Fast Fourier Sparsity Testing

Grigory Yaroslavtsev* Samson Zhou†

October 4, 2019

Abstract

A function $f : \mathbb{F}_2^n \to \mathbb{R}$ is *s-sparse* if it has at most s non-zero Fourier coefficients. Motivated by applications to fast sparse Fourier transforms over $\mathbb{F}_2^n$, we study efficient algorithms for the problem of approximating the ℓ_2-distance from a given function to the closest s-sparse function. While previous works (e.g., Gopalan *et al.* SICOMP 2011) study the problem of distinguishing s-sparse functions from those that are far from s-sparse under Hamming distance, to the best of our knowledge no prior work has explicitly focused on the more general problem of distance estimation in the ℓ_2 setting, which is particularly well-motivated for noisy Fourier spectra. Given the focus on efficiency, our main result is an algorithm that solves this problem with query complexity $\mathcal{O}(s)$ for constant accuracy and error parameters, which is only quadratically worse than applicable lower bounds.

1 Introduction

The *Fourier representation* of the function $f : \mathbb{F}_2^n \to \mathbb{R}$ is the function $\hat{f} : \mathbb{F}_2^n \to \mathbb{R}$ defined by the forward Fourier transform $\hat{f}(\alpha) = \mathbb{E}_{x \in \mathbb{F}_2^n}[f(x)\chi_\alpha(x)]$ and its inverse $f(x) = \sum_{\alpha \in \mathbb{F}_2^n} \hat{f}(\alpha)\chi_\alpha(x)$, where for each $\alpha \in \mathbb{F}_2^n$, the function $\chi_\alpha : \mathbb{F}_2^n \to \mathbb{R}$ is defined by $\chi_\alpha(x) = (-1)^{\sum_{i=1}^n \alpha_i x_i}$. The values $\hat{f}(\alpha)$ are the *Fourier coefficients* of f. When f has at most s non-zero Fourier coefficients, we say that it is *Fourier s-sparse*, or just *s-sparse* for short. The Fourier sparsity of functions plays an important role in many different areas of computer science, including error-correcting codes [GL89, AGS03], learning theory [KM93, LMN93], communication complexity [ZS09, BC99, MO09, TWXZ13], property testing [GOS+11, WY13], and parity decision tree complexity [ZS10, STlV14].

There has also been renewed interest in the Fourier sparsity of functions over various finite abelian groups with the recent development of specialized Fourier transform algorithms for such functions [HIKP12a, HIKP12b]. These algorithms improve on the efficiency of the standard Fast Fourier Transform algorithms for functions with sparse Fourier transforms by taking advantage of this sparsity itself. Since many functions (and/or signals) in practical applications *do* display Fourier sparsity, this line of research has yielded many exciting applications as well as theoretical contributions (see [HIK+13] for details). For example, much of the recent work on the sparse Fourier transform has focused on functions over fundamental domains, such as the line or the hypergrid. Meanwhile, a sparse Fourier transform for functions over $\mathbb{F}_2^n$ has been known for over twenty years as the Goldreich–Levin [GL89] and Kushilevitz–Mansour [KM93] algorithm. This algorithm can

*Indiana University, Bloomington & The Alan Turing Institute, London, UK. E-mail: gyarosla@iu.edu
†Indiana University, Bloomington. E-mail: samsonzhou@gmail.com

learn functions that are (close to) s-sparse, using time and query complexity $\mathrm{poly}(n, s)$. Since many classes of functions over $\mathbb{F}_2^n$ are known to be close to being s-sparse for a certain value of s (e.g., monotone functions, decision trees, r-DNF formulas, etc.), the sparse Fourier transform given by the GL/KM-algorithm is one of the cornerstones of computational learning theory.

One of the main limitations of the sparse Fourier transform as a technique is the fact that its efficiency is conditional on the assumption that the data of interest can be sparsely represented in the Fourier domain. Hence in order to reliably use sparse Fourier transform algorithms it is beneficial to have a way to *test* if a function is s-sparse or, more generally, to estimate the distance of a function to the closest s-sparse function. For such tasks *property testing* algorithms often come into play as a preprocessing step (see, e.g., [Ron08]) since they typically require a much smaller number of samples and other resources such as time and space. An important consideration when using property testing is the fact that presence of two kinds of noise in the data must be tolerated: small fraction of errors/outliers [PRR06] (noise of small Hamming weight) as well as arbitrary noise with small ℓ_p-norm [BRY14]. Since the performance of sparse FFT algorithms is conditioned on the sparsity under ℓ_2^2-distance, the subject of our study is to what extent can sparsity under ℓ_2^2 distance be tested. The fundamental reason why ℓ_2^2-distance plays a special role in the Fourier domain is its relation to the energy of the signal that is proportional to the sum of squares of the Fourier coefficients according to Parseval's theorem.

Formally, we define the ℓ_2^2-*distance* between f and g as $\mathrm{dist}_2^2(f, g) = \|f - g\|_2^2 = \frac{1}{2^n} \sum_{x \in \mathbb{F}_2^n} (f(x) - g(x))^2$ and the ℓ_2^2-distance between f and a class $\mathcal{P}$ of functions as $\mathrm{dist}_2^2(f, \mathcal{P}) = \min_{g \in \mathcal{P}} \mathrm{dist}_2^2(f, g)$. The distance to Fourier s-sparsity is the latter distance when $\mathcal{P}$ is the set of functions with Fourier sparsity at most s. We denote the class of all Fourier s-sparse functions as $\mathcal{F}_s$. Hence our main goal is to estimate $\mathrm{dist}_2^2(f, \mathcal{F}_s)$ up to an additive error $\pm\epsilon$. We refer to it as ℓ_2^2-distance estimation problem and to the closely related decision version as tolerant ℓ_2^2-testing.

We also reserve the name of non-tolerant ℓ_2^2-testing for an easier promise problem of distinguishing functions with Fourier sparsity at most s from those that are ϵ-far from having such sparsity, i.e., $\mathrm{dist}_2^2(f, \mathcal{F}_s) \geq \epsilon$. Note that when working with noisy Fourier spectra, where most of the Fourier coefficients are non-zero, this decision can be trivial when $s \ll 2^n$, as an ℓ_2^2-testing algorithm can just always reject. Hence the distance estimation problem described above can be substantially harder for such spectra. To simplify presentation, we call a class ϵ-testable with q queries if there exists an algorithm which makes q queries and achieves the above guarantee with constant probability. We will also use Hamming distance while keeping the rest of the definitions the same in order to describe some of the previous work in the area of property testing. In this case the distance between f and g is defined as $\Pr_{x \sim \mathbb{F}_2^n}[f(x) \neq g(x)]$ and all the definitions above are changed accordingly.

1.1 Previous work

The most direct approach for ℓ_2^2-distance estimation and ℓ_2^2-testing is to use the testing-by-learning approach established by Goldreich, Goldwasser, and Ron [GGR98]. Using the Goldreich–Levin / Kushilevitz–Mansour algorithm [GL89, KM93], we can learn an s-sparse function h that will be essentially as close to f as possible. We can then estimate the distance between f and h to get a good approximation of the distance from f to Fourier s-sparsity. This approach requires $\mathcal{O}(sn)$ queries in order to achieve constant error ϵ (see, e.g., the textbook exposition in [Gol01, O'D14]). An improvement to this approach would be to use hashing to reduce the dimension down to a subspace of size $\mathcal{O}(s^2)$ (thus introducing no collisions between the top s coefficients) and then run GL/KM within the subspace. The complexity of this approach would be $\mathcal{O}(s \log s)$ queries for

constant ϵ, where the $\log s$ factor results from using $\mathcal{O}\left(s^2\right)$ buckets to avoid collisions among the top s coefficients. Other related previous work (e.g. [BBG18] who study testing sparsity over known and unknown bases, including the Fourier basis) also incurs extra factors in query complexity. [1]

The first specialized algorithm for the problem of testing Fourier sparsity under Hamming distance was developed by Gopalan et al. [GOS+11]. They give a *non-tolerant* tester for Fourier s-sparsity under Hamming distance with a number of queries to f that is *independent* of n and *polynomial* in s and $1/\epsilon$. More precisely, the focus of [GOS+11] was a slightly different problem where the class $\mathcal{P}$ of interest is defined to contain only Boolean s-sparse functions. Below we will refer to this class as $\mathcal{F}_s^{0/1}$.[2] However, in fact [GOS+11] show that with some loss in parameters, this problem can be reduced to estimating ℓ_2^2-distance from $\mathcal{F}_s$, the problem that we study in this paper. Thus an implicit ingredient of the [GOS+11] algorithm is a ℓ_2^2-distance estimation algorithm for $\mathcal{F}_s$ with query complexity $\mathcal{O}\left(\text{poly}(s)\right)$ for any constant additive error.

An active line of previous work focuses on *tolerant* testing under Hamming distance. Wimmer and Yoshida [WY13] showed that the general approach of [GOS+11] can be extended to yield tolerant testers for Fourier s-sparsity of Boolean functions. Specifically, they give an algorithms that distinguish between functions that are $\epsilon/3$-close to Fourier s-sparse from those that are ϵ-far from Fourier s-sparse under Hamming distance, using $\text{poly}(s)$ queries. This allows one to approximate the distance to Fourier s-sparsity up to some *multiplicative* factor. The polynomial dependence on s is fairly large and the result does not extend to additive error. Algorithms for estimating the Hamming distance to Fourier s-sparsity up to an additive error can be also derived through a general framework of Hatami and Lovett [HL13]. However, the instantiation of the [HL13] framework results in power tower dependency on s.

