An Adaptive Sublinear-Time Block Sparse Fourier Transform

Volkan Cevher
EPFL

Michael Kapralov
EPFL

Jonathan Scarlett
EPFL

Amir Zandieh
EPFL

February 4, 2017

Abstract

The problem of approximately computing the k dominant Fourier coefficients of a vector X quickly, and using few samples in time domain, is known as the Sparse Fourier Transform (sparse FFT) problem. A long line of work on the sparse FFT has resulted in algorithms with $O(k \log n \log(n/k))$ runtime [Hassanieh *et al.*, STOC'12] and $O(k \log n)$ sample complexity [Indyk *et al.*, FOCS'14]. These results are proved using non-adaptive algorithms, and the latter $O(k \log n)$ sample complexity result is essentially the best possible under the sparsity assumption alone: It is known that even adaptive algorithms must use $\Omega((k \log(n/k))/ \log \log n)$ samples [Hassanieh *et al.*, STOC'12]. By *adaptive*, we mean being able to exploit previous samples in guiding the selection of further samples.

This paper revisits the sparse FFT problem with the added twist that the sparse coefficients approximately obey a (k_0, k_1)-block sparse model. In this model, signal frequencies are clustered in k_0 intervals with width k_1 in Fourier space, where $k = k_0 k_1$ is the total sparsity. Signals arising in applications are often well approximated by this model with $k_0 \ll k$.

Our main result is the first sparse FFT algorithm for (k_0, k_1)-block sparse signals with the sample complexity of $O^*(k_0 k_1 + k_0 \log(1+k_0) \log n)$ at constant signal-to-noise ratios, and sublinear runtime. A similar sample complexity was previously achieved in the works on *model-based compressive sensing* using random Gaussian measurements, but used $\Omega(n)$ runtime. To the best of our knowledge, our result is the first sublinear-time algorithm for model based compressed sensing, and the first sparse FFT result that goes below the $O(k \log n)$ sample complexity bound.

Interestingly, the aforementioned model-based compressive sensing result that relies on Gaussian measurements is non-adaptive, whereas our algorithm crucially uses *adaptivity* to achieve the improved sample complexity bound. We prove that adaptivity is in fact necessary in the Fourier setting: Any *non-adaptive* algorithm must use $\Omega(k_0 k_1 \log \frac{n}{k_0 k_1})$ samples for the (k_0, k_1)-block sparse model, ruling out improvements over the vanilla sparsity assumption. Our main technical innovation for adaptivity is a new randomized energy-based importance sampling technique that may be of independent interest.

Contents

1 Introduction

The discrete Fourier transform (DFT) is one of the most important tools in modern signal processing, finding applications in audio and video compression, radar, geophysics, medical imaging, communications, and many more. The best known algorithm for computing the DFT of a general signal of length n is the Fast Fourier Transform (FFT), taking $O(n \log n)$ time, which matches the trivial $\Omega(n)$ lower bound up to a logarithmic factor.

In recent years, significant attention has been paid to exploiting *sparsity* in the signal's Fourier spectrum, which is naturally the case for numerous of the above applications. By sparse, we mean that the signal can be well-approximated by a small number of Fourier coefficients. Given this assumption, the computational lower bound of $\Omega(n)$ no longer applies. Indeed, the DFT can be computed in *sublinear time*, while using a sublinear number of samples in the time domain [1, 2].

The problem of computing the DFT of signals that are approximately sparse in Fourier domain has received significant attention in several communities. The seminal work of [3, 4] in *compressive sensing* first showed that only $k \log^{O(1)} n$ samples in time domain suffice to recover a length n signal with at most k nonzero Fourier coefficients. A different line of research on the *Sparse Fourier Transform* (sparse FFT), with origins in computational complexity and learning theory, has resulted in algorithms that use $k \log^{O(1)} n$ samples and $k \log^{O(1)} n$ runtime (i.e., the runtime is *sublinear* in the length of the input signal). Many such algorithms have been proposed in the literature [5–23]; we refer the reader to the recent surveys [1, 2] for a more complete overview.

The best known runtime for computing the k-sparse FFT is due to Hassanieh *et al.* [24], and is given by $O(k \log n \log(n/k))$, asymptotically improving upon the FFT for all $k = o(n)$. The recent works of [22, 25] also show how to achieve the optimal sample complexity bound of $O(k \log n)$ in linear time, or in time $k \log^{O(1)} n$ at the expense of $\mathrm{poly}(\log \log n)$ factors. Intriguingly, the aforementioned algorithms are all *non-adaptive.* That is, these algorithms do not exploit existing samples in guiding the selection of the new samples to improve approximation quality. In the same setting, it is also known that adaptivity cannot improve the sample complexity by more than an $O(\log \log n)$ factor [24].

Despite the significant gains permitted by sparsity, designing an algorithm for handling *arbitrary* sparsity patterns may be overly generic; in practice, signals often exhibit more specific sparsity structures. A common example is *block sparsity*, where significant coefficients tend to cluster on known partitions, as opposed to being unrestricted in the signal spectrum. Other common examples include tree-based sparsity, group sparsity, and dispersive sparsity [26–29].

Such structured sparsity models can be captured via the *model-based* framework [26], where the number of sparsity patterns may be far lower than $\binom{n}{k}$. For the compressive sensing problem, this restriction has been shown to translate into a reduction in the sample complexity, even with non-adaptive algorithms. Specifically, one can achieve a sample complexity of $O(k + \log |\mathcal{M}|)$ with dense measurement matrices based on the Gaussian distribution, where $\mathcal{M}$ is the set of permitted sparsity patterns. Reductions in the sample complexity with other types of measurement matrices, e.g., sparse measurement matrices based on expanders, are typically less [30, 31]. Other benefits of exploiting model-based sparsity include faster recovery and improved noise robustness [26, 30].

Surprisingly, in stark contrast to the extensive work on exploiting model-based sparsity with general linear measurements, there are no existing sparse FFT algorithms exploiting such structure. This paper presents the first such algorithm, focusing on the special case of block sparsity. Even for this relatively simple sparsity model, achieving the desiderata turns out to be quite challenging, needing a whole host of new techniques, and intriguingly, requiring *adaptivity* in the sampling.

To clarify our contributions, we describe our model and the problem statement in more detail.

Model and Basic Definitions The Fourier transform of a signal $X \in \mathbb{C}^n$ is denoted by $\widehat{X}$, and defined as

$$\widehat{X}_f = \frac{1}{n} \sum_{i \in [n]} X_i \omega_n^{-ft}, \quad f \in [n],$$

where ω_n is the n-th root of unity. With this definition, Parseval's theorem takes the form $\|X\|^2 = n\|\widehat{X}\|_2^2$.

We are interested in computing the Fourier transform of signals that, in frequency domain, are well-approximated by a *block sparse* signal with k_0 blocks of width k_1, formalized as follows.

Definition 1.1 (Block sparsity)**.** Given a sequence $X \in \mathbb{C}^n$ and an even block width k_1, the *j-th interval* is defined as $I_j = \big((j-1/2)k_1, (j+1/2)k_1\big] \cap \mathbb{Z}$ for $j \in \big[\frac{n}{k_1}\big]$, and we refer to $\widehat{X}_{I_j}$ as the *j-th block*. We say that a signal is (k_0, k_1)*-block sparse* if it is non-zero within at most k_0 of these intervals.

Block sparsity is of direct interest in several applications [26, 32]; we highlight two examples here: (i) In spectrum sensing, cognitive radios seek to improve the utilization efficiency in a sparsely used wideband spectrum. In this setting, the frequency bands being detected are non-overlapping and predefined. (ii) In multi-sensor processing, a frequency-sparse signal is recorded by an array of sensors. Each sensor has the same dominant frequencies, but with different delays and amplitudes; hence, the frequency-domain signals of the sensors can be rearranged to produce a block-sparse signal. While such rearranging does not directly fall into the framework of the present paper, our techniques can be applied, and in fact simplified, for this latter setting.

Our goal is to output a list of frequencies and values estimating $\widehat{X}$, yielding an ℓ_2-distance to $\widehat{X}$ not much larger than that of the best (k_0, k_1)-block sparse approximation. Formally, we say that an output signal $\widehat{X}'$ satisfies the ℓ_2/ℓ_2 *block-sparse recovery guarantee* if

$$\|\widehat{X} - \widehat{X}'\|_2 \leq (1+\epsilon) \min_{\widehat{Y} \text{ is } (k_0,k_1)\text{-block sparse}} \|\widehat{X} - \widehat{Y}\|_2$$

for an input parameter $\epsilon > 0$.

The sample complexity and runtime of our algorithm are parameterized by the *signal-to-noise ratio* (SNR) of the input signal, defined as follows.

Definition 1.2 (Tail noise and signal-to-noise ratio (SNR))**.** We define the *tail noise level* as

$$\mathrm{Err}(\widehat{X}, k_0, k_1) := \min_{\substack{S \subset [\frac{n}{k_1}] \\ |S| = k_0}} \sum_{j \in [\frac{n}{k_1}] \setminus S} \|\widehat{X}_{I_j}\|_2^2, \tag{1}$$

and its normalized version as $\mu^2 := \frac{1}{k_0}\mathrm{Err}^2(\widehat{X}, k_0, k_1)$, representing the average noise level per block. The *signal-to-noise ratio* is defined as $\mathrm{SNR} := \frac{\|\widehat{X}\|^2}{\mathrm{Err}^2(\widehat{X}, k_0, k_1)}$.

Throughout the paper, we assume that both n and k_1 are powers of two. For n, this is a standard assumption in the sparse FFT literature. As for k_1, the assumption comes without too much loss of generality, since one can always round the block size up to the nearest power of two and then cover the original k_0 blocks with at most $2k_0$ larger blocks, thus yielding a near-identical recovery problem other than a possible increase in the SNR. We also assume that $\frac{n}{k_1}$ exceeds a large absolute constant; if this fails, our stated scaling laws can be obtained using the standard FFT.

We use $O^*(\cdot)$ notation to hide $\log\log \mathrm{SNR}$, $\log\log n$, and $\log\frac{1}{\epsilon}$ factors. Moreover, to simplify the notation in certain lemmas having free parameters that will be set in terms of ϵ, we assume throughout the paper that $\epsilon = \Omega\big(\frac{1}{\mathrm{poly}\log n}\big)$, and hence $\log\frac{1}{\epsilon} = O(\log\log n)$. This is done purely for convenience, and since the dependence on ϵ is not our main focus; the precise expressions with $\log\frac{1}{\epsilon}$ factors are easily inferred from the proofs. Similarly, since the low-SNR regime is not our key focus, we assume that $\mathrm{SNR} \geq 2$, and thus $\log \mathrm{SNR}$ is positive.

Contributions. We proceed by informally stating our main result; a formal statement is given in Section 5.2.

Theorem 1.1. (Upper bound – informal version) *There exists an adaptive algorithm for approximating the Fourier transform with (k_0, k_1)-block sparsity that achieves the ℓ_2/ℓ_2 guarantee for any constant $\epsilon > 0$, with a sample complexity of $O^*\big((k_0k_1 + k_0\log(1+k_0)\log n)\log \mathrm{SNR}\big)$, and a runtime of $O^*\big((k_0k_1\log^3 n + k_0\log(1+k_0)\log^2 n)\log \mathrm{SNR}\big)$.*

The sample complexity of our algorithm *strictly improves* upon the sample complexity of $O(k_0k_1\log n)$ (essentially optimal under the standard sparsity assumption) when $\log(1 + k_0)\log \mathrm{SNR} \ll k_1$ and $\log \mathrm{SNR} \ll \log n$ (e.g., $\mathrm{SNR} = O(1)$).

Our algorithm that achieves the above upper bound crucially uses adaptivity. This is in stark contrast with the standard sparse FFT, where we know how to achieve the optimal $O(k\log n)$ bound using non-adaptive sampling [25]. While relying on adaptivity can be viewed as a weakness, we provide a lower bound revealing that *adaptivity is essential* for obtaining the above sample complexity gains. We again state an informal version, which is formalized in Section 6.

Theorem 1.2. (Lower bound – informal version) *Any non-adaptive sparse FFT algorithm that achieves the ℓ_2/ℓ_2 sparse recovery guarantee with (k_0, k_1)-block sparsity must use $\Omega\big(k_0k_1\log\frac{n}{k_0k_1}\big)$ samples.*

To the best of our knowledge, these two theorems provide the first results along several important directions, giving **(a)** the first sublinear-time algorithm for model-based compressed sensing; **(b)** the first model-based result with provable sample complexity guarantees in the Fourier setting; **(c)** the first proven gap between the power of adaptive and non-adaptive sparse FFT algorithms; and **(d)** the first proven gap between the power of structured (Fourier basis) and unstructured (random Gaussian entries) matrices for model-based compressed sensing.

To see that **(d)** is true, note that the sample complexity $O(k_0 \log n + k_0 k_1)$ for block-sparse recovery can be achieved *non-adaptively* using Gaussian measurements [26], but we show that adaptivity is required in the Fourier setting.

Dependence of our results on SNR. The sample complexity and runtime of our upper bound depend logarithmically on the SNR of the input signal. This dependence is common for sparse FFT algorithms, and even for the case of standard sparsity, algorithms avoiding this dependence in the runtime typically achieve a suboptimal sample complexity [24, 33]. Moreover, all existing sparse FFT lower bounds consider the constant SNR regime (e.g., [24, 34, 35]).

We also note that our main result, as stated above, assumes that upper bounds on the SNR and the tail noise are known that are tight to within a constant factor (in fact, such tightness is not required, but the resulting bound replaces the true values by the assumed values). These assumptions can be avoided at the expense of a somewhat worse dependence on $\log \mathrm{SNR}$, but we prefer to present the algorithm in the above form for clarity. The theoretical guarantees for noise-robust compressive sensing algorithms often require similar assumptions [36].

Our techniques: At a high level, our techniques can be summarized as follows:

Upper bound. The high-level idea of our algorithm is to *reduce* the (k_0, k_1)-block sparse signal of length n to a number of downsampled $O(k_0)$-sparse signals of length $\frac{n}{k_1}$, and use standard sparse FFT techniques to locate their dominant values, thereby identifying the dominant blocks of the original signal. Once the blocks are located, their values can be estimated using hashing techniques. Despite the high-level simplicity, this is a difficult task requiring a variety of novel techniques, the most notable of which is an adaptive *importance sampling* scheme for allocating sparsity budgets to the downsampled signals. Further details are given in Section 2.

Lower bound. Our lower bound for non-adaptive algorithms follows the information-theoretic framework of [37], but uses a significantly different ensemble of *structured* approximately block-sparse signals occupying only a fraction $O\big(\frac{1}{k_0 k_1}\big)$ of the time domain. Hence, whereas the analysis of [37] is based on the difficulty of identifying one of (roughly) $\binom{n}{k}$ sparsity patterns, the difficulty in our setting is in *non-adaptively* finding where the signal is non-zero – one must take enough samples to cover the various possible time domain locations. The details are given in Section 6.

Interestingly, our upper bound uses adaptivity to circumvent the difficulty exploited in this lower bounding technique, by *first* determining where the energy lies, and *then* concentrating the rest of its samples on the "right" parts of the signal.

Notation: For an even number n, we define $[n] := \big(-\frac{n}{2}, \frac{n}{2}\big] \cap \mathbb{Z}$, where $\mathbb{Z}$ denotes the integers. When we index signals having a given length m, all arithmetic should be interpreted as returning values in $[m]$ according to modulo-m arithmetic. For $x, y \in \mathbb{C}$ and $\Delta \in \mathbb{R}$, we write $y = x \pm \Delta$ to denote $|y - x| \leq \Delta$. The support of a vector X is denoted by $\mathrm{supp}(X)$. For a number $a \in \mathbb{R}$, we write $|a|_+ := \max\{0, a\}$ to denote the positive part of a.

Organization: The paper is organized as follows. In Section 2, we provide an outline of our algorithm and the main challenges involved. We formalize our energy-based importance sampling scheme in Section 3, and provide the corresponding techniques for energy estimation in Section 4. The block-sparse FFT algorithm and its theoretical guarantees are given in Section 5, and the lower bound is presented and proved in Section 6. Several technical proofs are relegated to the appendices.

2 Overview of the Algorithm

One of our key technical contributions consists of a reduction from the (k_0, k_1)-block sparse recovery problem for signals of length n to $O(k_0)$-sparse recovery on a set of carefully-defined signals of *reduced length* n/k_1, in sublinear time. We outline this reduction below.

A basic candidate reduction to $O(k_0)$-sparse recovery consists of first convolving $\widehat{X}$ with a filter $\widehat{G}$ whose support approximates the indicator function of the interval $[-k_1/2, k_1/2]$, and then considering a new signal whose Fourier transform consists of samples of $\widehat{X} \star \widehat{G}$ at multiples of k_1. The resulting signal $\widehat{Z}$ of length n/k_1 **(a)** naturally represents $\widehat{X}$, as every frequency of this sequence is a (weighted) sum of the frequencies in the corresponding block, and **(b)** can be accessed in time domain using a small number of accesses to X (if G is compactly supported; see below).

This is a natural approach, but its vanilla version does not work: Some blocks in $\widehat{X}$ may entirely cancel out, not contributing to $\widehat{Z}$ at all, and other blocks may add up constructively and contribute an overly large amount of energy to $\widehat{Z}$. To overcome this challenge, we consider not one, but rather $2k_1$ reductions: For each $r \in [2k_1]$, we apply the above reduction to the *shift of X by $r \cdot \frac{n}{2k_1}$ in time domain*, and call the corresponding vector Z^r. We show that all shifts cumulatively capture the energy of X well, and the major contribution of the paper is an algorithm for locating the dominant blocks in $\widehat{X}$ from a small number of accesses to the Z^r's (via an importance sampling scheme).

Formal definitions: We formalize the above discussion in the following, starting with the notion of a *flat filter* that approximates a rectangle.

Definition 2.1 (Flat filter)**.** A sequence $G \in \mathbb{R}^n$ with Fourier transform $\widehat{G} \in \mathbb{R}^n$ symmetric about zero is called an *(n, B, F)-flat filter* if (i) $\widehat{G}_f \in [0, 1]$ for all $f \in [n]$; (ii) $\widehat{G}_f \geq 1 - \big(\frac{1}{4}\big)^{F-1}$ for all $f \in [n]$ such that $|f| \leq \frac{n}{2B}$; and (iii) $\widehat{G}_f \leq \big(\frac{1}{4}\big)^{F-1}\big(\frac{n}{B|f|}\big)^{F-1}$ for all $f \in [n]$ such that $|f| \geq \frac{n}{B}$.

The following lemma, proved in Appendix A.1, shows that it is possible to construct such a filter having $O(FB)$ support in time domain.

Lemma 2.1. (Compactly supported flat filter) *Fix the integers (n, B, F) with n a power of two, $B < n$, and $F \geq 2$ an even number. There exists an (n, B, F)-flat filter $\widehat{G} \in \mathbb{R}^n$, which (i) is* *supported on a length-$O(FB)$ window centered at zero in time domain, and (ii) has a total energy satisfying $\sum_{f\in[n]} |\widehat{G}_f|^2 \le \frac{3n}{B}$.*

Throughout the paper, we make use of the filter construction from Lemma 2.1, except where stated otherwise. To ease the analysis, we assume that G and $\widehat{G}$ are pre-computed and can be accessed in $O(1)$ time. Without this pre-computation, evaluating $\widehat{G}$ is non-trivial, but possible using semi-equispaced Fourier transform techniques (*cf.*, Section 4.2).

With the preceding definition, the set of $2k_1$ downsampled signals is given as follows.

Definition 2.2 (Downsampling)**.** Given integers (n, k_1), a parameter $\delta \in \big(0, \frac{1}{20}\big)$, and a signal $X \in \mathbb{C}^n$, we say that the set of signals $\{Z^r\}_{r\in[2k_1]}$ with $Z^r \in \mathbb{C}^{\frac{n}{k_1}}$ is a (k_1, δ)-*downsampling* of X if

$$Z^r_j = \frac{1}{k_1} \sum_{i\in[k_1]} (G \cdot X^r)_{j+\frac{n}{k_1}\cdot i},\ j \in \left[\frac{n}{k_1}\right]$$

for an $\big(n, \frac{n}{k_1}, F\big)$-flat filter with $F = 10 \log \frac{1}{\delta}$ and support $O\big(F\frac{n}{k_1}\big)$, where we define $X^r_i = X_{i+a_r}$ with $a_r = \frac{nr}{2k_1}$. Equivalently, in frequency domain, this can be written as

$$\widehat{Z}^r_j = (\widehat{X}^r \star \widehat{G})_{jk_1} = \sum_{f\in[n]} \widehat{G}_{f-k_1\cdot j} \widehat{X}_f \omega_n^{a_r\cdot f}, j \in \left[\frac{n}{k_1}\right] \tag{2}$$

by the convolution theorem and the duality of subsampling and aliasing (e.g., see Appendix C.1).

By the assumption of the bounded support of G, along with the choice of F, we immediately obtain the following lemma, showing that we do not significantly increase the sample complexity by working with $\{Z^r\}_{r\in[2k_1]}$ as opposed to X itself.

Lemma 2.2. (Sampling the downsampling signals) *Let $\{Z^r\}_{r\in[2k_1]}$ be a (k_1, δ)-downsampling of $X \in \mathbb{C}^n$ for some (n, k_1, δ). Then any single entry Z^r_i can be computed in $O\big(\log \frac{1}{\delta}\big)$ time using $O\big(\log \frac{1}{\delta}\big)$ samples of X.*

This idea of using $2k_1$ reductions fixes the above-mentioned problem of constructive and destructive cancellations: The $2k_1$ reduced signals Z^r ($r \in [2k_1]$) cumulatively capture all the energy of X well. That is, while the energy $|\widehat{Z}^r_j|^2_2$ can vary significantly as a function of r, we can tightly control the behavior of the sum $\sum_{r\in[2k_1]} |\widehat{Z}^r_j|^2_2$. This is formalized in the following.

Lemma 2.3. (Downsampling properties) *Fix (n, k_1), a parameter $\delta \in \big(0, \frac{1}{20}\big)$, a signal $X \in \mathbb{C}^n$, and a (k_1, δ)-downsampling $\{Z^r\}_{r\in[2k_1]}$ of X. The following conditions hold:*

1. *For all $j \in [\frac{n}{k_1}]$,*

$$\frac{\sum_{r\in[2k_1]} |\widehat{Z}^r_j|^2}{2k_1} \ge (1-\delta)\|\widehat{X}_{I_j}\|^2_2 - 3\delta \cdot \left(\|\widehat{X}_{I_j \cup I_{j-1} \cup I_{j+1}}\|^2_2 + \delta \sum_{j' \in [\frac{n}{k_1}]\setminus\{j\}} \frac{\|\widehat{X}_{I_{j'}}\|^2_2}{|j'-j|^{F-1}} \right).$$

2. *The total energy satisfies $(1-12\delta)\|\widehat{X}\|^2_2 \le \frac{\sum_{r\in[2k_1]} \|\widehat{Z}^r\|^2_2}{2k_1} \le 6\|\widehat{X}\|^2_2$.*

The proof is given in Appendix A.2.

Location via sparse FFT: We expect each Z^r to be approximately $O(k_0)$-sparse, as every block contributes *primarily* to one downsampled coefficient. At this point, a natural step is to run $O(k_0)$-sparse recovery on the signals Z^r to recover the dominant blocks. However, there are too many signals Z^r to consider! Indeed, if we were to run $O(k_0)$-sparse recovery on every Z^r, we would recover the locations of the blocks, but at the cost of $O(k_0 k_1 \log n)$ samples. This precludes any improvement on the vanilla sparse FFT.

It turns out, however, that it is possible to avoid running a k_0-sparse FFT on all $2k_1$ reduced signals, and to instead *allocate budgets* to them, some of which are far smaller than k_0, and some of which may be zero. This will be key in reducing the sample complexity.

Before formally defining budget allocation, we present the following definition and lemma, showing that we can use less samples to identify less of the dominant coefficients of a signal, or more samples to identify more dominant coefficients.

Definition 2.3. *(Covered frequency)* Given an integer m, a frequency component j of a signal $\widehat{Z} \in \mathbb{C}^m$ is called *covered* by budget s in the signal $\widehat{Z}$ if $|\widehat{Z}_j|^2 \geq \frac{\|\widehat{Z}\|_2^2}{s}$.

Lemma 2.4. (LocateReducedSignal guarantees – informal version) *There exists an algorithm such that if a signal $X \in \mathbb{C}^n$, a set of budgets $\{s^r\}_{r \in [2k_1]}$, and a confidence parameter p are given to it as input, then it outputs a list that, with probability at least $1-p$, contains any given j that is covered by s^r in $\widehat{Z}^r$ for some $r \in [2k_1]$, where $\{\widehat{Z}^r\}_{r \in [2k_1]}$ denotes the (k_1, δ)-downsampling of X. Moreover, the list size is $O\big(\sum_{r \in [2k_1]} s^r\big)$, the number of samples that the algorithm takes is $O\big(\sum_{r \in [2k_1]} s^r \log n\big)$, and the runtime is $O\big(\sum_{r \in [2k_1]} s^r \log^2 n\big)$.*[1]

The formal statement and proof are given in Appendix F, and reveal that s^r essentially dictates how many buckets we hash $\widehat{Z}^r$ into in order to locate the dominant frequencies (e.g., see [24, 25]).

Hence, the goal of budget allocation is to approximately solve the following covering problem:

$$\text{Minimize}_{\{s^r\}} \sum_{r \in [2k_1]} s^r \quad \text{subject to} \quad \sum_{\substack{j \text{ is covered by } s_r \\ \text{in } Z^r \text{ for some } r \in [2k_1]}} \|\widehat{X}_{I_j}\|_2^2 \geq (1-\alpha) \cdot \|\widehat{X}^*\|_2^2, \tag{3}$$

for a suitable constant $\alpha \in (0,1)$, where s^r is the budget allocated to $\widehat{Z}^r$, and $\widehat{X}^*$ is the best $(k_0.k_1)$-block sparse approximation of $\widehat{X}$. That is, we want to minimize the total budget while accounting for a constant proportion of the signal energy.

Challenges in budget allocation: Allocating the budgets is a challenging task, as each block in the spectrum of the signal may have very different energy concentration properties in time domain, or equivalently, different variations in $|\widehat{Z}_j^r|^2$ as a function of r. To see this more concretely, in Figure 1, we show three *hypothetical* examples of such variations, in the case that $k_0 = 2k_1 = 6$ and all of the blocks have equal energy, leading to equal column sums in the matrices.

[1] As stated in the formal version, additional terms in the runtime are needed when it comes to subtracting off a current estimate to form a residual signal.

$$2k_1 \Bigg\updownarrow \overset{\xleftrightarrow{\qquad k_0 \qquad}}{\begin{bmatrix} \mathbf{6} & 0 & 0 & 0 & 0 & 0 \\ 0 & \mathbf{6} & 0 & 0 & 0 & 0 \\ 0 & 0 & \mathbf{6} & 0 & 0 & 0 \\ 0 & 0 & 0 & \mathbf{6} & 0 & 0 \\ 0 & 0 & 0 & 0 & \mathbf{6} & 0 \\ 0 & 0 & 0 & 0 & 0 & \mathbf{6} \end{bmatrix}}$$

(a) Spiky energies

$$\begin{bmatrix} \mathbf{1} & \mathbf{1} & \mathbf{1} & \mathbf{1} & \mathbf{1} & \mathbf{1} \\ \mathbf{1} & \mathbf{1} & \mathbf{1} & \mathbf{1} & \mathbf{1} & \mathbf{1} \\ \mathbf{1} & \mathbf{1} & \mathbf{1} & \mathbf{1} & \mathbf{1} & \mathbf{1} \\ \mathbf{1} & \mathbf{1} & \mathbf{1} & \mathbf{1} & \mathbf{1} & \mathbf{1} \\ \mathbf{1} & \mathbf{1} & \mathbf{1} & \mathbf{1} & \mathbf{1} & \mathbf{1} \\ \mathbf{1} & \mathbf{1} & \mathbf{1} & \mathbf{1} & \mathbf{1} & \mathbf{1} \end{bmatrix}$$

(b) Flat energies

$$\begin{bmatrix} \mathbf{3} & \mathbf{5} & \mathbf{1} & 0 & 0 & 0 \\ \mathbf{1} & 0 & \mathbf{1} & 0 & 0 & \mathbf{3} \\ \mathbf{1} & 0 & \mathbf{1} & 0 & \mathbf{2} & 0 \\ 0 & \mathbf{1} & \mathbf{1} & \mathbf{6} & 0 & \mathbf{1} \\ 0 & 0 & \mathbf{1} & 0 & \mathbf{2} & \mathbf{1} \\ \mathbf{1} & 0 & \mathbf{1} & 0 & \mathbf{2} & \mathbf{1} \end{bmatrix}$$

(c) Mixed energies

Figure 1: Three hypothetical examples of matrices with (r, j)-th entry given by $|\widehat{Z}_j^r|^2$, i.e., each row corresponds to a single sequence Z^r, but only at the entries corresponding to the k_0 blocks in X.

In the first example, each block contributes to a different Z^r, and thus the blocks could be located by running 1-sparse recovery separately on the $2k_1$ signals. In stark contrast, in the second example, each block contributes equally to each Z^r, so we would be much better off running k_0-sparse recovery on a single (arbitrary) Z^r. Finally, in the third example, the best budget allocation scheme is completely unclear by inspection alone! We need to design an allocation scheme to handle all of these cases, and to do so *without even knowing the structure of the matrix.*

While the examples in Figure 1 may seem artificial, and are not necessarily feasible with the *exact* values given, we argue in Appendix E that situations exhibiting the same general behavior are entirely feasible.

Importance sampling: Our solution is to *sample r values with probability proportional to an estimate of $\|\widehat{Z}^r\|_2^2$*, and sample sparsity budgets from a carefully defined distribution (see Section 3, Algorithm 1). We show that sufficiently accurate estimates of $\|\widehat{Z}^r\|_2^2$ for all $r \in [2k_1]$ can be obtained using $O(k_0 k_1)$ samples of X via hashing techniques (*cf.*, Section 4); hence, what we are essentially doing is using these samples to determine where most of the energy of the signal is located, and then favoring the parts of the signal that appear to have more energy. This is exactly the step that makes our algorithm adaptive, and we prove that it produces a total budget in (3) of the form $O(k_0 \log(1 + k_0))$, on average.

Ideally, one would hope to solve (3) using a total budget of $O(k_0)$, since there are only k_0 blocks. However, the $\log(1 + k_0)$ factor is not an artifact of our analysis: We argue in Appendix E that very different sampling techniques would be needed to remove it in general. Specifically, we design a signal X for which the *optimal* solution to (3) indeed satisfies $\sum_{r\in[2k_1]} s^r = \Omega(k_0 \log(1 + k_0))$.

Iterative procedure and updating the residual: The techniques described above allow us to recover a list of blocks that contribute a constant fraction (e.g., 0.9) of the signal energy. We use $O(\log \mathrm{SNR})$ iterations of our main procedure to reduce the SNR to a constant, and then achieve $(1 + \epsilon)$-recovery with an extra "clean-up" step. Most of the techniques involved in this part are more standard, with a notable exception: Running a standard sparse FFT with budgets s^r on the reduced space (i.e., on the vectors Z^r) is not easy to implement in $k_0 k_1 \mathrm{poly}(\log n)$ time when Z^r are the *residual signals.* The natural approach is to subtract the current estimate $\widehat{\chi}$ of $\widehat{X}$ from our samples and essentially run on the residual, but subtraction in $k_0 k_1 \mathrm{poly}(\log n)$ time is not easy to achieve. Our solution crucially relies on a novel block semi-equispaced FFT (see Section 4.2), and the idea of letting the location primitives in the reduced space operate using common randomness (see Appendix F).

3 Location via Importance Sampling

As outlined above, our approach locates blocks by applying standard sparse FFT techniques to the downsampled signals arising from Definition 2.2. In this section, we present the techniques for assigning the corresponding sparsity budgets (*cf.*, (3)).

