alpha * \beta_i = b_i \forall i \in [k]\}$, for all $b = (b_1, \dots, b_k) \in \mathbb{Z}_{\mathcal{L}}^k$.

► **Remark 30.** Observe that the nonempty $C(b)$ are cosets of H . We will often refer to them as buckets.

► **Definition 31.** Let G be a group, with group operation \odot . A group automorphism on G is a bijective function $\psi : G \rightarrow G$ which satisfies the following.

$$\psi(x \odot y) = \psi(x) \odot \psi(y), \quad \forall x, y \in G.$$

The set of all automorphisms of G forms a group under the composition of functions and is denoted by $\text{Aut}(G)$.

► **Definition 32.** Let r_1, \dots, r_k be elements in \mathcal{G} . We call r_1, \dots, r_k to be **dependent** if there exists $\lambda_1, \dots, \lambda_k \in \mathbb{Z}_{\mathcal{L}}$ with at least one λ_j invertible in $\mathbb{Z}_{\mathcal{L}}$ such that $\sum_{i=1}^k \lambda_i r_i = 0$.

If there does not exist any such $\lambda_1, \dots, \lambda_k \in \mathbb{Z}_{\mathcal{L}}$ with at least one λ_j invertible in $\mathbb{Z}_{\mathcal{L}}$ such that $\sum_{i=1}^k \lambda_i r_i = 0$, then we call r_1, \dots, r_k to be **independent**.

For example, consider the group \mathbb{Z}_4 . The element 3 is dependent on the element 1, but it is independent of the element 2. Whereas, 2 is dependent on 3, as $2 = 2 \cdot 3 \pmod{4}$. So the set $\{2, 3\}$ is dependent.

► **Lemma 33 (Hoeffding's Inequality).** Let X_1, \dots, X_k be real independent random variables, each taking value in $[-1, 1]$. Then

$$\Pr \left[\left| \sum_{i=1}^k X_i - \mathbb{E} \sum_{i=1}^k X_i \right| \geq \epsilon \right] \leq 2 \exp \left(-\frac{\epsilon^2}{2k} \right).$$

The following structural result about an Abelian group and its dual will play a key role in our analysis. For a first reading, the proof of this lemma may be skipped without affecting the readability of the rest of the paper.

► **Lemma 34.** *Let A be a map from \mathcal{G} to \mathcal{G} . Let us define a map $\hat{A}: \hat{\mathcal{G}} \rightarrow \hat{\mathcal{G}}$ by $\hat{A}(\chi_r) = \chi_r \circ A$. Also, let us define another map $\hat{\hat{A}}: \mathcal{G} \rightarrow \mathcal{G}$ by $\chi_{\hat{\hat{A}}(r)} = \hat{A}(\chi_r)$. Then A is an automorphism if and only if \hat{A} is an automorphism if and only if $\hat{\hat{A}}$ is an automorphism.*

Proof.

Proof of the part [A is an automorphism if and only if \hat{A} is an automorphism]

When A is an automorphism. First let us show that \hat{A} is also a homomorphism. Observe

$$\begin{aligned} \hat{A}(\chi_{r_1} \chi_{r_2})(x) &= \hat{A}(\chi_{r_1+r_2})(x) \\ &= \chi_{r_1+r_2} \circ A(x) \\ &= \chi_{r_1}(A(x)) \chi_{r_2}(A(x)) \\ &= \hat{A}(\chi_{r_1})(x) \hat{A}(\chi_{r_2})(x), \end{aligned}$$

which shows that \hat{A} is indeed a homomorphism.