1.2 Our Contributions

We introduce two new algorithms for testing Fourier s-sparsity with respect to ℓ_2^2-distance. Our first main result shows that one can approximate the distance to Fourier s-sparsity in ℓ_2^2-distance with a number of non-adaptive queries that is in fact *linear* in s. This result is proved in Section 3.

Theorem 1.1 *(Approximating ℓ_2^2-distance to s-sparsity) For any $s \geq 1$ and $\epsilon > 0$, there is an algorithm that given non-adaptive query access to a function $f : \mathbb{F}_2^n \to \mathbb{R}$ with unit ℓ_2-norm takes at most $\mathcal{O}\left(\frac{s}{\epsilon^4} \log \frac{1}{\epsilon} \log \frac{1}{\delta}\right)$ queries and approximates $\text{dist}_2^2(f, \mathcal{F}_s)$ up to an additive error $\pm\epsilon$ with probability $1 - \delta$ and running time $\tilde{\mathcal{O}}\left(\frac{s}{\epsilon^4}\right)$ (see Section 3.)*

Here, the $\tilde{\mathcal{O}}$ notation suppresses polylogarithmic factors in s and $\frac{1}{\epsilon}$.

As mentioned before, the main challenge in testing Fourier s-sparsity with respect to ℓ_2^2-distance instead of Hamming distance seems to be the accurate estimation of a large number of possibly small nonzero Fourier coefficients using a small number of queries. Whereas a function can only be ϵ-far from Fourier s-sparsity with respect to Hamming distance by having a large number of nonzero Fourier coefficients, a function can be ϵ-far from Fourier s-sparsity with respect to ℓ_2^2-distance

[1] Also, since [BBG18] handles a much more general problem in order to handle arbitrary design matrices, the running time of their algorithms translates to polynomial in 2^n in our case, which can be prohibitively large for our application.

[2] While $\mathcal{F}_s^{0/1} \subseteq \mathcal{F}_s$ in general there is no known relationship between testing and distance estimation query complexities of classes and their subclasses.

by either having too many large Fourier coefficients or a large number of small nonzero Fourier coefficients.

Instead of estimating these small Fourier coefficients, we randomly partition the set of Fourier coefficients into a number of cosets by first picking a random subspace H and measuring the energy (the sum of the squared Fourier coefficients) in each coset. If H has sufficiently large codimension, then the top Fourier coefficients are partitioned into separate cosets, so the estimation of the energy in the top cosets is a good estimation of the energy of the top Fourier coefficients. To estimate the energy in each coset, we query the function at a number of random locations to obtain an empirical estimate within an additive factor of $\epsilon^2\|f\|_2^2$ with constant probability. We then bound the probability of two sources of errors: the hashing error, which originates from drawing a subspace in which large Fourier coefficients collide, and the estimation error, which results from inaccurate empirical estimations. Putting things together, we show that our estimator approximately captures the Fourier s-sparse function closest to f in ℓ_2^2-distance and hence gives a good approximation of the distance from f to the closest Fourier s-sparse function.

We also show a lower bound of $\Omega(\sqrt{s})$ for ℓ_2^2-testing of $\mathcal{F}_s$ for non-adaptive query algorithms.

Theorem 1.2 *For any $s \leq 2^{n-1}$, there exists a constant $c > 0$ such that any non-adaptive algorithm given query access to $f\colon \mathbb{F}_2^n \to \mathbb{R}$ such that $\|f\|_2^2 = 1 \pm \epsilon$ that distinguishes whether f is s-sparse or f is $\frac{1}{3}$-far from s-sparse in ℓ_2^2 with probability at least $2/3$ has to make at least $c\sqrt{s}$ queries to f (see Section 4.1).*

Our lower bound results from designing two distributions $\mathcal{D}_{YES}$ and $\mathcal{D}_{NO}$, where the distribution $\mathcal{D}_{YES}$ is the set of Fourier s-sparse functions whose Fourier coefficients are scaled Gaussian random variables whereas the $\mathcal{D}_{NO}$ distribution is the set of functions with support on *all* Fourier coefficients. The Fourier coefficients in the $\mathcal{D}_{NO}$ distribution are Gaussian random variables with a different scaling, such that the total variation distance between the $\mathcal{D}_{YES}$ and $\mathcal{D}_{NO}$ distributions restricted to a small query set is also small.

[GOS+11] gives an $\Omega(\sqrt{s})$ property testing lower bound for $\mathcal{F}_s^{0/1}$. Their results can be extended to $\mathcal{F}_s$, provided that $s \leq 2^{cn}$ for a specific constant $c > 0$, whereas our results covers the full range of values of s. Thus our results in Theorem 1.1 above are at most a quadratic factor away from optimal. We consider closing the quadratic gap in query complexity of ℓ_2^2-distance estimation for $\mathcal{F}_s$ as the main open problem posed by our work.

2 Preliminaries

For a finite set S we denote the uniform distribution over S as $U(S)$.

2.1 Fourier Analysis

We consider functions from $\mathbb{F}_2^n$ to $\mathbb{R}$. For any fixed $n \geq 1$, the space of these functions forms an inner product space with the inner product $\langle f, g\rangle = \mathbb{E}_{x\in\mathbb{F}_2^n}[f(x)g(x)] = \frac{1}{2^n}\sum_{x\in\mathbb{F}_2^n} f(x)g(x)$. The ℓ_2-norm of $f : \mathbb{F}_2^n \to \mathbb{R}$ is $\|f\|_2 = \sqrt{\langle f, f\rangle} = \sqrt{\mathbb{E}_x[f(x)^2]}$ and the ℓ_2-distance between two functions $f, g : \mathbb{F}_2^n \to \mathbb{R}$ is the ℓ_2-norm of the function $f - g$. We write $\mathrm{dist}_2(f, g) = \|f - g\|_2$. It is, in other words, $\|f - g\|_2 = \sqrt{\langle f - g, f - g\rangle} = \frac{1}{\sqrt{|\mathbb{F}_2^n|}}\sqrt{\sum_{x\in\mathbb{F}_2^n}(f(x) - g(x))^2}$.

For $\alpha \in \mathbb{F}_2^n$, the *character* $\chi_\alpha : \mathbb{F}_2^n \to \{-1, 1\}$ is the function defined by $\chi_\alpha(x) = (-1)^{\alpha\cdot x}$. The *Fourier coefficient* of $f : \mathbb{F}_2^n \to \mathbb{R}$ corresponding to α is $\hat{f}(\alpha) = \mathbb{E}_x[f(x)\chi_\alpha(x)]$. The *Fourier transform*

of f is the function $\hat{f} : \mathbb{F}_2^n \to \mathbb{R}$ that returns the value of each Fourier coefficient of f. The set of Fourier transforms of functions mapping $\mathbb{F}_2^n \to \mathbb{R}$ forms an inner product space with inner product $\left\langle \hat{f}, \hat{g} \right\rangle = \sum_{\alpha \in \mathbb{F}_2^n} \hat{f}(\alpha)\hat{g}(\alpha)$. The corresponding ℓ_2-norm is $\|\hat{f}\|_2 = \sqrt{\left\langle \hat{f}, \hat{f} \right\rangle} = \sqrt{\sum_{\alpha \in \mathbb{F}_2^n} \hat{f}(\alpha)^2}$. Note that the inner product and ℓ_2-norm are weighted differently for a function $f : \mathbb{F}_2^n \to \mathbb{R}$ and its Fourier transform $\hat{f} : \mathbb{F}_2^n \to \mathbb{R}$. We refer to the quantity $\hat{f}(\alpha)^2$ as the *energy* of a Fourier coefficient $\hat{f}(\alpha)$.

Fact 2.1 (Parseval's identity) *For any $f : \mathbb{F}_2^n \to \mathbb{R}$ it holds that $\|f\|_2 = \|\hat{f}\|_2 = \sqrt{\sum_{\alpha \in \mathbb{F}_2^n} \hat{f}(\alpha)^2}$.*

A function $f : \mathbb{F}_2^n \to \mathbb{R}$ is *Fourier s-sparse* for some sparsity s if the number of non-zero Fourier coefficients of f is at most s. We let $\mathcal{F}_s$ denote the set of all Fourier s-sparse functions.

2.2 Property Testing

We study algorithms that make queries to a given function f. In this setting two different query access models are typicaly considered. If all queries must be chosen in advance without access to the values of f, we call the corresponding algorithm *non-adaptive* or equivalently, using *non-adaptive queries*. Otherwise, the algorithm is *adaptive*, and uses *adaptive queries*, i.e. the queries made by the algorithm might depend on all previously queried values of f. In this paper, both our upper and lower bounds apply specifically to the non-adaptive query model.

We use the following standard definition of property testing under Hamming distance:

Definition 2.2 (Property testing [GGR98]) *An algorithm $\mathcal{A}$ is a property tester with parameter $\epsilon > 0$ for a class $\mathcal{C}$ of functions $f : \mathbb{F}_2^n \to \{-1, 1\}$ if given query access to f it distinguishes with probability at least $2/3$ whether $f \in \mathcal{C}$ or $\min_{g \in \mathcal{C}} \Pr_{x \sim \mathbb{F}_2^n}[f(x) \neq g(x)] \geq \epsilon$. If neither of the two conditions hold then $\mathcal{A}$ can output an arbitrary answer.*

The notions of ℓ_2^2-tester and distance approximator are defined below. In order to make ϵ be a scale-free parameter we assume that $\|f\|_2^2 = 1$ throughout this paper unless otherwise specified. For example, for Boolean functions $f : \mathbb{F}_2^n \to \{-1, 1\}$ this holds automatically and for real-valued functions this can be achieved by an appropriate scaling. The ℓ_2-distance from a function $f : \mathbb{F}_2^n \to \mathbb{R}$ to a class $\mathcal{C}$ of functions mapping $\mathbb{F}_2^n$ to $\mathbb{R}$ is $\mathrm{dist}_2(f, \mathcal{C}) = \min_{g \in \mathcal{C}} \|f - g\|_2$.