We use a novel procedure called *energy-based importance sampling*, which approximately samples r values with probability proportional to $\|\widehat{Z}^r\|^2$. Since these energies are not known exactly, we instead sample proportional to a general vector $\gamma = (\gamma^1, \ldots, \gamma^{2k_1})$, where we think of γ^r as approximating $\|\widehat{Z}^r\|^2$. The techniques for obtaining these estimates are deferred to Section 4.

The details are shown in Algorithm 1, where we repeatedly sample from the distribution w_q^r, corresponding to independently sampling r proportional to γ^r, and q from a truncated geometric distribution. The resulting sparsity level to apply to Z^r is selected to be $s^r = 10 \cdot 2^q$.

According to Definition 2.3, $s^r = 10 \cdot 2^q$ covers any given frequency j for which $|\widehat{Z}_j^r|^2 \geq \frac{\|\widehat{Z}^r\|_2^2}{10 \cdot 2^q}$. The intuition behind sampling q proportional to 2^{-q} is that this gives a high probability of producing small q values to cover the heaviest signal components, while having a small probability of producing large q values to cover the smaller signal components. We only want to do the latter rarely, since it costs significantly more samples.

Algorithm 1 Procedure for allocating sparsity budgets to the downsampled signals

```
procedure BudgetAllocation($\gamma, k_0, k_1, \delta, p$)
    $S \leftarrow \emptyset$
    for $i \in \{1, \ldots, \frac{10}{\delta} k_0 \cdot \log \frac{1}{p}\}$ do
        Sample $(r_i, q_i) \in [2k_1] \times \{1, \ldots, \log_2 \frac{10k_0}{\delta}\}$ with probability $w_q^r = \frac{2^{-q}}{1-\delta/(10k_0)} \frac{\gamma^r}{\|\gamma\|_1}$
        $S \leftarrow S \cup \{(r_i, q_i)\}$
    for $r \in [2k_1]$ do
        $q^* \leftarrow \max_{(r,q') \in S} \{q'\}$                ▷ By convention, $\max \emptyset = -\infty$
        $s^r \leftarrow 10 \cdot 2^{q^*}$
    return $s = [s^r]_{r \in [2k_1]}$
```

We first bound the expected total sum of budgets returned by BudgetAllocation.

Lemma 3.1. (BudgetAllocation budget guarantees) *For any integers k_0 and k_1, any positive vector $\gamma \in \mathbb{R}^{2k_1}$, and any parameters $p \in \left(0, \frac{1}{2}\right)$ and $\delta \in (0,1)$, if the procedure* BudgetAllocation *in Algorithm 1 is run with inputs $(\gamma, k_0, k_1, \delta, p)$, then the expected value of the total sum of budgets returned, $\{s^r\}_{r \in [2k_1]}$, satisfies $\mathbb{E}\big[\sum_{r \in [2k_1]} s^r\big] \leq 200 \frac{k_0}{\delta} \log \frac{k_0}{\delta} \log \frac{1}{p}$. The runtime of the procedure is $O\big(\frac{k_0}{\delta} \log \frac{1}{p} + k_1\big)$.*

Proof. Each time a new (r, q) pair is sampled, the sum of the s^r values increases by at most $10 \cdot 2^q$, and hence the overall expected sum is upper bounded by the number of trials $10k_0 \log \frac{1}{p}$ times the expected value of $10 \cdot 2^q$ for a single trial:

$$\begin{aligned}
\mathbb{E}\Big[\sum_{r \in [2k_1]} s^r \Big] &\le \frac{10}{\delta} k_0 \cdot \log \frac{1}{p} \sum_{r \in [2k_1]} \sum_{q=1}^{\log_2 \frac{10k_0}{\delta}} w_q^r \cdot 10 \cdot 2^q \\
&= \frac{100 k_0 \log \frac{1}{p}}{\delta} \cdot \frac{1}{1 - \delta/(10k_0)} \sum_{q=1}^{\log_2 \frac{10k_0}{\delta}} \sum_{r \in [2k_1]} \frac{\gamma^r}{\|\gamma\|_1} \\
&\le 200 \frac{k_0}{\delta} \log \frac{k_0}{\delta} \log \frac{1}{p},
\end{aligned}$$

where the second line follows from the definition of w_q^r, and the third line follows from $\frac{\delta}{10k_0} \le \frac{1}{2}$ (since $\delta \le 1$) and $\sum_{r \in [2k_1]} \frac{\gamma^r}{\|\gamma\|_1} = 1$.

Runtime: Note that sampling from w_q^r amounts to sampling q and r values independently, and the corresponding alphabet sizes are $O\big(\log \frac{k_0}{\delta}\big)$ and $O(k_1)$ respectively. The stated runtime follows since we take $O\big(\frac{k_0}{\delta} \log \frac{1}{p}\big)$ samples, and sampling from discrete distributions can be done in time linear in the alphabet size and number of samples [38]. The second loop in Algorithm 1 need not be done explicitly, since the maximum q value can be updated after taking each sample. □

As we discussed in Section 2, the $\log k_0$ term in the number of samples would ideally be avoided; however, in Appendix E.2, we argue that even the optimal solution to (3) can contain such a factor.

We now turn to formalizing the fact that the budgets returned by BUDGETALLOCATION are such that most of the dominant blocks are found. To do this, we introduce the following notion.

Definition 3.1 (Active frequencies)**.** Given (n, k_0, k_1), a signal $X \in \mathbb{C}^n$, a parameter $\delta \in (0, 1)$, and a (k_1, δ)-downsampling $\{Z^r\}_{r \in [2k_1]}$ of X, the set of *active frequencies* $\tilde{S}$ is defined as

$$\tilde{S} = \Big\{ j \in \Big[\frac{n}{k_1}\Big] \; : \; \sum_{r \in [2k_1]} \Big(|\widehat{Z}_j^r|^2 \cdot \frac{\gamma^r}{\|\widehat{Z}^r\|_2^2} \Big) \ge \delta \cdot \frac{\sum_{r \in [2k_1]} \|\widehat{Z}^r\|_2^2}{k_0} \Big\}. \tag{4}$$

Observe that if $\gamma^r = \|\widehat{Z}^r\|_2^2$, this reduces to $\sum_{r \in [2k_1]} |\widehat{Z}_j^r|^2 \ge \delta \cdot \frac{\sum_{r \in [2k_1]} \|\widehat{Z}^r\|_2^2}{k_0}$, thus essentially stating that the sum of the energies over $r \in [2k_1]$ for the given block index j is an $\Omega\big(\frac{\delta}{k_0}\big)$ fraction of the total energy. Combined with Lemma 2.3, this roughly amounts to $\|\widehat{X}_{I_j}\|_2^2$ exceeding an $\Omega\big(\frac{\delta}{k_0}\big)$ fraction of $\|\widehat{X}\|_2^2$.

To formalize and generalize this intuition, the following lemma states that the frequencies within $\tilde{S}$ account for most of the energy in X, as long as each γ^r approximates $\|\widehat{Z}^r\|_2^2$ sufficiently well.

Lemma 3.2. (Properties of active frequencies) *Fix (n, k_0, k_1), a parameter $\delta \in \big(0, \frac{1}{20}\big)$, a signal $X \in \mathbb{C}^n$, and a (k_1, δ)-downsampling $\{Z^r\}_{r \in [2k_1]}$ of X. Moreover, fix an arbitrary set $S^* \subseteq \big[\frac{n}{k_1}\big]$ of cardinality at most $10k_0$, and a vector $\gamma \in \mathbb{R}^{2k_1}$ satisfying*

$$\sum_{r \in [2k_1]} \Big| \|\widehat{Z}_{S^*}^r\|_2^2 - \gamma^r \Big|_+ \le 40\delta \sum_{r \in [2k_1]} \|\widehat{Z}^r\|_2^2. \tag{*}$$

Fix the set of active frequencies $\tilde{S}$ according to Definition 3.1, and define the signal $\widehat{X}_{\tilde{S}}$ to equal $\widehat{X}$ on all intervals $\{I_j\,; j \in \tilde{S}\}$ (see Definition 1.1), and zero elsewhere. Then $\|\widehat{X}_{S^* \setminus \tilde{S}}\|_2^2 \le 100\sqrt{\delta}\|\widehat{X}\|_2^2$.

The proof of Lemma 3.2 is given in Appendix B.1.

What remains is to show that if j is active, then j is covered by some s^r in $\widehat{Z}^r$ with high constant probability upon running Algorithm 1. This is formulated in the following.

Lemma 3.3. (BudgetAllocation covering guarantees) *Fix (n, k_0, k_1), the parameters $\delta \in (0,1)$ and $p \in \left(0, \frac{1}{2}\right)$, a signal $X \in \mathbb{C}^n$, and a (k_1, δ)-downsampling $\{Z^r\}_{r \in [2k_1]}$ of X. Moreover, fix a vector $\gamma \in \mathbb{R}^{2k_1}$ satisfying*

$$\|\gamma\|_1 \le 10 \sum_{r \in [2k_1]} \|\widehat{Z}^r\|_2^2. \tag{5}$$

Suppose that BudgetAllocation *in Algorithm 1 is run with inputs $(\gamma, k_0, k_1, \delta, p)$, and outputs the budgets $\{s^r\}_{r \in [2k_1]}$. Then for any active j (i.e., $j \in \tilde{S}$ as per Definition 3.1), the probability that there exists some $r \in [2k_1]$ such that j is covered by s^r in $\widehat{Z}^r$ is at least $1 - p$.*

Proof. Recall from Definition 2.3 that if a pair (r, q) is sampled in the first loop of BudgetAllocation, then j is covered provided that $|\widehat{Z}^r_j|^2 \ge \frac{\|\widehat{Z}^r\|_2^2}{10 \cdot 2^q}$. We therefore define

$$q_j^r = \min\left\{ q \in \mathbb{Z}_+ \,:\, |\widehat{Z}^r_j|^2 \ge \frac{\|\widehat{Z}^r\|_2^2}{10 \cdot 2^q} \right\}, \tag{6}$$

and note that the event described in the lemma statement is equivalent to some pair (r, q) being sampled with $q_j^r \le q$. Note that due to the range of q from which we sample (*cf.*, Algorithm 1), this can only occur if $q_j^r \le \log_2 \frac{10k_0}{\delta}$.

Taking a single sample: We first compute the probability of being covered for a *single* random sample of (q, r), denoting the corresponding probability by $\mathbb{P}_1[\cdot]$. Recalling from line 4 of Algorithm 1 that we sample each (q, r) with probability $w_q^r = \frac{2^{-q}}{1 - \delta/(10k_0)} \frac{\gamma^r}{\|\gamma\|_1}$, we obtain

$$\begin{aligned}
\mathbb{P}_1[j \text{ covered}] &= \sum_{r \in [2k_1]} \sum_{q_j^r \le q \le \log_2 \frac{10k_0}{\delta}} w_q^r \\
&= \frac{1}{1 - \delta/(10k_0)} \sum_{r \in [2k_1]} \sum_{q_j^r \le q \le \log_2 \frac{10k_0}{\delta}} 2^{-q} \frac{\gamma^r}{\|\gamma\|_1} \\
&\ge \frac{1}{2} \sum_{r \in [2k_1] \,:\, q_j^r \le \log_2 \frac{10k_0}{\delta}} 2^{-q_j^r} \frac{\gamma^r}{\|\gamma\|_1} \\
&= \frac{1}{2} \sum_{r \in [2k_1]} 2^{-q_j^r} \frac{\gamma^r}{\|\gamma\|_1} - \frac{1}{2} \sum_{r \in [2k_1] \,:\, q_j^r > \log_2 \frac{10k_0}{\delta}} 2^{-q_j^r} \frac{\gamma^r}{\|\gamma\|_1},
\end{aligned} \tag{7}$$

where the third line follows since $\delta/(10k_0) \le \frac{1}{2}$ due to the assumption that $\delta \le 1$.

Bounding the first term in (7): Observe from (6) that $2^{-q_j^r} \ge \frac{1}{2} \frac{10|\widehat{Z}^r_j|^2}{\|\widehat{Z}^r\|_2^2}$, and recall the definition of being active in (4). Combining these, we obtain the following when j is active:

$$\frac{1}{2}\sum_{r\in[2k_1]} 2^{-q_j^r}\frac{\gamma^r}{\|\gamma\|_1} \geq \frac{10}{4\|\gamma\|_1}\sum_{r\in[2k_1]}\frac{|\widehat{Z}_j^r|^2\gamma^r}{\|\widehat{Z}^r\|_2^2}$$
$$\geq \frac{10\delta}{4\|\gamma\|_1}\cdot\frac{\sum_{r\in[2k_1]}\|\widehat{Z}^r\|_2^2}{k_0} \geq \frac{\delta}{4k_0},$$

where the last inequality follows from the assumption on $\|\gamma\|_1$ in the lemma statement.

Bounding the second term in (7)**:** We have

$$\frac{1}{2}\sum_{r\in[2k_1]\,:\,q_j^r>\log_2\frac{10k_0}{\delta}} 2^{-q_j^r}\frac{\gamma^r}{\|\gamma\|_1} \leq \frac{1}{2}\sum_{r\in[2k_1]}\frac{\delta}{10k_0}\frac{\gamma^r}{\|\gamma\|_1} = \frac{\delta}{20k_0}.$$

Hence, we deduce from (7) that $\mathbb{P}_1[j \text{ covered}] \geq \frac{\delta}{5k_0}$.

Taking multiple independent samples: Since the sampling is done $\frac{10}{\delta}k_0\cdot\log\frac{1}{p}$ times independently, the overall probability of an active block j being covered satisfies

$$\mathbb{P}[j \text{ covered}] \geq 1-\left(1-\frac{\delta}{5k_0}\right)^{\frac{10}{\delta}k_0\cdot\log\frac{1}{p}} \geq 1-\exp\left(-\frac{10}{5}\cdot\log\frac{1}{p}\right) \geq 1-p,$$

where we have applied the inequality $1-\zeta \leq e^{-\zeta}$ for $\zeta \geq 0$. □

3.1 The Complete Location Algorithm

In Algorithm 2, we give the details of MultiBlockLocate, which performs the above-described energy-based importance sampling procedure, runs the sparse FFT location algorithm (see Appendix F) with the resulting budgets, and returns a list L containing the block indices that were identified.

MultiBlockLocate calls two primitives that are defined later in the paper, but their precise details are not needed in order to understand the location step:

- EstimateEnergies (see Section 4.4) provides us with a vector γ providing a good approximation of each $\|\widehat{Z}^r\|_2^2$, in the sense of satisfying the preconditions of Lemmas 3.2 and 3.3;
- LocateReducedSignals (see Appendix F) accepts the sparsity budgets $\{s^r\}$ and runs a standard s^r-sparse fast FFT algorithm on each downsampled signal Z^r in order to locate the dominant frequencies.

Note that in addition to X, these procedures accept a second signal $\widehat{\chi}$; this becomes relevant when we iteratively run the block sparse FFT (*cf.*, Section 5), representing previously-estimated components that are subtracted off to produce a residual.

The required guarantees on LocateReducedSignals are given in Lemma 2.4 (and more formally in Appendix E.2), and in order to prove our main result on MultiBlockLocate, we also need the following lemma ensuring that we can compute energy estimates satisfying the preconditions of Lemmas 3.2 and 3.3; the procedure and proof are presented in Section 4.4.

Lemma 3.4. (EstimateEnergies guarantees) *Given (n, k_0, k_1), the signals $X \in \mathbb{C}^n$ and $\widehat{\chi} \in \mathbb{C}^n$ with $\|\widehat{X}-\widehat{\chi}\|_2^2 \geq \frac{1}{\mathrm{poly}(n)}\|\widehat{\chi}\|_2$, and the parameter $\delta \in \left(\frac{1}{n},\frac{1}{20}\right)$, the procedure*

EstimateEnergies$(X, \widehat{\chi}, n, k_0, k_1, \delta)$ returns a vector $\gamma \in \mathbb{R}^{2k_1}$ such that, for any given set S^* of cardinality at most $10k_0$, we have the following with probability at least $\frac{1}{2}$:

1. $\sum_{r\in[2k_1]} \Big| \|\widehat{Z}^r_{S^*}\|_2^2 - \gamma^r \Big|_+ \le 40\delta \sum_{r\in[2k_1]} \|\widehat{Z}^r\|_2^2$;
2. $\|\gamma\|_1 \le 10 \sum_{r\in[2k_1]} \|\widehat{Z}^r\|_2^2$;

where $\{Z^r\}_{r\in[2k_1]}$ is the (k_1, δ)-downsampling of $X - \chi$ (see Definition 2.2).

Moreover, if $\widehat{\chi}$ is $(O(k_0), k_1)$-block sparse, then the sample complexity is $O(\frac{k_0 k_1}{\delta^2} \log \frac{1}{\delta} \log \frac{1}{\delta p})$, and the runtime is $O(\frac{k_0 k_1}{\delta^2} \log^2 \frac{1}{\delta} \log^2 n)$.

Remark 3.1. The preceding lemma ensures that the γ^r provide good approximations of $\|\widehat{Z}^r\|_2^2$ in a "restricted" and "one-sided" sense, while not over-estimating the total energy by more than a constant factor. Specifically, the first part concerns the energy of $\widehat{Z}^r$ restricted to a fixed set of size $O(k_0)$, and characterizes the extend to which the energies are under-estimated. It appears to be infeasible to characterize over-estimation in the same way (e.g., replacing $|\cdot|_+$ by $|\cdot|$), since several of the samples could be overly large due to spiky noise.

Remark 3.2. Here and subsequently, the poly(n) lower bounds regarding $(\widehat{X}, \widehat{\chi})$ are purely technical, resulting from extremely small errors when subtracting off $\widehat{\chi}$. See Section 4.2 for further details.

Algorithm 2 Multi-block sparse location

```
procedure MultiBlockLocate($X, \widehat{\chi}, n, k_0, k_1, \delta, p$)
    $a_r \leftarrow \frac{nr}{2k_1}$ for each $r \in [2k_1]$
    for $t \in \{1, \dots, 10 \log \frac{1}{p}\}$ do
        $\gamma \leftarrow$ EstimateEnergies($X, \widehat{\chi}, n, k_0, k_1, \delta$)        ▷ See Section 4.4
        $\mathbf{s}^{(t)} \leftarrow$ BudgetAllocation($\gamma, k_0, k_1, \delta, \frac{1}{2}\delta p$)        ▷ $\gamma = (\gamma^1, \dots, \gamma^{2k_1})$
    $\mathbf{s} \leftarrow \max_t \mathbf{s}^{(t)}$ (element-wise with respect to $r \in [2k_1]$)
    $L \leftarrow$ LocateReducedSignals($X, \widehat{\chi}, n, k_0, k_1, \mathbf{s}, \delta, \frac{1}{2}\delta p$)        ▷ See Appendix F
    return $L$
```

We are now in a position to provide our guarantees on MultiBlockLocate, namely, on the behavior of the list size, and on the energy that the components in the list capture.

Lemma 3.5. (MultiBlockLocate guarantees) *Given (n, k_0, k_1), the parameters $\delta \in \big(\frac{1}{n}, \frac{1}{20}\big)$ and $p \in \big(\frac{1}{n}, \frac{1}{2}\big)$, and the signals $X \in \mathbb{C}^n$ and $\chi \in \mathbb{C}^n$ with $\widehat{\chi}_0$ uniformly distributed over an arbitrarily length-$\frac{\|\widehat{\chi}\|^2}{\mathrm{poly}(n)}$ interval, the output L of the function MultiBlockLocate$(X, \widehat{X}, k_1, k_0, n, \delta, p)$ has the following properties for any set S^* of cardinality at most $10k_0$:*

1. $\mathbb{E}\big[|L|\big] = O\big(\frac{k_0}{\delta} \log \frac{k_0}{\delta} \log \frac{1}{p} \log^2 \frac{1}{\delta p}\big)$;
2. $\sum_{j\in S^*\setminus L} \|(\widehat{X} - \widehat{\chi})_{I_j}\|_2^2 \le 200\sqrt{\delta} \|\widehat{X} - \widehat{\chi}\|_2^2$ *with probability at least* $1 - p$.

Moreover, if $\widehat{\chi}$ is $(O(k_0), k_1)$-block sparse, and we have $\delta = \Omega\big(\frac{1}{\mathrm{poly}(\log n)}\big)$ and $p = \Omega\big(\frac{1}{\mathrm{poly}(\log n)}\big)$, then (i) the expected sample complexity is $O^*\big(\frac{k_0}{\delta} \log(1 + k_0) \log n + \frac{k_0 k_1}{\delta^2}\big)$, and the expected runtime *is* $O^*\big(\frac{k_0}{\delta}\log(1+k_0)\log^2 n + \frac{k_0 k_1}{\delta^2}\log^2 n + \frac{k_0 k_1}{\delta}\log^3 n\big)$*; (ii) if the procedure returns* L*, then we are guaranteed that the algorithm used* $O^*\big(|L|\cdot\log n + \frac{k_0 k_1}{\delta^2}\big)$ *samples and* $O^*\big(|L|\cdot\log^2 n + \frac{k_0 k_1}{\delta^2}\log^2 n + \frac{k_0 k_1}{\delta}\log^3 n\big)$ *runtime.*

Remark 3.1. The procedure MultiBlockLocate is oblivious to the choice of S^* in this theorem statement.

Proof. **First claim:** Note that in each iteration of the outer loop when we run BudgetAllocation$(\gamma, k_0, k_1, \delta, \frac{1}{2}\delta p)$, Lemma 3.1 implies that for any t, the following holds true:

$$\mathbb{E}\Big[\sum_{r\in[2k_1]} \mathbf{s}_r^{(t)}\Big] = O\Big(\frac{k_0}{\delta}\log\frac{k_0}{\delta}\log\frac{1}{\delta p}\Big),$$

where $\mathbf{s}_r^{(t)}$ is the r-th entry of the budget allocation vector $\mathbf{s}^t$ at iteration t. Therefore,

$$\mathbb{E}\Big[\sum_{r\in[2k_1]} \mathbf{s}_r\Big] = \mathbb{E}\Big[\sum_{r\in[2k_1]} \max_{t=1,\dots,10\log\frac{1}{p}} \mathbf{s}_r^t\Big] \le \sum_{t=1}^{10\log\frac{1}{p}} \mathbb{E}\Big[\sum_{r\in[2k_1]} \mathbf{s}_r^{(t)}\Big] = O\Big(k_0\log\frac{k_0}{\delta}\log\frac{1}{p}\log\frac{1}{\delta p}\Big). \tag{8}$$

We now apply Lemma 2.4, which is formalized in Appendix F; the assumption $\max_{r\in[2k_1]}\mathbf{s}_r = O\big(\frac{k_0}{\delta}\big)$ therein is satisfied due to the range of q from which we sample in BudgetAllocation. We set the target success probability to $1-\frac{1}{2}\delta p$, which guarantees that the size of the list returned by the function LocateReducedSignals is $O\big(\sum_{r\in[2k_1]}\mathbf{s}_r\log\frac{1}{\delta p}\big)$. Therefore, by (8), we have

$$\mathbb{E}[|L|] = O\Big(\frac{k_0}{\delta}\log\frac{k_0}{\delta}\log\frac{1}{p}\log^2\frac{1}{\delta p}\Big),$$

yielding the first statement of the lemma.

Second claim: Let $X' = X - \chi$, and consider the set S^* given in the theorem statement, and an arbitrary iteration t. By Lemma 3.4 in Section 4.4 (also stated above), the approximate energy vector γ in any given iteration of the outer loop satisfies

$$\sum_{r\in[2k_1]} \Big|\|\widehat{Z}^r_{S^*}\|_2^2 - \gamma^r\Big|_+ \le 40\delta \sum_{r\in[2k_1]} \|\widehat{Z}^r\|_2^2$$
$$\|\gamma\|_1 \le 10\sum_{r\in[2k_1]}\|\widehat{Z}^r\|_2^2 \tag{9}$$

with probability at least $\frac{1}{2}$. When this is the case, the vector γ meets the requirements of Lemmas 3.2 and 3.3. That means that the probability of having an energy estimate γ that meets these requirements in at least one iteration is lower bounded by $1-(\frac{1}{2})^{10\log\frac{1}{p}} \ge 1-p$.

We now consider an arbitrary iteration in which the above conditions on γ are satisfied. We write

$$\sum_{j\in S^*\setminus L} \|\widehat{X}'_{I_j}\|_2^2 = \sum_{j\in(S^*\cap\tilde{S})\setminus L} \|\widehat{X}'_{I_j}\|_2^2 + \sum_{j\in S^*\setminus(\tilde{S}\cup L)} \|\widehat{X}'_{I_j}\|_2^2. \tag{10}$$

The second term is bounded by

$$\sum_{j\in S^*\setminus(\tilde{S}\cup L)} \|\widehat{X}'_{I_j}\|_2^2 \le \sum_{j\in S^*\setminus\tilde{S}} \|\widehat{X}'_{I_j}\|_2^2 \le 100\sqrt{\delta}\|\widehat{X}'\|_2^2 \tag{11}$$

by Lemma 3.2, which uses the first condition on γ in (9).

We continue by calculating the expected value of the first term in (10) with respect to the randomness of BudgetAllocation and LocateReducedSignals:

$$\mathbb{E}\Big[\sum_{j\in(S^*\cap\tilde{S})\setminus L}\|\widehat{X}'_{I_j}\|_2^2\Big]=\mathbb{E}\Big[\sum_{j\in(S^*\cap\tilde{S})}\|\widehat{X}'_{I_j}\|_2^2\mathcal{I}\big[j\notin L\big]\Big]$$
$$\leq\sum_{j\in\tilde{S}}\|\widehat{X}'_{I_j}\|_2^2\cdot\Pr\big[j\notin L\big].\qquad(12)$$

We thus consider the probability $\Pr[j\notin L]$ for an arbitrary $j\in\tilde{S}$. If $j\in\tilde{S}$, then by Lemma 3.3 and the choice of the final parameter of $\frac{1}{2}\delta p$ passed to BudgetAllocation, there is at least one $r\in[2k_1]$ such that j is covered, with probability at least $1-\frac{1}{2}\delta p$. We also know from Lemma 2.4 that the failure probability of LocateReducedSignals for some covered j is at most $\frac{1}{2}\delta p$. A union bound on these two events gives

$$\Pr\big[j\notin L\big]\leq\delta p,\quad\forall j\in\tilde{S}.$$

Hence, we deduce from (12) that

$$\mathbb{E}\Big[\sum_{j\in(S^*\cap\tilde{S})\setminus L}\|\widehat{X}'_{I_j}\|_2^2\Big]\leq\delta p\cdot\|\widehat{X}'\|_2^2,$$

and Markov's inequality gives

$$\sum_{j\in(S^*\cap\tilde{S})\setminus L}\|\widehat{X}'_{I_j}\|_2^2\leq\delta\cdot\|\widehat{X}'\|_2^2$$

with probability at least $1-p$. Combining this with (10)–(11), and using the assumption $\delta\leq\frac{1}{20}$ to write $\delta\leq 100\sqrt{\delta}$, we complete the proof.

Sample complexity and runtime: We first consider the sample complexity and runtime as a function of the output L.

There are two operations that cost us samples. The first is the call to EstimateEnergies, which costs $O(\frac{k_0k_1}{\delta^2}\log^2\frac{1}{\delta})$ by Lemma 3.4. The second is the call to LocateReducedSignals; by Lemma 2.4 in Appendix F, with δp in place of p, this costs $O\big(\sum_{r\in[2k_1]}s^r\log\frac{1}{\delta p}\log\frac{1}{\delta}\log n\big)$ samples (recall that $\widehat{\chi}$ is $(O(k_0),k_1)$-block sparse by assumption), which is $O\big(|L|\log\frac{1}{\delta p}\log\frac{1}{\delta}\log n\big)$. Adding these contributions gives the desired result; the $\log\frac{1}{p}$ and $\log\frac{1}{\delta}$ factors are hidden in the $O^*(\cdot)$ notation, since we have assumed that δ and p behave as $\Omega\big(\frac{1}{\operatorname{poly}\log n}\big)$.

The time complexity follows by the a similar argument, with EstimateEnergies costing $O(\frac{k_0k_1}{\delta^2}\log^2\frac{1}{\delta}\log^2 n)$ by Lemma 3.4, and the call to LocateReducedSignals costing $O\big(|L|\log\frac{1}{\delta p}\log\frac{1}{\delta}\log^2 n+\frac{k_0k_1}{\delta}\log\frac{1}{\delta p}\log^3 n\big)$ by Lemma 2.4 in Appendix F. The complexity of EstimateEnergies dominates that and of calling BudgetAllocation, which is $O\big(k_1+\frac{k_0}{\delta}\log\frac{1}{p}\big)$ by Lemma 3.1.

The *expected* sample complexity and runtime follow directly from those depending on L, by simply substituting the expectation of $|L|$ given in the lemma statement. □

4 Energy Estimation

In this section, we provide the energy estimation procedure used in the MULTIBLOCKLOCATE procedure in Algorithm 2, and prove its guarantees that were used in the proof of Lemma 3.5. To do this, we introduce a variety of tools needed, including hashing and the semi-equispaced FFT. While such techniques are well-established for the standard sparsity setting [25], applying the existing semi-equispaced FFT algorithms *separately* for each Z^r in our setting would lead to a runtime of $k_0 k_1^2 \mathrm{poly}(\log n)$. Our techniques allow us to compute the required FFT values for *all* r in $k_0 k_1 \mathrm{poly}(\log n)$ time, as we detail in Section 4.2.

4.1 Hashing Techniques

The notion of hashing plays a central role in our estimation primitives, and in turn makes use of random permutations.

Definition 4.1 (Approximately pairwise-independent permutation)**.** Fix n, and let $\pi : [n] \to [n]$ be a random permutation. We say that π is *approximately pairwise-independent* if, for any $i, i' \in [n]$ and any integer t, we have $\mathbb{P}[|\pi(i) - \pi(i')| \le t] \le \frac{4t}{n}$.

It is well known that such permutations exist in the form of a simple modulo-n multiplication; we will specifically use the following lemma from [39].

Lemma 4.1. (Choice of permutation [39, Lemma 3.2]) *Let n be a power of two, and define $\pi(i) = \sigma \cdot i$, where σ is chosen uniformly at random from the odd numbers in $[n]$. Then π is an approximately pairwise-independent random permutation.*

We now turn to the notion of *hashing* a signal into buckets. We do this by applying the random permutation from Lemma 4.1 along with a random shift in time domain, and then applying a suitable filter according to Definition 2.1.

Definition 4.2 (Hashing)**.** Given integers (n, B), parameters $\sigma, \Delta \in [n]$, and the signals $X \in \mathbb{C}^n$ and $G \in \mathbb{C}^n$, we say that $U \in \mathbb{C}^B$ is an $(n, B, G, \sigma, \Delta)$*-hashing* of X if

$$U_b = \frac{B}{n} \sum_{i \in [\frac{n}{B}]} X_{\sigma(\Delta + j + B \cdot i)} G_{j + B \cdot i}, \quad j \in [B]. \tag{13}$$

Moreover, we define the following quantities:

- $\pi(j) = \sigma \cdot j$, representing the approximately pairwise random permutation;
- $h(j) = \mathrm{round}\big(j \frac{B}{n}\big)$, representing the bucket in $[B]$ into which a frequency j hashes;
- $o_j(j') = \pi(j') - h(j) \frac{n}{B}$, representing the offset associated with two frequencies (j, j').

With these definitions, we have the following lemma, proved in Appendix C.1. Note that here we write the exact Fourier transform of U as $\widehat{U}^*$, since later we will use $\widehat{U}$ for its *near-exact* counterpart to simplify notation.