To show injectivity, let $\hat{A}(\chi_r) = 1$. Then,

$$\begin{aligned} \hat{A}(\chi_r)(x) = 1 \quad \forall x \in \mathcal{G} &\Rightarrow \chi_r(A(x)) = 1 \quad \forall x \in \mathcal{G} \\ &\Rightarrow \chi_r(y) = 1 \quad \forall y \in \mathcal{G}, \quad \text{where } y = A(x) \\ &\Rightarrow \chi_r = \chi_0, \end{aligned}$$

where the equality in the second last line holds because A is an automorphism. Since χ_0 is the identity element of $\hat{\mathcal{G}}$, so \hat{A} is injective.

Since the domain and the range of \hat{A} are same, so \hat{A} is bijective, and hence, it is an automorphism on $\hat{\mathcal{G}}$.

When \hat{A} is an automorphism. To show that A is a homomorphism, we observe that, for all $r \in \mathcal{G}$,

$$\begin{aligned} \hat{A}(\chi_r)(x + y) &= \hat{A}(\chi_r)(x) \hat{A}(\chi_r)(y) = \chi_r(A(x)) \chi_r(A(y)) = \chi_r(A(x) + A(y)) \\ &\Rightarrow \chi_r(A(x + y)) = \chi_r(A(x) + A(y)) \\ &\Rightarrow \chi_r(A(x + y) - A(x) - A(y)) = 0. \end{aligned}$$

Therefore, $A(x + y) = A(x) + A(y)$, hence A is a homomorphism.

Now let us show that A is injective. To do that, let $A(x) = 0$. Then,

$$\hat{A}(\chi_r)(x) = \chi_r(A(x)) = \chi_r(0) = 1 \quad \forall r \in \mathcal{G},$$

which is only possible if $x = 0$, since \hat{A} is an automorphism and maps characters to characters. So, A is injective.

Since the domain and the range of A are same, so A is bijective, and hence, A is an automorphism.

Proof of the part [\hat{A} is an automorphism if and only if $\hat{\hat{A}}$ is an automorphism]

When \hat{A} is an automorphism. Then, for each $x \in \mathcal{G}$,

$$\begin{aligned} \chi_{\hat{\hat{A}}(r_1+r_2)}(x) &= \hat{A}(\chi_{r_1+r_2})(x) = \chi_{r_1+r_2}(A(x)) = \chi_{r_1}(A(x)) \chi_{r_2}(A(x)) = \hat{A}(\chi_{r_1})(x) \hat{A}(\chi_{r_2})(x) \\ &\Rightarrow \chi_{\hat{\hat{A}}(r_1+r_2)}(x) = \chi_{\hat{\hat{A}}(r_1)}(x) \chi_{\hat{\hat{A}}(r_2)}(x) \\ &\Rightarrow \chi_{\{\hat{\hat{A}}(r_1+r_2) - \hat{\hat{A}}(r_1) - \hat{\hat{A}}(r_2)\}}(x) = 0, \end{aligned}$$

which implies that $\widehat{A}(r_1 + r_2) = \widehat{A}(r_1) + \widehat{A}(r_2)$ for all $r_1, r_2 \in \mathcal{G}$. So \widehat{A} is a homomorphism.

Now, let $\widehat{A}(r) = 0$. Then, for all $x \in \mathcal{G}$,

$$\chi_{\widehat{A}(r)}^{\widehat{A}}(x) = 1 \Rightarrow \widehat{A}(\chi_r)(x) = 1 \Rightarrow \chi_r(A(x)) = 1.$$

Since \widehat{A} is an automorphism, so A is an automorphism by the first part of the proof, therefore $\chi_r(y) = 1$ for all $y \in \mathcal{G}$, where $y = A(x)$. So $r = 0$, hence \widehat{A} is injective.

Since the domain and the range of \widehat{A} is same, so \widehat{A} is bijective, and hence, it is an automorphism.