Definition 2.3 (ℓ_2^2-testing [BRY14]) *An algorithm $\mathcal{A}$ is an ℓ_2^2-tester with parameter $\epsilon > 0$ for a class $\mathcal{C}$ of functions $f : \mathbb{F}_2^n \to \mathbb{R}$ if given query access to f with unit ℓ_2-norm it distinguishes with probability at least $2/3$ whether $f \in \mathcal{C}$ or $\mathrm{dist}_2^2(f, \mathcal{C}) \geq \epsilon$.*

In order to simplify presentation we say that a function f is ϵ-far from a class $\mathcal{C}$ in some distance (e.g. Hamming or ℓ_2^2) if the closest function from $\mathcal{C}$ is at distance at least ϵ from f.

Generalizing the notion of ℓ_2^2-testing we define a notion of ℓ_2^2-distance approximation as follows:

Definition 2.4 (ℓ_2^2-distance approximator) *An algorithm $\mathcal{A}$ is an ℓ_2^2-distance approximator with parameter $\epsilon > 0$ for a class $\mathcal{C}$ of functions $f : \mathbb{F}_2^n \to \mathbb{R}$ if given query access to f with unit ℓ_2-norm it outputs an estimate ξ such that with probability at least $2/3$ it holds that $\left|\xi - \mathrm{dist}_2^2(f, \mathcal{C})\right| \leq \epsilon$.*

2.3 Fourier Hashing

We use notation $H \leq \mathbb{F}_2^n$ to denote a *subspace* H of $\mathbb{F}_2^n$. For $H \leq \mathbb{F}_2^n$ we use notation $H^\perp$ for the *orthogonal subspace* of H. Given $a \in \mathbb{F}_2^n$, the *coset* $a + H$ is defined by the set of points $a + H := \{a + h | h \in H\}$.

Definition 2.5 *For a subspace $H \leq \mathbb{F}_2^n$, an element $a \in H^\perp$, and a function $f : \mathbb{F}_2^n \to \mathbb{R}$, define the* projected function $f|_{a+H} : \mathbb{F}_2^n \to \mathbb{R}$ *to be the function that satisfies $f|_{a+H}(z) = \mathop{\mathbb{E}}_{x \in H^\perp}\left[f(x+z)\chi_a(x)\right]$ for each $z \in \mathbb{F}_2^n$. Given a subset $A \subseteq H^\perp$, we define $f|_{A+H} = \sum_{a \in A} f|_{a+H}$.*

From this definition, we observe that the values $f|_{a+H}(z)$ can all be computed simultaneously

Proposition 2.6 *The set of queries $\{f(x+z)\}_{x \in H^\perp}$ can be used to compute $f|_{a+H}(z)$ for* each *of the cosets $a + H$ of H simultaneously.*

We give more details about the number of queries required for computation of $f|_{a+H}$ in Lemma 3.8. We note that the projection of f onto the cosets of a linear subspace H yields a partition of the Fourier spectrum of f. Moreover, the projection of f to a coset $a + H$ is a function that zeroes out all Fourier coefficients not in $a + H$.

We now recall the following Poisson summation formula. For a reference, see Section 3.3 in [O'D14]. We also give the proof of Proposition 2.7 in Appendix A.2, for completeness.

Proposition 2.7 (Poisson Summation Formula) *Fix any subspace $H \leq \mathbb{F}_2^n$ and element $a \in \mathbb{F}_2^n$. Then for the projected function $f|_{a+H}$:*

(1) $f|_{a+H}(z) = \sum_{\beta \in a+H} \hat{f}(\beta)\chi_\beta(z)$

(2) $\widehat{f|_{a+H}}(\alpha) = \begin{cases} \hat{f}(\alpha) & \text{if } \alpha \in a + H \\ 0 & \text{otherwise.} \end{cases}$

Proposition 2.7 allows the following definition.

Definition 2.8 *The total energy of $f|_{a+H}$ is defined as $\sum_{\alpha \in a+H} \hat{f}(\alpha)^2 = \|\hat{f}|_{a+H}\|_2^2$.*

Fact 2.9 *When $H \leq \mathbb{F}_2^n$ is drawn uniformly at random from the set of subspaces of codimension d, then for any distinct $a, b \in \mathbb{F}_2^n \setminus \{0\}$, it holds that $\Pr[b \in a + H] = 2^{-d}$.*

Fact 2.9 allows one to think of the projections $\{f|_{a+H}\}_{a \in H^\perp}$ as a hashing process applied to the Fourier coefficients of f. In fact, it is also known (for example, by Proposition 2.9 in [GOS+11]) that random projections correspond to a *pairwise independent* hashing process.

3 ℓ_2^2-Distance Approximation and Sparsity Testing

Recall that the property testing model, initiated by [GGR98], requires an algorithm to accept objects that have some property $\mathcal{P}$ and reject objects that are at Hamming distance at least ϵ from having property $\mathcal{P}$ for some input parameter $\epsilon > 0$. In particular, in the property testing problem for s-sparsity, one would like to differentiate whether a given function $f : \mathbb{F}_2^n \to \mathbb{R}$ with $||f||_2 = 1$ is in the class $\mathcal{F}_s$ of Fourier s-sparse functions, or has distance at least ϵ from $\mathcal{F}_s$.

Problem 3.1 (Property Testing for s-Sparsity) *Let $\mathcal{F}_s$ be the class of s-sparse functions mapping from $\mathbb{F}_2^n$ to $\mathbb{R}$. Given query access to a function $f : \mathbb{F}_2^n \to \mathbb{R}$ with $||f||_2 = 1$ and parameter $\epsilon > 0$, we call an algorithm $\mathcal{A}$ a property tester with query complexity q if using at most q queries, $\mathcal{A}$ accepts f if $f \in \mathcal{F}_s$ and rejects if $\min_{g \in \mathcal{F}_s} ||f - g||_2^2 \geq \epsilon$.*

We now define the problem of energy estimation for the top s Fourier coefficients, which also allows to solve the property testing problem. Note that this energy estimation problem for functions with unit ℓ_2-norm is equivalent to the ℓ_2^2-distance approximation problem in Definition 2.4 since both are defined in terms of additive error approximation.

Problem 3.2 (Energy Estimation of top s Fourier Coefficients) *Let $\mathcal{F}_s$ be the class of s-sparse functions mapping from $\mathbb{F}_2^n$ to $\mathbb{R}$. Given non-adaptive query access to a function $f : \mathbb{F}_2^n \to \mathbb{R}$ with $||f||_2 = 1$ and parameters $s > 0$ and $0 < \epsilon \leq 1$, we call an algorithm $\mathcal{A}$ an ϵ-estimator of the energy of the top s Fourier coefficients if using at most q queries, $\mathcal{A}$ outputs ξ such that $\left|\xi - \max_{|S|=s} \sum_{\alpha \in S} \hat{f}(\alpha)^2\right| \leq \epsilon$.*

The energy estimation problem can be used to solve the property testing problem above easily with roughly the same query complexity (see Fact A.1).

Our Algorithm 1 estimates the energy of the top s Fourier coefficients by first picking a random subspace H of codimension $d = \log \frac{2s}{\epsilon^4}$ uniformly at random. The intuition is that by picking the codimension to be large enough, the top s Fourier coefficients are partitioned into cosets with only a few collisions, so the estimation of the energy in the top s cosets is a good estimation of the energy of the top s Fourier coefficients. To estimate the energy in the top s cosets, Algorithm 1 samples $\gamma = \mathcal{O}\left(\frac{s}{\epsilon^4}\|f\|_2^2\right)$ pairs $(x, x+z)$ to obtain an empirical estimate of the energy in each coset within an additive factor of $\epsilon^2\|f\|_2^2$ with constant probability. This yields the proof of Theorem 1.1. Similarly, Algorithm 2 gives a property tester for s-sparsity. The success probability for each of these algorithms can be increased to $1 - \delta$ for any $\delta > 0$ by taking the median of $\mathcal{O}\left(\log \frac{1}{\delta}\right)$ parallel repetitions.

Algorithm 1: ENERGY ESTIMATION(ϵ, s)

Draw $H \leq \mathbb{F}_2^n$ of codimension $d = \log \frac{2s}{\epsilon^4}$ uniformly at random;
for $j = 1$ *to* $\ell = \mathcal{O}\left(\log \frac{1}{\epsilon}\right)$ **do**
  $\mathcal{I}_j \leftarrow$ set of pairs $(x, x+z)$ of size $\gamma = \mathcal{O}\left(\frac{s}{\epsilon^4}\|f\|_2^2\right)$, where $x \sim U(\mathbb{F}_2^n), z \sim U(H^\perp)$;
  for *each* $a \in H^\perp$ **do**
    $y_{a+H}^{(j)} \leftarrow 0$;
    for *each* $(x, x+z) \in \mathcal{I}_j$ **do**
      $y_{a+H}^{(j)} \leftarrow y_{a+H}^{(j)} + \frac{1}{|\mathcal{I}_j|}\chi_a(z)f(x)f(x+z)$
    end
  end
end
Return: $\xi := \max_{S \subseteq H^\perp : |S|=s} \sum_{a \in S} \text{median}\left(y_{a+H}^{(1)}, y_{a+H}^{(2)}, \ldots, y_{a+H}^{(\ell)}\right)$.