Lemma 4.2. (Fourier transform of hashed signal) *Fix (n, B) and the signals $X \in \mathbb{C}^n$ and $G \in \mathbb{C}^n$ with the latter symmetric about zero. If U is an $(n, B, G, \sigma, \Delta)$-hashing of X, then its exact Fourier transform $\widehat{U}^*$ is given by*

$$\widehat{U}^*_b = \sum_{f \in [n]} \widehat{X}_f \widehat{G}_{\sigma f - b\frac{n}{B}} \omega_n^{\sigma \Delta f}, \quad b \in [B].$$

We conclude this subsection by stating the following technical lemma regarding approximately pairwise independent permutations and flat filters.

Lemma 4.3. (Additional filter property) *Fix n, and let G be an (n, B, F)-flat filter. Let $\pi(\cdot)$ be an approximately pairwise-independent random permutation (*cf.*, Definition 4.1), and for $f, f' \in [n]$, define $o_f(f') = \pi(f') - \frac{n}{B}\mathrm{round}\big(\pi(f)\frac{B}{n}\big)$. Then for any $x \in \mathbb{C}^n$ and $f \in [n]$, we have*

$$\sum_{f' \neq f} |\widehat{X}_{f'}|^2 \mathbb{E}_\pi\big[|\widehat{G}_{o_f(f')}|^2\big] \le \frac{10}{B} \|\widehat{X}\|^2. \tag{14}$$

The proof is given in Appendix C.2.

4.2 Semi-Equispaced FFT

Algorithm 3 Semi-equispaced inverse FFT for approximating the inverse Fourier transform, with standard sparsity (top) and block sparsity (bottom)

```
procedure SemiEquiInverseFFT($\widehat{X}, n, k, \zeta$)
    $\widehat{G} \leftarrow$ Filter($n, k, \zeta$)        ▷ See [25, Sec. 12]; same as proof of Lemma 4.5
    $\widehat{Y}_i \leftarrow (\widehat{X} \star \widehat{G})_{\frac{in}{2k}}$ for each $i \in [2k]$
    $Y \leftarrow$ InverseFFT($\widehat{Y}$)
    return $\{Y_j\}_{|j| \le \frac{k}{2}}$
procedure SemiEquiInverseBlockFFT($\widehat{X}, n, k_0, k_1, c$)
    $\widehat{G} \leftarrow$ Filter($n, k_1, n^{-c}$)        ▷ See proof of Lemma 4.5
    for $j \in \big[\frac{2n}{k_1}\big]$ such that $(\widehat{X} \star \widehat{G})_{\frac{k_1}{2}j}$ may be non-zero ($O(ck_0 \log n)$ in total) do
        $\widetilde{Y}^b_j \leftarrow \frac{k_1}{2} \sum_{l=1}^{\frac{n}{2k_1}} \widehat{X}_{b+2k_1 l} \widehat{G}_{\frac{k_1}{2}j-(b+2k_1 l)}$ for each $b \in [2k_1]$
        $(\widehat{Y}^1_j, \dots, \widehat{Y}^{2k_1}_j) \leftarrow$ IFFT($\widetilde{Y}^1_j, \dots, \widetilde{Y}^{2k_1}_j$)
    for $r \in [2k_1]$ do
        $\widehat{Y}^r \leftarrow (\widehat{Y}^r_1, \dots, \widehat{Y}^r_{n/k_1})$
        $Y^r \leftarrow$ SemiEquiInverseFFT($\widehat{Y}^r, \frac{n}{k_1}, k_0, n^{-(c+1)}$)
    return $\{Y^r_j\}_{r \in [2k_1], |j| \le \frac{k_0}{2}}$
```

One of the steps of our algorithm will be to take the inverse Fourier transform of our current estimate of the spectrum, so that it can be subtracted off and we can work with the residual. The *semi-equispaced inverse FFT* provides an efficient method for doing this, and is based on the application of the standard inverse FFT to a filtered and downsampled signal.

We start by describing an existing technique of this type for *standard* sparsity; the details are shown in the procedure SemiEquiInverseFFT in Algorithm 3, and the resulting guarantee from [25, Sec. 12] is stated as follows.[2]

Lemma 4.4. (SemiEquiInverseFFT guarantees [25, Lemma 12.1, Cor. 12.2]) *(i) Fix n and a parameter $\zeta > 0$. If $\widehat{X} \in \mathbb{C}^n$ is k-sparse for some k, then* SemiEquiInverseFFT*$(\widehat{X}, n, k, \zeta)$ returns a set of values $\{Y_j\}_{|j| \le k/2}$ in time $O(k \log \frac{n}{\zeta})$, satisfying*

$$|Y_j - X_j| \le \zeta \|X\|_2.$$

(ii) Given two additional parameters $\sigma, \Delta \in [n]$ with σ odd, it is possible to compute a set of values $\{Y_j\}$ for all j equaling $\sigma j' + \Delta$ for some j' with $|j'| \le k/2$, with the same running time and approximation guarantee.

For the block-sparse setting, we need to adapt the techniques of [25], making use of a *two-level* scheme that calls SemiEquiInverseFFT. The resulting procedure, SemiEquiInverseBlockFFT, is described in Algorithm 3. The main result of the procedure is the following analog of Lemma 4.4.

Lemma 4.5. (SemiEquiInverseBlockFFT guarantees) *(i) Fix (n, k_0, k_1), a (k_0, k_1)-block sparse signal $\widehat{X} \in \mathbb{C}^n$, and a constant $c \ge 1$. Define the shifted signals $\{X^r\}_{r \in [2k_1]}$ with $X^r_i = X_{i + \frac{nr}{2k_1}}$. The procedure* SemiEquiInverseBlockFFT*$(\widehat{X}, n, k_0, k_1, c)$ returns a set of values Y^r_j for all $r \in [2k_1]$ and $|j| \le \frac{k_0}{2}$ in time $O(c^2 k_0 k_1 \log^2 n)$, satisfying*

$$|Y^r_j - X^r_j| \le 2n^{-c} \|X\|_2. \tag{15}$$

(ii) Given two additional parameters $\sigma, \Delta \in \left[\frac{n}{k_1}\right]$ with σ odd, it is possible to compute a set of values Y^r_j for all $r \in [2k_1]$ and j equaling $\sigma j' + \Delta$ (modulo $\frac{n}{k_1}$) for some $|j'| \le \frac{k_0}{2}$, with the same running time and approximation guarantee.

The proof is given in Appendix C.3.

Remark 4.1. In the preceding lemmas, the signal sparsity and the number of values we wish to estimate will not always be identical. However, this can immediately be resolved by letting the parameter k_0 therein equal the maximum of the two.

4.3 Combining the Tools

In Algorithm 4, we describe two procedures combining the above tools. The first, HashToBins, accepts the signal $\widehat{X}$ and its current estimate $\widehat{\chi}$, uses SemiEquiInverseFFT to approximate the relevant entries of χ, and computes a hashing of $X - \chi$ as per Definition 4.2. The second, HashToBinsReduced, is analogous, but instead accepts a (k_1, δ)-downsampling of X, and uses SemiEquiInverseBlockFFT. It will prove useful to allow the function to hash into a different number of buckets for differing r values, and hence accept $\{G^r\}_{r \in [2k_1]}$ and $\{B^r\}_{r \in [2k_1]}$ as inputs. For simplicity, Algorithm 4 states the procedures without precisely giving the parameters passed to the semi-equispaced FFT, but the details are given in the proof of the following.

[2] Note that the roles of time and frequency are reversed here compared to [25].

Algorithm 4 Hash to bins functions for original signal (top) and reduced signals (bottom)

```
procedure HashToBins($X, \widehat{\chi}, G, n, B, \sigma, \Delta$)
    Compute $\{\chi_i\}$ using SemiEquiInverseFFT with input $(\widehat{\chi}, n, O(FB), n^{-c'})$
                ▷ See Lemma 4.4; $F$ equals the parameter of filter $G$, and $c'$ is a large constant
    $U_X \leftarrow (n, B, G, \sigma, \Delta)$-hashing of $X$                    ▷ See Definition 4.2
    $U_\chi \leftarrow (n, B, G, \sigma, \Delta)$-hashing of $\chi$
    $\widehat{U} \leftarrow$ FFT of $U_X - U_\chi$
    return $\widehat{U}$
procedure HashToBinsReduced($\{Z_X^r\}_{r\in[2k_1]}, \widehat{\chi}, \{G^r\}_{r\in[2k_1]}, n, k_1, \{B^r\}_{r\in[2k_1]}, \sigma, \Delta$)
    $B_{\max} \leftarrow \max_{r\in[2k_1]} B^r$
    $k_0 \leftarrow$ minimal value such that $\widehat{\chi}$ is $(k_0, k_1)$-block sparse
    Compute $\{\chi_i\}$ using SemiEquiInverseBlockFFT; input $(\widehat{\chi}, n, O(F_{\max}B_{\max} + k_0), k_1, c')$
        ▷ See Lemma 4.5; $F_{\max}$ equals the maximal parameter of the filters $\{G^r\}$, and $c'$ is a large constant
    $\{Z_\chi^r\}_{r\in[2k_1]} \leftarrow (k_1, \delta)$-downsampling of $\chi$                    ▷ See Definition 2.2
    for $r \in [2k_1]$ do
        $U_X^r \leftarrow \left(\frac{n}{k_1}, B^r, G^r, \sigma, \Delta\right)$-hashing of $Z_X^r$                    ▷ See Definition 4.2
        $U_\chi^r \leftarrow \left(\frac{n}{k_1}, B^r, G^r, \sigma, \Delta\right)$-hashing of $Z_\chi^r$
        $\widehat{U}^r \leftarrow$ FFT of $U_X^r - U_\chi^r$
    return $\{\widehat{U}^r\}_{r\in[2k_1]}$
```

Lemma 4.6. (HashToBins and HashToBinsReduced guarantees) *(i) Fix (n,k,B,F), an (n,B,F)-flat filter G supported on an interval of length $O(FB)$, a signal $X \in \mathbb{C}^n$, a k-sparse signal $\widehat{\chi}$ For any (σ,Δ), the procedure* HashToBins$(X, \widehat{\chi}, G, n, B, \sigma, \Delta)$ *returns a sequence $\widehat{U}$ such that*

$$\|\widehat{U} - \widehat{U}^*\|_\infty \le n^{-c}\|\widehat{\chi}\|_2,$$

where $\widehat{U}^*$ is the exact Fourier transform of the (n,B,G,σ,Δ)-hashing of $X - \chi$ (see Definition 4.2), and $c = c' + O(1)$ for c' in Algorithm 4. Moreover, the sample complexity is $O(FB)$, and the runtime is $O(cF(B+k)\log n)$.

(ii) Fix (n, k_0, k_1) and the parameters $(\{B^r\}_{r\in[2k_1]}, F, \delta)$. For each $r \in [2k_1]$, fix an $\left(\frac{n}{k_1}, B^r, F\right)$-flat filter G^r supported on an interval of length $O(FB^r)$. Moreover, fix a signal $X \in \mathbb{C}^n$ and its (k_1,δ)-downsampling $\{Z^r\}_{r\in[2k_1]}$ with $\delta \in \left(\frac{1}{n}, \frac{1}{20}\right)$, and a (k_0,k_1)-block sparse signal $\widehat{\chi}$. For any (σ,Δ), the procedure HashToBinsReduced$(\{Z^r\}_{r\in[2k_1]}, \widehat{\chi}, \{G^r\}_{r\in[2k_1}, n, k_1, \{B^r\}_{r\in[2k_1}, \sigma, \Delta)$ *returns a set of sequences $\{\widehat{U}^r\}_{r\in[2k_1]}$ such that*

$$\|\widehat{U}^r - \widehat{U}^{*r}\|_\infty \le n^{-c}\|\widehat{\chi}\|_2, \quad r \in [2k_1],$$

where $\widehat{U}^{*r}$ is the exact Fourier transform of the $\left(\frac{n}{k_1}, B^r, G^r, \sigma, \Delta\right)$-hashing for the (k_1,δ)-downsampling of $X - \chi$, and $c = c' + O(1)$ for c' in Algorithm 4. Moreover, the sample complexity is $O\big(F\sum_{r\in[2k_1]} B^r \log\frac{1}{\delta}\big)$, and the runtime is $O\big(c^2(B_{\max}F + k_0)k_1\log^2 n\big)$ with $B_{\max} = \max_{r\in[2k_1]} B^r$.

The proof is given in Appendix C.4.

Remark 4.2. Throughout the paper, we consider c in Lemma 4.6 to be a large absolute constant. Specifically, various results make assumptions such as $\|\widehat{X} - \widehat{\chi}\|_2 \ge \frac{1}{\mathrm{poly}(n)}\|\widehat{\chi}\|_2$, and the results hold true when c is sufficiently large compared to implied exponent in the poly(n) notation. Essentially, the n^{-c} error term is so small that it can be thought of as zero, but we nevertheless handle it explicitly for completeness.

4.4 Estimating the Downsampled Signal Energies

We now come to the main task of this section, namely, approximating the energy of each $\widehat{Z}^r$. To do this, we hash into $B = \frac{4}{\delta^2} \cdot k_0$ buckets (*cf.*, Definition 4.2), and form the estimate as the energy of the hashed signal. The procedure is shown in Algorithm 5.

Before stating the guarantees of Algorithm 5, we provide the following lemma characterizing the approximation quality for an *exact* hashing of a signal, as opposed to the approximation returned by HashToBinsReduced. Intuitively, the first part states that we can accurately estimate the top coefficients well without necessarily capturing the noise, and the second part states that, in expectation, we do not over-estimate the total signal energy by more than a small constant factor.

Algorithm 5 Procedure for estimating energies of downsampled signals

```
procedure EstimateEnergies($X, \widehat{\chi}, n, k_0, k_1, \delta$)
    $B \leftarrow \frac{4}{\delta^2} \cdot k_0$
    $F \leftarrow 10 \log \frac{1}{\delta}$
    $H \leftarrow (\frac{n}{k_1}, B, F)$-flat filter                              ▷ See Definition 2.1
    $\Delta \leftarrow$ uniform random sample from $[\frac{n}{k_1}]$
    $\sigma \leftarrow$ uniform random sample from odd numbers in $[\frac{n}{k_1}]$
    $\{Z^r\}_{r \in [2k_1]} \leftarrow (k_1, \delta)$-downsampling of $X - \chi$          ▷ See Definition 2.2
    $\mathbf{H} \leftarrow (H, \dots, H)$
    $\mathbf{B} \leftarrow (B, \dots, B)$
    $\{\widehat{U}^r\}_{r \in [2k_1]} \leftarrow$ HashToBinsReduced($\{Z^r\}_{r \in [2k_1]}, \widehat{\chi}, \mathbf{H}, n, k_1, \mathbf{B}, \sigma, \Delta$)   ▷ See Section 4.1
    for $r \in [2k_1]$ do
        $\gamma^r \leftarrow \|\widehat{U}^r\|_2^2$
    return $\gamma$                                              ▷ Length-$2k_1$ vector of $\gamma^r$ values
```

Lemma 4.7. (Properties of exact hashing) *Fix the integers (m, B), the parameters $\delta \in \big(0, \frac{1}{20}\big)$ and $F' \ge 10 \log \frac{1}{\delta}$, and the signal $Y \in \mathbb{C}^m$ and (m, B, F')-flat filter H (*cf.*, Definition 2.1). Let U be an $(m, B, H, \sigma, \Delta)$-hashing of Y for uniformly random $\sigma, \Delta \in [m]$ with σ odd, and let $\pi(\cdot)$ be defined as in Definition 4.2. Then, letting $\widehat{U}^*$ denote the exact Fourier transform of U, we have the following:*

1. *For any set $S \subset [m]$,*

$$\mathbb{E}_{\Delta,\pi}\Big[\Big|\|\widehat{Y}_S\|_2^2 - \|\widehat{U}^*\|_2^2\Big|_+\Big] \le \bigg(10\sqrt{\frac{|S|}{B}} + 15\frac{|S|}{B} + 2\delta^2\bigg)\|\widehat{Y}\|_2^2,$$

where $|.|_+ = \max\{.\,, 0\}$, and $\|\widehat{Y}_S\|_2^2$ denotes $\sum_{j \in S} |\widehat{Y}_j|^2$.

2. *We have*

$$\mathbb{E}_{\Delta,\pi}\big[\|\widehat{U}^*\|_2^2\big] \le 3\|\widehat{Y}\|_2^2.$$

The proof is given in Appendix C.5.

We now present the following lemma, showing that the procedure EstimateEnergies provides us with an estimator satisfying the preconditions of Lemmas 3.2 and 3.3.

Lemma 3.4 (EstimateEnergies guarantees – re-stated from Section 3.1) *Given (n, k_0, k_1), the signals $X \in \mathbb{C}^n$ and $\widehat{\chi} \in \mathbb{C}^n$ with $\|\widehat{X} - \widehat{\chi}\|_2^2 \geq \frac{1}{\mathrm{poly}(n)}\|\widehat{\chi}\|_2$, and the parameter $\delta \in \left(\frac{1}{n}, \frac{1}{20}\right)$, the procedure* EstimateEnergies$(X, \widehat{\chi}, n, k_0, k_1, \delta)$ *returns a vector $\gamma \in \mathbb{R}^{2k_1}$ such that, for any given set S^* of cardinality at most $10k_0$, we have the following with probability at least $\frac{1}{2}$:*

1. $\sum_{r\in[2k_1]} \left| \|\widehat{Z}^r_{S^*}\|_2^2 - \gamma^r \right|_+ \leq 40\delta \sum_{r\in[2k_1]} \|\widehat{Z}^r\|_2^2$;
2. $\|\gamma\|_1 \leq 10 \sum_{r\in[2k_1]} \|\widehat{Z}^r\|_2^2$;

where $\{Z^r\}_{r\in[2k_1]}$ is the (k_1, δ)-downsampling of $X - \chi$ (see Definition 2.2).

Moreover, if $\widehat{\chi}$ is $(O(k_0), k_1)$-block sparse, then the sample complexity is $O(\frac{k_0 k_1}{\delta^2} \log^2 \frac{1}{\delta})$, and the runtime is $O(\frac{k_0 k_1}{\delta^2} \log^2 \frac{1}{\delta} \log^2 n)$.

Proof. **Analysis for the exact hashing sequence:** We start by considering the case that the call to HashToBinsReduced is replaced by an evaluation of the exact hashing sequence $\widehat{U}^{*r}$, i.e., Definition 4.2 applied to Z^r resulting from the (k_1, δ)-downsampling of $X - \chi$. In this case, by applying Lemma 4.7 with $\widehat{Y} = \widehat{Z}^r$, $B = \frac{4}{\delta^2}k_0$ and $S = S^*$ (and hence $|S| \leq 10k_0$), the right-hand side of the first claim therein becomes $(5\delta + (\frac{15}{4} + 2)\delta^2)\|\widehat{Z}^r\|_2^2 \leq 6\delta\|\widehat{Z}^r\|_2^2$, since $\delta \leq \frac{1}{20}$. By applying the lemma separately for each $r \in [2k_1]$ with $\widehat{Y} = \widehat{Z}^r$, and summing the corresponding expectations in the two claims therein over r, we obtain $\sum_{r\in[2k_1]} \mathbb{E}\big[\big|\|\widehat{Z}^r_{S^*}\|_2^2 - \|\widehat{U}^*\|_2^2\big|_+\big] \leq 6\delta \sum_{r\in[2k_1]} \|\widehat{Z}^r\|_2^2$ and $\sum_{r\in[2k_1]} \mathbb{E}\big[\|\widehat{U}^*\|_2^2\big] \leq 3\sum_{r\in[2k_1]} \|\widehat{Z}^r\|_2^2$. We apply Markov's inequality with a factor of 6 in the former and 3 in the latter, to conclude that the quantities $\gamma^{*r} = \|\widehat{U}^{*r}\|_2^2$ satisfy

$$\sum_{r\in[2k_1]} \left| \|\widehat{Z}^r_{S^*}\|_2^2 - \gamma^{*r} \right|_+ \leq 36\delta \sum_{r\in[2k_1]} \|\widehat{Z}^r\|_2^2 \tag{16}$$

$$\|\gamma^*\|_1 \leq 9 \sum_{r\in[2k_1]} \|\widehat{Z}^r\|_2^2, \tag{17}$$

with probability at least 1/2.

Incorporating $\mathbf{\frac{1}{n^c}}$ error from use of semi-equispaced FFT in HashToBinsReduced: Since $\widehat{U}^r$ is computed using HashToBinsReduced, the energy vector γ is different from the exact one γ^*, and we write

$$\sum_{r\in[2k_1]} \left| \|\widehat{Z}^r_{S^*}\|_2^2 - \gamma^r \right|_+ \leq \sum_{r\in[2k_1]} \left| \|\widehat{Z}^r_{S^*}\|_2^2 - \gamma^{*r} \right|_+ + \left|\gamma^r - \gamma^{*r}\right|. \tag{18}$$

By substituting $\gamma^r = \|\widehat{U}^r\|_2^2$ and $\gamma^{*r} = \|\widehat{U}^{*r}\|_2^2$, and using the identity $\big|\|a\|_2^2 - \|b\|_2^2\big| \leq 2\|a - b\|_2 \cdot \|b\|_2 + \|a - b\|_2^2$, we can write

$$\sum_{r\in[2k_1]} \left|\gamma^r - \gamma^{*r}\right| \leq \sum_{r\in[2k_1]} \Big(2\|\widehat{U}^r - \widehat{U}^{*r}\|_2\|\widehat{U}^{*r}\|_2 + \|\widehat{U}^r - \widehat{U}^{*r}\|_2^2\Big). \tag{19}$$

Upper bounding the ℓ_2 norm by the ℓ_∞ norm times the vector length, we have $\|\widehat{U}^r - \widehat{U}^{*r}\|_2 \leq \sqrt{n}\|\widehat{U}^r - \widehat{U}^{*r}\|_\infty \leq n^{-c+1/2}\|\widehat{\chi}\|_2$, where the second inequality follows from Lemma 4.6. Moreover, from the definition of $\widehat{U}^{*r}$ resulting from Definition 4.2 applied to Z^r, along with the filter property

$\|\widehat{G}\|_\infty$ in Definition 2.1, it follows that $\|\widehat{U}^{*r}\|_2 \le \|\widehat{G}\|_\infty \|\widehat{Z}^r\|_1 \le \sqrt{n}\|\widehat{Z}^r\|_2$. Combining these into (19) gives

$$\begin{aligned}\sum_{r\in[2k_1]} \big|\gamma^r - \gamma^{*r}\big| &\le \sum_{r\in[2k_1]} \Big(2n^{-c+1}\|\widehat{\chi}\|_2\|\widehat{Z}^r\|_2 + n^{-2c+1}\|\widehat{\chi}\|_2^2\Big) \\ &\le 2n^{-c+2}\sqrt{\sum_{r\in[2k_1]}\|\widehat{\chi}\|_2^2 \cdot \sum_{r\in[2k_1]}\|\widehat{Z}^r\|_2^2} + n^{-2c+1}k_1\|\widehat{\chi}\|_2^2 \\ &\le 2n^{-c+3}\|\widehat{\chi}\|_2\sqrt{\sum_{r\in[2k_1]}\|\widehat{Z}^r\|_2^2} + n^{-2c+2}\|\widehat{\chi}\|_2^2. \qquad (20)\end{aligned}$$

where the second line is by Cauchy-Schwarz, and the third by $k_1 \le n$.

By the second part of Lemma 2.3 and the assumption $\delta \le \frac{1}{20}$, we have $\sum_{r\in[2k_1]}\|\widehat{Z}^r\|_2^2 \ge \frac{1}{4}\|\widehat{X} - \widehat{\chi}\|_2^2 \ge \frac{1}{4n^{c'}}\|\widehat{\chi}\|_2^2$, where the second equality holds for some $c' > 0$ by the assumption $\|\widehat{X} - \widehat{\chi}\|_2^2 \ge \frac{1}{\mathrm{poly}(n)}\|\widehat{\chi}\|_2$. Hence, (20) gives

$$\sum_{r\in[2k_1]} \big|\gamma^r - \gamma^{*r}\big| \le 4\big(n^{-c+3}n^{c'/2} + n^{-2c+2}n^{c'}\big)\sum_{r\in[2k_1]}\|\widehat{Z}^r\|_2^2. \qquad (21)$$

Since we have chosen $\delta > 1/n$, the coefficient to the summation is upper bounded by 4δ when c is sufficiently large, thus yielding the first part of the lemma upon combining with (16).

To prove the second part, note that by the triangle inequality,

$$\begin{aligned}\|\gamma\|_1 &\le \|\gamma^*\|_1 + \Big|\|\gamma\|_1 - \|\gamma^*\|_1\Big| \\ &\le 9\sum_{r\in[2k_1]}\|\widehat{Z}^r\|_2^2 + \sum_{r\in[2k_1]}\big|\gamma^r - \gamma^{*r}\big|,\end{aligned} \qquad (22)$$

where we have applied (17). Again applying (21) and noting that the coefficient to the summation is less than one for sufficiently large c, the second claim of the lemma follows.

Sample complexity and runtime: The only step that uses samples is the call to HashToBinsReduced. By Lemma 4.6 and the choices $B = \frac{4}{\delta^2}k_0$ and $F = 10\log\frac{1}{\delta}$, this uses $O\big(k_1 F B \log\frac{1}{\delta}\big) = O\big(\frac{k_0 k_1}{\delta^2}\log^2\frac{1}{\delta}\big)$ samples per call. The time complexity follows by the same argument along the assumption that $\widehat{\chi}$ is $(O(k_0), k_1)$-block sparse, with an additional $\log^2 n$ factor following from Lemma 4.6. Note that the call to HashToBinsReduced dominates the computation of γ^r, which is $O(k_1 B)$, □

5 The Block-Sparse Fourier Transform

In this section, we combine the tools from the previous sections to obtain the full sublinear-time block sparse FFT algorithm, and provide its guarantees.

5.1 Additional Estimation Procedures

Before stating the final algorithm, we note the main procedures that it relies on: MultiBlockLocate, PruneLocation, and EstimateValues. We presented the first of these in Section 3. The latter two are somewhat more standard, and hence we relegate them to the appendices. However, for the sake of readability, we provide some intuition behind them here, and state their guarantees.

We begin with PruneLocation. The procedure MultiBlockLocate gives us a list of block indices containing the dominant signal blocks with high probability, with a list size $L = O^*\big(k_0 \log k_0\big)$. Estimating the values of all of these blocks in every iteration would not only cost $O^*(k_0 k_1 \log k_0)$ samples, but would also destroy the sparsity of the input signal: Most of the blocks correspond to noise, and thus the estimation error may dominate the values being estimated. The PruneLocation primitive is designed to alleviate these issues, pruning L to a list that contains mostly "signal" blocks, i.e., blocks that contain a large amount of energy. Some false positives and false negatives occur, but are controlled by Lemma 5.1 below. The procedure is given in Algorithm 8 in Appendix G.

The following lemma shows that with high probability, the pruning algorithm retains most of the energy in the head elements, while removing most tail elements.

Lemma 5.1. (PruneLocation guarantees) *Given (n, k_0, k_1), a list of block indices L, the parameters $\theta > 0$, $\delta \in \big(\frac{1}{n}, \frac{1}{20}\big)$ and $p \in (0,1)$, and the signals $X \in \mathbb{C}^n$ and $\widehat{\chi} \in \mathbb{C}^n$ with $\|\widehat{X} - \widehat{\chi}\|_2 \geq \frac{1}{\mathrm{poly}(n)} \|\widehat{\chi}\|_2$, the output L' of* PruneLocation$(X, \widehat{\chi}, L, n, k_0, k_1, \delta, p, \theta)$ *has the following properties:*

a. *Let S_{tail} denote the tail elements in signal $\widehat{X} - \widehat{\chi}$, defined as*

$$S_{\mathrm{tail}} = \Big\{ j \in \Big[\frac{n}{k_1}\Big] \;:\; \|(\widehat{X} - \widehat{\chi})_{I_j}\|_2 \leq \sqrt{\theta} - \sqrt{\frac{\delta}{k_0}} \|\widehat{X} - \widehat{\chi}\|_2 \Big\},$$

where I_j is defined in Definition 1.1. Then, we have

$$\mathbb{E}\Big[\big|L' \cap S_{\mathrm{tail}}\big|\Big] \leq \delta p \cdot |L \cap S_{\mathrm{tail}}|.$$

b. *Let S_{head} denote the head elements in signal $\widehat{X} - \widehat{\chi}$, defined as*

$$S_{\mathrm{head}} = \Big\{ j \in \Big[\frac{n}{k_1}\Big] \;:\; \|(\widehat{X} - \widehat{\chi})_{I_j}\|_2 \geq \sqrt{\theta} + \sqrt{\frac{\delta}{k_0}} \|\widehat{X} - \widehat{\chi}\|_2 \Big\}.$$

Then, we have

$$\mathbb{E}\Big[\sum_{j \in (L \cap S_{\mathrm{head}}) \setminus L'} \|(\widehat{X} - \widehat{\chi})_{I_j}\|_2^2 \Big] \leq \delta p \sum_{j \in L \cap S_{\mathrm{head}}} \|(\widehat{X} - \widehat{\chi})_{I_j}\|_2^2.$$

Moreover, provided that $\|\widehat{\chi}\|_0 = O(k_0 k_1)$, the sample complexity is $O(\frac{k_0 k_1}{\delta} \log \frac{1}{\delta p} \log \frac{1}{\delta})$, and the runtime is $O(\frac{k_0 k_1}{\delta} \log \frac{1}{\delta p} \log \frac{1}{\delta} \log n + k_1 \cdot |L| \log \frac{1}{\delta p})$.

The proof is given in Appendix G.

We are left with the procedure EstimateValues, which is a standard procedure for estimating the values at the frequencies within the blocks after they have been located. The details are given in Algorithm 9 in Appendix H.

Lemma 5.2. (EstimateValues guarantees) *For any integers (n, k_0, k_1), list of block indices L, parameters $\delta \in \big(\frac{1}{n}, \frac{1}{20}\big)$ and $p \in (0, 1/2)$, and signals $X \in \mathbb{C}^n$ and $\widehat{\chi} \in \mathbb{C}^n$ with $\|\widehat{X} - \widehat{\chi}\|_2 \geq \frac{1}{\mathrm{poly}(n)} \|\widehat{\chi}\|_2$, the output W of the function* EstimateValues$(X, \widehat{\chi}, L, n, k_0, k_1, \delta, p)$ *has the following property:*

$$\sum_{f \in \bigcup_{j \in L} I_j} |W_f - (\widehat{X} - \widehat{\chi})_f|^2 \le \delta \frac{|L|}{3k_0} \|\widehat{X} - \widehat{\chi}\|_2^2$$

with probability at least $1-p$, where I_j is the j-th block. Moreover, provided that $\|\widehat{\chi}\|_0 = O(k_0 k_1)$, the sample complexity is $O(\frac{k_0 k_1}{\delta} \log \frac{1}{p} \log \frac{1}{\delta})$, and the runtime is $O(\frac{k_0 k_1}{\delta} \log \frac{1}{p} \log \frac{1}{\delta} \log n + k_1 \cdot |L| \log \frac{1}{p})$.

The proof is given in Appendix H.

5.2 Statement of the Algorithm and Main Result

Our overall block-sparse Fourier transform algorithm is given in Algorithm 6. It first calls ReduceSNR, which performs an iterative procedure that picks up high energy components of the signal, subtracts them from the original signal, and then recurses on the residual signal $X^{(i)} = X - \chi^{(i)}$. Once this is done, the procedure RecoverAtConstSNR performs a final "clean-up" step to obtain the $(1+\epsilon)$-approximation guarantee.