When \widehat{A} is an automorphism. Then, for each $x \in \mathcal{G}$,

$$\begin{aligned} \widehat{A}(\chi_{r_1+r_2})(x) &= \chi_{\widehat{A}(r_1+r_2)}^{\widehat{A}}(x) = \chi_{\widehat{A}(r_1)+\widehat{A}(r_2)}^{\widehat{A}}(x) = \chi_{\widehat{A}(r_1)}^{\widehat{A}}(x) \chi_{\widehat{A}(r_2)}^{\widehat{A}}(x) \\ &\Rightarrow \widehat{A}(\chi_{r_1} \chi_{r_2})(x) = \widehat{A}(\chi_{r_1})(x) \widehat{A}(\chi_{r_2})(x), \end{aligned}$$

which shows that \widehat{A} is a homomorphism.

Now, let $\widehat{A}(\chi_r)(x) = 1$ for all $x \in \mathcal{G}$. Then, for all $x \in \mathcal{G}$,

$$\chi_{\widehat{A}(r)}^{\widehat{A}}(x) = 1 \Rightarrow \widehat{A}(r) = 0 \Rightarrow r = 0,$$

since \widehat{A} is an automorphism. Therefore, \widehat{A} is injective.

Since the domain and the range of \widehat{A} is same, so \widehat{A} is bijective. Hence, \widehat{A} is an automorphism. \blacktriangleleft

► **Definition 35** (Boolean function isomorphism). *Let $f, g : \mathcal{G} \rightarrow \{-1, +1\}$ be Boolean valued functions. Then f is isomorphic to g if there exists an automorphism A on \mathcal{G} such that $f = g \circ A$.*

► **Definition 36** (Automorphism distance). *Let $f, g : \mathcal{G} \rightarrow \{-1, +1\}$ be Boolean valued functions. The automorphism distance between f and g is defined by*

$$\text{dist}_{\mathcal{G}}(f, g) = \min_{A \in \text{Aut}(\mathcal{G})} \delta(f, g \circ A),$$

where $\delta(f, g \circ A)$ is the fractional Hamming distance between f and $g \circ A$.

► **Definition 37** (ϵ -close and ϵ -far from isomorphic). *Let $f, g : \mathcal{G} \rightarrow \{-1, +1\}$ be two Boolean valued functions. Then f is said to be ϵ -close (ϵ -far) from being isomorphic to g if $\text{dist}_{\mathcal{G}}(f, g) \leq \epsilon$ ($\text{dist}_{\mathcal{G}}(f, g) \geq \epsilon$). That is, f is ϵ -close to being isomorphic to g if there exists an automorphism A on \mathcal{G} such that $\delta(f, g \circ A) \leq \epsilon$; and f is ϵ -far from being isomorphic to g if for all automorphism A on \mathcal{G} , $\delta(f, g \circ A) \geq \epsilon$.*

► **Lemma 38.** *For any automorphism $A : \mathcal{G} \rightarrow \mathcal{G}$, we have*

$$\widehat{f \circ A}(\chi_r) = \widehat{f}(\chi_{\widehat{A^{-1}}(r)}).$$

Proof. Observe

$$\begin{aligned}
 \widehat{f \circ A}(\chi_r) &= \frac{1}{|\mathcal{G}|} \sum_{x \in \mathcal{G}} f \circ A(x) \chi_r(x) \\
 &= \frac{1}{|\mathcal{G}|} \sum_{x \in \mathcal{G}} f(A(x)) \chi_r(x) \\
 &= \frac{1}{|\mathcal{G}|} \sum_{y \in \mathcal{G}} f(y) \chi_r(A^{-1}(y)) && \text{where } y = A(x) \\
 &= \frac{1}{|\mathcal{G}|} \sum_{y \in \mathcal{G}} f(y) \widehat{A^{-1}}(\chi_r)(y) && \text{by Lemma 34} \\
 &= \frac{1}{|\mathcal{G}|} \sum_{y \in \mathcal{G}} f(y) \chi_{\widehat{A^{-1}(r)}}(y) && \text{by Lemma 34} \\
 &= \widehat{f}(\chi_{\widehat{A^{-1}(r)}}).
 \end{aligned}$$