Algorithm 2: Fast Fourier Sparsity Test (FFST)(ϵ, s)

Let f be some function with known $||f||_2$.

Let ξ be the output of Algorithm 1 on input $\frac{\epsilon}{2}$ and sparsity s.

If $\xi \leq \left(1 - \frac{\epsilon}{2}\right) ||f||_2^2$, reject.

Otherwise, accept.

Our analysis deals with two possible sources of error in the energy estimation. In Section 3.1, we consider the error caused by collisions in the hashing scheme and in Section 3.2, we consider the error caused by sampling variance in the energy estimates. Note that we perform worst-case analysis (over all possible sets of size s) for the hashing error as in the last step of the algorithm we adaptively select the largest subset.

3.1 Hashing Error

We first analyze the error introduced into our estimator by hashing the Fourier coefficients across multiple cosets (assuming all estimates of energies in the cosets are exact). Thus the first technical component of the analysis of the sparsity distance approximator shows that for a random choice of subspace H of codimension $\log \frac{2s}{\epsilon^4}$, the union of the top s cosets of H has total energy that is close to the sum of the Fourier mass of the s coefficients largest in magnitude.

Let $\mathcal{E}_1 \geq \cdots \geq \mathcal{E}_{2^n}$ be the true values of the energies of the 2^n Fourier coefficients corresponding to the function $f : \mathbb{F}_2^n \to \mathbb{R}$. Let h be some pairwise independent hash function with domain $[2^n]$ and range $[2^d]$, which can be viewed as partitioning the 2^n Fourier coefficients across the 2^d cosets, which we refer to as buckets. We denote the overall energy in the i-th bucket as y_i, where we assume that the hash function is clear from the context. Let the buckets be indexed in the non-increasing order by energy, so that $y_1 \geq y_2 \geq \cdots \geq y_{2^d}$. Furthermore, let y_i^* denote the energy fo the largest coefficient hashing into the i-th bucket. Formally, if index i corresponds to coset $a + H$, then we let

$$y_i = \sum_{\beta \in a+H} \hat{f}(\beta)^2, \qquad y_i^* = \max_{\beta \in a+H} \hat{f}(\beta)^2.$$

Definition 3.3 (Hashing Error) *We define the* hashing error *of h as $err_h^s(\mathcal{E}_1, \ldots, \mathcal{E}_{2^n}) = \sum_{i=1}^s y_i - \mathcal{E}_i$ to be the difference between the overall energy in the top s buckets and the energy of the top s coefficients.*

Note that the hashing error is always non-negative as there are at most s buckets containing the top s Fourier coefficients. The contribution to the energy of the i^{th} bucket from the largest Fourier coefficient hashing into this bucket is denoted as y_i^*. We have:

$$err_h^s(\mathcal{E}_1, \ldots, \mathcal{E}_{2^n}) = \sum_{i=1}^s y_i - \mathcal{E}_i = \sum_{i=1}^s y_i - y_i^* + \sum_{i=1}^s y_i^* - \mathcal{E}_i \leq \sum_{i=1}^s y_i - y_i^*,$$

where we used the fact that $\sum_{i=1}^s y_i^* \leq \sum_{i=1}^s \mathcal{E}_i$.

We can bound the hashing error across any set of s buckets, rather than just the hashing error across the buckets containing the top s Fourier coefficients.

Lemma 3.4 (Expected Hashing Error Bound) *Let $H \leq \mathbb{F}_2^n$ be a subspace of codimension d drawn uniformly at random. Let $z_i = y_i - y_i^*$ be the "collision error" in the i^{th} bucket. Then*

$$\mathop{\mathbb{E}}_{H}\left[\sum_{i=1}^{s} z_i\right] \leq \sqrt{\frac{2s}{2^d}}||f||_2^2.$$

Proof : By the Cauchy-Schwarz inequality, $\sum_{i=1}^{s} z_i \leq \sqrt{s}\sqrt{\sum_{i=1}^{s} z_i^2}$. Let δ_{jk} be the indicator variable for the event that Fourier coefficients $\mathcal{E}_j$ and $\mathcal{E}_k$ collide and let D_j be the indicator variable for the event that $\mathcal{E}_j$ is not the largest coefficient in its hash bucket. Then we have:

$$\sum_{i=1}^{s} z_i^2 \leq \sum_{i=1}^{2^d} z_i^2 = \sum_{i=1}^{2^d}(y_i - y_i^*)^2 = \sum_{j,k\in[2^n]} \mathcal{E}_j\mathcal{E}_k\delta_{jk}D_jD_k \leq \sum_{j,k\in[2^n]} \mathcal{E}_j\mathcal{E}_k\delta_{jk}D_j,$$

where the first inequality holds since the s buckets is a subset of the 2^d buckets, and the second inequality holds because either $D_k = 0$ or $D_k = 1$.

Taking expectation over H we have:

$$\mathop{\mathbb{E}}_{H}\left[\sum_{j,k\in[2^n]} \mathcal{E}_j\mathcal{E}_k\delta_{jk}D_j\right] = \mathop{\mathbb{E}}_{H}\left[\sum_{j\in[2^n]} \mathcal{E}_j^2D_j\right] + \mathop{\mathbb{E}}_{H}\left[\sum_{j\neq k\in[2^n]} \mathcal{E}_j\mathcal{E}_k\delta_{jk}D_j\right] \leq \mathop{\mathbb{E}}_{H}\left[\sum_{j\in[2^n]} \mathcal{E}_j^2D_j\right] + \frac{\left(\sum_{j=1}^{2^n}\mathcal{E}_j\right)^2}{2^d},$$

where we used Fact 2.9 and pairwise independence, so that $\mathop{\mathbb{E}}_{H}[\delta_{jk}] = \frac{1}{2^d}$. Note that by Fact 2.9, pairwise independence and a union bound, $\Pr[D_j] \leq \frac{j-1}{2^d}$ and hence for the first term we have:

$$\mathop{\mathbb{E}}_{H}\left[\sum_{j\in[2^n]} \mathcal{E}_j^2D_j\right] \leq \sum_{j=1}^{2^n}\frac{j-1}{2^d}\mathcal{E}_j^2 \leq \frac{1}{2^d}\sum_{j=1}^{2^n}\sum_{k=1}^{j-1}\mathcal{E}_j\mathcal{E}_k \leq \frac{\left(\sum_{j=1}^{2^n}\mathcal{E}_j\right)^2}{2^d}$$

Putting things together, we have

$$\mathop{\mathbb{E}}_{H}\left[\sum_{i=1}^{s} z_i\right] \leq \sqrt{s}\cdot\mathop{\mathbb{E}}_{H}\left[\sqrt{\sum_{i=1}^{s} z_i^2}\right] \leq \sqrt{s}\cdot\sqrt{\mathop{\mathbb{E}}_{H}\left[\sum_{i=1}^{s} z_i^2\right]} \leq \sqrt{\frac{2s}{2^d}}\sum_{j=1}^{2^n}\mathcal{E}_j = \sqrt{\frac{2s}{2^d}}||f||_2^2,$$

where we recall that the first inequality is by Cauchy-Schwarz, the second is by Jensen and the third is from the bound on $\mathop{\mathbb{E}}_{H}\left[\sum_{i=1}^{s} z_i^2\right]$ derived above. □

Now we give an upper bound on the variance of the difference between the energies of the top s buckets and their respective largest Fourier coefficients:

Lemma 3.5 (Variance of the Hashing Error) *Let $H \leq \mathbb{F}_2^n$ be a subspace of codimension d drawn uniformly at random. Let $z_i = y_i - y_i^*$ be the "collision error" in the i^{th} bucket. Then*

$$\mathop{\mathrm{Var}}_{H}\left[\sum_{i=1}^{s} z_i\right] \leq \frac{2||f||_2^4}{2^d}.$$

Proof : By pairwise independence we have:

$$\operatorname*{Var}_H\left[\sum_{i=1}^{s} z_i\right] \leq \operatorname*{Var}_H\left[\sum_{i=1}^{2^d} z_i\right] = \sum_{i=1}^{2^d} \operatorname*{Var}_H[z_i] \leq \sum_{i=1}^{2^d} \mathop{\mathbb{E}}_H\left[z_i^2\right] \leq \frac{2(\sum_{i=1}^{n} \mathcal{E}_i)^2}{2^d} = \frac{2||f||_2^4}{2^d},$$

where the last inequality follows using the same argument as in the proof of Lemma 3.4. □

We now give a bound on the hashing error.

Corollary 3.6 *For $2^d = \frac{2s}{\epsilon^4}$ and $0 < \epsilon \leq 1/2$, then with probability at least 15/16 over the possible choices of H:*

$$err_h^s(\mathcal{E}_1, \ldots, \mathcal{E}_{2^n}) = \sum_{i=1}^{s} y_i - \mathcal{E}_i \leq 5\epsilon^2 ||f||_2^2.$$

Proof : Let $z_i = y_i - y_i^*$ be the collision error in the i^{th} bucket and let $Z = \sum_{i=1}^{s} z_i$ and $\|\mathcal{E}\|_1 = \sum_{i=1}^{n} \mathcal{E}_i$. From Lemma 3.4, Lemma 3.5, and Chebyshev's inequality, we have that for any $\alpha > 0$:

$$\Pr\left[Z \geq \sqrt{\frac{2s}{2^d}}||f||_2^2 + \alpha\sqrt{\frac{2}{2^d}}||f||_2^2\right] \leq \frac{1}{\alpha^2}.$$

For $2^d = \frac{2s}{\epsilon^4}$ we have $\Pr[Z \geq (1 + \frac{\alpha}{\sqrt{s}})\epsilon^2||f||_2^2] \leq 1/\alpha^2$. Recall that as we already argued above:

$$\sum_{i=1}^{s} y_i - \mathcal{E}_i = \sum_{i=1}^{s} y_i - y_i^* + \sum_{i=1}^{s} y_i^* - \mathcal{E}_i \leq \sum_{i=1}^{s} y_i - y_i^*,$$

since $\sum_{i=1}^{s} y_i^* \leq \sum_{i=1}^{s} \mathcal{E}_i$. Taking $\alpha = 4$ and noting that $s \geq 1$, it follows that

$$\sum_{i=1}^{s} y_i - \mathcal{E}_i \leq 5\epsilon^2||f||_2^2$$

with probability at least 15/16. □

3.2 Estimation Error

We now analyze the error introduced to our estimator through sampling used to approximate the true bucket energies. Our intuition is the following standard fact to estimate the total energy via sampling.