Algorithm 6 Block-sparse Fourier transform.

```
procedure BlockSparseFT($X, n, k_0, k_1, \mathrm{SNR}', \nu^2, \epsilon$)
                                        ▷ $X \in \mathbb{C}^n$ is approximately $(k_0, k_1)$-block sparse
                                        ▷ $(\mathrm{SNR}', \nu^2)$ are upper bounds on $(\mathrm{SNR}, \mu^2)$ from Definition 1.2
                                        ▷ $\epsilon$ is the parameter for $(1 + O(\epsilon))$-approximate recovery
    $\widehat{\chi} \leftarrow$ ReduceSNR($X, n, k_0, k_1, \mathrm{SNR}', \nu^2$).
    $\widehat{\chi} \leftarrow$ RecoverAtConstSNR($X, \widehat{\chi}, n, k_0, k_1, \nu^2, \epsilon$).
    return $\widehat{\chi}$
procedure ReduceSNR($X, n, k_0, k_1, \mathrm{SNR}', \nu^2$)      ▷ Iteratively locate/estimate to reduce SNR
    $T \leftarrow \log \mathrm{SNR}'$
    $\delta \leftarrow$ small absolute constant
    $p \leftarrow \frac{\delta}{\log^2 \frac{k_0}{\delta} \log^4 \mathrm{SNR}'}$                ▷ Failure probability for subroutines
    $\widehat{\chi}^{(0)} \leftarrow 0$                                 ▷ $\widehat{\chi}^{(t)}$ is our current estimate of $\widehat{X}$
    for $t \in \{1, \dots, T\}$ do
        $L \leftarrow$ MultiBlockLocate($X, \widehat{\chi}^{(t-1)}, n, k_1, k_0, \delta, p$)
        $\theta \leftarrow 10 \cdot 2^{-t} \cdot \nu^2 \mathrm{SNR}'$                       ▷ Threshold for pruning
        $L' \leftarrow$ PruneLocation($X, \widehat{\chi}^{(t-1)}, L, n, k_0, k_1, \delta, p, \theta$)
        $\widehat{\chi}^{(t)} \leftarrow \widehat{\chi}^{(t-1)}+$ EstimateValues($X, \widehat{\chi}^{(t-1)}, L', n, k_0, k_1, \delta, p$)
    return $\widehat{\chi}^T$
procedure RecoverAtConstSNR($X, \widehat{\chi}, n, k_0, k_1, \epsilon$)          ▷ A final "clean-up" step
    $\eta \leftarrow$ small absolute constant
    $p \leftarrow \frac{\eta\epsilon}{\log^2 \frac{k_0}{\epsilon}}$                 ▷ Upper bound on failure probability for subroutines
    $L \leftarrow$ MultiBlockLocate($X, \widehat{\chi}, n, k_1, k_0, \epsilon^2, p$)
    $\theta \leftarrow 200\epsilon\nu^2$
    $L' \leftarrow$ PruneLocation($X, \widehat{\chi}, L, n, k_0, k_1, \epsilon, p, \theta$)
    $W \leftarrow$ EstimateValues($X, \widehat{\chi}, L', n, 3k_0/\epsilon, k_1, \epsilon, p$)
    $\widehat{\chi}' \leftarrow W + \widehat{\chi}$
    return $\widehat{\chi}'$
```

With these definitions in place, we can now state our final result, which formalizes Theorem 1.1.

Theorem 1.1 (Upper bound – formal version) *Given (n, k_0, k_1), the parameter $\epsilon \in \left(\frac{1}{n}, \frac{1}{20}\right)$, and the signal $X \in \mathbb{C}^n$, if X, SNR', μ^2, and ν^2 satisfy the following for (μ^2, SNR) given in Definition 1.2:*

1. $\mu^2 \leq \nu^2$;
2. $\|\widehat{X}\|_2^2 \leq (k_0\nu^2) \cdot \mathrm{SNR}'$;
3. $\mathrm{SNR}' = O(\mathrm{poly}(n))$;
4. $\mu^2 \geq \frac{\|\widehat{X}\|_2^2}{\mathrm{poly}(n)}$;

then with probability at least 0.8*, the procedure* $\text{BlockSparseFT}(X, n, k_0, k_1, \mathrm{SNR}, \nu^2, \epsilon)$ *satisfies the following: (i) The output $\widehat{\chi}$ satisfies*

$$\|\widehat{X} - \widehat{\chi}\|_2^2 \leq k_0(\mu^2 + O(\epsilon\nu^2)).$$

(ii) The sample complexity is is $O^*(k_0 \log(1+k_0) \log \mathrm{SNR}' \log n + k_0 k_1 \log \mathrm{SNR}' + \frac{k_0}{\epsilon^2} \log(1+k_0) \log n + \frac{k_0 k_1}{\epsilon^4})$, and the runtime is $O^*(k_0 \log(1+k_0) \log \mathrm{SNR}' \log^2 n + k_0 k_1 \log \mathrm{SNR}' \log^3 n + \frac{k_0}{\epsilon^2} \log(1+k_0) \log^2 n + \frac{k_0 k_1}{\epsilon^4} \log^2 n + \frac{k_0 k_1}{\epsilon^2} \log^3 n)$.

The assumptions of the theorem are essentially that we know upper bounds on the tail noise μ^2 and SNR. Moreover, in order to get the $(1+\epsilon)$-approximation guarantee, the former upper bound should be tight to within a constant factor.

In the remainder of the section, we provide the proof of Theorem 1.1, deferring the technical details to the appendices.

Guarantees for ReduceSNR and RecoverAtConstSNR. The following theorem proves the success of the function ReduceSNR. We again recall the definitions of Err^2, μ^2, and SNR in Definition 1.2.

Lemma 5.3. (ReduceSNR guarantees) *Given (n, k_0, k_1), parameters (ν, SNR'), and a signal $X \in \mathbb{C}^n$, if X, SNR', and ν^2 satisfy the following for (μ^2, SNR) given in Definition 1.2:*

1. $\mu^2 \leq \nu^2$;
2. $\|\widehat{X}\|_2^2 \leq (k_0\nu^2) \cdot \mathrm{SNR}'$;
3. $\mathrm{SNR}' = O(\mathrm{poly}(n))$;
4. $\nu^2 \geq \frac{\|\widehat{X}\|_2^2}{\mathrm{poly}(n)}$;

then the procedure $\text{ReduceSNR}(X, n, k_0, k_1, \mathrm{SNR}', \nu^2)$ *satisfies the following guarantees with probability at least* 0.9 *when the constant δ therein is sufficiently small: (i) The output $\widehat{\chi}^T$ satisfies*

$$\widehat{\chi}^T \text{ is } (3k_0, k_1)\text{-block sparse}$$
$$\|\widehat{X} - \widehat{\chi}^T\|_2^2 \leq 100 k_0 \nu^2.$$

(ii) The number of samples used is $O^*(k_0 \log(1+k_0) \log \mathrm{SNR}' \log n + k_0 k_1 \log \mathrm{SNR}')$, and the runtime is $O^*(k_0 \log(1+k_0) \log \mathrm{SNR}' \log^2 n + k_0 k_1 \log \mathrm{SNR}' \log^3 n)$.

The proof is given in Appendix D.1

The following theorem proves the success of the function RecoverAtConstSNR.

Lemma 5.4. (RecoverAtConstSNR guarantees) *Given* (n, k_0, k_1)*, parameters* $\nu^2 \geq \frac{\|\widehat{X}\|_2^2}{\text{poly}(n)}$ *and* $\epsilon \in \left(\frac{1}{n}, \frac{1}{20}\right)$*, and the signals* $X \in \mathbb{C}^n$ *and* $\widehat{\chi} \in \mathbb{C}^n$ *satisfying*

1. $\text{Err}^2(\widehat{X} - \widehat{\chi}, 10k_0, k_1) \leq k_0\nu^2$*;*

2. $\|\widehat{X} - \widehat{\chi}\|_2^2 \leq 100k_0\nu^2$*;*

the procedure RecoverAtConstSNR$(X, \widehat{\chi}, n, k_0, k_1, \epsilon)$ *satisfies the following guarantees with probability at least* 0.9 *when the constant* η *therein is sufficiently small: (i) The output* $\widehat{\chi}'$ *satisfies*

$$\|\widehat{X} - \widehat{\chi}'\|_2^2 \leq \text{Err}^2(\widehat{X} - \widehat{\chi}, 10k_0, k_1) + (4 \cdot 10^5)\epsilon k_0\nu^2. \tag{23}$$

(ii) If $\widehat{\chi}$ *is* $(O(k_0), k_1)$*-block sparse, then the number of samples used is* $O^*(\frac{k_0}{\epsilon^2}\log(1+k_0)\log n + \frac{k_0k_1}{\epsilon^4})$ *and the runtime is* $O^*(\frac{k_0}{\epsilon^2}\log(1+k_0)\log^2 n + \frac{k_0k_1}{\epsilon^4}\log^2 n + \frac{k_0k_1}{\epsilon^2}\log^3 n)$.

The proof is given in Appendix D.2.

Proof of Theorem 1.1. We are now in a position to prove Theorem 1.1 via a simple combination of Lemmas 5.3 and 5.4.

Success event associated with ReduceSNR: Define a successful run of ReduceSNR$(X, n, k_0, k_1, \text{SNR}, \nu^2)$ to mean mean the following conditions on the output $\widehat{\chi}^T$:

$$\|\widehat{X} - \widehat{\chi}^T\|_2^2 \leq 100k_0\nu^2$$
$$\text{Err}^2(\widehat{X} - \widehat{\chi}^T, 10k_0, k_1) \leq k_0\mu^2.$$

By Lemma 5.3, it follows that the probability of having a successful run of ReduceSNR is at least 0.9. Note that the second condition is not explicitly stated in Lemma 5.3, but it follows by using $3k_0$ blocks to cover the parts where $\widehat{\chi}^T$ is non-zero, and k_0 blocks to cover the dominant blocks of $\widehat{X}$, in accordance with Definition 1.2.

Success event associated with RecoverAtConstSNR: Define a successful run of RecoverAtConstSNR$(X, \widehat{\chi}, k_0, k_1, n, \epsilon)$ to mean the following conditions on the output $\widehat{\chi}$:

$$\|\widehat{X} - \widehat{\chi}'\|_2^2 \leq \text{Err}^2(\widehat{X} - \widehat{\chi}, 10k_0, k_1) + (4 \cdot 10^5)\epsilon k_0\nu^2.$$

Conditioning on event of having a successful run of ReduceSNR, by Lemma 5.4, it follows that the probability of having a successful run to RecoverAtConstSNR is at least 0.9.

By a union bound, the aforementioned events occur simultaneously with probability at least 0.8, as desired. Moreover, the sample complexity and runtime are a direct consequence of summing the contributions from Lemmas 5.3 and 5.4.

6 Lower Bound

Our upper bound in Theorem 1.1, in several scaling regimes, provides a strict improvement over standard sparse FFT algorithms in terms of sample complexity. The corresponding algorithm is inherently *adaptive*, which raises the important question of whether adaptivity is necessary in order to achieve these improvements. In this section, we show that the answer is affirmative, by proving the following formalization of Theorem 1.2.

Theorem 1.2 (Lower bound – formal version) *Fix (n, k_0, k_1) and $C > 0$, and suppose that there exists a non-adaptive algorithm that, when given a signal Y with Fourier transform $\widehat{Y}$, outputs a signal $\widehat{Y}'$ satisfying the following ℓ_2/ℓ_2-guarantee with probability at least $\frac{1}{2}$:*

$$\|\widehat{Y} - \widehat{Y}'\|_2^2 \le C \min_{\widehat{Y}^* \text{ is } (k_0,k_1)-\text{block sparse}} \|\widehat{Y} - \widehat{Y}^*\|_2^2. \tag{24}$$

Then the number of samples taken by the algorithm must behave as $\Omega\big(k_0 k_1 \log \frac{n}{k_0 k_1}\big)$.

Hence, for instance, if $k_0 = O(1)$ and $\mathrm{SNR} = O(1)$ then our adaptive algorithm uses $O(k_1 + \log n)$ samples, whereas any non-adaptive algorithm must use $\Omega\big(k_1 \log \frac{n}{k_1}\big)$ samples.

The remainder of this section is devoted to the proof of Theorem 1.2. Throughout the section, we let $k = k_0 k_1$ denote the total sparsity.

High-level overview: Our analysis follows the information-theoretic framework of [37]. However, whereas [37] considers a signal with k arbitrary dominant frequency locations and uniform noise, we consider signals where the $k = k_0 k_1$ dominant frequencies are (nearly) contiguous, and both the noise and signal are concentrated on an $O\big(\frac{1}{k}\big)$ fraction of the time domain.

As a result, while the difficulty in [37] arises from the fact that the algorithm needs to recover roughly $\log \frac{n}{k}$ bits per frequency location for k such locations, our source of difficulty is different. In our signal, there are only roughly $\log \frac{n}{k}$ bits to be learned about the location of *all* the blocks in frequency domain, but the signal is tightly concentrated on an $O\big(\frac{1}{k}\big)$ fraction of the input space. As a consequence, any non-adaptive algorithm is bound to waste most of its samples on regions of the input space where the signal is zero, and only an $O\big(\frac{1}{k}\big)$ fraction of its samples can be used to determine the single frequency that conveys the location of the blocks. In the presence of noise, this results in a lower bound on sample complexity of $\Omega\big(k \log \frac{n}{k}\big)$.

Information-theoretic preliminaries: We will make use of standard results from information theory, stated below. Here and subsequently, we use the notations $H(X)$, $H(Y|X)$, $I(X;Y)$ and $I(X;Y|U)$ for the (conditional) Shannon entropy and (conditional) mutual information (e.g., see [40]).

We first state Fano's inequality, a commonly-used tool for proving lower bounds by relating a conditional entropy to an error probability.

Lemma 6.1. (Fano's Inequality [40, Lemma 7.9.1]) *Fix the random variables (X, Y) with X being discrete, let X' be an estimator of X such that $X \to Y \to X'$ forms a Markov chain (i.e., X and X' are conditionally independent given Y), and define $P_\mathrm{e} := \mathbb{P}[X' \neq X]$. Then*

$$H(Y|X) \le 1 + P_\mathrm{e} \log |\mathcal{X}|,$$

where $\mathcal{X} = \mathrm{supp}(X)$. Consequently, if X is uniformly distributed, then

$$I(X;Y) \geq -1 + (1 - P_{\mathrm{e}}) \log |\mathcal{X}|.$$

The next result gives the formula for the capacity of a complex-valued additive white Gaussian noise channel, often referred to as the Shannon-Hartley theorem. Here and subsequently, $\mathrm{CN}(\mu, \sigma^2)$ denotes the complex normal distribution.

Lemma 6.2. (Complex Gaussian Channel Capacity [40, Thm. 2.8.1]) *For $Z \sim \mathrm{CN}(0, \sigma_z^2)$ and any complex random variable X with $\mathbb{E}[|X|^2] = \sigma_x^2$, we have*

$$I(X; X + Z) \leq \log\left(1 + \frac{\sigma_x^2}{\sigma_z^2}\right),$$

with equality if $X \sim \mathrm{CN}(0, \sigma_x^2)$.

The following lemma states the data processing inequality, which formalizes the statement that processing a channel output cannot increase the amount of information revealed about the input.

Lemma 6.3. (Data Processing Inequality [40, Thm. 2.8.1]) *For any random variables (X, Y, Z) such that $X \to Y \to Z$ forms a Markov chain, we have $I(X;Z) \leq I(X;Y)$.*

Finally, the following lemma bounds the mutual information between two vectors in terms of the individual mutual information terms between components of those vectors.

Lemma 6.4. (Mutual Information for Vectors [40, Lemma 7.9.2]) *For any random vectors $\mathbf{X} = (X_1, \dots, X_n)$ and $\mathbf{Y} = (Y_1, \dots, Y_n)$, if the entries of $\mathbf{Y}$ are conditionally independent given $\mathbf{X}$, then $I(\mathbf{X};\mathbf{Y}) \leq \sum_{i=1}^n I(X_i;Y_i)$.*

A communication game: We consider a communication game consisting of channel coding with a state known at both the encoder (Alice) and decoder (Bob), where block-sparse recovery is performed at the decoder. Recalling that we are in the non-adaptive setting, by Yao's minimax principle, we can assume that the samples are deterministic and require probability-$\frac{1}{2}$ recovery over a random ensemble of signals, as opposed to randomizing the samples and requiring constant-probability recovery for any given signal in the ensemble. Hence, we denote the *fixed* sampling locations by $\mathcal{A}$.

We now describe our hard input distribution. Each signal in the ensemble is indexed by two parameters (u, f^*), and is given by

$$X_t = \begin{cases} \omega^{f^* t} & t \in \{u, \dots, u + \frac{C'n}{k} - 1\} \\ 0 & \text{otherwise}, \end{cases} \tag{25}$$

where $C' > 0$ is a constant that will be chosen later, and where all indices are modulo-n. Hence, each signal is non-zero only in a window of length $\frac{C'n}{k}$, and within that window, the signal oscillates at a rate dictated by f^*. Specifically, u specifies where the signal is non-zero in time domain, and f^* specifies where the energy is concentrated in frequency domain. We restrict the values of u and f^* to the following sets:

$$\mathcal{U} = \left\{\frac{C'n}{k}, \frac{2C'n}{k} \dots, \left(\frac{k}{C'} - 1\right)\frac{C'n}{k}, n\right\}$$
$$\mathcal{F} = \left\{k, 2k, \dots, \left(\frac{n}{k} - 1\right)k, n\right\}.$$

The communication game is as follows:

1. Nature selects a *state* U and a *message* F uniformly from $\mathcal{U}$ and $\mathcal{F}$, respectively.
2. An *encoder* maps (U, F) to the signal X according to (25).
3. A *state-dependent channel* adds independent $\mathrm{CN}(0, \alpha)$ noise to X_t for each $t \in \{u, \dots, u + \frac{C'n}{k} - 1\}$, while keeping the other entries noiseless. This is written as $Y_t = X_t + W_t$, where
$$W_t \sim \begin{cases} \mathrm{CN}(0, \alpha) & \{u, \dots, u + \frac{C'n}{k} - 1\} \\ 0 & \text{otherwise.} \end{cases} \tag{26}$$
The channel output is given by $Y = \{Y_t\}_{t \in \mathcal{A}}$ for the sampling locations $\mathcal{A}$.
4. A *decoder* receives U and Y, applies (k_0, k_1)-block sparse recovery to Y to obtain a signal $\widehat{Y}'$, and then selects F' to be the frequency $f' \in \mathcal{F}$ such that the energy in $\widehat{Y}'$ within the length-k window centered at f' is maximized:
$$F' = \operatorname*{arg\,max}_{f' \in \mathcal{F}} \|\widehat{Y}'_{I_k(f')}\|_2^2, \tag{27}$$
where $I_k(f') = \{f' + \Delta \,:\, \Delta \in [k]\}$.

We observe that if adaptivity were allowed, then the knowledge of U at the decoder would make the block-sparse recovery easy – one could let all of the samples lie within the window given in (25). The problem is that we are in the *non-adaptive* setting, and hence we must take enough samples to account for all of the possible choices of U.

We denote the subset of $\mathcal{A}$ falling into $\{u, \dots, u + \frac{C'n}{k} - 1\}$ by $\mathcal{A}_u$, its cardinality by m_u and the total number of measurements by $m = |\mathcal{A}| = \sum_u m_u$. Moreover, we let $X_{\mathcal{A}_u}$ and $Y_{\mathcal{A}_u}$ denote the sub-vectors of X and Y indexed by $\mathcal{A}_u$.

Information-theoretic analysis: We first state the following lemma.

Lemma 6.5. (Mutual information bound) *In the setting described above, the conditional mutual information $I(F; Y|U)$ satisfies $I(F; Y|U) \le \frac{C'm}{k} \log\left(1 + \frac{1}{\alpha}\right)$, where α is variance of the additive Gaussian noise within $\mathcal{A}_u$.*

Proof. We have

$$\begin{aligned} I(F;Y|U) &= \frac{C'}{k}\sum_u I(F;Y|U=u) \\ &= \frac{C'}{k}\sum_u I(F;Y_{\mathcal{A}_u}|U=u) \\ &\le \frac{C'}{k}\sum_u I(X_{\mathcal{A}_u};Y_{\mathcal{A}_u}|U=u) \\ &\le \frac{C'}{k}\sum_u \sum_{t\in\mathcal{A}_u} I(X_t;Y_t|U=u) \\ &\le \frac{C'}{k}\sum_u m_u \log\Big(1+\frac{1}{\alpha}\Big) \\ &= \frac{C'm}{k}\log\Big(1+\frac{1}{\alpha}\Big), \end{aligned} \tag{28}$$

where:

- Line 1 follows since U is uniform on a set of cardinality $\frac{k}{C'}$;
- Line 2 follows since given $U=u$, only the entries of Y indexed by $\mathcal{A}_u$ are dependent on F (*cf.*, (25));
- Line 3 follows by noting that given $U=u$ we have the Markov chain $F\to X_{\mathcal{A}_u}\to Y_{\mathcal{A}_u}$, and applying the data processing inequality (Lemma 6.3);
- Line 4 follows from Lemma 6.4 and (26), where the conditional independence assumption holds because we have assumed the random variables W_t are independent;
- Line 5 follows from the Shannon-Hartley Theorem (Lemma 6.2); in our case, the signal power is exactly one by (25), and the average noise energy is α by construction.

□

Next, defining $\delta_u := \Pr[F'\neq F\,|\,U=u]$, Fano's inequality (Lemma 6.1) gives

$$I(F;Y|U=u) \ge -1 + (1-\delta_u)\log\frac{n}{k},$$

and averaging both sides over U gives

$$I(F;Y|U) \ge -1 + (1-\delta)\log\frac{n}{k},$$

where $\delta := \mathbb{E}[\delta_U] = \Pr[F'\neq F]$.

Hence, and by Lemma 6.5, if we can show that our ℓ_2/ℓ_2-error guarantee (24) gives $F'=F$ constant probability, then we can conclude that

$$m \ge \frac{k\big((1-\delta)\log\frac{n}{k}-1\big)}{C'\log\big(1+\frac{1}{\alpha}\big)} = \Omega\Big(k\log\frac{n}{k}\Big).$$

We therefore conclude the proof of Theorem 1.2 by proving the following lemma.

Lemma 6.6. (Probability of error characterization) *Fix (n, k_0, k_1) and $C > 0$. If the (k_0, k_1)-block sparse recovery algorithm used in the above communication game satisfies* (24) *with probability at least $\frac{1}{2}$, then there exist choices of C' and α such that the decoder's estimate of F' according to* (27) *satisfies $F' = F$ with probability at least $\frac{1}{4}$.*

Proof. By the choice of estimator in (27), it suffices to show that $\widehat{Y}'$, the output of the block-sparse Fourier transform algorithm, has more than half of its energy within the length-k window $I_k(f^*)$ centered of f^*. We show this in three steps.

Characterizing the energy of $\widehat{X}$ within $I_k(f^*)$: The Fourier transform of X in (25) is a shifted sinc function of "width" $\frac{k}{C'}$ centered at f^* when the time window is centered at zero, and more generally, has the same magnitude as this sinc function. Hence, by letting C' be suitably large, we can ensure that an arbitrarily high fraction of the energy of $\widehat{X}$ falls within the length-k window centered at $f^* \in \mathcal{F}$. Formally, we have

$$\|\widehat{X}_{I_k(f^*)}\|_2^2 \geq (1-\eta)\|\widehat{X}\|_2^2 \tag{29}$$

for $\eta \in (0,1)$ that we can make arbitrarily small by choosing C' large.

Characterizing the energy of $\widehat{Y}$ within $I_k(f^*)$: We now show that, when the noise level α in (26) it sufficiently small, the energy in $\widehat{Y}$ within $I_k(f^*)$ is also large with high probability:

$$\sum_{f \in I_k(f^*)} |\widehat{Y}_f|^2 \geq (1-2\eta)\|\widehat{X}\|_2^2. \tag{30}$$

To prove this, we first note that $|\widehat{Y}_f|^2 = |\widehat{X}_f + \widehat{W}_f|^2$ for all $f \in [n]$, from which it follows that

$$\left| \sum_{f \in I_k(f^*)} |\widehat{Y}_f|^2 - \sum_{f \in I_k(f^*)} |\widehat{X}_f|^2 \right| \leq \sum_{f \in I_k(f^*)} |\widehat{W}_f|^2 + 2 \sum_{f \in I_k(f^*)} |\widehat{X}_f| \cdot |\widehat{W}_f|.$$

Upper bounding the summation over $|\widehat{W}_f|^2$ by the total noise energy, and upper bounding the summation over $|\widehat{X}_f| \cdot |\widehat{W}_f|$ using the Cauchy-Schwarz inequality, we obtain

$$\left| \sum_{f \in I_k(f^*)} |\widehat{Y}_f|^2 - \sum_{f \in I_k(f^*)} |\widehat{X}_f|^2 \right| \leq \|\widehat{W}\|_2^2 + 2\|\widehat{X}\|_2 \cdot \|\widehat{W}\|_2. \tag{31}$$

We therefore continue by bounding the *total* noise energy $\|\widehat{W}\|_2^2$; the precise distribution of the noise across different frequencies is not important for our purposes.

Recall that every non-zero entry of X has magnitude one, and every non-zero time-domain entry of W is independently distributed as $\mathrm{CN}(0,\alpha)$. Combining these observations gives $\mathbb{E}[\|W\|_2^2] = \alpha\|X\|_2^2$, or equivalently $\mathbb{E}[\|\widehat{W}\|_2^2] = \alpha\|\widehat{X}\|_2^2$ by Parseval. Therefore, by Markov's inequality, we have $\|\widehat{W}\|_2^2 \leq 4\alpha\|\widehat{X}\|_2^2$ with probability at least $\frac{3}{4}$. When this occurs, (31) gives

$$\left| \sum_{f \in I_k(f^*)} |\widehat{Y}_f|^2 - \sum_{f \in I_k(f^*)} |\widehat{X}_f|^2 \right| \leq 4(\alpha + \sqrt{\alpha})\|\widehat{X}\|_2^2. \tag{32}$$

If we choose $\alpha = \frac{\eta^2}{100}$, then we have $4(\alpha + \sqrt{\alpha}) = \frac{\eta^2}{25} + \frac{4\eta}{10} \leq \eta$. In this case, by (29) and (32), the length-k window $I_k(f^*)$ centered at f^* satisfies (30).

Characterizing the energy of $\widehat{Y}'$ within $I_k(f^*)$: The final step is to prove that (30) and (24) imply the following with constant probability for a suitable choice of η:

$$\sum_{f \in I_k(f^*)} |\widehat{Y}'_f|^2 > \frac{1}{2}\|\widehat{Y}'\|_2^2, \tag{33}$$

where $\widehat{Y}'$ is the output of the block-sparse recovery algorithm. This clearly implies that $F = F'$, due to our choice of estimator in (27).

As a first step towards establishing (33), we rewrite (30) as

$$\sum_{f \in [n] \setminus I_k(f^*)} |\widehat{Y}_f|^2 \le \|\widehat{Y}\|_2^2 - (1 - 2\eta)\|\widehat{X}\|_2^2. \tag{34}$$

We can interpret (34) as an error term $\|\widehat{Y} - \widehat{Y}^*\|_2^2$ for a signal $\widehat{Y}^*$ coinciding with $\widehat{Y}$ within $I_k(f^*)$ and being zero elsewhere. Since $I_k(f^*)$ contains k contiguous elements, this signal is (k_0, k_1)-block sparse, and hence if the guarantee in (24) holds, then combining with (34) gives

$$\|\widehat{Y} - \widehat{Y}'\|_2^2 \le C\Big(\|\widehat{Y}\|_2^2 - (1 - 2\eta)\|\widehat{X}\|_2^2\Big). \tag{35}$$

We henceforth condition on both (24) and the above-mentioned event $\|\widehat{W}\|_2^2 \le 4\alpha\|\widehat{X}\|_2^2$. Since the former occurs with probability at least $\frac{1}{2}$ by assumption, and the latter occurs with probability at least $\frac{3}{4}$, their intersection occurs with probability at least $\frac{1}{4}$.

Next, we write the conditions in (30) and (35) in terms of $\|\widehat{Y}\|_2^2$, rather than $\|\widehat{X}\|_2^2$. Since $\widehat{X} = \widehat{Y} - \widehat{W}$, we can use the triangle inequality to write $\|\widehat{X}\|_2 \ge \|\widehat{Y}\|_2 - \|\widehat{W}\|_2$, and combining this with $\|\widehat{W}\|_2^2 \le 4\alpha\|\widehat{X}\|_2^2$, we obtain $\|\widehat{X}\|_2 \ge \frac{\|\widehat{Y}\|_2}{1+2\sqrt{\alpha}}$. Hence, we can weaken (30) and (35) to

$$\sum_{f \in I_k(f^*)} |\widehat{Y}_f|^2 \ge \frac{1 - 2\eta}{(1 + 2\sqrt{\alpha})^2}\|\widehat{Y}\|_2^2 \ge 0.99\|\widehat{Y}\|_2^2 \tag{36}$$

$$\|\widehat{Y} - \widehat{Y}'\|_2^2 \le C\Big(1 - \frac{1 - 2\eta}{(1 + 2\sqrt{\alpha})^2}\Big)\|\widehat{Y}\|_2^2 \le 0.01\|\widehat{Y}\|_2^2, \tag{37}$$

where the second step in each equation holds for sufficiently small η due to the choice $\alpha = \frac{\eta^2}{100}$.

It only remains to use (36)–(37) to bound the left-hand side of (33). To do this, we first note that by interpreting both (36) and (37) as bounds on $\|\widehat{Y}\|_2^2$, and using $\|\widehat{Y} - \widehat{Y}'\|_2^2 \ge \|(\widehat{Y} - \widehat{Y}')_{I_k(f^*)}\|_2^2$ in the latter, we have

$$\sum_{f \in I_k(f^*)} |\widehat{Y}_f - \widehat{Y}'_f|^2 \le 0.02 \sum_{f \in I_k(f^*)} |\widehat{Y}_f|^2,$$

since $\frac{0.01}{0.99} \le 0.02$. Taking the square root and applying the triangle inequality to the ℓ_2-norm on the left-hand side, we obtain

$$\sum_{f \in I_k(f^*)} |\widehat{Y}'_f|^2 \ge (1 - \sqrt{0.02})^2 \sum_{f \in I_k(f^*)} |\widehat{Y}_f|^2. \tag{38}$$

Next, writing $\|\widehat{Y}'\|_2 = \|\widehat{Y} + (\widehat{Y}' - \widehat{Y})\|_2$, and applying the triangle inequality followed by (37), we have $\|\widehat{Y}'\|_2 \le 1.1\|\widehat{Y}\|_2$, and hence $\|\widehat{Y}\|_2 \ge 0.9\|\widehat{Y}'\|_2$. Squaring and substituting into (36), we obtain

$$\sum_{f \in I_k(f^*)} |\widehat{Y}_f|^2 \ge 0.8\|\widehat{Y}'\|_2^2. \tag{39}$$

Finally, combining (38) and (39) yields (33), and we have thus shown that (33) holds (and hence

$F = F'$) with probability at least $\frac{1}{4}$. □

7 Acknowledgements

The second author would like to thank Piotr Indyk for helpful discussions.

A Omitted Proofs from Section 2

A.1 Proof of Lemma 2.1

Our filter construction is similar to [39], but we prove and utilize different properties, and hence provide the details for completeness.

Definition A.1 (Rectangular pulse)**.** For an even integer B', let $\mathrm{rect}^{B'}$ denote the rectangular pulse of width $B'-1$, i.e.

$$\mathrm{rect}^{B'}_t = \begin{cases} 1, & \text{if } |t| < \frac{B'}{2} \\ 0 & \text{otherwise.} \end{cases}$$

For an integer $B' > 0$ a power of 2, define the length-n signal

$$W = \left(\frac{n}{B'-1} \cdot \mathrm{rect}^{B'}\right) * \cdots * \left(\frac{n}{B'-1} \cdot \mathrm{rect}^{B'}\right), \tag{40}$$

where the convolution is performed F times. As noted in [39], we have $\mathrm{supp}(W^F) \subseteq [-F \cdot B', F \cdot B']$, and the Fourier transform is given by

$$\widehat{W}_f = \left(\frac{1}{B'-1} \sum_{|f'|<\frac{B'}{2}} \omega_n^{ff'}\right)^F = \left(\frac{\sin(\pi(B'-1)f/n)}{(B'-1)\sin(\pi f/n)}\right)^F \tag{41}$$

for $f \neq 0$, and $W_0 = 1$.