3 The subgroup H^\perp

Throughout this section, \mathcal{G} denotes the finite Abelian group $\mathbb{Z}_{p_1^{m_1}} \times \cdots \times \mathbb{Z}_{p_n^{m_n}}$, where p_i are primes for all $i \in [n]$ and not necessarily distinct, $\langle x \rangle$ denotes the subgroup generated by x , and $\omega_{\mathcal{L}}$ denotes a primitive \mathcal{L}^{th} root of unity and $\mathcal{L} = LCM\{p_1^{m_1}, \dots, p_n^{m_n}\}$. Let H be a subgroup of \mathcal{G} .

Let us look at the definition of H^\perp .

► **Definition 39** (Definition 1). *Let H be a subgroup of \mathcal{G} . Then H^\perp is the subgroup given by*

$$H^\perp := \{x \in \mathcal{G} : x * y = 0 \ \forall y \in H\},$$

where

$$x * y := \left(\sum_{i=1}^T \frac{\mathcal{L}}{p_i^{m_i}} x^{(i)} \cdot y^{(i)} \right) \pmod{\mathcal{L}},$$

and $\mathcal{L} = LCM\{p_1^{m_1}, \dots, p_n^{m_n}\}$.

► **Remark 40.** Observe that H^\perp is also a subgroup of \mathcal{G} .

▷ **Claim 41.** For $z \in H^\perp$,

$$\sum_{\beta \in H} \omega_{\mathcal{L}}^{\beta * z} = |H|,$$

where $\omega_{\mathcal{L}}$ is a primitive \mathcal{L}^{th} root of unity of order \mathcal{L} , and $|H|$ denotes the order of the subgroup H .

Also, for $z \notin H^\perp$,

$$\sum_{\beta \in H} \omega_{\mathcal{L}}^{\beta * z} = 0.$$

Proof.

Case 1: When $z \in H^\perp$. Then $z * x = 0$ for all $x \in H$. Hence

$$\sum_{\beta \in H} \omega_{\mathcal{L}}^{\beta * z} = |H|.$$

Case 2: When $z \notin H^\perp$. Let $\sum_{\beta \in H} \chi_\beta(z) = A$. Since $z \notin H^\perp$, so, by Definition 39, there exists $\gamma \in H$ such that $\gamma * z \neq 0$, that is, $\chi_\gamma(z) \neq 1$. (see Observation 23) Then,

$$\begin{aligned} \chi_\gamma(z) \times A &= \chi_\gamma(z) \sum_{\beta \in H} \chi_\beta(z) \\ &= \sum_{\beta \in H} \chi_{\beta+\gamma}(z) \\ &= \sum_{\gamma' \in H} \chi_{\gamma'}(z), & \gamma' = \beta + \gamma \\ &= A, \end{aligned}$$

which implies that

$$A(\chi_\gamma(z) - 1) = 0 \Rightarrow A = 0,$$

since $\chi_\gamma(z) \neq 1$. ◁

We need the following isomorphism result:

► **Lemma 42.** H^\perp is isomorphic to the quotient group \mathcal{G}/H .

Proof. We will show that $\widehat{H^\perp}$ is isomorphic to $\widehat{\mathcal{G}/H}$, which implies that H^\perp is isomorphic to the quotient group \mathcal{G}/H , as $G \equiv \widehat{G}$ for any group G .

The set of characters of H^\perp is given by

$$\text{Ann}_{\mathcal{G}}(H) = \{\chi \in \widehat{\mathcal{G}} : \chi(y) = 1 \ \forall y \in H\},$$

where $\text{Ann}_{\mathcal{G}}(H)$ is known as the annihilator of the subgroup H in \mathcal{G} . From Definition 39, observe that for $x \in \mathcal{G}$, $\chi_x(y) = \omega_{\mathcal{L}}^{x*y} = 1$ if and only if $x * y = 0 \Leftrightarrow x \in H^\perp$.