Fact 3.7 (Fact 2.5 in [GOS+11]) $\sum_{\alpha \in a+H} \hat{f}(\alpha)^2 = \mathop{\mathbb{E}}_{x \in \mathbb{F}_2^n, z \in H^\perp}[\chi_a(z) f(x) f(x+z)]$.

Using Fact 3.7, the energy $\sum_{\alpha \in a+H} \hat{f}(\alpha)^2$ in each bucket $a + H$ can be approximated by repeatedly querying f using the following Lemma 3.8, whose proof is similar to Proposition 2.6 in [GOS+11]. We include the full proofs to formalize the dependency on $||f||_2$.

In the language of Lemma 3.8, suppose x_i is the energy of bucket $a + H$ and $\mathcal{I}_j$ is a set of pairs $(x, x + z)$ of size γ, as in Algorithm 1. Then the estimate $y_{i,j}$ corresponding to a sample $\mathcal{I}_j$ is:

$$y_{i,j} = \frac{1}{|\mathcal{I}_j|} \sum_{(x, x+z) \in \mathcal{I}_j} \chi_a(x) f(z)(x+z).$$

We now bound the expected squared distance between $y_{i,j}$ and x_i by the inverse of the sample size.

Lemma 3.8 *Given a subspace $H \leq \mathbb{F}_2^n$, let $x_1 \geq x_2 \geq \ldots \geq x_{2^d}$ be the true energies in each of the buckets and $y_{i,j}$ be the estimate of x_i given sample $\mathcal{I}_j$ of size γ. Then using $\gamma = \mathcal{O}\left(\frac{s}{\epsilon^4}||f||_2^2\right)$ queries to f,*

$$\mathbb{E}\left[|y_{i,j} - x_i|^2\right] \leq \frac{\epsilon^4}{s}||f||_2^2,$$

where the expectation is taken over possible samples $\mathcal{I}_j$.

Proof : Given a subspace $H \leq \mathbb{F}_2^n$, let $x, y \in \mathbb{F}_2^n$ so that $|f(x)f(y)| \leq \frac{f^2(x)+f^2(y)}{2} \leq \frac{1}{2}||f||_2^2$. Thus, an empirical estimation of $\mathop{\mathbb{E}}_{x\in\mathbb{F}_2^n, z\in H^\perp}[\chi_a(z)f(x)f(x+z)]$, with $\mathcal{O}\left(\frac{1}{\epsilon^2}\log\frac{1}{\delta}\right)$ queries to f, is within an additive factor of $\epsilon||f||_2^2$ by standard Chernoff bounds.

Let C be a constant such that $\frac{C}{\epsilon^2}\log\frac{1}{\delta}$ samples suffice to estimate $y_{i,j} - x_i$ within an additive $\epsilon||f||_2^2$ with probability at least $1-\delta$. Equivalently for any $\theta > 0$, the probability that $|y_{i,j} - x_i| \geq \theta$ using γ samples is at most $e^{-\frac{\gamma\theta^2}{C||f||_2^4}}$. Then we have:

$$\begin{aligned}
\mathbb{E}\left[|y_{i,j} - x_i|^2\right] &= \int_0^\infty \mathbf{Pr}\left[|y_{i,j} - x_i|^2 \geq t\right]\, dt \\
&= \int_0^\infty \mathbf{Pr}\left[|y_{i,j} - x_i| \geq \sqrt{t}\right]\, dt \\
&\leq \int_0^\infty e^{-\frac{\gamma t}{C||f||_2^4}}\, dt = \frac{C||f||_2^4}{\gamma}.
\end{aligned}$$

Hence, for $\gamma = \mathcal{O}\left(\frac{s}{\epsilon^4}||f||_2^2\right)$, we have $\mathbb{E}\left[|y_{i,j} - x_i|^2\right] \leq \frac{\epsilon^4}{s}||f||_2^2$, as desired. □

Note that the estimate $y_{i,j}$ is exactly the estimate $y^{(j)}_{a+H}$ in Algorithm 1, where bucket $a+H$ is the i^{th} largest Fourier coefficient. We use two different notations to refer to the same quantity since it is more convenient to use the notation $y_{i,j}$ to index estimates by magnitude of Fourier coefficient, whereas the notation $y^{(j)}_{a+H}$ is more convenient to index by coset. Moreover, observe that we can obtain estimates $y_{i,j}$ of the energies x_i simultaneously, by Proposition 2.6.

As before, let y_i^* denote the contribution to the energy of the i^{th} bucket from the largest Fourier coefficient hashing into this bucket.

Lemma 3.9 *Let $\epsilon > 0$ and H be a random subspace of codimension $d = \log\frac{2s}{\epsilon^4}$ and let $x_1 \geq x_2 \geq \ldots \geq x_{2^d}$ be the true energies in each of the buckets. Let $\ell = \mathcal{O}\left(\log\frac{1}{\epsilon}\right)$ be the number of random samples. Then for any $\eta > 0$,*

$$\Pr[|y_i^* - x_i|^2 \geq \eta] \leq \left(\frac{2e\epsilon^2||f||_2^2}{s\eta}\right)^{\ell/2},$$

where the probability is taken over all samples of size ℓ.

Proof : By applying Markov's inequality to Lemma 3.8, it follows that for each pair of i and j,

$$\Pr\left[|y_{i,j} - x_i|^2 \geq \eta\right] \leq \frac{\epsilon^4||f||_2^2}{s\eta}.$$

Then the probability that at least half of the ℓ samples returns such estimates is

$$\Pr\left[|\{j : |y_{i,j} - x_i|^2 \geq \eta\}| > \frac{\ell}{2}\right] \leq \binom{\ell}{\ell/2}\left(\frac{\epsilon^4||f||_2^2}{s\eta}\right)^{\ell/2} \leq \left(\frac{2e\epsilon^4||f||_2^2}{s\eta}\right)^{\ell/2},$$

where the second inequality follows from the well-known bound on the binomial coefficient $\binom{n}{k} \leq \left(\frac{n\cdot e}{k}\right)^k$ for all $1 \leq k \leq n$. □

Lemma 3.10 *Let H be a random subspace of codimension $d = \log\frac{2s}{\epsilon^4}$. Then the expected value of the estimation error satisfies*

$$\mathop{\mathbb{E}}_{H}\left[\sum_{i=1}^{s}|y_i^* - x_i|^2\right] \leq \epsilon^2 \cdot ||f||_2^2.$$

Proof : Let $\beta = \frac{2e\epsilon^4||f||_2^2}{s\epsilon^{4/\ell}}$. Then:

$$\begin{aligned}
\mathop{\mathbb{E}}_{H}\left[\sum_{i=1}^{s}|y_i^* - x_i|^2\right] &= \mathop{\mathbb{E}}_{H}\left[\int_0^\infty \min(s, |\{a\colon |y_a^* - x_a|^2 \geq \eta|\})d\eta\right] \\
&\leq \int_0^\infty \min(s, \mathbb{E}\left[|\{i\colon |y_i^* - x_i|^2 \geq \eta|\}\right])d\eta \\
&\leq \int_0^\infty \min\left(s, 2^d\left(\frac{2e\epsilon^4||f||_2^2}{s\eta}\right)^{\ell/2}\right)d\eta \\
&\leq \int_0^\beta s\, d\eta + \int_\beta^\infty 2^d\left(\frac{2e\epsilon^4||f||_2^2}{s\eta}\right)^{\ell/2}d\eta,
\end{aligned}$$

where the second inequality follows from Lemma 3.9. Thus,

$$\begin{aligned}
\mathop{\mathbb{E}}_{H}\left[\sum_{i=1}^{s}|y_i^* - x_i|^2\right] &\leq \frac{2e\epsilon^4||f||_2^2}{\epsilon^{4/\ell}} + 2^d\left(\frac{2e\epsilon^4||f||_2^2}{s}\right)^{\ell/2}\frac{2}{\ell-2}\left(\frac{1}{\beta}\right)^{\ell/2-1} \\
&= \frac{2e\epsilon^4||f||_2^2}{\epsilon^{4/\ell}} + 2^d\left(\frac{2e\epsilon^4||f||_2^2}{s}\right)\frac{\epsilon^2}{\epsilon^{4/\ell}}\frac{2}{\ell-2}.
\end{aligned}$$

Hence for $\ell = \Theta\left(\log\frac{1}{\epsilon}\right)$, we have $\mathop{\mathbb{E}}_{H}\left[\sum_{i=1}^{s}|y_i^* - x_i|^2\right] \leq \epsilon^2 \cdot ||f||_2^2$. □

3.3 Proof of Theorem 1.1

Recall that our algorithm returns an estimate ξ of the sum of the s buckets with the largest energy. Since the estimation error is small by Lemma 3.10, ξ is a good estimate of the actual sum of the s buckets with the largest energy. Because the hashing error is small by Corollary 3.6, ξ is also a good approximation of the energy of the s Fourier coefficients $\beta_1, \ldots, \beta_s$ with the largest energy. Consider the function f^* whose values are the same as f at the Fourier coefficients $\{\beta_i\}$ but are zero elsewhere and note that by Parseval's identity, f^* is the s-sparse function closest to f. Hence, ξ is a good estimate of $||f^*||_2^2$.