Lemma A.1. (Properties of W) *For every even $F \geq 2$, the following hold for the signal W defined in* (40)–(41)*:*

1 $\widehat{W}_f \in [0,1]$ *for all $f \in [n]$;*

2 *There exists an absolute constant $C \geq 0$ such that for every $\lambda > 1$,*

$$\sum_{f \in [n],\, |f| \geq \frac{\lambda \cdot n}{2B'}} \widehat{W}_f \leq (C/\lambda)^{F-1} \sum_{f \in [n]} \widehat{W}_f.$$

Proof. First note that the maximum of $\widehat{W}_f$ is achieved at 0 and equals 1. Since F is even by assumption, we have from (41) that $\widehat{W}_f \geq 0$ for all f. These two facts establish the first claim.

To prove the second claim, note that for all $f \in [n]$, we have

$$\begin{aligned} \widehat{W}_f = \left|\frac{\sin(\pi(B'-1)f/n)}{(B'-1)\sin(\pi f/n)}\right|^F &\leq \left|\frac{1}{(B'-1)\sin(\pi f/n)}\right|^F \qquad \text{(since } |\sin(\pi x)| \leq 1) \\ &\leq \left|\frac{1}{(B'-1)2|f|/n}\right|^F \qquad \text{(since } |\sin(\pi x)| \geq 2|x| \text{ for } |x| \leq 1/2). \end{aligned} \tag{42}$$

We claim that this can be weakened to

$$\widehat{W}_f \le \left(\frac{n}{B'|f|}\right)^F. \tag{43}$$

For $f \in [-n/B', n/B']$ the right-hand side is at least one, and hence this claim follows directly from the first claim above. On the other hand, if $|f| \ge n/B'$, we have

$$\begin{aligned} 2(B'-1)|f|/n &= 2B'|f|/n - 2|f|/n \\ &\ge 2B'|f|/n - 1 \quad (\text{since } |f| \le n/2) \\ &\ge B'|f|/n \quad (\text{since } |f| \ge n/B'), \end{aligned}$$

and hence (43) follows from (42).

Using (43), we have

$$\sum_{|f| \ge \frac{\lambda \cdot n}{2B'}} \widehat{W}_f \le \sum_{|f| \ge \frac{\lambda \cdot n}{2B'}} \left(\frac{n}{B'|f|}\right)^F = O(\lambda^{-F+1}) \cdot \frac{n}{B'}. \tag{44}$$

At the same time, for any $f \in [-\frac{n}{2B'}, \frac{n}{2B'}]$, we have

$$\begin{aligned} \widehat{W}_f &= \left|\frac{\sin(\pi(B'-1)f/n)}{(B'-1)\sin(\pi f/n)}\right|^F \\ &\ge \left|\frac{2(B'-1)f/n}{(B'-1)\sin(\pi f/n)}\right|^F \qquad (\text{since } |\sin(\pi x)| \ge 2|x| \text{ for } |x| \le 1/2) \\ &\ge \left|\frac{2(B'-1)f/n}{(B'-1)\pi(f/n)}\right|^F \qquad (\text{since } |\sin(\pi x)| \le \pi|x|) \\ &= \left(\frac{2}{\pi}\right)^F. \end{aligned}$$

This means that

$$\sum_{f \in [n]} \widehat{W}_f \ge \sum_{f \in [-\frac{n}{2B'}, \frac{n}{2B'}]} \widehat{W}_f \ge \left(\frac{2}{\pi}\right)^F \cdot \frac{n}{B'}. \tag{45}$$

Putting (44) together with (45), we get

$$\sum_{f \in [n],\, |f| \ge \frac{\lambda \cdot n}{2B'}} \widehat{W}_f = O(\lambda^{-F+1}) \cdot \frac{n}{B'} \le C' \cdot \left(\frac{\pi}{2}\right)^F \lambda^{-F+1} \cdot \sum_{f \in [n]} W_f$$

for an absolute constant $C' > 0$. The desired claim follows since $C'(\pi/2)^F = ((C')^{1/(F-1)} \cdot (\pi/2)^{F/(F-1)}))^{F-1} \le (C' \cdot (\pi/2)^2)^{F-1}$ (due to the assumption that $F \ge 2$). □

We now fix an integer B, and define $\widehat{G}$ by

$$\widehat{G}_f = \frac{1}{Z} \sum_{\Delta = -\frac{3n}{4B}}^{\frac{3n}{4B}} \widehat{W}_{f-\Delta}.$$

where $Z = \sum_{f \in [n]} \widehat{W}_f$. By interpreting this as a convolution with a rectangle, we obtain that the inverse Fourier transform G_t is obtained via the multiplication of W_t with a sinc pulse.

We proceed by showing that, upon identifying $B' = 8CB$ (where B' was used in defining $\widehat{W}$, and C is the implied constant in Lemma A.1), this filter satisfies the claims of Lemma 2.1. We start with the three properties in Definition 2.1.

Proof of [Lemma 2.1 (filter property 1)]: For every f, we have

$$\widehat{G}_f = \frac{1}{Z} \cdot \sum_{\Delta=-\frac{3n}{4B}}^{\frac{3n}{4B}} \widehat{W}_{f-\Delta} \leq \frac{1}{Z} \sum_{\Delta \in [n]} \widehat{W}_{f-\Delta} = 1.$$

Similarly, the non-negativity of $\widehat{G}$ follows directly from that of $\widehat{W}$. □

Proof of [Lemma 2.1 (filter property 2)]: For every $f \in [n]$ with $|f| \leq \frac{n}{2B}$, we have

$$\begin{aligned}
\widehat{G}_f &= \frac{1}{Z} \cdot \sum_{\Delta=-\frac{3n}{4B}}^{\frac{3n}{4B}} \widehat{W}_{f-\Delta} \\
&= 1 - \frac{1}{Z} \sum_{|\Delta| > \frac{3n}{4B}} \widehat{W}_{f-\Delta} \\
&\geq 1 - \frac{2}{Z} \sum_{f' > \frac{n}{4B}} \widehat{W}_{f'} \quad \text{(since } |f| \leq \frac{n}{2B} \text{ and } W \text{ is symmetric)} \\
&= 1 - \frac{2}{Z} \sum_{f' > \frac{B'}{2B} \cdot \frac{n}{2B'}} \widehat{W}_{f'} \\
&\geq 1 - \Big(2C \cdot \frac{B}{B'}\Big)^{F-1}. \quad \text{(by Lemma A.1)}
\end{aligned}$$

Since $B'/B = 8C$ by our choice of B' above, we get $\widehat{G}_f \geq 1 - (1/4)^{F-1}$, as required. □

Proof of [Lemma 2.1 (filter property 3)]: For every $f \in [n]$ with $|f| \geq \frac{n}{B}$, we have

$$\begin{aligned}
\widehat{G}_f &= \frac{1}{Z} \cdot \sum_{\Delta=-\frac{3n}{4B}}^{\frac{3n}{4B}} \widehat{W}_{f-\Delta} \\
&\leq \frac{1}{Z} \cdot \sum_{f' \,:\, |f'| \geq |f| - \frac{3n}{4B}} \widehat{W}_{f'}. \quad \text{(by } |f| \geq \frac{n}{B}\text{)}
\end{aligned}$$

Defining $\zeta \geq 1$ such that $|f| = (3+\zeta)\frac{n}{4B}$, this becomes

$$\begin{aligned}
\widehat{G}_f &\leq \frac{1}{Z} \cdot \sum_{f' \,:\, |f'| \geq \frac{\zeta n}{4B}} \widehat{W}_{f'} \\
&= \frac{1}{Z} \cdot \sum_{f' \,:\, |f'| \geq \frac{\zeta B'}{2B} \cdot \frac{n}{2B'}} \widehat{W}_{f'} \\
&\leq \Big(\frac{2CB}{\zeta B'}\Big)^{F-1} \quad \text{(by Lemma A.1)} \\
&= \Big(\frac{1}{4\zeta}\Big)^{F-1} \quad \text{(since } B' = 8CB\text{)}.
\end{aligned}$$

Rearranging the definition of ζ, we obtain $\zeta = \frac{4B|f|}{n} - 3$, and hence $\zeta \geq \frac{B|f|}{n}$ due to the fact that $|f| \geq \frac{n}{B}$. Therefore, $\widehat{G}_f \leq \big(\frac{n}{4B|f|}\big)^{F-1}$. □

Proof of [Lemma 2.1 (additional property 1)]: We have already shown that W is supported on a window of length $O(FB') = O(FB)$ centered at zero. The same holds for G since it is obtained via a pointwise multiplication of W with a sinc pulse. $\square$

Proof of [Lemma 2.1 (additional property 2)]: Since $\widehat{G}_f \in [0,1]$, the total energy across $|f| < \frac{n}{B}$ is at most $\frac{2n}{B}$. On the other hand, we have from the third property in Definition 2.1 that

$$\begin{aligned}\sum_{|f| \ge \frac{n}{B}} |\widehat{G}_f|^2 &\le 2 \sum_{f \ge \frac{n}{B}} \Big(\frac{1}{4}\Big)^{2(F-1)} \Big(\frac{n}{Bf}\Big)^{2(F-1)} \\ &\le \frac{1}{8} \sum_{f \ge \frac{n}{B}} \Big(\frac{n}{Bf}\Big)^2 \quad \text{(since } F \ge 2\text{)} \\ &\le \frac{1}{8} \cdot \frac{n}{B} \sum_{f=1}^{\infty} \frac{1}{f^2} \\ &\le \frac{n}{B} \quad \Big(\text{since } \sum_{f=1}^{\infty} \frac{1}{f^2} < 8\Big).\end{aligned}$$

Combining this with the contribution from $|f| < \frac{n}{B}$ concludes the proof. $\square$

A.2 Proof of Lemma 2.3

We are interested in the behavior of $\sum_{r \in [2k_1]} |\widehat{Z}_j^r|^2$ for each j (first part), and summed over all j (second part). We therefore begin with the following lemma, bounding this summation in terms of the signal X and the filter G.

Lemma A.2. (Initial downsampling bound) *For any integers (n, k_1), parameter $\delta \in \big(0, \frac{1}{20}\big)$, signal $X \in \mathbb{C}^n$ and its corresponding (k_1, δ)-downsampling $\{Z^r\}_{r \in [2k_1]}$, the following holds for all $j \in [\frac{n}{k_1}]$:*

$$\left| \frac{1}{2k_1} \sum_{r \in [2k_1]} |\widehat{Z}_j^r|^2 - \sum_{f=1}^{n} |\widehat{G}_{f-k_1 j}|^2 \cdot |\widehat{X}_f|^2 \right| \le 3\delta \sum_{f=1}^{n} |\widehat{G}_{f-k_1 j}| \cdot |\widehat{X}_f|^2.$$

Proof. **Directly evaluating the sum:** Using the definition of the signals $\widehat{Z}^r$ in (2), we write

$$\begin{aligned}\sum_{r \in [2k_1]} |\widehat{Z}_j^r|^2 &= \sum_{r \in [2k_1]} \Big(\sum_{f=1}^{n} \widehat{G}_{f-k_1 j} \cdot \widehat{X}_f \cdot \omega_n^{a_r f} \Big)^{\dagger} \Big(\sum_{f'=1}^{n} \widehat{G}_{f'-kj} \cdot \widehat{X}_{f'} \cdot \omega_n^{a_r f'} \Big) \\ &= \sum_{f=1}^{n} \sum_{f'=1}^{n} \widehat{G}^*_{f-k_1 j} \cdot \widehat{X}^*_f \cdot \widehat{G}_{f'-k_1 j} \cdot \widehat{X}_{f'} \cdot \Big(\sum_{r \in [2k_1]} \omega_n^{a_r (f'-f)} \Big)\end{aligned}$$

where $(\cdot)^{\dagger}$ denotes the complex conjugate. Since $a_r = \frac{nr}{2k_1}$, the term $\sum_{r \in [2k_1]} \omega_n^{a_r (f'-f)}$ is equal to $2k_1$ if $f - f'$ is a multiple of $2k_1$ (including $f = f'$) and zero otherwise, yielding

$$\sum_{r \in [2k_1]} |\widehat{Z}_j^r|^2 = 2k_1 \cdot \sum_{f=1}^{n} \Bigg(|\widehat{G}_{f-k_1 j}|^2 \cdot |\widehat{X}_f|^2 + \sum_{\substack{j' \in [\frac{n}{2k_1}] \\ j' \ne 0}} \widehat{G}^*_{f-k_1 j} \cdot \widehat{X}^*_f \cdot \widehat{G}_{f-2k_1 j' - k_1 j} \cdot \widehat{X}_{f-2k_1 j'} \Bigg). \quad (46)$$

Without loss of generality, we can assume that $j = 0$; otherwise, we can simply consider a version of X shifted in frequency domain by $k_1 j$. Setting $j = 0$ in (46) and applying the triangle inequality, we obtain

$$\left|\sum_{r\in[2k_1]}|\widehat{Z}_0^r|^2-2k_1\cdot\sum_{f=1}^{n}|\widehat{G}_f|^2\cdot|\widehat{X}_f|^2\right|\le 2k_1\cdot\sum_{\substack{|j'|\le\frac{n}{2(2k_1)}\\ j'\neq 0}}\sum_{f=1}^{n}\left|\widehat{G}_f^*\cdot\widehat{X}_f^*\cdot\widehat{G}_{f-2k_1j'}\cdot\widehat{X}_{f-2k_1j'}\right|. \tag{47}$$

Bounding the right-hand side of (47)**:** We write

$$\begin{aligned}&\sum_{\substack{|j'|\le\frac{n}{2(2k_1)}\\ j'\neq 0}}\sum_{f=1}^{n}\left|\widehat{G}_f^*\cdot\widehat{X}_f^*\cdot\widehat{G}_{f-2k_1j'}\cdot\widehat{X}_{f-2k_1j'}\right|\\ &\quad=\sum_{\substack{|j'|\le\frac{n}{2(2k_1)}\\ j'\neq 0}}\sum_{f=1}^{n}\left(|\widehat{G}_f|^{1/2}\cdot|\widehat{G}_{f-2k_1j'}|^{1/2}\right)\cdot|\widehat{G}_f|^{1/2}\cdot|\widehat{X}_f^*|\cdot|\widehat{G}_{f-2k_1j'}|^{1/2}\cdot|\widehat{X}_{f-2k_1j'}|. \end{aligned}\tag{48}$$

In Lemma A.3 below, we show that

$$|\widehat{G}_f|^{1/2}\cdot|\widehat{G}_{f-2k_1j'}|^{1/2}\le\frac{(\frac{1}{2})^{F-1}}{|j'|^{(F-1)/2}}$$

for all $f\in[n]$ and all $|j'|\le\frac{n}{2(2k_1)}$ with $j'\neq 0$. Definition 2.2 ensures that $\left(\frac{1}{2}\right)^{F-1}\le\delta$, and substitution into (48) gives

$$\begin{aligned}&\sum_{\substack{|j'|\le\frac{n}{2(2k_1)}\\ j'\neq 0}}\sum_{f=1}^{n}\left|\widehat{G}_f^*\cdot\widehat{X}_f^*\cdot\widehat{G}_{f-2k_1j'}\cdot\widehat{X}_{f-2k_1j'}\right|\\ &\qquad\le\delta\cdot\sum_{\substack{|j'|\le\frac{n}{2(2k_1)}\\ j'\neq 0}}\frac{1}{|j'|^{(F-1)/2}}\sum_{f=1}^{n}|\widehat{G}_f|^{1/2}\cdot|\widehat{X}_f^*|\cdot|\widehat{G}_{f-2k_1j'}|^{1/2}\cdot|\widehat{X}_{f-2k_1j'}|. \end{aligned}\tag{49}$$

Next, we apply the Cauchy-Schwarz inequality to upper bound the inner summation over f above for any fixed $j'\in[\frac{n}{2k_1}]$, yielding

$$\begin{aligned}\sum_{f=1}^{n}|\widehat{G}_f|^{1/2}\cdot|\widehat{X}_f^*|\cdot|\widehat{G}_{f-2k_1j'}|^{1/2}\cdot|\widehat{X}_{f-2k_1j'}|&\le\sqrt{\sum_{f=1}^{n}|\widehat{G}_f|\cdot|\widehat{X}_f^*|^2}\cdot\sqrt{\sum_{f=1}^{n}|\widehat{G}_{f-2k_1j'}|\cdot|\widehat{X}_{f-2k_1j'}|^2}\\ &=\sum_{f=1}^{n}|\widehat{G}_f|\cdot|\widehat{X}_f^*|^2,\end{aligned}\tag{50}$$

where we used the fact that $\left\{|\widehat{G}_{f-2k_1j'}|\cdot|\widehat{X}_{f-2k_1j'}|^2\right\}_{f=1}^{n}$ is a permutation of $\left\{|\widehat{G}_f|\cdot|\widehat{X}_f^*|^2\right\}_{f=1}^{n}$.

Wrapping up: Substituting (50) into (49) gives

$$\begin{aligned}\sum_{\substack{|j'|\le\frac{n}{2(2k_1)}\\ j'\neq 0}}\sum_{f=1}^{n}\left|\widehat{G}_f^*\cdot\widehat{X}_f^*\cdot\widehat{G}_{f-2k_1j'}\cdot\widehat{X}_{f-2k_1j'}\right|\\ &\le \delta\cdot\sum_{\substack{|j'|\le\frac{n}{2(2k_1)}\\ j'\neq 0}}\frac{1}{|j'|^{(F-1)/2}}\sum_{f=1}^{n}|\widehat{G}_f|\cdot|\widehat{X}_f|^2\\ &\le 3\delta\sum_{f=1}^{n}|\widehat{G}_f|\cdot|\widehat{X}_f|^2,\end{aligned}$$

where the last inequality follows from the fact that $\sum_{\substack{|j'|\le\frac{n}{2(2k_1)}\\ j'\neq 0}}\frac{1}{|j'|^{(F-1)/2}}\le 2\sum_{j'=1}^{\infty}\frac{1}{|j'|^{(F-1)/2}}$, which is upper bounded by 3 for $F\ge 8$, a condition guaranteed by Definition 2.2. We therefore obtain the following bound from (47):

$$\left|\frac{1}{2k_1}\cdot\sum_{r\in[2k_1]}|\widehat{Z}_0^r|^2-\sum_{f=1}^{n}|\widehat{G}_f|^2\cdot|\widehat{X}_f|^2\right|\le 3\delta\sum_{f=1}^{n}|\widehat{G}_f|\cdot|\widehat{X}_f|^2.$$

The lemma follows by recalling that the choice $j=0$ was without loss of generality, with the general case amounting to replacing $\widehat{Z}_0$ by $\widehat{Z}_j$ and $\widehat{G}_f$ by $\widehat{G}_{f-k_1j}$. □

In the preceding proof, we made use of the following technical result bounding the product of the filter G evaluated at two locations separated by some multiple of $2k_1$.

Lemma A.3. (Additional filter property) *Given (n,k_1) and a parameter $F\ge 2$, if G is an $(n,\frac{n}{k_1},F)$-flat filter, then the following holds for all $f\in[n]$ and all $j\in\left[\frac{n}{k_1}\right]$ with $|j'|\le\frac{n}{2(2k_1)}$ and $j'\neq 0$:*

$$|\widehat{G}_f|^{1/2}\cdot|\widehat{G}_{f-2k_1j'}|^{1/2}\le\frac{(\frac{1}{2})^{F-1}}{|j'|^{(F-1)/2}}.$$

Proof. For clarity, let f_1 and f_2 denote the frequencies corresponding to f and $f-2k_1j'$ respectively, defined in the range $(-n/2,n/2]$ according to modulo-n arithmetic. By definition, f_1-f_2 is equal to $2k_1j'$ modulo-n, and since $|j'|\le\frac{n}{2(2k_1)}$, we have $|2k_1j'|\le\frac{n}{2}$. This immediately implies that the distance $\Delta=|f_1-f_2|$ according to *regular arithmetic* is lower bounded by the distance according to modulo-n arithmetic: $\Delta\ge 2k|j'|$.

Since f_1 and f_2 are at a distance Δ according to regular arithmetic, it must be the case that either $|f_1|\ge\frac{\Delta}{2}$ or $|f_2|\ge\frac{\Delta}{2}$. Moreover, since $j'\neq 0$, we have, from the above-established fact $\Delta\ge 2k_1|j'|$, that $\frac{\Delta}{2}\ge k_1$, and hence we can apply the third filter property in Definition 2.1 to conclude that $|G_{f_\nu}|\le\left(\frac{1}{4}\right)^{F-1}\left(\frac{2k_1}{\Delta}\right)^{F-1}$ for either $\nu=1$ or $\nu=2$. Substituting $\Delta\ge 2k_1|j'|$, upper bounding $G_{f_{\nu'}}\le 1$ (*cf.*, Definition 2.1) for the index $\nu'\in\{1,2\}$ differing from ν, and taking the square root, we obtain the desired result. □

We are now in a position to prove the claims of Lemma 2.3

Proof of first part of Lemma 2.3: Recall from Lemma A.2 that

$$\left|\frac{1}{2k_1}\sum_{r\in[2k_1]}|\widehat{Z}_j^r|^2-\sum_{f=1}^n|\widehat{G}_{f-k_1j}|^2\cdot|\widehat{X}_f|^2\right|\leq 3\delta\sum_{f=1}^n|\widehat{G}_{f-k_1j}|\cdot|\widehat{X}_f|^2. \tag{51}$$

We proceed by lower bounding $\sum_{f=1}^n|\widehat{G}_{f-k_1j}|^2\cdot|\widehat{X}_f|^2$ and upper bounding $\sum_{f=1}^n|\widehat{G}_{f-k_1j}|\cdot|\widehat{X}_f|^2$. Starting with the former, recalling that $I_j=\big((j-1/2)k_1,(j+1/2)k_1\big]\cap\mathbb{Z}$, we have

$$\begin{aligned}\sum_{f=1}^n|\widehat{G}_{f-k_1j}|^2\cdot|\widehat{X}_f|^2&\geq\sum_{f\in I_j}|\widehat{G}_{f-k_1j}|^2\cdot|\widehat{X}_f|^2\\&\geq\left(1-\left(\frac{1}{4}\right)^{F-1}\right)^2\|\widehat{X}_{I_j}\|_2^2\\&\geq(1-\delta)\|\widehat{X}_{I_j}\|_2^2,\end{aligned} \tag{52}$$

where the second line is by the second filter property in Definition 2.1, and the third line is by the choice of F in Definition 2.2.

Next, we upper bound $\sum_{f=1}^n|\widehat{G}_{f-k_1j}|\cdot|\widehat{X}_f|^2$ as folows:

$$\sum_{f=1}^n|\widehat{G}_{f-k_1j}|\cdot|\widehat{X}_f|^2\leq\sum_{f\in I_j\cup I_{j-1}\cup I_{j+1}}|\widehat{G}_{f-k_1j}|\cdot|\widehat{X}_f|^2+\sum_{f\in[n]\,:\,|f-k_1j|\geq\frac{3k_1}{2}}|\widehat{G}_{f-k_1j}|\cdot|\widehat{X}_f|^2. \tag{53}$$

By the third property in Definition 2.1, the filter decays as $|\widehat{G}_f|\leq(\frac{1}{4})^{F-1}(\frac{k_1}{|f|})^{F-1}$ for $|f|\geq k_1$, and therefore the second term in (53) is bounded by

$$\begin{aligned}\sum_{|f|\geq k_1j+\frac{3k_1}{2}}|\widehat{G}_{f-k_1j}|\cdot|\widehat{X}_f|^2&\leq\left(\frac{1}{4}\right)^{F-1}\cdot\sum_{j'\in[\frac{n}{k_1}]\,:\,|j'-j|\geq 2}\frac{\|\widehat{X}_{I_{j'}}\|_2^2}{(|j'-j|-1)^{F-1}}\\&\leq\left(\frac{1}{2}\right)^{F-1}\cdot\sum_{j'\in[\frac{n}{k_1}]\setminus\{j\}}\frac{\|\widehat{X}_{I_{j'}}\|_2^2}{|j'-j|^{F-1}}\\&\leq\delta\cdot\sum_{j'\in[\frac{n}{k_1}]\setminus\{j\}}\frac{\|\widehat{X}_{I_{j'}}\|_2^2}{|j'-j|^{F-1}},\end{aligned} \tag{54}$$

where the second line follows from $|j'-j|-1\geq\frac{|j'-j|}{2}$, and the third line follows since the choice of F in Definition 2.2 ensures that $\left(\frac{1}{2}\right)^{F-1}\leq\delta$. We bound the term $\sum_{f\in I_j\cup I_{j-1}\cup I_{j+1}}|\widehat{G}_{f-k_1j}|\cdot|\widehat{X}_f|^2$ in (53) using the first property in Definition 2.1, namely, $\widehat{G}_f\leq 1$:

$$\sum_{f\in I_j\cup I_{j-1}\cup I_{j+1}}|\widehat{G}_{f-k_1j}|\cdot|\widehat{X}_f|^2\leq\|\widehat{X}_{I_j\cup I_{j-1}\cup I_{j+1}}\|_2^2. \tag{55}$$

Hence, combining (53)–(55), we obtain

$$\sum_{f=1}^n|\widehat{G}_{f-k_1j}|\cdot|\widehat{X}_f|^2\leq\|\widehat{X}_{I_j\cup I_{j-1}\cup I_{j+1}}\|_2^2+\delta\cdot\sum_{j'\in[\frac{n}{k_1}]\setminus\{j\}}\frac{\|\widehat{X}_{I_{j'}}\|_2^2}{|j'-j|^{F-1}}. \tag{56}$$

The first claim of the lemma follows by combining (51), (52), and (56). □

Proof of second part of Lemma 2.3: By following the same steps as those used to handle (53), we obtain the following analog of (56) with $|\widehat{G}_f|^2$ in place of $|\widehat{G}_f|$:

$$\sum_{f=1}^{n} |\widehat{G}_{f-k_1 j}|^2 \cdot |\widehat{X}_f|^2 \le \|\widehat{X}_{I_j \cup I_{j-1} \cup I_{j+1}}\|_2^2 + \delta \cdot \sum_{j' \in [\frac{n}{k_1}] \setminus \{j\}} \frac{\|\widehat{X}_{I_{j'}}\|_2^2}{|j'-j|^{2(F-1)}}. \tag{57}$$

Combining (51), (56), and (57), we obtain

$$\begin{aligned} \frac{\sum_{r \in [2k_1]} |\widehat{Z}_j^r|^2}{2k_1} &\le \|\widehat{X}_{I_j \cup I_{j-1} \cup I_{j+1}}\|_2^2 + \delta \cdot \sum_{j' \in [\frac{n}{k_1}] \setminus \{j\}} \frac{\|\widehat{X}_{I_{j'}}\|_2^2}{|j'|^{2(F-1)}} \\ &\quad + 3\delta \cdot \left(\|\widehat{X}_{I_j \cup I_{j-1} \cup I_{j+1}}\|_2^2 + \delta \sum_{j' \in [\frac{n}{k_1}] \setminus \{j\}} \frac{\|\widehat{X}_{I_{j'}}\|_2^2}{|j'-j|^{F-1}} \right), \end{aligned}$$

and summing over $j \in [n]$ gives

$$\begin{aligned} \frac{1}{2k_1} \sum_{r \in [2k_1]} \|\widehat{Z}^r\|^2 &\le \sum_{j \in [n]} \left((1+3\delta) \|\widehat{X}_{I_j \cup I_{j-1} \cup I_{j+1}}\|_2^2 + (3\delta^2 + \delta) \sum_{j' \in [\frac{n}{k_1}] \setminus \{j\}} \frac{\|\widehat{X}_{I_{j'}}\|_2^2}{|j'-j|^{F-1}} \right) \\ &= 3(1+3\delta) \|\widehat{X}\|_2^2 + (3\delta^2 + \delta) \sum_{j \in [n]} \sum_{j' \in [\frac{n}{k_1}] \setminus \{j\}} \frac{\|\widehat{X}_{I_{j'}}\|_2^2}{|j'-j|^{F-1}} \end{aligned} \tag{58}$$

The double summation is upper bounded by $\sum_{j' \in [n]} \|\widehat{X}_{I_{j'}}\|_2^2 \cdot 2 \sum_{\Delta=1}^{\infty} \frac{1}{\Delta^{F-1}} = 2\|\widehat{X}\|_2^2 \cdot \sum_{\Delta=1}^{\infty} \frac{1}{\Delta^{F-1}}$, which in turn is upper bounded by $3\|\widehat{X}\|_2^2$ for $F \ge 4$, a condition guaranteed by Definition 2.2. We can therefore upper bound (58) by $\|\widehat{X}\|_2^2 (3(1+3\delta) + 3(3\delta^2 + \delta))$, which is further upper bounded by $6\|\widehat{X}\|_2^2$ for $\delta \le \frac{1}{20}$, as is assumed in Definition 2.2.

For the lower bound, we sum the first part of the lemma over all j, yielding

$$\sum_{r \in [2k_1]} \|\widehat{Z}^r\|^2 \ge \frac{\sum_{r \in [2k_1]} |\widehat{Z}_j^r|^2}{2k_1} \ge (1-\delta)\|\widehat{X}\|_2^2 - 3\delta \cdot \left(3\|\widehat{X}\|_2^2 + \delta \sum_{j \in [\frac{n}{k_1}]} \sum_{j' \in [\frac{n}{k_1}] \setminus \{j\}} \frac{\|\widehat{X}_{I_{j'}}\|_2^2}{|j'-j|^{F-1}} \right).$$

We showed above that the double summation is upper bounded by $3\|\widehat{X}\|_2^2$, yielding an lower bound of $(1 - \delta - 9\delta - 3\delta^2)\|\widehat{X}\|_2^2$. This is lower bounded by $(1-12\delta)\|\widehat{X}\|_2^2$ for $\delta \le \frac{1}{20}$. □

B Omitted Proofs from Section 3

B.1 Proof of Lemma 3.2

Note that for any $j \in \left[\frac{n}{k_1}\right]$, solving the first part of Lemma 2.3 for $\|\widehat{X}_{I_j}\|_2^2$ gives

$$\|\widehat{X}_{I_j}\|_2^2 \le \frac{1}{1-\delta} \left(\frac{1}{2k_1} \sum_{r \in [2k_1]} |\widehat{Z}_j^r|^2 + 3\delta \cdot \left(\|\widehat{X}_{I_j \cup I_{j-1} \cup I_{j+1}}\|_2^2 + \delta \sum_{j' \in [\frac{n}{k_1}] \setminus \{j\}} \frac{\|\widehat{X}_{I_{j'}}\|_2^2}{|j'-j|^{F-1}} \right) \right). \tag{59}$$

We will sum both sides over $j \in S^* \backslash \tilde{S}$; we proceed by analyzing the resulting terms.

Second term in (59) **summed over** $j \in S^*\backslash\tilde{S}$**:** We have

$$\begin{aligned}
&\sum_{j\in S^*\backslash\tilde{S}} 3\delta\cdot\bigg(\|\widehat{X}_{I_j\cup I_{j-1}\cup I_{j+1}}\|_2^2 + \delta\sum_{j'\in[\frac{n}{k_1}]\backslash\{j\}}\frac{\|\widehat{X}_{I_{j'}}\|_2^2}{|j'-j|^{F-1}}\bigg)\\
&\qquad\le 9\delta\|\widehat{X}\|_2^2 + 3\delta^2\sum_{j\in S^*\backslash\tilde{S}}\sum_{j'\in[\frac{n}{k_1}]\backslash\{j\}}\frac{\|\widehat{X}_{I_{j'}}\|_2^2}{|j'-j|^{F-1}}\\
&\qquad\le 9\delta\|\widehat{X}\|_2^2 + 10\delta^2\|\widehat{X}\|_2^2 \le 10\delta\|\widehat{X}\|_2^2, \qquad (60)
\end{aligned}$$

where the last line follows by expanding the double summation to all $j, j' \in \left[\frac{n}{k_1}\right]$ with $j \neq j'$, noting that $2\sum_{\Delta=1}^{\infty}\frac{1}{\Delta^{F-1}} \le 2.5$ for $F \ge 4$ (a condition guaranteed by Definition 2.2), and then applying the assumption $\delta \le \frac{1}{20}$.