Let us define a group homomorphism $\mathcal{F} : \widehat{\mathcal{G}/H} \rightarrow \text{Ann}_{\mathcal{G}}(H)$ by $\mathcal{F}(\zeta) = \zeta \circ q$, where $q : \mathcal{G} \rightarrow \mathcal{G}/H$ is the quotient map and $\widehat{\mathcal{G}/H}$ is the quotient group homomorphism defined by $q(r) = r + H$.

- (**\mathcal{F} is injective**) Let $\zeta \in \widehat{\mathcal{G}/H}$ such that $\zeta \circ q = \tilde{0}$, where $\tilde{0} = (0, \dots, 0)[T \text{ times}]$ is the identity element of \mathcal{G} . So, $\zeta(r + H) = \zeta \circ q(r) = 1$ for all $r \in \mathcal{G}$, which implies ζ is the identity element of \mathcal{G}/H . Therefore, \mathcal{F} is injective.
- (**\mathcal{F} is surjective**) Let $\psi \in \text{Ann}_{\mathcal{G}}(H)$. Let $\zeta : \mathcal{G}/H \rightarrow \mathbb{C}$ by $\zeta(r + H) = \psi(r)$ for all $r \in \mathcal{G}$. Since $\chi_{r+H}(x) = \omega_{\mathcal{L}}^{r*x+H*x}$, so any character χ_{r+H} of \mathcal{G} is a character of \mathcal{G}/H if $H * x = 0$ for all $x \in \mathcal{G}$ (as then, the value of $\omega_{\mathcal{L}}^{r*x+H*x}$ will be determined only by the coset representatives). Since $\psi(r + H) = \psi(r)\psi(H) = \psi(r)$ and H is the identity element of \mathcal{G}/H , so ζ is a character of \mathcal{G}/H . Also, $\psi = \zeta \circ q$. Therefore, $\mathcal{F}(\zeta) = \zeta \circ q = \psi$. Hence, \mathcal{F} is surjective.

Therefore, \mathcal{F} is an isomorphism, which implies that H^\perp is isomorphic to the quotient group \mathcal{G}/H . ◀

► **Corollary 43.** $|H| \times |H^\perp| = |\mathcal{G}|$.

Proof. Follows from the fact that $|H^\perp| = \frac{|\mathcal{G}|}{|H|}$, since H^\perp is isomorphic to the quotient group \mathcal{G}/H by Lemma 42, and $|\mathcal{G}/H| = \frac{|\mathcal{G}|}{|H|}$ by Lagrange's theorem. ◀