For each random sample $\mathcal{I}_j$ of size $\gamma = \mathcal{O}\left(\frac{s\|f\|_2^2}{\epsilon^4}\right)$, let $y_{a+H}^{(i)}$ be the corresponding estimate of $(\hat{f}|_{a+H})^2$. Let $S^* = \text{argmax}_{|S|=s} \sum_{a \in S} \text{median}\{y_{a+H}^{(1)}, y_{a+H}^{(2)}, \ldots, y_{a+H}^{(\ell)}\}$, where $\ell = \mathcal{O}\left(\log \frac{1}{\epsilon}\right)$ is the number of repetitions. Let $\beta_{f|a+H}^* = \text{argmax}_{\alpha \in a+H} \hat{f}(\alpha)^2$ and define the function $h : \mathbb{F}_2^n \to \mathbb{R}$ by setting

$$\hat{h}(\beta_{f|a+H}^*) = \mathsf{sgn}(\hat{f}(\beta_{f|a+H}^*)) \cdot \text{median}\left\{\sqrt{y_{a+H}^{(i)}}\right\}$$

for each $a \in S^*$ to be the only non-zero Fourier coefficients of h. Let $\beta_1, \beta_2, \ldots, \beta_s$ be defined so that

$$\hat{f}(\beta_1), \hat{f}(\beta_2), \ldots, \hat{f}(\beta_s)$$

are the largest s Fourier coefficients of f. Define the function $f^* : \mathbb{F}_2^n \to \mathbb{R}$ by setting

$$\hat{f^*}(\beta_i) = \hat{f}(\beta_i)$$

for each $1 \leq i \leq s$ to be the only non-zero Fourier coefficients of f^*.

Lemma 3.11 *Let ξ be the output of Algorithm 1 and f^* and h be defined as above. Then*

$$\left|\xi - ||f^*||_2^2\right| \leq 2||f^* - h||_2||f||_2.$$

Proof : Observe that Algorithm 1 outputs

$$\xi = \sum_{a \in S^*} \text{median}\left\{y_{a+H}^{(1)}, y_{a+H}^{(2)}, \ldots, y_{a+H}^{(i)}\right\} = \sum_{a \in S^*} \hat{h}(\beta_{f|a+H}^*)^2 = ||h||_2^2.$$

Let $g : \mathbb{F}_2^n \to \mathbb{R}$ be the s-sparse function defined by setting

$$\hat{g}(\beta_{f|a+H}^*) = \hat{f}(\beta_{f|a+H}^*)$$

for each $a \in S^*$ to be the only non-zero Fourier coefficients of f^*. Therefore,

$$\left|\xi - ||f^*||_2^2\right| = \left|||h||_2^2 - ||f^*||_2^2\right| = \left(|||h||_2 - ||f^*||_2|\right)\left(|||h||_2 + ||f^*||_2|\right).$$

By triangle inequality, $|||h||_2 - ||f^*||_2| \leq ||f^* - h||_2$ and $|||h||_2 + ||f^*||_2| \leq ||h||_2 + ||f^*||_2$. Thus,

$$\left|\xi - ||f^*||_2^2\right| \leq ||f^* - h||_2(||h||_2 + ||f^*||_2).$$

Since $||h||_2 + ||f^*||_2 \leq 2||f||_2$, then it remains to bound $||f^* - h||_2$. □

Lemma 3.12 *Let ξ be the output of Algorithm 1 and f^* be defined as above. Then with probability at least $\frac{7}{8}$,*

$$\left|\xi - ||f^*||_2^2\right| \leq 14\epsilon||f||_2^2.$$

Proof : Let $g : \mathbb{F}_2^n \to \mathbb{R}$ be the s-sparse function defined by setting

$$\hat{g}(\beta_{f|a+H}^*) = \hat{f}(\beta_{f|a+H}^*)$$

for each $a \in S^*$ to be the only non-zero Fourier coefficients of f^*. Then by triangle inequality,

$$||f^* - h||_2 \leq ||f^* - g||_2 + ||g - h||_2.$$

Recall that $\mathcal{E}_1 \geq \cdots \geq \mathcal{E}_{2^n}$ are the true values of the energies of the 2^n Fourier coefficients corresponding to function $f : \mathbb{F}_2^n \to \mathbb{R}$ and y_i^* is the contribution to the energy of the i^{th} bucket from the largest Fourier coefficient hashing into this bucket. Let S be the set of indices corresponding to the buckets with nonzero energy in f^* and g and observe that $|S| \leq s$. Thus, $||f^* - g||_2^2$ is at most $\sum_{i \in S}(y_i - \mathcal{E}_i)$, where y_i is the total energy in the i^{th} bucket. By Corollary 3.6, $\sum_{i \in S}(y_i - \mathcal{E}_i) \leq 5\epsilon^2 ||f||_2^2$ with probability at least $\frac{15}{16}$.

On the other hand, $||g - h||_2^2 \leq 16\epsilon^2 ||f||_2^2$ with probability at least $\frac{15}{16}$ by Lemma 3.10 and Markov's inequality. Thus, $||f^* - h||_2 \leq \left(\sqrt{5} + 4\right)\epsilon ||f||_2 \leq 7\epsilon ||f||_2$ and by Lemma 3.11, $\left|\xi - ||f^*||_2^2\right| \leq 14\epsilon ||f||_2^2$ with probability at least $\frac{7}{8}$. □

By Lemma 3.8, it suffices to use $\mathcal{O}\left(\frac{s}{\epsilon^2}\|f\|_2^2 \log \frac{1}{\epsilon} \log \frac{1}{\delta}\right)$ queries to bound the sampling error. Hence, the query complexity follows as we assume $\|f\|_2^2 = 1$.

Algorithm 1 runs through $\ell = \mathcal{O}\left(\log \frac{1}{\epsilon}\right)$ iterations, each time sampling f at $\gamma = \mathcal{O}\left(\frac{s}{\epsilon^4}\right)$ pairs of points and updating each of the $2^d = \mathcal{O}\left(\frac{s}{\epsilon^4}\right)$ cosets. Hence, Algorithm 1 runs in $\mathcal{O}\left(\frac{s^2}{\epsilon^8} \log \frac{1}{\epsilon}\right)$ time.

We do not attempt to optimize runtime in Algorithm 1, as further optimizations can be made using standard sparse Hadamard transform techniques, e.g. page 163 in [Gol00] or in [Lev95, Pri] to update the empirical estimation of each coset, which improves the total running time to $\mathcal{O}\left(\frac{s}{\epsilon^4} \log \frac{s}{\epsilon^4} \log \frac{1}{\epsilon}\right)$.

4 Lower Bounds for ℓ_2^2-Testing of s-Sparsity

To the best of our knowledge the only lower bound known for the s-sparsity testing problem is due to [GOS+11]. Formally, they construct a hard distribution that is far from s-sparse in Hamming distance but since the support of the distribution is Boolean functions this also implies a lower bound under ℓ_2^2. Under ℓ_2^2-distance their Theorem 2 can be restated as follows:

Theorem 4.1 (Lower bound for ℓ_2^2 testing of Fourier sparsity [GOS+11]) *Fix any constant $\tau > 0$. Let $C(\tau) = \mathcal{O}(\log 1/\tau)$ and $s \leq 2^{n/C(\tau)}$. There exists a constant $c(\tau)$ so that any algorithm, which given non-adaptive query access to $f : \mathbb{F}_2^n \to \{-1, 1\}$, that distinguishes s-sparse functions from functions that are $c(\tau)$-far from s-sparse in ℓ_2^2 distance with probability at least $2/3$ requires $\Omega(\sqrt{s})$ queries.*

Below we extend this result to larger values of s for non-adaptive testers of real-valued functions.

4.1 $\Omega(\sqrt{s})$ Lower Bound for Non-adaptive Testers

We show a lower bound by designing two distributions $\mathcal{D}_{YES}$ and $\mathcal{D}_{NO}$, the former supported on the class of interest and the latter being far from it, such that the total variation distance between these distributions restricted to the query set is at most δ. This implies that the query set cannot distinguish the two distributions with probability greater than $\frac{1+\delta}{2}$.

Definition 4.2 (Total Variation Distance) *The total variation distance between two random variables P_1 and P_2 with corresponding probability density functions $p_1(x), p_2(x) \in \mathbb{R}^n$ is defined as $d_{TV}(P_1, P_2) = \frac{1}{2}\int_{\mathbb{R}^n} |p_1(x) - p_2(x)|\, dx$.*

Theorem 4.3 *For any $s \leq 2^{n-1}$, there exists a constant $c > 0$ such that any non-adaptive algorithm given query access to $f\colon \mathbb{F}_2^n \to \mathbb{R}$ such that $\|f\|_2^2 = 1 \pm \epsilon$ that distinguishes whether f is s-sparse or f is $\frac{1}{3}$-far from s-sparse in ℓ_2^2 with probability at least $2/3$ has to make at least $c\sqrt{s}$ queries to f.*

Proof : We define two distributions $\mathcal{D}_{YES}$ and $\mathcal{D}_{NO}$ where $\mathcal{D}_{YES}$ is supported on s-sparse functions only and $\mathcal{D}_{NO}$ is supported on functions that are far from s-sparse. Then by Yao's principle it suffices to show that if the size of the query set Q is at most $c\sqrt{s}$ then the total variation distance between the two distributions restricted on the query set $d_{TV}(\mathcal{D}_{YES}(Q), \mathcal{D}_{NO}(Q)) < 1/3$.