First term in (59) **summed over** $j \in S^*\backslash\tilde{S}$**:** We first rewrite the sum of squares in terms of a weighted sum of fourth moments:

$$\begin{aligned}
\sum_{j\in S^*\backslash\tilde{S}}\frac{1}{2k_1}\sum_{r\in[2k_1]}|\widehat{Z}_j^r|^2 &= \frac{1}{2k_1}\sum_{r\in[2k_1]}\|\widehat{Z}^r_{S^*\backslash\tilde{S}}\|_2^2 = \sum_{r\in[2k_1]}\|\widehat{Z}^r\|_2\cdot\frac{\|\widehat{Z}^r_{S^*\backslash\tilde{S}}\|_2^2}{\|\widehat{Z}^r\|_2}\\
&\le \frac{1}{2k_1}\sqrt{\bigg(\sum_{r\in[2k_1]}\|\widehat{Z}^r\|_2^2\bigg)\bigg(\sum_{r\in[2k_1]}\frac{\|\widehat{Z}^r_{S^*\backslash\tilde{S}}\|_2^4}{\|\widehat{Z}^r\|_2^2}\bigg)}, \qquad (61)
\end{aligned}$$

by Cauchy-Schwarz applied to the length-$2k_1$ vectors containing entries $\|\widehat{Z}^r\|_2$ and $\frac{\|\widehat{Z}^r_{S^*\backslash\tilde{S}}\|_2^2}{\|\widehat{Z}^r\|_2}$.

The second summation inside the square root is upper bounded as

$$\begin{aligned}
\sum_{r\in[2k_1]}\frac{\|\widehat{Z}^r_{S^*\backslash\tilde{S}}\|_2^4}{\|\widehat{Z}^r\|_2^2} &\le \sum_{r\in[2k_1]}\|\widehat{Z}^r_{S^*}\|_2^2\cdot\frac{\|\widehat{Z}^r_{S^*\backslash\tilde{S}}\|_2^2}{\|\widehat{Z}^r\|_2^2}\\
&\le \sum_{r\in[2k_1]}\gamma^r\cdot\frac{\|\widehat{Z}^r_{S^*\backslash\tilde{S}}\|_2^2}{\|\widehat{Z}^r\|_2^2} + \sum_{r\in[2k_1]}\Big|\|\widehat{Z}^r_{S^*}\|_2^2-\gamma^r\Big|_+\cdot\frac{\|\widehat{Z}^r_{S^*\backslash\tilde{S}}\|_2^2}{\|\widehat{Z}^r\|_2^2}, \qquad (62)
\end{aligned}$$

where the first inequality follows since $\|\widehat{Z}^r_{S^*\backslash\tilde{S}}\|_2^2 \le \|\widehat{Z}^r_{S^*}\|_2^2$ and the second inequality uses $\|\widehat{Z}^r_{S^*}\|_2^2 \le \gamma^r + \big|\|\widehat{Z}^r_{S^*}\|_2^2-\gamma^r\big|_+$.

Now observe that by definition of $\tilde{S}$ (Definition 3.1), for every $j \notin \tilde{S}$, we have

$$\sum_{r\in[2k_1]}\Big(|\widehat{Z}_j^r|^2\cdot\frac{\gamma^r}{\|\widehat{Z}^r\|_2^2}\Big) \le \delta\cdot\frac{\sum_{r\in[2k_1]}\|\widehat{Z}^r\|_2^2}{k_0},$$

and summing both sides over all $j \in S^*\backslash\tilde{S}$ gives

$$\sum_{r\in[2k_1]}\gamma^r\cdot\frac{\|\widehat{Z}^r_{S^*\backslash\tilde{S}}\|_2^2}{\|\widehat{Z}^r\|_2^2} \le \frac{\delta|S^*\backslash\tilde{S}|}{k_0}\sum_{r\in[2k_1]}\|\widehat{Z}^r\|_2^2 \le 10\delta\sum_{r\in[2k_1]}\|\widehat{Z}^r\|_2^2,$$

since $|S^*| \le 10k_0$ by assumption. Applying this to the first term in (62), as well as $\frac{\|\widehat{Z}^r_{S^*\backslash\tilde{S}}\|_2^2}{\|\widehat{Z}^r\|_2^2} \le 1$ for the second term, we obtain

$$\sum_{r\in[2k_1]} \frac{\|\widehat{Z}^r_{S^*\setminus\tilde{S}}\|_2^4}{\|\widehat{Z}^r\|_2^2} \le 10\delta \sum_{r\in[2k_1]} \|\widehat{Z}^r\|_2^2 + \sum_{r\in[2k_1]} \Big| \|\widehat{Z}^r_{S^*}\|_2^2 - \gamma^r \Big|_+$$
$$\le 50\delta \sum_{r\in[2k_1]} \|\widehat{Z}^r\|_2^2, \tag{63}$$

where we have applied the assumption (*) of the lemma.

Finally, substituting (63) into (61) yields

$$\sum_{j\in S^*\setminus\tilde{S}} \frac{1}{2k_1} \sum_{r\in[2k_1]} |\widehat{Z}^r_j|^2 \le \frac{1}{2k_1}\sqrt{\Big(\sum_{r\in[2k_1]} \|\widehat{Z}^r\|_2^2\Big)\Big(\sum_{r\in[2k_1]} \frac{\|\widehat{Z}^r_{S^*\setminus\tilde{S}}\|_2^4}{\|\widehat{Z}^r\|_2^2}\Big)}$$
$$\le \frac{1}{2k_1}\sqrt{\Big(\sum_{r\in[2k_1]} \|\widehat{Z}^r\|_2^2\Big)\Big(50\delta \sum_{r\in[2k_1]} \|\widehat{Z}^r\|_2^2\Big)} \qquad \text{(by (63))}$$
$$\le \frac{\sqrt{50\delta}}{2k_1} \sum_{r\in[2k_1]} \|\widehat{Z}^r\|_2^2 \le 43\sqrt{\delta}\|\widehat{X}\|_2^2, \tag{64}$$

where the last inequality uses the fact that $\frac{\sum_{r\in[2k_1]}\|\widehat{Z}^r\|_2^2}{2k_1} \le 6\|X\|_2^2$ by the second part of Lemma 2.3. The proof is concluded by substituting (60) and (64) into (59), and using the assumption $\delta \le \frac{1}{20}$ to deduce that $\frac{1}{1-\delta}\big(43\sqrt{\delta} + 10\delta\big) \le 100\sqrt{\delta}$.

C Omitted Proofs from Section 4

C.1 Proof of Lemma 4.2

The (exact) Fourier transform of U, denoted by $\widehat{U}^*$, is given by

$$\widehat{U}^*_j = \frac{1}{B}\sum_{b\in[B]} U_b \omega_B^{-bj}$$
$$= \frac{1}{n}\sum_{b\in[B]}\sum_{i'\in[\frac{n}{B}]} X_{\sigma(\Delta+b+B\cdot i')} G_{b+B\cdot i'} \omega_B^{-bj}$$
$$= \frac{1}{n}\sum_{i\in[n]} X_{\sigma(\Delta+i)} G_i \omega_n^{-ijn/B}, \tag{65}$$

where we used the fact that $\omega_B^{(\cdot)}$ is periodic with period B, and then applied $\omega_B = \omega_n^{n/B}$. We see that (65) is the Fourier transform of the signal $\{X_{\sigma(\Delta+i)}G_i\}_{i\in[n]}$ evaluated at frequency jn/B, and hence, since multiplication and convolution are dual under the Fourier transform, we obtain

$$\widehat{U}^*_j = (\widehat{Y} \star \widehat{G})_{jn/B}, \tag{66}$$

where $Y_i = X_{\sigma(\Delta+i)}$. Now, by standard Fourier transform properties, we have $\widehat{Y}_f = \widehat{X}_{\sigma^{-1}f}\omega_n^{\Delta f}$, and substitution into (66) gives

$$\begin{aligned}\widehat{U}^*_j &= \sum_{f\in[n]} \widehat{X}_{\sigma^{-1}f}\widehat{G}_{j\frac{n}{B}-f}\omega_n^{\Delta f}\\ &= \sum_{f\in[n]} \widehat{X}_f \widehat{G}_{\sigma f-\frac{n}{B}j}\omega_n^{\sigma\Delta f},\end{aligned}$$

where we have used the assumed symmetry of G about zero.

C.2 Proof of Lemma 4.3

For brevity, let $\Psi = \sum_{f'\neq f}|\widehat{X}_{f'}|^2\mathbb{E}_\pi\big[|G_{o_f(f')}|^2\big]$ denote the left-hand side of (14). Following the approach of [39, Lemma 3.3], we define the intervals $\mathcal{F}_t = \big(\pi(f)-\frac{n}{B}2^t, \pi(f)+\frac{n}{B}2^t\big]$ for $t=1,\dots,\log_2 b$, and write

$$\begin{aligned}\Psi &\le \sum_{f'\neq f}|\widehat{X}_{f'}|^2\sum_{t=1}^{\log_2 B}\mathbb{P}[\pi(f')\in\mathcal{F}_t\backslash\mathcal{F}_{t-1}]\max_{f''\,:\,\pi(f'')\in\mathcal{F}_t\backslash\mathcal{F}_{t-1}}|\widehat{G}_{o_f(f'')}|^2\\ &\le \frac{4}{B}\sum_{f'\neq f}|\widehat{X}_{f'}|^2\bigg(2+\sum_{t=2}^{\log_2 B}2^t\max_{f''\,:\,\pi(f'')\in\mathcal{F}_t\backslash\mathcal{F}_{t-1}}|\widehat{G}_{o_f(f'')}|^2\bigg),\end{aligned}\tag{67}$$

where the second line follows by (i) upper bounding $\mathbb{P}[\pi(f')\in\mathcal{F}_t\backslash\mathcal{F}_{t-1}]\le\mathbb{P}[\pi(f')\in\mathcal{F}_t]$ and applying the approximate pairwise independence property (*cf.*, Definition 4.1); (ii) using the fact that there are at most $\frac{n}{B}\cdot 2^{t+1}$ integers within $\mathcal{F}_t$, and applying $|\widehat{G}_f|\le 1$ for the case $t=1$.

To handle the term containing $|\widehat{G}_{o_f(f'')}|^2$, we use the triangle inequality to write

$$\begin{aligned}|o_f(f'')| &\ge |\pi(f)-\pi(f'')| - \Big|\pi(f)-\frac{n}{b}\text{round}\Big(\pi(f)\frac{B}{n}\Big)\Big|\\ &\ge |\pi(f)-\pi(f'')| - \frac{n}{B}.\end{aligned}$$

For any f'' with $\pi(f'')\notin\mathcal{F}_{t-1}$, we have $|\pi(f)-\pi(f'')|\ge\frac{n}{B}2^{t-1}$, and hence $|o_f(f'')|\ge\frac{n}{B}(2^{t-1}-1)$. As a result, for $t\ge 2$, the third property in Definition 2.1 gives

$$\widehat{G}_{o_f(f'')}\le\Big(\frac{1}{4}\Big)^{F-1}\Big(\frac{1}{2^{t-1}-1}\Big)^{F-1}\le\Big(\frac{1}{4}\Big)^{F-1}\Big(\frac{1}{2^{t-2}}\Big)^{F-1}=\Big(\frac{1}{2^t}\Big)^{F-1},$$

and hence

$$\sum_{t=2}^{\log_2 B}2^t\max_{f''\,:\,\pi(f'')\in\mathcal{F}_t\backslash\mathcal{F}_{t-1}}|\widehat{G}_{o_f(f'')}|^2\le\sum_{t=2}^{\infty}\Big(\frac{1}{2^t}\Big)^{2F-1}.$$

This sum is less than $\frac{1}{2}$ for all $F\ge 2$, and hence substitution into (67) gives $\Psi\le\frac{10}{B}\|\widehat{X}\|_2^2$, as desired.

C.3 Proof of Lemma 4.5

We use techniques resembling those used for a (k_1,ϵ)-downsampling in Section 2, but with the notable difference of using a more rapidly-decaying filter with bounded support in *frequency* domain.

Choice of filter: We let $G\in\mathbb{R}^n$ be the filter used in [25] (as opposed to that used in Definition 2.1), satisfying the following:

- There exists an ideal filter G' satisfying $G'_f \in [0,1]$ for all f, and
$$G'_f = \begin{cases} 1 & |f| \leq \frac{n}{2k_1} \\ 0 & |f| \geq \frac{n}{k_1}, \end{cases} \tag{68}$$
such that $\|G - G'\|_2 \leq n^{-c}$;
- $\widehat{G}$ is supported on a window of length $O(ck_1 \log n)$ centered at zero;
- Each entry of $\widehat{G}$ can be computed in time $O(1)$.

Intuition behind the proof. Before giving the details, we provide the intuition for the proof. Recall that our goal is to compute X^r_j for $|j| \leq k_0/2$, for all $r \in [2k_1]$. To do this, we first note that X^0_j, for $|j| \leq k_0/2$ (i.e., for only one value of r, namely 0), can be computed via a reduction to standard semi-equispaced FFT (Lemma 4.4) on an input signal of length $2n/k_1$. To achieve this, consider the signal $X \cdot G$ aliased to length $2n/k_1$, which is close to X on all points j such that $|j| \leq n/(2k_1)$. In order to compute X^0_j for $|j| \leq k_0/2$, it essentially suffices (modulo boundary issues; see below) to calculate $(X \cdot G)_j$ for $|j| \leq k_0/2$. We show below that this can be achieved using Lemma 4.4, because multiplication followed by aliasing are dual to convolution and subsampling: the input (k_0, k_1)-block sparse signal of length n can be naturally mapped to an $O(k_0 \log n)$-sparse signal in a reduced space with $\approx n/k_1$ points, in which standard techniques (Lemma 4.4) can be applied.

This intuition only shows how to compute the values of X^r_j for $r = 0$ and $|j| \leq k_0/2$, but we need the values for all $r \in [2k_1]$. As we show below, the regular structure of the set of shifts that we are interested in allows us use the standard FFT on a suitably defined set of length-$2k_1$ signals, without increasing the runtime by a k_1 factor. It is interesting to note that our runtime is $O(\log n)$ worse than the runtime of Lemma 4.4 due to the two-level nature of our scheme; this is for reasons similar to the $\log^d n$ scaling of runtime of high-dimensional semi-equispaced FFT (e.g. [22, 41]).

We now give the formal proof of the lemma.

Computing a convolved signal: Here we show that we can efficiently compute the values $\widehat{Y}^r_j = (\widehat{X}^r \star \widehat{G})_{\frac{k_1}{2}j}$ at all $j \in \left[\frac{2n}{k_1}\right]$ where it is non-zero, for all values of $r \in [2k_1]$. We will later show that applying Lemma 4.4 to these signals (as a function of j) gives accurate estimates of the desired values of X.

Note that in the definition of $\widehat{Y}^r_j$, each non-zero block is convolved with a filter of support $O(ck_1 \log n)$, and so contributes to at most $O(c \log n)$ values of j. Since there are k_0 non-zero blocks, there are $O(ck_0 \log n)$ values of j for which the result is non-zero.

The procedure is as follows:

1. For all j such that $\widehat{Y}^r_j$ may be non-zero ($O(ck_0 \log n)$ in total), compute $\widetilde{Y}^b_j = \frac{k_1}{2} \sum_{l=1}^{\frac{n}{2k_1}} \widehat{X}_{b+2k_1 l} \widehat{G}_{\frac{k_1}{2}j-(b+2k_1 l)}$ for $b \in [2k_1]$. That is, alias the signal $\{\widehat{X}_f \widehat{G}_{\frac{k_1}{2}j-f}\}$ down to length $2k_1$, and normalize by $\frac{2}{k_1}$ (for later convenience). Since $\widehat{G}$ is supported on an interval of length $O(ck_1 \log n)$, this can be done in time $O(c \log n)$ per entry, for a total of $O(ck_1 \log n)$ per j value, or $O(c^2 k_0 k_1 \log^2 n)$ overall.

2. Compute the length-$2k_1$ inverse FFT of $\widetilde{Y}_j = (\widetilde{Y}_j^1, \dots, \widetilde{Y}_j^{2k_1})$ to obtain $\widehat{Y}_j \in \mathbb{C}^{2k_1}$. This can be done in time $O(k_1 \log(1+k_1))$ per j value, or $O(k_0 k_1 \log(1+k_1) \log n)$ overall.

We now show that $\widehat{Y}_j^r = \frac{k_1}{2}(X^r \star G)_{\frac{k_1}{2}j}$ for $r = 1, \dots, 2k_1$. By the definition of the inverse Fourier transform, we have

$$\begin{aligned}
\widehat{Y}_j^r &= \sum_{b=1}^{2k_1} \widetilde{Y}_j^b \omega_{2k_1}^{rb} \\
&= \frac{k_1}{2} \sum_{b=1}^{2k_1} \sum_{l=1}^{\frac{n}{2k_1}} \widehat{X}_{b+2k_1 l} \widehat{G}_{\frac{k_1}{2}j-(b+2k_1 l)} \omega_{2k_1}^{rb} \\
&= \frac{k_1}{2} \sum_{f=1}^{n} \widehat{X}_f \widehat{G}_{\frac{k_1}{2}j-f} \omega_{2k_1}^{rf} \\
&= \frac{k_1}{2} \sum_{f=1}^{n} \widehat{X}_f \widehat{G}_{\frac{k_1}{2}j-f} \omega_n^{rf \cdot \frac{n}{2k_1}} \\
&= \frac{k_1}{2} \sum_{f=1}^{n} \widehat{X}_f^r \widehat{G}_{\frac{k_1}{2}j-f} \quad \text{(since } X_{(\cdot)}^r = X_{(\cdot)+\frac{nr}{2k_1}} \text{ by Definition 2.2)},
\end{aligned}$$

where the second line is by the definition of $\widetilde{Y}^b$, the third by the periodicity of ω_{2k_1}, and the fifth since translation and phase shifting are dual under the Fourier transform. Hence, $\widehat{Y}_j^r = \frac{k_1}{2}(\widehat{X}^r \star \widehat{G})_{\frac{k_1}{2}j}$.

Applying the standard semi-equispaced FFT: For $r \in [2k_1]$, define $\widehat{Y}^r = (\widehat{Y}_1^r, \dots, \widehat{Y}_{n/k_1}^r)$. We have already established that the support of each $\widehat{Y}^r$ is a subset of a set having size at most $k' = O(ck_0 \log n)$. We can therefore apply Lemma 4.4 with $\zeta = n^{-(c+1)}$ to conclude that we can evaluate Y_j^r for $|j| \le \frac{k'}{2}$ satisfying

$$|Y_j^r - Y_j^{*r}| \le n^{-(c+1)} \|Y^r\|_2, \tag{69}$$

where Y^{*r} is the exact inverse Fourier transform of $\widehat{Y}^r$. Moreover, this can be done in time $O\big(k' \log \frac{n/k_1}{n^{-(c+1)}}\big) = O(c^2 k_0 \log^2 n)$ per r value, or $O(c^2 k_0 k_1 \log^2 n)$ overall.

Proof of first part of lemma: It remains to show that the above procedure produces estimates of the desired X values of the form (15).

Recall that $\widehat{Y}_j^r = \frac{k_1}{2}(\widehat{X}^r \star \widehat{G})_{\frac{k_1}{2}j}$. By the convolution theorem and the fact that subsampling and aliasing are dual (e.g., see Appendix C.1), the inverse Fourier transform of $\widehat{Y}^r$ satisfies the following when $|j| \le \frac{n}{k_1}$:

$$\begin{aligned}
Y_j^r &= \sum_{i \in [\frac{k_1}{2}]} (G \cdot X^r)_{j+\frac{2n}{k_1}i} \\
&= G_j X_j^r + \sum_{i \in [\frac{k_1}{2}],\, i \neq 0} (G \cdot X^r)_{j+\frac{2n}{k_1}i} \\
&= \left(G'_j X_j^r + \sum_{i \in [\frac{k_1}{2}],\, i \neq 0} (G' \cdot X^r)_{j+\frac{2n}{k_1}i} \right) \pm \|G - G'\|_2 \|X\|_2 \\
&= X_j^r \pm n^{-c} \|X\|_2,
\end{aligned} \tag{70}$$

where the last line follows from the definition of G' in (68) and the assumption $\|G - G'\|_2 \le n^{-c}$.

Combining (69) and (70) and using the triangle inequality, we obtain

$$|Y^r_j - X^r_j| \le n^{-(c+1)} \|Y^r\|_2 + n^{-c}\|X\|_2.$$

Since we have already shown that we can efficiently compute Y^r_j for $|j| \le \frac{k'}{2}$ with $k' = O(ck_0 \log n)$, it only remains to show that $\|Y^r\|_2 \le n\|X\|^2$. To do this, we use the first line of (70) to write

$$\begin{aligned} |Y^r_j| &\le \sum_{i \in [\frac{k_1}{2}]} |G_{j+\frac{2n}{k_1}i}| \cdot |X^r_{j+\frac{2n}{k_1}i}| \\ &\le 2 \sum_{i \in [\frac{k_1}{2}]} |X^r_{j+\frac{2n}{k_1}i}| \\ &\le \sqrt{2k_1 \sum_{i \in [\frac{k_1}{2}]} |X^r_{j+\frac{2n}{k_1}i}|^2}, \end{aligned} \tag{71}$$

where the first line is the triangle inequality, the second line follows since the first filter assumption above ensures that $|G_j| \le 2$ for all j, and the third line follows since the squared ℓ_1-norm is upper bounded by the squared ℓ_2-norm times the vector length.

Squaring both sides of (71) and summing over all j gives $\|Y^r\|_2^2 \le 2k_1\|X\|^2 \le n^2\|X\|^2$ (under the trivial assumption $n \ge 2$), thus completing the proof.

Proof of second part of lemma: In the proof of the first part, we applied Lemma 4.4 to signals of length $\frac{2n}{k_1}$. It follows directly from the arguments in [25, Cor. 12.2] that since we can approximate the entries of X^r_j for all $|j| \le \frac{k_0}{2}$, we can do the same for all j equaling $\sigma j' + b$ modulo-$\frac{2n}{k_1}$ for some $|j'| \le \frac{k_0}{2}$. Specifically, this follows since the multiplication by σ and shift by b simply amounts to a phase shift and a linear change of variables $f \to \sigma^{-1} f$ in frequency domain, both of which can be done in constant time.

However, the second part of the lemma regards indices modulo-$\frac{n}{k_1}$, as opposed to modulo-$\frac{2n}{k_1}$. To handle the former, we note that for any integer a, we either have $a \bmod \frac{n}{k_1} = a \bmod \frac{2n}{k_1}$ or $a \bmod \frac{n}{k_1} = \left(a + \frac{n}{k_1}\right) \bmod \frac{2n}{k_1}$. Hence, we obtain the desired result by simply performing two calls to the first part, one with a universal shift of $\frac{n}{k_1}$.

C.4 Proof of Lemma 4.6

C.4.1 First Part

Since U_X is computed according to X itself in Algorithm 4, we only need to compute the error in U_χ.

In the definition of hashing in Definition 4.2, since G has support $O(FB)$, we see that the values of X used correspond to a permutation of an interval having length $k' = O(FB)$. We can therefore apply the second part of Lemma 4.4 with sparsity k' and parameter $\zeta = n^{-c'}$ for some $c' > 0$, ensuring an ℓ_∞-guarantee of $n^{-c'}\|\chi\|_2$ for the signal χ.

Since $\widehat{U}$ is computed from these values using (13) followed by the FFT, we readily obtain via the relation $\|v\|_\infty \le \|v\|_2 \le \sqrt{m}\|v\|_\infty$ (for $v \in \mathbb{C}^m$) and Parseval's theorem that $\widehat{U}$ has an ℓ_∞-guarantee of $n^{-(c'-O(1))}\|\chi\|_2$, which can be made to equal $n^{-c}\|\widehat{\chi}\|_2$ by choosing $c' = c + O(1)$.

Sample complexity and runtime: The only operation that consumes samples from the signal X is the hashing operation applied to X. From the definition of hashing in Definition 4.2, and the fact that the filter G has support $O(FB)$, we find that the sample complexity is also $O(FB)$.

The runtime is dominated by the application of the semi-equispaced FFT, which is $O(cF(\|\widehat{\chi}\|_0 + B)\log n)$ by Lemma 4.4. In particular, this dominates the $O(B \log B)$ time to perform the FFT in Algorithm 4, and the hashing operation, whose time complexity is the same as the sample complexity.

C.4.2 Second Part

Recall the definition of a (k_1, δ)-downsampling of a signal X from (2):

$$Z_j^r = \frac{1}{k_1} \sum_{i \in [k_1]} (G \cdot X^r)_{j + \frac{n}{k_1} \cdot i}, \; j \in \left[\frac{n}{k_1}\right].$$

For each $r \in [2k_1]$. Now in order to compute $\left(\frac{n}{k_1}, B^r, G^r, \sigma, \Delta\right)$-hashing of $\widehat{Z}^r$, by Definition 4.2, it is needed to have the samples of Z_j^r at points $j = \sigma(j' + \Delta) \mod \frac{n}{k_1}$ for $|j'| \leq FB^r$ (because G^r is supported on $[-FB^r, +FB^r]$). Note that FB^r is further upper bounded by $O(FB_{\max})$, as a result, in the second part of Lemma 4.5, it suffices to set the sparsity level to $O(FB_{\max} + k_0)$; here we add k_0 in accordance with Remark 4.2 and the fact that $\widehat{\chi}$ is (k_0, k_1)-block sparse. The reason that this works is that $\widehat{Z}^r$ has length $\frac{n}{k_1}$ and one sample of Z_j^r can be computed from $X^r_{j+i\frac{n}{k_1}} = X_j^{r+2i}$ for $|i| \leq F$ as per Definition 2.2 and the fact that the filter G is supported on $[-F\frac{n}{k_1}, +F\frac{n}{k_1}]$. Therefore all we need is $X_{j'}^{r'}$ for each $r' \in [2k_1]$ and for all $j' = \sigma(j + \Delta) \mod \frac{n}{k_1}$ with $|j| \leq F_{\max} B_{\max}$.

Applying the second part of Lemma 4.5 with sparsity $O(F_{\max} B_{\max} + k_0)$ and parameter $\zeta = n^{-c'}$ for some $c' > 0$, ensuring an ℓ_∞-guarantee of $2n^{-c'}\|\chi\|_2$ on the computed values of χ. By an analogous argument to the first case, this implies an ℓ_∞-guarantee of $n^{-c}\|\chi\|_2$ on the FFT $\widehat{U}^r$ of the hashing of Z_χ^r, with $c = c' + O(1)$.

Sample complexity and runtime: We take $O(FB^r)$ samples of the r-th downsampled signal each time we do the hashing, separately for each $r \in [2k_1]$. By Lemma 2.2, accessing a single sample of Z_X^r costs us $O(\log \frac{1}{\delta})$ samples of X. Hence, the sample complexity is $O\big(F \sum_{r \in [2k_1]} B^r \log \frac{1}{\delta}\big)$.

We now turn to the runtime. By Lemma 4.5, the call to SemiEquiInverseBlockFFT with $O(FB_{\max} + k_0)$ in place of k_0 takes time $O\big(c^2(FB_{\max} + k_0)k_1 \log^2 n\big)$. The hashing operation's runtime matches its sample complexity, and since we have assumed $\delta \geq \frac{1}{n}$, its contribution is dominated by the preceding term.

C.5 Proof of Lemma 4.7

First part of lemma: We start with the following upper bound on the expression inside the expectation:

$$\left| \|\widehat{Y}_S\|_2^2 - \|\widehat{U}^*\|_2^2 \right|_+ \leq \left| \|\widehat{Y}_S\|_2^2 - \|\widehat{U}^*_{h(S \setminus S_{coll})}\|_2^2 \right|_+$$

where $h(S) = \{h(j) : j \in S\}$ with $h(j) = \text{round}\big(\pi(j)\frac{B}{m}\big)$, denoting the bucket into which element j hashes. We define S_{coll} to be a subset of S containing the elements that collide with each other, i.e.,

$S_{\mathrm{coll}} = \{j \in S \,|\, h(j) \cap h(S\backslash\{j\}) \neq \emptyset\}$, yielding

$$\begin{aligned}
\Big|\|\widehat{Y}_S\|_2^2 - \|\widehat{U}^*\|_2^2\Big|_+ &\leq \Big|\sum_{j\in S} |\widehat{Y}_j|^2 - \sum_{b\in h(S\backslash S_{\mathrm{coll}})} |\widehat{U}^*_b|^2\Big|_+ \\
&= \Big|\sum_{j\in S_{\mathrm{coll}}} |\widehat{Y}_j|^2 + \sum_{j\in S\backslash S_{\mathrm{coll}}} \Big(|\widehat{Y}_j|^2 - |\widehat{U}^*_{h(j)}|^2\Big)\Big|_+ \\
&\leq \sum_{j\in S_{\mathrm{coll}}} |\widehat{Y}_j|^2 + \sum_{j\in S} \Big||\widehat{Y}_j|^2 - |\widehat{U}^*_{h(j)}|^2\Big|_+, \qquad (72)
\end{aligned}$$

where the final line follows from the inequality $[a+b]_+ \leq |a| + [b]_+$.

Bounding the first term in (72): We start by evaluating the expected value of the term corresponding to S_{coll} over the random permutation π:

$$\begin{aligned}
\mathbb{E}_\pi\Big[\sum_{j\in S_{\mathrm{coll}}} |\widehat{Y}_j|^2\Big] &\leq \mathbb{E}_\pi\Big[\sum_{j\in S} |\widehat{Y}_j|^2 \mathcal{I}[j \in S_{\mathrm{coll}}]\Big] \\
&\leq \sum_{j\in S} |\widehat{Y}_j|^2 \sum_{j'\in S\backslash\{j\}} \Pr[h(j) = h(j')] \\
&\leq \sum_{j\in S}\sum_{j'\in S} |\widehat{Y}_j|^2 \frac{4}{B} \\
&= \frac{4|S|}{B} \sum_{j\in S} |\widehat{Y}_j|^2, \qquad (73)
\end{aligned}$$

where the second line follows from the union bound, and the third line follows since π is approximately pairwise independent as per Definition 4.1.