References

- 1 Noga Alon and Eric Blais. Testing Boolean Function Isomorphism. In *Proceedings of the 14th International Workshop on Randomization and Computation, RANDOM*, pages 394–405, 2010. doi:10.1007/978-3-642-15369-3_30.
- 2 Noga Alon, Eric Blais, Sourav Chakraborty, David García-Soriano, and Arie Matsliah. Nearly tight bounds for testing function isomorphism. *SIAM J. Comput.*, 42(2):459–493, 2013. doi:10.1137/110832677.
- 3 Prashanth Amireddy, Amik Raj Behera, Manaswi Paraashar, Srikanth Srinivasan, and Madhu Sudan. Local correction of linear functions over the boolean cube. In *Proceedings of the 56th Annual ACM Symposium on Theory of Computing*, pages 764–775, 2024. doi:10.1145/3618260.3649746.
- 4 Nikhil Bansal and Makrand Sinha. k-Forrelation Optimally Separates Quantum and Classical Query Complexity. In *Proceedings of the 53rd Annual ACM SIGACT Symposium on Theory of Computing, STOC*, pages 1303–1316, 2021. doi:10.1145/3406325.3451040.
- 5 A. Bemporad. Efficient Conversion of Mixed Logical Dynamical Systems Into an Equivalent Piecewise Affine Form. *IEEE Transactions on Automatic Control*, 49(5):832–838, 2004. doi:10.1109/TAC.2004.828315.
- 6 Arnab Bhattacharyya, Eldar Fischer, Hamed Hatami, Pooya Hatami, and Shachar Lovett. Every locally characterized affine-invariant property is testable. In *Proceedings of the 45th ACM Symposium on Theory of Computing, STOC*, pages 429–436, 2013. doi:10.1145/2488608.2488662.
- 7 Arnab Bhattacharyya, Elena Grigorescu, and Asaf Shapira. A Unified Framework for Testing Linear-Invariant Properties. *Random Struct. Algorithms*, 46(2):232–260, 2015. doi:10.1002/RSA.20507.
- 8 Eric Blais and Ryan O’Donnell. Lower Bounds for Testing Function Isomorphism. In *Proceedings of the 25th Annual IEEE Conference on Computational Complexity, CCC*, pages 235–246, 2010. doi:10.1109/CCC.2010.30.
- 9 Eric Blais, Amit Weinstein, and Yuichi Yoshida. Partially symmetric functions are efficiently isomorphism testable. *SIAM J. Comput.*, 44(2):411–432, 2015. doi:10.1137/140971877.
- 10 Sourav Chakraborty, Eldar Fischer, David García-Soriano, and Arie Matsliah. Junto-symmetric functions, hypergraph isomorphism and crunching. In *Proceedings of the 27th Conference on Computational Complexity, CCC 2012, Porto, Portugal, June 26-29, 2012*, pages 148–158. IEEE Computer Society, 2012. doi:10.1109/CCC.2012.28.
- 11 Sourav Chakraborty, David García-Soriano, and Arie Matsliah. Nearly tight bounds for testing function isomorphism. In *Proceedings of the 22nd Annual ACM-SIAM Symposium on Discrete Algorithms, SODA*, pages 1683–1702, 2011. doi:10.1137/1.9781611973082.130.
- 12 Mei-Chu Chang. A polynomial bound in Freiman’s theorem. *Duke Mathematical Journal*, 113(3):399–419, 2002.
- 13 Eshan Chattopadhyay, Pooya Hatami, Shachar Lovett, and Avishay Tal. Pseudorandom Generators from the Second Fourier Level and Applications to AC0 with Parity Gates. In *Proceedings of the 10th Innovations in Theoretical Computer Science Conference, ITCS*, volume 124, pages 22:1–22:15, 2019. doi:10.4230/LIPIcs.ITCS.2019.22.
- 14 Jung Hee Cheon, Hyunsook Hong, Joohee Lee, and Jooyoung Lee. An efficient affine equivalence algorithm for multiple s-boxes and a structured affine layer. In *International Conference on Selected Areas in Cryptography*, pages 299–316, 2016. doi:10.1007/978-3-319-69453-5_17.
- 15 Tsun-Ming Cheung, Hamed Hatami, Rosie Zhao, and Itai Zilberstein. Boolean functions with small approximate spectral norm. *Discrete Analysis*, to appear, 2022.
- 16 V. Ciriani. Synthesis of SPP three-level logic networks using affine spaces. *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, 22(10):1310–1323, 2003. doi:10.1109/TCAD.2003.818121.
- 17 Paul J. Cohen. On a Conjecture of Littlewood and Idempotent Measures. *American Journal of Mathematics*, 82(2):191–212, 1960.