We now define the $\mathcal{D}_{YES}$ distribution. For each $z \in 2^{[n]}$ let $\mathbf{g}_z \sim N(0,1)$ be an independent zero mean and unit variance Gaussian random variable. Let $\mathbf{S} \subseteq 2^{[n]}$ be a random subset of fixed size s chosen uniformly at random from the collection of all subsets of size exactly s. Our distribution $\mathcal{D}_{YES}$ corresponds to a random family of functions $f_{\mathbf{S}}$ defined as follows:

$$f_{\mathbf{S}}(x) := \frac{1}{\sqrt{s}} \sum_{z \in \mathbf{S}} \mathbf{g}_z \chi_z(x).$$

The distribution $\mathcal{D}_{NO}$ is defined similarly, except that we fix $S = 2^{[n]}$, i.e. we set:

$$f(x) = \frac{1}{2^{n/2}} \sum_{z \in 2^{[n]}} \mathbf{g}'_z \chi_z(x),$$

where $\mathbf{g}'_z \sim N(0,1)$ are again independent and identically distributed standard normal variables.

Note that by standard Chernoff bounds with high probability functions sampled from both distributions satisfy $\|f\|_2^2 = 1 \pm \epsilon$. Furthermore by Chernoff bounds, with high probability functions in the support of $\mathcal{D}_{NO}$ are at least $\frac{1}{3}$-far in ℓ_2^2 from s-sparse for $s \leq 2^{n-1}$ (their expected distance is at least $1/2$). Consider any non-adaptive randomized algorithm that makes q queries. By Yao's principle we can fix the set of queries to form a set $Q \subseteq \mathbb{F}_2^n$ or size q. The values of $f_{\mathbf{S}}$ on Q form a vector with (possibly correlated) zero mean Gaussian entries.

Fix any S of size s. If $x = y$ then we have:

$$\mathbb{E}_{\mathbf{g}}[f_S(x) f_S(y)] = \mathbb{E}_{\mathbf{g}}[f_S(x)^2] = \frac{1}{s} \mathbb{E}_{\mathbf{g}} \left[\left(\sum_{z_1 \in S} \mathbf{g}_{z_1} \chi_{z_1}(x) \right)^2 \right] = \frac{1}{s} \left(\sum_{z_1 \in S} \mathbb{E}_{\mathbf{g}}[\mathbf{g}_{z_1}^2] \right) = 1.$$

Computing the values of the off-diagonal entries in the covariance matrix of f_S for $x \neq y$ we have:

$$\begin{aligned}
\mathbb{E}_{\mathbf{g}}[f_S(x) f_S(y)] &= \frac{1}{s} \mathbb{E}_{\mathbf{g}} \left[\sum_{z_1 \in S} \mathbf{g}_{z_1} \chi_{z_1}(x) \sum_{z_2 \in S} \mathbf{g}_{z_2} \chi_{z_2}(y) \right] \\
&= \frac{1}{s} \mathbb{E}_{\mathbf{g}} \left[\sum_{z \in S} \mathbf{g}_z^2 \chi_z(x) \chi_z(y) + \sum_{z_1 \neq z_2 \in S} \mathbf{g}_{z_1} \chi_{z_1}(x) \mathbf{g}_{z_2} \chi_{z_2}(y) \right] \\
&= \frac{1}{s} \left(\sum_{z \in S} \chi_z(x) \chi_z(y) \mathbb{E}_{\mathbf{g}} \left[\mathbf{g}_z^2 \right] + \sum_{z_1 \neq z_2 \in S} \chi_{z_1}(x) \chi_{z_2}(y) \mathbb{E}_{\mathbf{g}} \left[\mathbf{g}_{z_1} \mathbf{g}_{z_2} \right] \right) \\
&= \frac{1}{s} \left(\sum_{z \in S} \chi_z(x) \chi_z(y) + \sum_{z_1 \neq z_2 \in S} \chi_{z_1}(x) \chi_{z_2}(y) \mathbb{E}_{\mathbf{g}} \left[\mathbf{g}_{z_1} \right] \mathbb{E}_{\mathbf{g}} [\mathbf{g}_{z_2}] \right)
\end{aligned}$$

$$= \frac{1}{s} \sum_{z \in S} \chi_z(x)\chi_z(y)$$

Let $\xi_1, \ldots, \xi_q$ be the inputs in the query set Q. For any fixed $z \in 2^{[n]}$ define $a_z \in \{-1, 1\}^q$ to be a column vector with entries $a_{z,i} = \chi_z(\xi_i)$. Then the covariance matrix of $f_{\mathbf{S}}(\xi_1), \ldots f_{\mathbf{S}}(\xi_q)$ under the distribution $\mathcal{D}_{YES}$ is given by a random family of matrices $M_{\mathbf{S}} \in \mathbb{R}^{q \times q}$ defined as follows:

$$M_{\mathbf{S}} = \frac{1}{s} \sum_{z \in \mathbf{S}} a_z a_z^T .$$

Similarly for $\mathcal{D}_{NO}$ the covariance matrix of $f(\xi_1), \ldots, f(\xi_q)$ is $\frac{1}{2^n} \sum_{z \in 2^{[n]}} a_z a_z^T = I$.

The following standard fact allows to bound the total variation distance between two zero mean Gaussians with known covariance matrices.

Fact 4.4 *(See e.g. Corollary 2.14 in [DKK+16]) Let $\delta > 0$ be sufficiently small and let $\mathcal{N}(0, \Sigma_1)$ and $\mathcal{N}(0, \Sigma_2)$ be normal distributions with zero mean and covariance matrices Σ_1 and Σ_2 respectively. If $\|I - \Sigma_2^{-1/2} \Sigma_1 \Sigma_2^{-1/2}\|_F \leq \delta$ then:*

$$d_{TV}(\mathcal{N}(0, \Sigma_1), \mathcal{N}(0, \Sigma_2)) \leq \mathcal{O}(\delta) .$$

Using the lemma above and setting $\Sigma_1 = M_{\mathbf{S}}$ and $\Sigma_2 = I$ in order to show an upper bound on the total variation distance it suffices to bound the expected Frobenius norm of the difference $\mathbb{E}_{\mathbf{S}}[\|I - M_{\mathbf{S}}\|_F]$.

We have:

$$\begin{aligned}
\mathbb{E}_{\mathbf{S}}\left[\left\|I - \frac{1}{s}\sum_{z \in \mathbf{S}} a_z a_z^T\right\|_F\right] &= \mathbb{E}_{\mathbf{S}}\left[\sum_{1 \leq i,j \leq q}\left(\delta_{ij} - \frac{1}{s}\sum_{z \in \mathbf{S}} \chi_z(\xi_i)\chi_z(\xi_j)\right)^2\right] \\
&= \sum_{1 \leq i \leq q} \mathbb{E}_{\mathbf{S}}\left[\left(1 - \frac{1}{s}\sum_{z \in \mathbf{S}} \chi_z(\xi_i)^2\right)^2\right] + \sum_{1 \leq i \neq j \leq q} \mathbb{E}_{\mathbf{S}}\left[\left(\frac{1}{s}\sum_{z \in \mathbf{S}} \chi_z(\xi_i)\chi_z(\xi_j)\right)^2\right] \\
&= \frac{1}{s^2} \sum_{1 \leq i \neq j \leq q} \mathbb{E}_{\mathbf{S}}\left[\left(\sum_{z_1 \in \mathbf{S}} \chi_{z_1}(\xi_i)\chi_{z_1}(\xi_j)\right)\left(\sum_{z_2 \in \mathbf{S}} \chi_{z_2}(\xi_i)\chi_{z_2}(\xi_j)\right)\right] \\
&= \frac{1}{s^2} \sum_{1 \leq i \neq j \leq q} \mathbb{E}_{\mathbf{S}}\left[\sum_{z \in \mathbf{S}} \chi_z(\xi_i)^2 \chi_z(\xi_j)^2\right] + \mathbb{E}_{\mathbf{S}}\left[\sum_{z_1 \neq z_2 \in \mathbf{S}} \chi_{z_1}(\xi_i)\chi_{z_1}(\xi_j)\chi_{z_2}(\xi_i)\chi_{z_2}(\xi_j)\right] \\
&\leq \frac{q^2}{s}
\end{aligned}$$

Thus if $q < \sqrt{\delta s}$ we have $d_{TV}(\mathcal{D}_{YES}(Q), \mathcal{D}_{NO}(Q)) \leq \mathcal{O}(\delta)$. By picking δ to be a sufficiently small constant it follows that no algorithm that makes less than $c\sqrt{s}$ queries for some constant $c > 0$ can distinguish $\mathcal{D}_{YES}$ and $\mathcal{D}_{NO}$ with high probability. □

Acknowledgements

The authors would like to thank Piotr Indyk and Eric Price for multiple helpful discussions of this topic as well as Andrew Arnold, Arturs Backurs, Eric Blais, Michael Kapralov and Krzysztof Onak for their participation in earlier versions of this work.