Bounding the second term in (72): We apply Lemma 4.2 to obtain $\widehat{U}^*_{h(j)} = \sum_{j'\in[m]} \widehat{Y}_{j'} \widehat{H}_{o_j(j')} \omega_m^{\sigma\Delta j'}$ with $o_j(j') = \pi(j') - h(j)\frac{m}{B}$. We write this as $\widehat{U}^*_{h(j)} = \widehat{Y}_j \widehat{H}_{o_j(j)} \omega_m^{\sigma\Delta j} + \mathrm{err}_j$ with $\mathrm{err}_j := \sum_{j'\in[m]\backslash\{j\}} \widehat{Y}_{j'} \widehat{H}_{o_j(j')} \omega_m^{\sigma\Delta j'}$, yielding

$$\begin{aligned}
\sum_{j\in S} \Big||\widehat{Y}_j|^2 - |\widehat{U}^*_{h(j)}|^2\Big|_+ &\leq \sum_{j\in S} \Big||\widehat{Y}_j|^2 - |\widehat{Y}_j \widehat{H}_{o_j(j)} \omega_m^{\sigma\Delta j} + \mathrm{err}_j|^2\Big| \\
&\leq \sum_{j\in S} \Big(\big||\widehat{Y}_j|^2 - |\widehat{Y}_j H_{o_j(j)}|^2\big| + |\mathrm{err}_j|^2 + 2|\mathrm{err}_j| \cdot |\widehat{Y}_j \widehat{H}_{o_j(j)}|\Big) \qquad (74)
\end{aligned}$$

by $|\xi|_+ \leq |\xi|$ and the triangle inequality. We have by definition that $|o_j(j)| \leq \frac{m}{2B}$, and hence item 2 in Definition 2.1 yields $H_{o_j(j)} \geq 1 - \big(\frac{1}{4}\big)^{F'-1}$, which in turn implies $H^2_{o_j(j)} \geq 1 - 2\big(\frac{1}{4}\big)^{F'-1}$. Combining this with $H_f \leq 1$ from item 1 in Definition 2.1, we can weaken (74) to

$$\sum_{j\in S} \Big||\widehat{Y}_j|^2 - |\widehat{U}^*_{h(j)}|^2\Big|_+ \leq \sum_{j\in S} \Big(2|\mathrm{err}_j| \cdot |\widehat{Y}_j| + |\mathrm{err}_j|^2 + 2\Big(\frac{1}{4}\Big)^{F'-1} |\widehat{Y}_j|^2\Big). \qquad (75)$$

We proceed by bounding the expected value of $|\mathrm{err}_j|^2$. We first take the expectation over Δ, using Parseval's theorem to write

$$\mathbb{E}_\Delta[|\mathrm{err}_j|^2] = \sum_{j'\in[m]\backslash\{j\}} |\widehat{Y}_{j'}|^2 |\widehat{H}_{o_j(j')}|^2.$$

Taking the expectation over π, we obtain

$$\begin{aligned}\mathbb{E}_{\Delta,\pi}[|\mathrm{err}_j|^2] &= \mathbb{E}_\pi\Big[\sum_{j'\in[m]\setminus\{j\}} |\widehat{Y}_{j'}|^2|\widehat{H}_{o_j(j')}|^2\Big] \\ &= \sum_{j'\in[m]\setminus\{j\}} |\widehat{Y}_{j'}|^2\mathbb{E}_\pi[|\widehat{H}_{o_j(j')}|^2 \\ &\le \frac{10}{B}\sum_{j'\in[m]\setminus\{j\}} |\widehat{Y}_{j'}|^2 \le \frac{10}{B}\|\widehat{Y}\|_2^2.\end{aligned}$$

where the final line follows from Lemma 4.3.

Substituting into (75), and using Jensen's inequality to write $\mathbb{E}[|\mathrm{err}_j|] \le \sqrt{\mathbb{E}[|\mathrm{err}_j|^2]}$, we obtain

$$\begin{aligned}\mathbb{E}_{\Delta,\pi}\Big[\sum_{j\in S}\Big||\widehat{Y}_j|^2 - |\widehat{U}^*_{h(j)}|^2\Big|_+\Big] &\le 2\sum_{j\in S}\sqrt{\frac{10}{B}\|\widehat{Y}\|_2^2}\cdot|\widehat{Y}_j| + \frac{10}{B}\sum_{j\in S}\|\widehat{Y}\|_2^2 + 2\Big(\frac{1}{4}\Big)^{F'-1}\sum_{j\in S}|\widehat{Y}_j|^2 \\ &\le 10\sqrt{\frac{|S|}{B}}\|\widehat{Y}\|_2^2 + \bigg(10\frac{|S|}{B} + 2\delta^2\bigg)\|\widehat{Y}\|_2^2, \qquad (76)\end{aligned}$$

where the second line follows from the fact that $\|v\|_1 \le \sqrt{|S|}\|v\|_2$ for any $v \in \mathbb{C}^{|S|}$, as well as $\left(\frac{1}{4}\right)^{F'-1} \le \delta$ by the choice of F'. The claim follows by substituting (73) and (76) into (72).

Second part of lemma: By the definition of $\widehat{U}^*$ (*cf.*, Definition 4.2), we have

$$\begin{aligned}\mathbb{E}_\Delta\Big[\|\widehat{U}^*\|_2^2\Big] &= \mathbb{E}_\Delta\Big[\sum_{b\in[B]}\Big|\sum_{j\in[m]}\widehat{Y}_j\widehat{H}_{\pi(j)-b\frac{m}{B}}\omega_m^{\Delta j}\Big|^2\Big] \\ &= \sum_{b\in[B]}\sum_{j\in[m]}|\widehat{Y}_j|^2|\widehat{H}_{\pi(j)-b\frac{m}{B}}|^2\end{aligned}$$

by Parseval. Taking the expectation with respect to π, we obtain

$$\begin{aligned}\mathbb{E}_{\Delta,\pi}\Big[\|\widehat{U}^*\|_2^2\Big] &= \sum_{b\in[B]}\mathbb{E}_\pi\Big[\sum_{j\in[m]}|\widehat{Y}_j|^2|\widehat{H}_{\pi(j)-b\frac{m}{B}}|^2\Big] \\ &= \sum_{b\in[B]}\sum_{j\in[m]}|\widehat{Y}_j|^2\mathbb{E}_\pi\Big[|\widehat{H}_{\pi(j)-b\frac{m}{B}}|^2\Big] \\ &\le \sum_{b\in[B]}\frac{3}{B}\sum_{j\in[m]}|\widehat{Y}_j|^2 = 3\|\widehat{Y}\|_2^2.\end{aligned}$$

where the final line follows by noting that $\pi(j) - b\frac{m}{B}$ is uniformly distributed over $[m]$, and applying the second part of Lemma 2.1.

D Omitted Proofs from Section 5

D.1 Proof of Lemma 5.3

Note on $\frac{1}{\mathrm{poly}(n)}$ assumptions in lemmas: Throughout the proof, we apply Lemmas 3.5, 5.1, and 5.2. The first of these assumes that $\widehat{\chi}_0$ uniformly distributed over an arbitrarily length-$\Omega\big(\frac{\|\widehat{\chi}\|^2}{\mathrm{poly}(n)}\big)$ interval, and the latter two use the assumption $\|\widehat{X} - \widehat{\chi}\|_2 \ge \frac{1}{\mathrm{poly}(n)}\|\widehat{\chi}\|_2^2$.

We argue that these assumptions are trivial and can be ignored. To see this, we apply a minor technical modification to the algorithm as follows. Suppose the implied exponent to the $\mathrm{poly}(n)$ notation is c'. By adding a noise term to $\widehat{\chi}_0$ on each iteration uniform in $[-n^{-c'+10}\|\widehat{\chi}\|^2, n^{c'+10}\|\widehat{\chi}\|^2]$, we immediately satisfy the first assumption above, and we also find that the probability of $\|\widehat{X}-\widehat{\chi}\|_2 < \frac{1}{\mathrm{poly}(n)}\|\widehat{\chi}\|_2^2$ is at most $O(n^{-10})$, and the additional error in the estimate is $O(n^{c'+10}\|\widehat{\chi}\|^2)$. Since we only do $O(\log \mathrm{SNR}') = O(n)$ iterations (by the assumption $\mathrm{SNR}' = O(\mathrm{poly}(n))$), this does not affect the result because the accumulated noise added to $\widehat{\chi}_0$ which we denote by $\mathrm{err}(\widehat{\chi}_0)$, does not exceed $\frac{\|\widehat{X}\|_2^2}{\mathrm{poly}(n)}$ which by the final assumption of the lemma implies that $\mathrm{err}(\widehat{\chi}_0) \le \nu^2$.

Overview of the proof: We introduce the *approximate support* set of the input signal $\widehat{X}$, given by the union of the top k_0 blocks of the signal and the blocks whose energy is more than the tail noise level:

$$S_0 := \left\{ j \in \left[\frac{n}{k_1}\right] : \|\widehat{X}_{I_j}\|_2^2 \ge \mu^2 \right\} \cup \left(\operatorname*{arg\,min}_{\substack{S \subset [\frac{n}{k_1}] \\ |S| = k_0}} \sum_{j \in [\frac{n}{k_1}] \setminus S} \|\widehat{X}_{I_j}\|_2^2 \right). \tag{77}$$

From the definition of μ^2 in Definition 1.2, we readily obtain $|S_0| \le 2k_0$. For each $t = 1, 2, ..., T$, define the set S_t as

$$S_t = S_{t-1} \cup L'_t,$$

where L'_t is the output of PruneLocation at iteration t of ReduceSNR. S_t containes the set of the head elements of $\widehat{X}$ plus every element that is modified by the algorithm so far which can be both potetially large in the residual.

We prove by induction on the iteration number $t = 1, \dots, T$ that there exist events $\mathcal{E}_0 \supseteq \mathcal{E}_1 \supseteq ... \supseteq \mathcal{E}_T$ such that conditioned on $\mathcal{E}_t$, the following conditions hold true:

a. $|S_t| \le 2k_0 + \frac{tk_0}{T}$;

b. $\|\widehat{\chi}^{(t)}_{I_j}\|_2^2 = 0$ for all $j \in \left[\frac{n}{k_1}\right] \backslash S_t$;

c. $\|\widehat{X} - \widehat{\chi}^{(t)}\|_2^2 \le 99 \cdot \mathrm{SNR}'(k_0\nu^2)/2^t$;

and for each $t \le T$, we have $\Pr[\mathcal{E}_{t+1}|\mathcal{E}_t] \ge 1 - \frac{1}{10T}$.

Base case of the induction: We have already deduced that $|S_0| \le 2k_0$ and defined $\widehat{\chi}^{(0)} = 0$, and we have $\|\widehat{X} - \widehat{\chi}^{(0)}\|_2^2 = \|\widehat{X}\|_2^2 \le \mathrm{SNR}' \cdot (k_0\mu^2)/2^0$ by the two assumptions of the lemma. Hence, we can let $\mathcal{E}_0$ be the trivial event satisfying $\Pr[\mathcal{E}_0] = 1$.

Inductive step: We seek to define an event $\mathcal{E}_{t+1}$ that occurs with probability at least $1 - \frac{1}{10T}$ conditioned on $\mathcal{E}_t$, and such that the induction hypotheses **a**, **b**, and **c** are satisfied for $t+1$ conditioned on $\mathcal{E}_{t+1}$. To do this, we will introduce three events $\mathcal{E}_{\mathrm{loc},t}$, $\mathcal{E}_{\mathrm{prune},t}$, and $\mathcal{E}_{\mathrm{est},t}$, and set $\mathcal{E}_{t+1} = \mathcal{E}_{\mathrm{loc},t} \cap \mathcal{E}_{\mathrm{prune},t} \cap \mathcal{E}_{\mathrm{est},t} \cap \mathcal{E}_t$. Throughout the following, we let δ, θ, and p be chosen as in Algorithm 6

Success event associated with MultiBlockLocate: Let $\mathcal{E}_{\mathrm{loc},t}$ be the event of having a successful run of MultiBlockLocate$(X, \widehat{\chi}^{(t)}, n, k_1, k_0, \delta, p)$ at iteration $t+1$ of the algorithm, meaning the following conditions on the output L:

$$|L| \le C \cdot T \cdot \frac{k_0}{\delta} \log \frac{k_0}{\delta} \log^3 \frac{1}{\delta p} \tag{78}$$

$$\sum_{j \in S_t \setminus L} \|(\widehat{X} - \widehat{\chi}^{(t)})_{I_j}\|_2^2 \le 0.1 \|\widehat{X} - \widehat{\chi}^{(t)}\|_2^2, \tag{79}$$

where C is a constant to be specified shortly, for a small enough δ. To show this, we invoke Lemma 3.5 with $S^* = S_t$. Note that inductive hypothesis (**a**) implies $|S_t| \le (2 + \frac{t}{\log \mathrm{SNR}'})k_0 \le 3k_0$. By the first part of Lemma 3.5, we have $\mathbb{E}\big[|L|\big] \le C' \frac{k_0}{\delta} \log \frac{k_0}{\delta} \log \frac{1}{p} \log^2 \frac{1}{\delta p}$ for an absolute constant C', and hence (78) follows with $C = 100C'$ and probability at least $1 - \frac{1}{100T}$, by Markov's inequality.

By the second part of Lemma 3.5, (79) holds with probability at least $1 - p$ provided that $\delta \le \big(\frac{0.1}{200}\big)^2$, so by the union bound, the event $\mathcal{E}_{\mathrm{loc},t}$ occurs with probability at least $\Pr[\mathcal{E}_{\mathrm{loc},t} | \mathcal{E}_t] \ge 1 - p - \frac{1}{100T}$.

Success event associated with PruneLocation: Let $\mathcal{E}_{\mathrm{prune},t}$ be the event of having a successful run of PruneLocation$(X, \widehat{\chi}^{(t)}, L, k_0, k_1, \delta, p, n, \theta)$ at iteration $t+1$ of the algorithm, meaning the following conditions on the output L':

$$|L' \setminus S_t| \le \frac{k_0}{T} \tag{80}$$

$$\sum_{j \in [\frac{n}{k_1}] \setminus L'} \|(\widehat{X} - \widehat{\chi}^{(t)})_{I_j}\|_2^2 \le 0.2 \|\widehat{X} - \widehat{\chi}^{(t)}\|_2^2 + k_0(\mu^2 + 33\nu^2 \mathrm{SNR}'/2^{t+1}). \tag{81}$$

The probability of (80) **holding:** In order to bound $|L' \setminus S_t|$, first recall that the set S_{tail}, defined in Lemma 5.1 part (**a**), has the following form:

$$S_{\mathrm{tail}} = \Big\{ j \in \Big[\frac{n}{k_1}\Big] \,:\, \|(\widehat{X} - \widehat{\chi}^{(t)})_{I_j}\|_2 \le \sqrt{\theta} - \sqrt{\frac{\delta}{k_0}} \|\widehat{X} - \widehat{\chi}^{(t)}\|_2 \Big\}.$$

By substituting $\theta = 10 \cdot 2^{-(t+1)} \cdot \nu^2(\mathrm{SNR}')$ and using $\|\widehat{X} - \widehat{\chi}^{(t)}\|_2^2 \le 99 \cdot \mathrm{SNR}'(k_0\nu^2)/2^t$ from part **c** of the inductive hypothesis, we have

$$\begin{aligned}
&\sqrt{\theta} - \sqrt{\frac{\delta}{k_0}} \|\widehat{X} - \widehat{\chi}^{(t)}\|_2 \\
&\quad \ge \sqrt{10 \cdot 2^{-(t+1)} \cdot \nu^2(\mathrm{SNR}')} - \sqrt{\frac{\delta}{k_0}} \sqrt{99 \cdot \mathrm{SNR}'(k_0\nu^2)/2^t} \\
&\quad \ge \sqrt{9 \cdot \nu^2(\mathrm{SNR}')/2^{t+1}},
\end{aligned}$$

where the last inequality holds when δ is sufficiently small. Hence,

$$S_{\mathrm{tail}} \supseteq \Big\{ j \in \Big[\frac{n}{k_1}\Big] \,:\, \|(\widehat{X} - \widehat{\chi}^{(t)})_{I_j}\|_2^2 \le 9 \cdot \nu^2(\mathrm{SNR}')/2^{t+1} \Big\}. \tag{82}$$

Now, to prove that (80) holds with high probability, we write

$$|L' \setminus S_t| = |(L' \cap S_{\mathrm{tail}}) \setminus S_t| + |L' \setminus (S_{\mathrm{tail}} \cup S_t)|. \tag{83}$$

To upper bound the first term, note that by the first part of Lemma 5.1, we have

$$\mathbb{E}\Big[\big|L' \cap S_{\mathrm{tail}}\big|\Big] \le \delta p \cdot |L|,$$

and hence by Markov's inequality, the following holds with probability at least $1 - \frac{1}{100T}$:

$$\begin{aligned}\big|(L' \cap S_{\mathrm{tail}}) \backslash S_t\big| &\le \big|L' \cap S_{\mathrm{tail}}\big| \\ &\le 100T\delta p \cdot |L| \\ &\le 100T\delta p \cdot CT\frac{k_0}{\delta}\log\frac{k_0}{\delta}\log^3\frac{1}{\delta p} \\ &= 100CT^2 p \cdot k_0 \log\frac{k_0}{\delta}\log^3\frac{1}{\delta p} \\ &= \frac{100C\delta \cdot k_0 \log^3 \frac{1}{\delta p}}{\log\frac{k_0}{\delta} \cdot \log^2 \mathrm{SNR}'},\end{aligned}$$

where the third line follows from (78) (we condition on $\mathcal{E}_{\mathrm{loc},t}$), and the fifth line follows from $T = \log \mathrm{SNR}'$ and the choice $p = \frac{\delta}{\log^2 \frac{k_0}{\delta} \log^4 \mathrm{SNR}'}$ in Algorithm 6. Again using this choice of p, we find that $\frac{100C\delta \log^3 \frac{1}{\delta p}}{\log \frac{k_0}{\delta} \cdot \log \mathrm{SNR}'} \le 1$ for sufficiently small δ regardless of the values (k_0, SNR') because the dependence of $1/p$ on k_0 and SNR' is logarithmic therefore in the numerator we have $\log^3 \log k_0$ and $\log^3 \log \mathrm{SNR}'$ while the denuminator has $\log k_0$ and $\log^2 \mathrm{SNR}'$ which means that the ratio is upper bounded and can be made arbitraryly small by choosing a small enough constant δ. Hence,

$$\big|(L' \cap S_{\mathrm{tail}}) \backslash S_t\big| \le \frac{k_0}{\log \mathrm{SNR}'}$$

with probability at least $1 - \frac{1}{100T}$.

We now show that the second term in (83) is zero, by showing that $S_{\mathrm{tail}} \cup S_t = [\frac{n}{k_1}]$. To see this, note that the term $\nu^2(\mathrm{SNR}')/2^{t+1}$ in the bound on S_{tail} in (82) satisfies

$$\nu^2(\mathrm{SNR}')/2^{t+1} \ge \frac{1}{2}\nu^2 \ge \frac{1}{2}\mu^2, \tag{84}$$

by applying $t \le T = \log \mathrm{SNR}'$, followed by the first assumption of the lemma. Hence,

$$S_{\mathrm{tail}} \backslash S_t \supseteq \Big\{ j \in \Big[\frac{n}{k_1}\Big] \backslash S_t \,:\, \|(\widehat{X} - \widehat{\chi}^{(t)})_{I_j}\|_2^2 \le 4\mu^2 \Big\}.$$

By part **b** of the inductive hypothesis, we have $\|(\widehat{X} - \widehat{\chi}^{(t)})_{I_j}\|_2^2 = \|\widehat{X}_{I_j}\|_2^2$ for all $j \notin S_t$, and hence

$$S_{\mathrm{tail}} \backslash S_t \supseteq \Big\{ j \in \Big[\frac{n}{k_1}\Big] \backslash S_t \,:\, \|\widehat{X}_{I_j}\|_2^2 \le 4\mu^2 \Big\}.$$

But from (77), we know that S_0 (and hence S_t) contains all j with $\|\widehat{X}_{I_j}\|_2^2 > 4\mu^2$, so we obtain $S_{\mathrm{tail}} \backslash S_t \supset \big[\frac{n}{k_1}\big] \backslash S_t$, and hence $S_{\mathrm{tail}} \cup S_t = (S_{\mathrm{tail}} \backslash S_t) \cup S_t = \big[\frac{n}{k_1}\big]$, as required.

Therefore, the probability of (80) holding conditioned on $\mathcal{E}_t$ and $\mathcal{E}_{\mathrm{loc},t}$ is at least $1 - \frac{1}{100T}$.

The probability of (81) **holding:** To show (81), we use the second part of Lemma 5.1. The set S_{head} therein is defined as

$$S_{\mathrm{head}} = \Big\{ j \in \Big[\frac{n}{k_1}\Big] \,:\, \|(\widehat{X} - \widehat{\chi}^{(t)})_{I_j}\|_2 \ge \sqrt{\theta} + \sqrt{\frac{\delta}{k_0}}\|\widehat{X} - \widehat{\chi}^{(t)}\|_2 \Big\}.$$

By substituting $\theta = 10 \cdot 2^{-(t+1)} \cdot \nu^2(\mathrm{SNR}')$ and using $\|\widehat{X} - \widehat{\chi}^{(t)}\|_2^2 \leq 99 \cdot \mathrm{SNR}'(k_0\nu^2)/2^t$ from part **c** of the inductive hypothesis, we have

$$\begin{aligned}
&\sqrt{\theta} + \sqrt{\frac{\delta}{k_0}}\|\widehat{X} - \widehat{\chi}\|_2 \\
&\quad= \sqrt{10 \cdot 2^{-(t+1)} \cdot \nu^2(\mathrm{SNR}')} + \sqrt{\frac{\delta}{k_0}}\sqrt{99 \cdot \mathrm{SNR}'(k_0\nu^2)/2^t} \\
&\quad\leq \sqrt{11 \cdot \nu^2(\mathrm{SNR}')/2^{t+1}}
\end{aligned}$$

for sufficiently small δ, and hence

$$S_{\mathrm{head}} \supseteq \Big\{ j \in \Big[\frac{n}{k_1}\Big] \,:\, \|(\widehat{X} - \widehat{\chi})_{I_j}\|_2^2 \geq 11 \cdot \nu^2(\mathrm{SNR}')/2^{t+1} \Big\}. \tag{85}$$

Next, we write

$$\begin{aligned}
\sum_{j \in [\frac{n}{k_1}] \backslash L'} \|(\widehat{X} - \widehat{\chi}^{(t)})_{I_j}\|_2^2 = &\sum_{j \in (S_t \cap S_{\mathrm{head}} \cap L) \backslash L'} \|(\widehat{X} - \widehat{\chi}^{(t)})_{I_j}\|_2^2 + \sum_{j \in (S_t \cap S_{\mathrm{head}}) \backslash (L' \cup L)} \|(\widehat{X} - \widehat{\chi}^{(t)})_{I_j}\|_2^2 \\
&+ \sum_{j \in S_t \backslash (S_{\mathrm{head}} \cup L')} \|(\widehat{X} - \widehat{\chi}^{(t)})_{I_j}\|_2^2 + \sum_{j \in [\frac{n}{k_1}] \backslash (L' \cup S_t)} \|(\widehat{X} - \widehat{\chi}^{(t)})_{I_j}\|_2^2,
\end{aligned} \tag{86}$$

and we proceed by upper bounding the four terms.

Bounding the first term in (86)**:** By the second part of Lemma 5.1 and the use of Markov, we have

$$\sum_{j \in (S_t \cap S_{\mathrm{head}} \cap L) \backslash L'} \|(\widehat{X} - \widehat{\chi}^{(t)})_{I_j}\|_2^2 \leq \delta \sum_{j \in L \cap S_{\mathrm{head}}} \|(\widehat{X} - \widehat{\chi}^{(t)})_{I_j}\|_2^2 \leq \delta \|\widehat{X} - \widehat{\chi}^{(t)}\|_2^2$$

with probability at least $1 - p$.

Bounding the second term in (86)**:** Conditioned on $\mathcal{E}_{\mathrm{loc},t}$, we have

$$\sum_{j \in (S_t \cap S_{\mathrm{head}}) \backslash (L \cup L')} \|(\widehat{X} - \widehat{\chi}^{(t)})_{I_j}\|_2^2 \leq \sum_{j \in S_t \backslash L} \|(\widehat{X} - \widehat{\chi}^{(t)})_{I_j}\|_2^2 \leq 0.1 \|\widehat{X} - \widehat{\chi}^{(t)}\|_2^2,$$

where we have applied (79).

Bounding the third term in (86)**:** We have

$$\begin{aligned}
\sum_{j \in S_t \backslash (S_{\mathrm{head}} \cup L')} \|(\widehat{X} - \widehat{\chi}^{(t)})_{I_j}\|_2^2 &\leq \sum_{j \in S_t \backslash S_{\mathrm{head}}} \|(\widehat{X} - \widehat{\chi}^{(t)})_{I_j}\|_2^2 \\
&\leq |S_t \backslash S_{\mathrm{head}}| \cdot \max_{j \in S_t \backslash S_{\mathrm{head}}} \|(\widehat{X} - \widehat{\chi}^{(t)})_{I_j}\|_2^2 \\
&\leq |S_t| (11 \cdot \nu^2 \mathrm{SNR}'/2^{t+1})
\end{aligned}$$

by (85). Part **a** of the inductive hypothesis implies that $|S_t| \leq 3k_0$, and hence

$$\sum_{j \in (L \cap S_t) \backslash (S_{\mathrm{head}} \cup L')} \|(\widehat{X} - \widehat{\chi}^{(t)})_{I_j}\|_2^2 \leq 33 k_0 \cdot \nu^2 \mathrm{SNR}'/2^{t+1}.$$

Bounding the fourth term in (86):

$$\begin{aligned}\sum_{j\in[\frac{n}{k_1}]\setminus(L'\cup S_t)} \|(\widehat{X}-\widehat{\chi}^{(t)})_{I_j}\|_2^2 &\le \sum_{[\frac{n}{k_1}]\setminus S_t} \|(\widehat{X}-\widehat{\chi}^{(t)})_{I_j}\|_2^2 \\ &= \sum_{[\frac{n}{k_1}]\setminus S_t} \|\widehat{X}_{I_j}\|_2^2 \le k_0\mu^2,\end{aligned}$$

where the equality follows from part **b** of the inductive hypothesis, and the final step holds since S_t contains all elements with $\|\widehat{X}_{I_j}\|_2^2 \ge \mu^2$ (*cf.*, (77)).

Adding the above four contributions and applying the union bound, we find that conditioned on $\mathcal{E}_t$ and $\mathcal{E}_{\mathrm{loc},t}$, (81) holds with probability at least $\Pr[\mathcal{E}_{\mathrm{prune},t}|\mathcal{E}_t\cap\mathcal{E}_{\mathrm{loc},t}] \ge 1-p-\frac{1}{100T}$, provided that δ is a sufficiently small constant ($\delta \le 0.1$).

Success event associated with EstimateValues: Let $\mathcal{E}_{\mathrm{est},t}$ be the event of having a successful run of EstimateValues$(X, \widehat{\chi}^{(t)}, L', k_0, k_1, \delta, p)$ at iteration $t+1$ of the algorithm conditioned on $\mathcal{E}_t$, meaning the following conditions on the output signal W:

$$\begin{gathered} W_f = 0 \text{ for all } f\notin\mathcal{F} \\ \sum_{j\in L'} \|(\widehat{X}-\widehat{\chi}^{(t)}-W)_{I_j}\|^2 \le \delta\|\widehat{X}-\widehat{\chi}^{(t)}\|_2^2, \end{gathered} \tag{87}$$

where $\mathcal{F}$ contains the frequencies within the blocks indexed by L'. By Lemma 5.2 and the fact that $|L'|\le 3k_0$ conditioned on $\mathcal{E}_{\mathrm{prune},t}$, $\mathcal{E}_{\mathrm{loc},t}$, and $\mathcal{E}_t$, it immediately follows that $\mathcal{E}_{\mathrm{est},t}$ occurs with probability at least $\Pr[\mathcal{E}_{\mathrm{est},t}|\mathcal{E}_{\mathrm{prune},t}\cap\mathcal{E}_{\mathrm{loc},t}\cap\mathcal{E}_t] \ge 1-p$.

Combining the events: We can now wrap everything up as follows:

$$\begin{aligned}\Pr\big[\mathcal{E}_{t+1}\big|\mathcal{E}_t\big] &= \Pr\big[\mathcal{E}_{\mathrm{loc},t}\cap\mathcal{E}_{\mathrm{prune},t}\cap\mathcal{E}_{\mathrm{est},t}\cap\mathcal{E}_t\big|\mathcal{E}_t\big] \\ &= \Pr\big[\mathcal{E}_{\mathrm{est},t}\big|\mathcal{E}_{\mathrm{loc},t}\cap\mathcal{E}_{\mathrm{prune},t}\cap\mathcal{E}_t\big]\Pr\big[\mathcal{E}_{\mathrm{prune},t}\big|\mathcal{E}_{\mathrm{loc},t}\cap\mathcal{E}_t\big]\Pr\big[\mathcal{E}_{\mathrm{loc},t}\big|\mathcal{E}_t\big].\end{aligned}$$

Substituting the probability bounds into the above equation, we have

$$\Pr\big[\mathcal{E}_{t+1}\big|\mathcal{E}_t\big] \ge 1-3p-\frac{2}{100T} \ge 1-\frac{1}{20T},$$

by the choice of p in Algorithm 6 along with $T=\log\mathrm{SNR}$.

Now we show that the event $\mathcal{E}_{t+1}=\mathcal{E}_{\mathrm{loc},t}\cap\mathcal{E}_{\mathrm{prune},t}\cap\mathcal{E}_{\mathrm{est},t}\cap\mathcal{E}_t$ implies the induction hypothesis. Conditioned on $\mathcal{E}_{\mathrm{prune},t}\cap\mathcal{E}_t$, we have (80), which immediately gives part **a**. Conditioned on $\mathcal{E}_{\mathrm{loc},t}\cap\mathcal{E}_{\mathrm{est},t}$, from the definition $S_{t+1}=S_t\cup L'$, part **b** of the inductive hypothesis follows from the fact that only elements in L' are updated. Finally, conditioned on $\mathcal{E}_t\cap\mathcal{E}_{\mathrm{prune},t}\cap\mathcal{E}_{\mathrm{est},t}$, we have

$$\begin{aligned}\|\widehat{X}-\widehat{\chi}^{(t+1)}\|_2^2 &= \sum_{j\in L'}\|(\widehat{X}-\widehat{\chi}^{(t+1)})_{I_j}\|_2^2 + \sum_{j\in[\frac{n}{k_1}]\setminus L'}\|(\widehat{X}-\widehat{\chi}^{(t+1)})_{I_j}\|_2^2 \\ &= \sum_{j\in L'}\|(\widehat{X}-\widehat{\chi}^{(t)}-W)_{I_j}\|_2^2 + \sum_{j\in[\frac{n}{k_1}]\setminus L'}\|(\widehat{X}-\widehat{\chi}^{(t+1)})_{I_j}\|_2^2 \\ &\le (0.2+\delta)\|\widehat{X}-\widehat{\chi}^{(t)}\|_2^2 + k_0(\mu^2+33\nu^2\mathrm{SNR}'/2^{t+1}) \\ &\le 99\nu^2k_0\mathrm{SNR}'/2^{t+1},\end{aligned}$$

where the second line holds since W is non-zero only for the blocks indexed by L', the third line follows from (81) and (87), and the last line holds for sufficiently small δ from part **c** of the induction hypothesis, and the upper bound $\mu^2 \le 2\nu^2(\mathrm{SNR}')/2^{t+1}$ given in (84).

The first part of the lemma now follows from a union bound over the T iterations and the fact that the accumulated error $\mathrm{err}(\widehat{\chi}_0) \le \nu^2$, and by noting that the three parts of the induction hypothesis immediately yield the two claims therein. We conclude by analyzing the sample complexity and runtime.

Sample complexity: We have from Lemma 3.5 that the *expected* sample complexity of MultiBlockLocate in a given iteration is $O^*\big(\frac{k_0}{\delta}\log(1+k_0)\log n + \frac{k_0k_1}{\delta^2}\big)$, and multiplying by the number $T = O(\log \mathrm{SNR}')$ of iterations gives a total of $O^*\big(\log \mathrm{SNR}'\big(\frac{k_0}{\delta}\log(1+k_0)\log n + \frac{k_0k_1}{\delta^2}\big)\big)$. Hence, by Markov's inequality, this is also the total sample complexity across all calls to MultiBlockLocate with probability at least $1-\frac{1}{100}$; this probability can be combined with the union bound that we applied above. Since $\delta = \Omega(1)$, the above sample complexity simplifies to $O^*(k_0\log(1+k_0)\log\mathrm{SNR}'\log n + k_0k_1\log\mathrm{SNR}')$.

By Lemma 5.1, the sample complexity of PruneLocation is $O(\frac{k_0k_1}{\delta}\log\frac{1}{\delta p}\log\frac{1}{\delta})$, and by Lemma 5.2, the sample complexity of EstimateValues is $O(\frac{k_0k_1}{\delta}\log\frac{1}{p}\log\frac{1}{\delta})$. Substituting the choices of δ and p, these behave as $O^*(k_0k_1)$ per iteration, or $O^*(k_0k_1\log\mathrm{SNR}')$ overall.