- 18 Lingguo Cui and Yuanda Cao. A new S-box structure named Affine-Power-Affine. *International Journal of Innovative Computing, Information and Control*, 3(3):751–759, 2007.
- 19 Swarnalipa Datta, Arijit Ghosh, Chandrima Kayal, Manaswi Paraashar, and Manmatha Roy. Spectral norm, economical sieve, and linear invariance testing of boolean functions, 2025.
- 20 Swarnalipa Datta, Arijit Ghosh, Chandrima Kayal, Manaswi Paraashar, and Manmatha Roy. Testing isomorphism of boolean functions over finite abelian groups, 2025. [arXiv:2507.07654](#).
- 21 S. Dautovic and L. Novak. A comment on “Boolean functions classification via fixed polarity Reed-Muller form”. *IEEE Transactions on Computers*, 55(8):1067–1069, 2006. doi:10.1109/TC.2006.114.
- 22 David Steven Dummit and Richard M Foote. *Abstract Algebra*, volume 3. Wiley Hoboken, 2004.
- 23 Orr Dunkelman, Nathan Keller, and Adi Shamir. Minimalism in cryptography: The even-mansour scheme revisited. In *Advances in Cryptology – EUROCRYPT 2012: 31st Annual International Conference on the Theory and Applications of Cryptographic Techniques, Cambridge, UK, April 15-19, 2012. Proceedings 31*, pages 336–354, 2012. doi:10.1007/978-3-642-29011-4_21.
- 24 I. M. Duursma, C. Rentería, and H. Tapia-Recillas. Reed-muller codes on complete intersections. *Applicable Algebra in Engineering, Communication and Computing*, 11(6):455–462, 2001. doi:10.1007/S002000000047.
- 25 Vitaly Feldman, Parikshit Gopalan, Subhash Khot, and Ashok Kumar Ponnuswami. New Results for Learning Noisy Parities and Halfspaces. In *Proceedings of the 47th Annual IEEE Symposium on Foundations of Computer Science, FOCS*, pages 563–574, 2006. doi:10.1109/FOCS.2006.51.
- 26 Eldar Fischer, Guy Kindler, Dana Ron, Shmuel Safra, and Alex Samorodnitsky. Testing Juntas. *J. Comput. Syst. Sci.*, 68(4):753–787, 2004. doi:10.1016/J.JCSS.2003.11.004.
- 27 Michael A. Forbes and Zander Kelley. Pseudorandom Generators for Read-Once Branching Programs, in Any Order. In *59th IEEE Annual Symposium on Foundations of Computer Science, FOCS 2018, Paris, France, October 7-9, 2018*, pages 946–955, 2018. doi:10.1109/FOCS.2018.00093.
- 28 Uma Girish, Avishay Tal, and Kewen Wu. Fourier Growth of Parity Decision Trees. In *Proceedings of the 36th Computational Complexity Conference, CCC*, volume 200, pages 39:1–39:36, 2021. doi:10.4230/LIPIcs.CCC.2021.39.
- 29 Oded Goldreich and Leonid A Levin. A hard-core predicate for all one-way functions. In *Proceedings of the twenty-first annual ACM symposium on Theory of computing*, pages 25–32, 1989. doi:10.1145/73007.73010.
- 30 Parikshit Gopalan, Ryan O’Donnell, Rocco A Servedio, Amir Shpilka, and Karl Wimmer. Testing fourier dimensionality and sparsity. *SIAM Journal on Computing*, 40(4):1075–1100, 2011. doi:10.1137/100785429.
- 31 Ben Green and Tom Sanders. Boolean functions with small spectral norm. *Geometric and Functional Analysis*, 18:144–162, 2008.
- 32 Ben Green and Tom Sanders. A quantitative version of the idempotent theorem in harmonic analysis. *Annals of Mathematics*, 168(3):1025–1054, 2008.
- 33 Elena Grigorescu, Tali Kaufman, and Madhu Sudan. 2-Transitivity Is Insufficient for Local Testability. In *Proceedings of the 23rd Annual IEEE Conference on Computational Complexity, CCC*, pages 259–267, 2008. doi:10.1109/CCC.2008.31.
- 34 Elena Grigorescu, Swastik Kopparty, and Madhu Sudan. Local decoding and testing for homomorphisms. In *International Workshop on Approximation Algorithms for Combinatorial Optimization*, pages 375–385. Springer, 2006. doi:10.1007/11830924_35.
- 35 Elena Grigorescu, Karl Wimmer, and Ning Xie. Tight Lower Bounds for Testing Linear Isomorphism. In *Proceedings of the 17th International Workshop on Randomization and Computation, RANDOM*, pages 559–574, 2013. doi:10.1007/978-3-642-40328-6_39.