References

[AGS03] Adi Akavia, Shafi Goldwasser, and Shmuel Safra. Proving hard-core predicates using list decoding. In *44th Symposium on Foundations of Computer Science (FOCS 2003), 11-14 October 2003, Cambridge, MA, USA, Proceedings*, pages 146–157, 2003. 1

[BBG18] Siddharth Barman, Arnab Bhattacharyya, and Suprovat Ghoshal. Testing sparsity over known and unknown bases. In *Proceedings of the 35th International Conference on Machine Learning, ICML*, pages 500–509, 2018. 1.1, 1

[BC99] Anna Bernasconi and Bruno Codenotti. Spectral analysis of boolean functions as a graph eigenvalue problem. *IEEE Trans. Computers*, 48(3):345–351, 1999. 1

[BRY14] Piotr Berman, Sofya Raskhodnikova, and Grigory Yaroslavtsev. L_p-testing. In *Symposium on Theory of Computing, STOC 2014, New York, NY, USA, May 31 - June 03, 2014*, pages 164–173, 2014. 1, 2.3

[DKK+16] Ilias Diakonikolas, Gautam Kamath, Daniel M. Kane, Jerry Li, Ankur Moitra, and Alistair Stewart. Robust estimators in high dimensions without the computational intractability. In *IEEE 57th Annual Symposium on Foundations of Computer Science, FOCS 2016, 9-11 October 2016, Hyatt Regency, New Brunswick, New Jersey, USA*, pages 655–664, 2016. 4.4

[GGR98] Oded Goldreich, Shafi Goldwasser, and Dana Ron. Property testing and its connection to learning and approximation. *J. ACM*, 45(4):653–750, 1998. 1.1, 2.2, 3

[GL89] Oded Goldreich and Leonid A. Levin. A hard-core predicate for all one-way functions. In *Proceedings of the 21st Annual ACM Symposium on Theory of Computing, May 14-17, 1989, Seattle, Washigton, USA*, pages 25–32, 1989. 1, 1.1

[Gol00] Oded Goldreich. Modern cryptography, probabilistic proofs and pseudorandomness, 2000. 3.3

[Gol01] Oded Goldreich. *The Foundations of Cryptography - Volume 1, Basic Techniques*. Cambridge University Press, 2001. 1.1

[GOS+11] Parikshit Gopalan, Ryan O'Donnell, Rocco A. Servedio, Amir Shpilka, and Karl Wimmer. Testing fourier dimensionality and sparsity. *SIAM J. Comput.*, 40(4):1075–1100, 2011. 1, 1.1, 1.2, 2.3, 3.7, 3.2, 4, 4.1

[HIK+13] Haitham Hassanieh, Piotr Indyk, Michael Kapralov, Dina Katabi, Eric Price, and Lixin Shi. Sfft: Sparse fast fourier transform. http://groups.csail.mit.edu/netmit/sFFT/index.html, 2013. [Online; accessed 07-July-2015]. 1

[HIKP12a] Haitham Hassanieh, Piotr Indyk, Dina Katabi, and Eric Price. Nearly optimal sparse fourier transform. In *Proceedings of the 44th Symposium on Theory of Computing Conference, STOC 2012, New York, NY, USA, May 19 - 22, 2012*, pages 563–578, 2012. 1

[HIKP12b] Haitham Hassanieh, Piotr Indyk, Dina Katabi, and Eric Price. Simple and practical algorithm for sparse fourier transform. In *Proceedings of the Twenty-Third Annual ACM-SIAM Symposium on Discrete Algorithms, SODA 2012, Kyoto, Japan, January 17-19, 2012*, pages 1183–1194, 2012. 1

[HL13] Hamed Hatami and Shachar Lovett. Estimating the distance from testable affine-invariant properties. In *54th Annual IEEE Symposium on Foundations of Computer Science, FOCS 2013, 26-29 October, 2013, Berkeley, CA, USA*, pages 237–242, 2013. 1.1

[KM93] Eyal Kushilevitz and Yishay Mansour. Learning decision trees using the fourier spectrum. *SIAM J. Comput.*, 22(6):1331–1348, 1993. 1, 1.1

[Lev95] Leonid A Levin. Randomness and nondeterminism. In *Proceedings of the International Congress of Mathematicians*, pages 1418–1419. Springer, 1995. 3.3

[LMN93] Nathan Linial, Yishay Mansour, and Noam Nisan. Constant depth circuits, fourier transform, and learnability. *J. ACM*, 40(3):607–620, 1993. 1

[MO09] Ashley Montanaro and Tobias Osborne. On the communication complexity of XOR functions. *CoRR*, abs/0909.3392, 2009. 1

[O'D14] Ryan O'Donnell. *Analysis of Boolean Functions*. Cambridge University Press, 2014. 1.1, 2.3

[Pri] Eric Price. Private communication. 3.3

[PRR06] Michal Parnas, Dana Ron, and Ronitt Rubinfeld. Tolerant property testing and distance approximation. *J. Comput. Syst. Sci.*, 72(6):1012–1042, 2006. 1

[Ron08] Dana Ron. Property testing: A learning theory perspective. *Foundations and Trends in Machine Learning*, 1(3):307–402, 2008. 1

[STlV14] Amir Shpilka, Avishay Tal, and Ben lee Volk. On the structure of boolean functions with small spectral norm. In *Innovations in Theoretical Computer Science, ITCS'14, Princeton, NJ, USA, January 12-14, 2014*, pages 37–48, 2014. 1

[TWXZ13] Hing Yin Tsang, Chung Hoi Wong, Ning Xie, and Shengyu Zhang. Fourier sparsity, spectral norm, and the log-rank conjecture. In *54th Annual IEEE Symposium on Foundations of Computer Science, FOCS 2013, 26-29 October, 2013, Berkeley, CA, USA*, pages 658–667, 2013. 1

[WY13] Karl Wimmer and Yuichi Yoshida. Testing linear-invariant function isomorphism. In *Proceedings of the 40th International Conference on Automata, Languages, and Programming - Volume Part I*, ICALP'13, pages 840–850, 2013. 1, 1.1

[ZS09] Zhiqiang Zhang and Yaoyun Shi. Communication complexities of symmetric XOR functions. *Quantum Information & Computation*, 9(3):255–263, 2009. 1

[ZS10] Zhiqiang Zhang and Yaoyun Shi. On the parity complexity measures of boolean functions. *Theor. Comput. Sci.*, 411(26-28):2612–2618, 2010. 1

A Appendix

A.1 Basic Facts

Fact A.1 (Reduction of Property Testing to Energy Estimation of Top s Fourier Coefficients) *Suppose we are given query access to some function $f : \mathbb{F}_2^n \to \mathbb{R}$ with $||f||_2^2 = 1$. Given an energy estimator of the top s Fourier coefficients that uses $q_s(\epsilon)$ queries, there exists a property tester for s-sparsity with parameter ϵ that uses $q_s\left(\frac{\epsilon}{2}\right)$ queries, where $q_s(\cdot)$ is some function that depends on ϵ.*

Proof : Let $\mathcal{F}_s$ be the class of s-sparse functions mapping from $\mathbb{F}_2^n$ to $\mathbb{R}$. Trivially if $f \in \mathcal{F}_s$, then the sum of the top s Fourier coefficients is $||f||_2^2$ and so an $\frac{\epsilon}{2}$-energy estimator of the top s Fourier coefficients outputs a value ξ with $\xi \geq ||f||_2^2 - \frac{\epsilon}{2}||f||_2^2$.

On the other hand, if for any s-sparse function g, it holds that $||f-g||_2^2 \geq \epsilon||f||_2^2$, then the energy of the top s Fourier coefficients of f is at most $(1-\epsilon)||f||_2^2$. Then an $\frac{\epsilon}{2}$-energy estimator of the top s Fourier coefficients outputs a value ξ with

$$\left|\xi - \max_{|S|=s} \sum_{\alpha \in S} \hat{f}(\alpha)^2\right| \leq \frac{\epsilon}{2}||f||_2^2,$$

so the energy estimator outputs a value ξ with $\xi \leq ||f||_2^2 - \frac{\epsilon}{2}||f||_2^2$.

Thus, the energy estimator can differentiate whether $f \in \mathcal{F}_s$ or f is ϵ-far from s-sparsity, using $q_s\left(\frac{\epsilon}{2}\right)$ queries. □

A.2 Poisson Summation Formula

Recall the proof of the Poisson summation formula:

Proof of Proposition 2.7: For any $z \in \mathbb{F}_2^n$, we have that

$$\begin{aligned} f|_{a+H}(z) &= \mathop{\mathbb{E}}_{x \in H^\perp}\Big[\sum_{\beta \in \mathbb{F}_2^n} \hat{f}(\beta)\chi_\beta(x+z)\cdot\chi_a(x)\Big] \\ &= \sum_{\beta \in \mathbb{F}_2^n} \hat{f}(\beta)\chi_\beta(z)\cdot \mathop{\mathbb{E}}_{x \in H^\perp}\left[\chi_{\beta+a}(x)\right]. \end{aligned}$$

Since $\mathop{\mathbb{E}}_{x \in H^\perp}\Big[\chi_{\beta+a}(x)\Big]$ equals 1 when $\beta + a \in H$ and 0 otherwise, we obtain

$$f|_{a+H}(z) = \sum_{\beta \in a+H} \hat{f}(\beta)\chi_\beta(z)$$

and hence:

$$\begin{aligned} \widehat{f|_{a+H}}(\alpha) &= \mathop{\mathbb{E}}_{x \in \mathbb{F}_2^n}\left[f|_{a+H}(x)\chi_\alpha(x)\right] = \mathop{\mathbb{E}}_{x \in \mathbb{F}_2^n}\left[\sum_{\beta \in a+H} \hat{f}(\beta)\chi_\beta(x)\chi_\alpha(x)\right] \\ &= \sum_{\beta \in a+H}\left(\hat{f}(\beta)\mathop{\mathbb{E}}_{x \in \mathbb{F}_2^n}\left[\chi_\beta(x)\chi_\alpha(x)\right]\right) = \hat{f}(\alpha) \end{aligned}$$

□