- 36 Xiang-Dong Hou. Classification of cosets of the Reed-Muller code $R(m-3, m)$. *Discrete Mathematics*, 128(1):203–224, 1994. doi:10.1016/0012-365X(94)90113-9.
- 37 Tali Kaufman and Madhu Sudan. Algebraic Property Testing: The Role of Invariance. In *Proceedings of the 40th Annual ACM Symposium on Theory of Computing, STOC*, pages 403–412, 2008. doi:10.1145/1374376.1374434.
- 38 Eyal Kushilevitz and Yishay Mansour. Learning Decision Trees Using the Fourier Spectrum. *SIAM J. Comput.*, 22(6), 1993. doi:10.1137/0222080.
- 39 Raghu Meka, Omer Reingold, and Avishay Tal. Pseudorandom generators for width-3 branching programs. In *Proceedings of the 51st Annual ACM SIGACT Symposium on Theory of Computing, STOC*, pages 626–637, 2019. doi:10.1145/3313276.3316319.
- 40 Ryan O’Donnell. *Analysis of Boolean Functions*. Cambridge University Press, 2014.
- 41 Ran Raz and Avishay Tal. Oracle Separation of BQP and PH. *Journal of the ACM*, 69(4):30:1–30:21, 2022. doi:10.1145/3530258.
- 42 Omer Reingold, Thomas Steinke, and Salil P. Vadhan. Pseudorandomness for Regular Branching Programs via Fourier Analysis. In *Proceedings of the 17th International Workshop on Randomization and Computation, RANDOM*, pages 655–670, 2013. doi:10.1007/978-3-642-40328-6_45.
- 43 Amir Shpilka, Avishay Tal, and Ben I. Volk. On the Structure of Boolean Functions with Small Spectral Norm. *Comput. Complex.*, 26(1), 2017. doi:10.1007/S00037-015-0110-Y.
- 44 Avishay Tal. Tight Bounds on the Fourier Spectrum of AC0. In *Proceedings of the 32nd Computational Complexity Conference, CCC*, volume 79, pages 15:1–15:31, 2017. doi:10.4230/LIPIcs.CCC.2017.15.
- 45 Avishay Tal. Towards Optimal Separations between Quantum and Randomized Query Complexities. In *61st IEEE Annual Symposium on Foundations of Computer Science, FOCS 2020, Durham, NC, USA, November 16-19, 2020*, pages 228–239, 2020. doi:10.1109/FOCS46700.2020.00030.
- 46 Hing Yin Tsang, Chung Hoi Wong, Ning Xie, and Shengyu Zhang. Fourier Sparsity, Spectral Norm, and the Log-Rank Conjecture. In *Proceedings of the 54th Annual IEEE Symposium on Foundations of Computer Science, FOCS*, pages 658–667, 2013. doi:10.1109/FOCS.2013.76.
- 47 Karl Wimmer and Yuichi Yoshida. Testing Linear-Invariant Function Isomorphism. In *Proceedings of the 40th International Colloquium on Automata, Languages and Programming, ICALP*, volume 7965, pages 840–850, 2013. doi:10.1007/978-3-642-39206-1_71.
- 48 Boyan Yordanov, Jana Tumova, Ivana Cerna, Jiří Barnat, and Calin Belta. Temporal Logic Control of Discrete-Time Piecewise Affine Systems. *IEEE Transactions on Automatic Control*, 57(6):1491–1504, 2012. doi:10.1109/TAC.2011.2178328.