Runtime: By Lemma 3.5, the expected runtime of MultiBlockLocate in a given iteration is $O^*\big(\frac{k_0}{\delta}\log(1+k_0)\log^2 n + \frac{k_0k_1}{\delta^2}\log^2 n + \frac{k_0k_1}{\delta}\log^3 n\big)$. Moreover, by Lemma 5.1, the runtime of PruneLocation as a function of $|L|$ is $O(\frac{k_0k_1}{\delta}\log\frac{1}{\delta p}\log\frac{1}{\delta}\log n + k_1\cdot|L|\log\frac{1}{\delta p})$, and substituting $\mathbb{E}\big[|L|\big] = O\big(\frac{k_0}{\delta}\log\frac{k_0}{\delta}\log\frac{1}{p}\log^2\frac{1}{\delta p}\big)$ from Lemma 3.5, this becomes $O^*\big(\frac{k_0k_1}{\delta}\log n\big)$ in expectation, by absorbing the $\log\frac{1}{\delta}$ and $\log\frac{1}{p}$ factors into the $O^*(\cdot)$ notation.

Summing the preceding per-iteration expected runtimes, multiplying by the number of iterations T, and substituting the choices of T, p and δ, we find that their combined expectation is $O^*(k_0\log k_0\log\mathrm{SNR}'\log^2 n + k_0k_1\log\mathrm{SNR}'\log^3 n)$. Hence, by Markov's inequality, this is also the total sample complexity across all calls to MultiBlockLocate and PruneLocation with probability at least $1-\frac{1}{100}$. A union bound over this failure event, $\bar{\mathcal{E}}_T$ and the $1/\mathrm{poly}(n)$ probability failure event arising from random perturbations of $\widehat{\chi}_0$ yields the required bound of 0.9 on the success probability.

By Lemma 5.2, the runtime of EstimateValues is $O(\frac{k_0k_1}{\delta}\log\frac{1}{p}\log\frac{1}{\delta}\log n + k_1\cdot|L'|\log\frac{1}{p})$, which behaves as $O(\frac{k_0k_1}{\delta}\log\frac{1}{p}\log\frac{1}{\delta}\log n)$ conditioned on $\mathcal{E}_{\mathrm{prune},t}\cap\mathcal{E}_t$ (see (80) and recall that $|S_t|\le 3k_0$). By our choices of p and δ, this simplifies to $O^*(k_0k_1\log n)$ per iteration, or $O^*(k_0k_1\log\mathrm{SNR}'\log n)$ overall.

D.2 Proof of Lemma 5.4

The proof resembles that of Lemma 5.3, but is generally simpler, and has some differing details. We provide the details for completeness.

Note on $\frac{1}{\mathrm{poly}(n)}$ assumptions in lemmas: We use an analogous argument to the start of Section D.1 to handle the assumptions $|\widehat{X}_0 - \widehat{\chi}_0|_2 \geq \frac{1}{\mathrm{poly}(n)}\|\widehat{\chi}\|_2^2$ and $\|\widehat{X} - \widehat{\chi}\|_2 \geq \frac{1}{\mathrm{poly}(n)}\|\widehat{\chi}\|_2^2$ in Lemmas 3.5, 5.1, and 5.2. Specifically, we add a noise term to $\widehat{\chi}_0$ uniform in $[-n^{-c'+10}\|\widehat{\chi}\|^2, n^{c'+10}\|\widehat{\chi}\|^2]$. This does not affect the result, since the noise added to $\widehat{\chi}_0$ which we denote by $\mathrm{err}(\widehat{\chi}_0)$, does not exceed $\frac{\|\widehat{X}\|_2^2}{\mathrm{poly}(n)}$ which by the assumptions of the lemma implies that $\mathrm{err}(\widehat{\chi}_0) \leq \epsilon\nu^2$.

Overview of the proof: We first introduce the *approximate support* set of the input signal $\widehat{X} - \widehat{\chi}$, given by the top $10k_0$ blocks of the signal:

$$S_0 := \operatorname*{arg\,min}_{\substack{S\subset[\frac{n}{k_1}]\\ |S|=10k_0}} \sum_{j\notin S} \|(\widehat{X} - \widehat{\chi})_{I_j}\|_2^2 \tag{88}$$

We also introduce another set indexing blocks whose energy is sufficiently large:

$$S_\epsilon = \left\{ j \in \left[\frac{n}{k_1}\right] \,:\, \|(\widehat{X} - \widehat{\chi})_{I_j}\|_2^2 \geq \epsilon \frac{\mathrm{Err}^2(\widehat{X} - \widehat{\chi}, 10k_0, k_1)}{k_0} \right\} \cup S_0. \tag{89}$$

It readily follows from this definition and Definition 1.2 that $|S_\epsilon \backslash S_0| \leq k_0/\epsilon$.

The function calls three other primitives, and below, we show that each of them succeeds with high probability by introducing suitable success events. Throughout, we let θ, p, and η be as chosen in Algorithm 6

Success event of the location primitive: Let $\mathcal{E}_{\mathrm{loc}}$ be the event of having a successful run of $\textsc{MultiBlockLocate}(X, \widehat{\chi}, k_1, k_0, n, \epsilon^2, p)$, meaning the following conditions on the output L:

$$|L| \leq C\frac{k_0}{\epsilon^2}\log\frac{k_0}{\epsilon^2}\log^3\frac{1}{\epsilon^2 p} \tag{90}$$

$$\sum_{j\in S_0\backslash L} \|(\widehat{X} - \widehat{\chi})_{I_j}\|_2^2 \leq 200\epsilon\|\widehat{X} - \widehat{\chi}\|_2^2, \tag{91}$$

where C is a constant to be specified shortly. To verify these conditions, we invoke Lemma 3.5 with $S^* = S_0$. By the first part of Lemma 3.5, we have $\mathbb{E}\big[|L|\big] \leq C'\frac{k_0}{\epsilon^2}\log\frac{k_0}{\epsilon}\log^3\frac{1}{\epsilon p}$ for an absolute constant C', and hence (90) follows with $C = 100C'$ and probability at least $1 - \frac{1}{100}$, by Markov's inequality.

By the second part of Lemma 3.5 with $\delta = \epsilon^2$, (91) holds with probability at least $1 - p$, so by the union bound, the event $\mathcal{E}_{\mathrm{loc}}$ occurs with probability at least $1 - p - \frac{1}{100}$.

Success event of the pruning primitive: Let $\mathcal{E}_{\mathrm{prune}}$ be the event of having a successful run of $\textsc{PruneLocation}(X, \widehat{\chi}, L, n, k_0, k_1, \epsilon, p, \theta)$, meaning the following conditions on the output L':

$$|L'\backslash S_0| \leq \frac{2k_0}{\epsilon} \tag{92}$$

$$\sum_{j\in[\frac{n}{k_1}]\backslash L'} \|(\widehat{X} - \widehat{\chi})_{I_j}\|_2^2 \leq 300\epsilon\|\widehat{X} - \widehat{\chi}\|_2^2 + 6000\epsilon\nu^2 k_0 + \mathrm{Err}^2(\widehat{X} - \widehat{\chi}, 10k_0, k_1). \tag{93}$$

Bounding the probability of (92)**:** In order to bound $|L'\backslash S_0|$, first note that the set S_{tail}, defined in Lemma 5.1 part (**a**), has the following form:

$$S_{\text{tail}} = \Big\{ j \in \Big[\frac{n}{k_1}\Big] \ : \ \|(\widehat{X} - \widehat{\chi})_{I_j}\|_2 \le \sqrt{\theta} - \sqrt{\frac{\epsilon}{k_0}}\|\widehat{X} - \widehat{\chi}\|_2 \Big\}.$$

By substituting $\theta = 200 \cdot \epsilon\nu^2$ (*cf.*, Algorithm 6) and using the assumption $\|\widehat{X} - \widehat{\chi}\|_2^2 \le 100k_0\nu^2$ in the lemma, we have

$$\begin{aligned}\sqrt{\theta} - \sqrt{\frac{\epsilon}{k_0}}\|\widehat{X} - \widehat{\chi}\|_2 &= \sqrt{200 \cdot \epsilon\nu^2} - \sqrt{\frac{\epsilon}{k_0}}\|\widehat{X} - \widehat{\chi}\|_2 \\ &\ge \sqrt{200 \cdot \epsilon\nu^2} - \sqrt{100 \cdot \epsilon\nu^2} \\ &\ge \sqrt{16 \cdot \epsilon\nu^2}. \end{aligned} \tag{94}$$

Hence,

$$S_{\text{tail}} \supseteq \Big\{ j \in \Big[\frac{n}{k_1}\Big] \ : \ \|(\widehat{X} - \widehat{\chi}^{(t)})_{I_j}\|_2^2 \le 16\epsilon \cdot \nu^2 \Big\}.$$

Now, to prove that (92) holds with high probability, we write

$$\begin{aligned}|L'\backslash S_0| &= |(L' \cap S_\epsilon)\backslash S_0| + |L'\backslash(S_0 \cup S_\epsilon)| \\ &\le |(L' \cap S_\epsilon)\backslash S_0| + |(L' \cap S_{\text{tail}})\backslash S_\epsilon| + |L'\backslash(S_{\text{tail}} \cup S_\epsilon)|.\end{aligned} \tag{95}$$

We first upper bound the first term as follows:

$$|(L' \cap S_\epsilon)\backslash S_0| \le |S_\epsilon\backslash S_0| \le k_0/\epsilon,$$

which follows directly from the definition of S_ϵ. To upper bound the second term in (95), note that by Lemma 5.1 part (**a**) with $\delta = \epsilon$,

$$\mathbb{E}\Big[\big|L' \cap S_{\text{tail}}\big|\Big] \le \epsilon p \cdot |L|,$$

and hence by Markov's inequality, the following holds with probability at least $1 - \frac{1}{100}$:

$$\begin{aligned}\big|(L' \cap S_{\text{tail}})\backslash S_\epsilon\big| &\le \big|L' \cap S_{\text{tail}}\big| \\ &\le 100\epsilon p \cdot |L| \\ &\le 100\epsilon p \cdot C\frac{k_0}{\epsilon}\log\frac{k_0}{\epsilon}\log^3\frac{1}{\epsilon p} \\ &= 100Cp \cdot k_0 \log\frac{k_0}{\epsilon}\log^3\frac{1}{\epsilon p} \\ &= \frac{100C\eta\epsilon \cdot k_0\log^3\frac{1}{\epsilon p}}{\log\frac{k_0}{\epsilon}},\end{aligned}$$

where the third line follows from (90) (we condition on $\mathcal{E}_{\text{loc}}$), and the fifth line follows from and the choice $p = \frac{\eta\epsilon}{\log^2\frac{k_0}{\epsilon}}$ in Algorithm 6. Again using this choice of p, we find that $\frac{100C\eta\epsilon\log^3\frac{1}{\epsilon p}}{\log\frac{k_0}{\delta}} \le 1$ for sufficiently small η regardless of the value of k_0 because the dependence of $1/p$ on k_0 is logarithmic therefore in the numerator we have $\log^3\log k_0$ while the denuminator has $\log k_0$ which means that the ratio is upper bounded and can be made arbitraryly small by choosing a small enough constant η. Hence

$$\big|(L' \cap S_{\text{tail}})\backslash S_\epsilon\big| \le k_0$$

with probability at least $1 - \frac{1}{100}$.

We now show that the third term in (95) is zero, by showing that $S_{\text{tail}} \cup S_\epsilon = [\frac{n}{k_1}]$. To see this, note that the term ν^2 in the definition of S_{tail} is more than $\frac{\text{Err}^2(\widehat{X}-\widehat{\chi},10k_0,k_1)}{k_0}$ by the first assumption of the lemma, and hence

$$S_{\text{tail}} \backslash S_\epsilon \supset \Big\{ j \in \Big[\frac{n}{k_1}\Big] \backslash S_\epsilon \,:\, \|(\widehat{X} - \widehat{\chi})_{I_j}\|_2^2 \le 16\epsilon \frac{\text{Err}^2(\widehat{X} - \widehat{\chi}, 10k_0, k_1)}{k_0} \Big\}$$

However, the definition of S_ϵ in (89) reveals that the condition upper bounding $\|(\widehat{X} - \widehat{\chi})_{I_j}\|_2^2$ is redundant, and $S_{\text{tail}} \backslash S_\epsilon \supset \big[\frac{n}{k_1}\big] \backslash S_\epsilon$, and hence $S_{\text{tail}} \cup S_\epsilon = (S_{\text{tail}} \backslash S_\epsilon) \cup S_\epsilon = \big[\frac{n}{k_1}\big]$.

Bounding the probability of (93)**:** To show (93), we use the second part of Lemma 5.1. The set S_{head} therein is defined as

$$S_{\text{head}} = \Big\{ j \in \Big[\frac{n}{k_1}\Big] \,:\, \|(\widehat{X} - \widehat{\chi})_{I_j}\|_2 \ge \sqrt{\theta} + \sqrt{\frac{\epsilon}{k_0}} \|\widehat{X} - \widehat{\chi}\|_2 \Big\}.$$

By substituting $\theta = 200\epsilon \cdot \nu^2$ (*cf.*, Algorithm 6) and using the assumption $\|\widehat{X} - \widehat{\chi}\|_2^2 \le 100k_0\nu^2$ in the lemma, we have

$$\sqrt{\theta} + \sqrt{\frac{\epsilon}{k_0}} \|\widehat{X} - \widehat{\chi}\|_2 = \sqrt{200\epsilon \cdot \nu^2} + \sqrt{\frac{\epsilon}{k_0}} \|\widehat{X} - \widehat{\chi}\|_2 \le \sqrt{600\epsilon \cdot \nu^2},$$

and hence

$$S_{\text{head}} \supseteq \Big\{ j \in \Big[\frac{n}{k_1}\Big] \,:\, \|(\widehat{X} - \widehat{\chi})_{I_j}\|_2^2 \ge 600\epsilon \cdot \nu^2 \Big\}. \tag{96}$$

Next, we write

$$\begin{aligned} \sum_{j \in [\frac{n}{k_1}] \backslash L'} \|(\widehat{X} - \widehat{\chi})_{I_j}\|_2^2 = & \sum_{j \in (S_0 \cap S_{\text{head}} \cap L) \backslash L'} \|(\widehat{X} - \widehat{\chi})_{I_j}\|_2^2 + \sum_{j \in (S_0 \cap S_{\text{head}}) \backslash (L' \cup L)} \|(\widehat{X} - \widehat{\chi})_{I_j}\|_2^2 \\ & + \sum_{j \in S_0 \backslash (S_{\text{head}} \cup L')} \|(\widehat{X} - \widehat{\chi})_{I_j}\|_2^2 + \sum_{j \in [\frac{n}{k_1}] \backslash (L' \cup S_0)} \|(\widehat{X} - \widehat{\chi})_{I_j}\|_2^2, \end{aligned} \tag{97}$$

and we proceed by upper bounding the four terms.

Bounding the first term in (97)**:** By part **b** of Lemma 5.1, the choice $\delta = \epsilon$, and the use of Markov, we have

$$\sum_{j \in (S_0 \cap S_{\text{head}} \cap L) \backslash L'} \|(\widehat{X} - \widehat{\chi})_{I_j}\|_2^2 \le \epsilon \sum_{j \in L \cap S_{\text{head}}} \|(\widehat{X} - \widehat{\chi})_{I_j}\|_2^2 \le \epsilon \|\widehat{X} - \widehat{\chi}\|_2^2$$

with probability at least $1 - p$.

Bounding the second term in (97)**:** Conditioned on $\mathcal{E}_{\text{loc}}$, we have

$$\begin{aligned} \sum_{j \in (S_0 \cap S_{\text{head}}) \backslash (L \cup L')} \|(\widehat{X} - \widehat{\chi})_{I_j}\|_2^2 \le & \sum_{j \in S_0 \backslash L} \|(\widehat{X} - \widehat{\chi})_{I_j}\|_2^2 \\ \le & \; 200\epsilon \|\widehat{X} - \widehat{\chi}\|_2^2, \end{aligned}$$

where we have applied (91).

Bounding the third term in (97)**:** We have

$$\begin{aligned}\sum_{j\in S_0\backslash(S_{\mathrm{head}}\cup L')} \|(\widehat{X}-\widehat{\chi})_{I_j}\|_2^2 &\le \sum_{j\in S_0\backslash S_{\mathrm{head}}} \|(\widehat{X}-\widehat{\chi})_{I_j}\|_2^2 \\ &\le |S_0\backslash S_{\mathrm{head}}|\cdot \max_{j\in S_0\backslash S_{\mathrm{head}}} \|(\widehat{X}-\widehat{\chi})_{I_j}\|_2^2 \\ &\le |S_0|(600\epsilon\cdot\nu^2)\end{aligned}$$

by (96). We have by definition that $|S_0| = 10k_0$ (*cf.*, (88)), and hence

$$\sum_{j\in(L\cap S_0)\backslash(S_{\mathrm{head}}\cup L')} \|(\widehat{X}-\widehat{\chi})_{I_j}\|_2^2 \le 6000k_0\epsilon\cdot\nu^2.$$

Bounding the fourth term in (97)**:** We have

$$\begin{aligned}\sum_{j\in[\frac{n}{k_1}]\backslash(L'\cup S_0)} \|(\widehat{X}-\widehat{\chi})_{I_j}\|_2^2 &\le \sum_{j\in[\frac{n}{k_1}]\backslash S_0} \|(\widehat{X}-\widehat{\chi})_{I_j}\|_2^2 \\ &= \mathrm{Err}^2(\widehat{X}-\widehat{\chi}, 10k_0, k_1),\end{aligned}$$

which follows from the definition of S_0 in (88), along with Definition 1.2.

Hence by the union bound, it follows that $\mathcal{E}_{\mathrm{prune}}$ holds with probability at least $1-p-\frac{1}{1000}$ conditioned on $\mathcal{E}_{\mathrm{loc}}$.

Success event of estimation primitive: Let $\mathcal{E}_{\mathrm{est}}$ be the event of having a successful run of EstimateValues$(X,\widehat{\chi},L,n,3k_0/\epsilon,k_1,\epsilon,p)$, meaning the following conditions on the output, W:

$$\begin{gathered} W_f = 0 \text{ for all } f\notin\mathcal{F} \\ \sum_{j\in L'} \|(\widehat{X}-\widehat{\chi}-W)_{I_j}\|^2 \le \epsilon\|\widehat{X}-\widehat{\chi}\|_2^2, \end{gathered} \tag{98}$$

where $\mathcal{F}$ contains the frequencies within the blocks indexed by L'. Since the assumption of the theorem implies that $\|\widehat{\chi}\|_0 = O(k_0k_1)$, by Lemma 5.2 (with $\delta=\epsilon$ and $3k_0/\epsilon$ in place of k_0) and the fact that conditioned on $\mathcal{E}_{\mathrm{prune}}$ we have $|L'|\le 3k_0/\epsilon$ (*cf.*, (92)), it follows that $\mathcal{E}_{\mathrm{est}}$ occurs with probability at least $1-p$.

Combining the events: We can now can wrap everything up.

Letting $\mathcal{E}$ denote the overall success event corresponding to the claim of the lemma, we have

$$\begin{aligned}\Pr[\mathcal{E}] &= \Pr\big[\mathcal{E}_{\mathrm{loc}}\cap\mathcal{E}_{\mathrm{prune}}\cap\mathcal{E}_{\mathrm{est}}\big] \\ &= \Pr\big[\mathcal{E}_{\mathrm{est}}\big|\mathcal{E}_{\mathrm{loc}}\cap\mathcal{E}_{\mathrm{prune}}\big]\Pr\big[\mathcal{E}_{\mathrm{prune}}\big|\mathcal{E}_{\mathrm{loc}}\big]\Pr\big[\mathcal{E}_{\mathrm{loc}}\big].\end{aligned}$$

By the results that we have above, along with the union bound, it follows that

$$\Pr\big[\mathcal{E}\big] \ge 1 - 2/100 - 3p \ge 0.95$$

for sufficiently small η in Algorithm 6. A union bound over $\bar{\mathcal{E}}$ and the 1/poly(n) probability failure event arising from random perturbations of $\widehat{\chi}_0$ yields the required bound of 0.9 on the success probability.

What remains is to first show that the statement of the lemma follows from $\mathcal{E}_{\mathrm{loc}}\cap\mathcal{E}_{\mathrm{prune}}\cap\mathcal{E}_{\mathrm{est}}$.

To do this, we observe that, conditioned on these events,

$$
\begin{aligned}
\|\widehat{X} - \widehat{\chi}'\|_2^2 &= \sum_{j\in L'} \|(\widehat{X} - \widehat{\chi}')_{I_j}\|_2^2 + \sum_{j\in[\frac{n}{k_1}]\setminus L'} \|(\widehat{X} - \widehat{\chi}')_{I_j}\|_2^2 + \mathrm{err}(\widehat{\chi}_0) \\
&= \sum_{j\in L'} \|(\widehat{X} - \widehat{\chi} - W)_{I_j}\|_2^2 + \sum_{j\in[\frac{n}{k_1}]\setminus L'} \|(\widehat{X} - \widehat{\chi})_{I_j}\|_2^2 + \mathrm{err}(\widehat{\chi}_0) \\
&\le \epsilon\|\widehat{X} - \widehat{\chi}\|_2^2 + 300\epsilon\|\widehat{X} - \widehat{\chi}\|_2^2 + 6000\epsilon\nu^2 k_0 + \mathrm{Err}^2(\widehat{X} - \widehat{\chi}, 10k_0, k_1) + \epsilon\nu^2 \\
&\le (4\cdot 10^5)\epsilon\nu^2 k_0 + \mathrm{Err}^2(\widehat{X} - \widehat{\chi}, 10k_0, k_1),
\end{aligned}
$$

where the second line follows since $\widehat{\chi}' = \widehat{\chi} + W$ and W is non-zero only within the blocks indexed by L', the third line follows from (93) and (98), and the final line follows from the assumption $\|\widehat{X} - \widehat{\chi}\|_2^2 \le 100k_0\nu^2$ in the lemma.

Sample complexity: We condition on $\mathcal{E}_{\mathrm{loc}} \cap \mathcal{E}_{\mathrm{prune}} \cap \mathcal{E}_{\mathrm{est}}$ and calculate the number of samples used.

By Lemma 3.5 with $\delta = \epsilon^2$, the sample complexity of MultiBlockLocate is $O^*(|L|\cdot\log n + \frac{k_0k_1}{\epsilon^4})$ which behaves $O^*(\frac{k_0}{\epsilon^2}\log(1+k_0)\log n + \frac{k_0k_1}{\epsilon^4}\log\frac{1}{p})$ conditioned on $\mathcal{E}_{\mathrm{loc}}$ (see (91)).

By Lemma 5.1 with $\delta = \epsilon$, the sample complexity of PruneLocation is $O(\frac{k_0k_1}{\epsilon}\log\frac{1}{\epsilon p}\log\frac{1}{\epsilon})$. Moreover, by Lemma 5.2 with $\delta = \epsilon$, the sample complexity of EstimateValues is $O(\frac{k_0k_1}{\epsilon^2}\log\frac{1}{p}\log\frac{1}{\epsilon})$.

The claim follows by summing the three terms and noting from the choice of p in Algorithm 6 that, up to $\log\log\frac{k_0}{\epsilon}$ factors, we can replace p by ϵ in the above calculations.

Runtime: As in sample complexity analysis, we condition on $\mathcal{E}_{\mathrm{loc}} \cap \mathcal{E}_{\mathrm{prune}} \cap \mathcal{E}_{\mathrm{est}}$.

By Lemma 3.5 with $\delta = \epsilon^2$, the runtime of MultiBlockLocate is $O^*\big(|L|\cdot\log^2 n + \frac{k_0k_1}{\epsilon^4}\log^2 n + \frac{k_0k_1}{\epsilon^2}\log^3 n\big)$, which behaves as $O^*\big(\frac{k_0}{\epsilon^2}\log\frac{k_0}{\epsilon}\cdot\log^2 n + \frac{k_0k_1}{\epsilon^4}\log^2 n + \frac{k_0k_1}{\epsilon^2}\log^3 n\big)$ conditioned on $\mathcal{E}_{\mathrm{loc}}$ (see (90)).

By Lemma 5.1 with $\delta = \epsilon$, the runtime of PruneLocation is $O(\frac{k_0k_1}{\epsilon}\log\frac{1}{\epsilon p}\log\frac{1}{\epsilon}\log n + k_1\cdot|L|\log\frac{1}{\epsilon p})$, which behaves as $O(\frac{k_0k_1}{\epsilon}\log\frac{1}{\epsilon p}\log\frac{1}{\epsilon}\log n + k_1\cdot\frac{k_0}{\epsilon^2}\log\frac{k_0}{\epsilon}\log^4\frac{1}{\epsilon p})$ conditioned on $\mathcal{E}_{\mathrm{loc}}$ (see (91)).

By Lemma 5.2 with $\delta = \epsilon$, the runtime of EstimateValues is $O(\frac{k_0k_1}{\epsilon}\log\frac{1}{p}\log\frac{1}{\epsilon}\log n + k_1\cdot|L'|\log\frac{1}{p})$, which behaves as $O(\frac{k_0k_1}{\epsilon}\log\frac{1}{p}\log\frac{1}{\epsilon}\log n)$ conditioned on $\mathcal{E}_{\mathrm{prune}}$ (see (92) and recall that $|S_0| = 10k_0$).

The claim follows by summing the above terms and replacing p by ϵ, with the $\log\log n$, $\log\log\mathrm{SNR}'$ and $\log\frac{1}{\epsilon}$ terms absorbed into the $O^*(\cdot)$ notation.

E Discussion on Energy-Based Importance Sampling

Here we provide further discussing on the adaptive energy-based importance sampling scheme described in Sections 2–3. Recall from Definition 2.2 that given the signal X and filter G, we are considering downsampled signals of the form $\widehat{Z}_j^r = (\widehat{X}^r \star \widehat{G})_{jk_1}$ with $X_i^r = X_{i+\frac{nr}{2k_1}}$ for $r \in [2k_1]$, and recall from (3) that the goal of energy-based importance sampling is to approximately solve the covering problem

$$\text{Minimize}_{\{s^r\}} \quad \sum_{r\in[2k_1]} s^r \quad \text{subject to} \quad \sum_{\substack{j\,:\,|\widehat{Z}^r_j|^2 \ge \frac{\|\widehat{Z}^r\|_2^2}{s^r} \\ \text{for some } r\in[2k_1]}} \|\widehat{X}_{I_j}\|_2^2 \ge (1-\alpha)\|\widehat{X}^*\|_2^2 \tag{99}$$

for suitable $\alpha \in (0,1)$, where $\widehat{X}^*$ is the best $(k_0.k_1)$-block sparse approximation of $\widehat{X}$.

To ease the discussion, we assume throughout this appendix that the filter G is a width-$\frac{n}{k_1}$ rectangle in time domain, corresponding to a sinc pulse of "width" k_1 in frequency domain. Such a filter is less tight than the one we use (see the proof of Lemma 2.1), but similar enough for the purposes of the discussion.

E.1 Examples – Flat vs. Spiky Energies

We begin by providing two examples for the 1-block sparse case, demonstrating how the energies can vary with r. An illustration of the energy in each $\widehat{Z}^r$ is illustrated in Figure 2 in two different cases – one in which X is a sinc pulse (i.e., rectangular in frequency domain), and one in which X is constant (i.e., a delta function in frequency domain). Both of the signals are $(1,k_1)$-block sparse with $k_1 = 16$, and the signal energy is the same in both cases. However, the sinc pulse gives significantly greater variations in $|\widehat{Z}^r_j|^2$ as a function of r. In fact, these examples demonstrate two extremes that can occur – in one case, the energy exhibits no variations, and in the other case, the energy is $O(k_1)$ times its expected value for an $O\big(\frac{1}{k_1}\big)$ fraction of the r values.

The second example above is, of course, an extreme case of a $(1,k_1)$-block sparse signal, because it is also $(1,1)$-block sparse. Nevertheless, one also observes a similar flatness in time domain for other signals; e.g., one could take the first example above and randomize the signs, as opposed to letting them all be positive.

E.2 The $\log(1+k_0)$ factor

Here we provide an example demonstrating that, as long as we rely solely on frequencies being covered according to Definition 2.3, after performing the budget allocation, the extra $\log(1+k_0)$ factor in our analysis is unavoidable. Specifically, we argue that for a certain signal X, the *optimal* solution to (3) satisfies $\sum_{r\in[2k_1]} s^r = \Omega(k_0 \log(1+k_0))$. However, we do not claim that this $\log(1+k_0)$ factor is unavoidable for *arbitrary* sparse FFT algorithms.

We consider a scenario where $k_0 = \Theta(k_1) = o(n)$, and for concreteness, we let both k_0 and k_1 behave as $\Theta(n^{0.1})$; hence, $\log(1+k_0) = O(\log k_0)$

Constructing a base signal: We first specify a base signal $W \in \mathbb{C}^n$ that will be used to construct the approximately (k_0,k_1)-block sparse signal. Specifically, we fix the integers C and L,

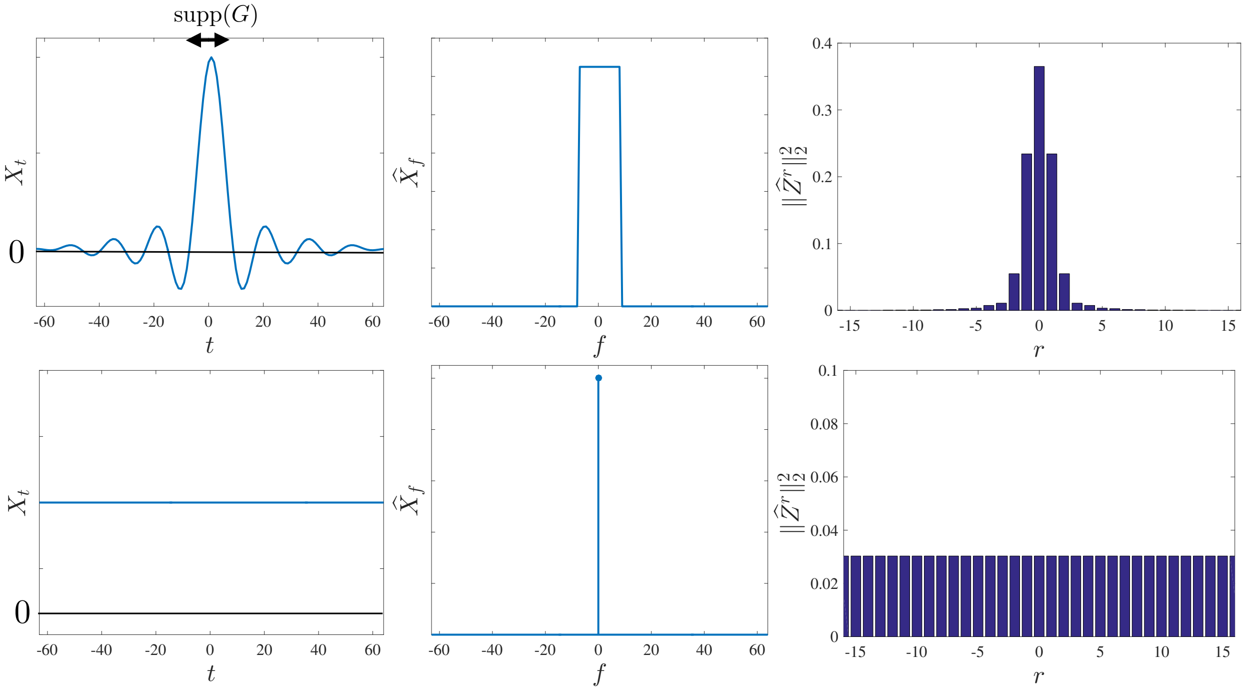

Figure 2: Behavior of $\|\widehat{Z}^r\|_2^2$ as a function of r for a sinc function (top) and a rectangular function (bottom), both of which are $(1, 16)$-block sparse.

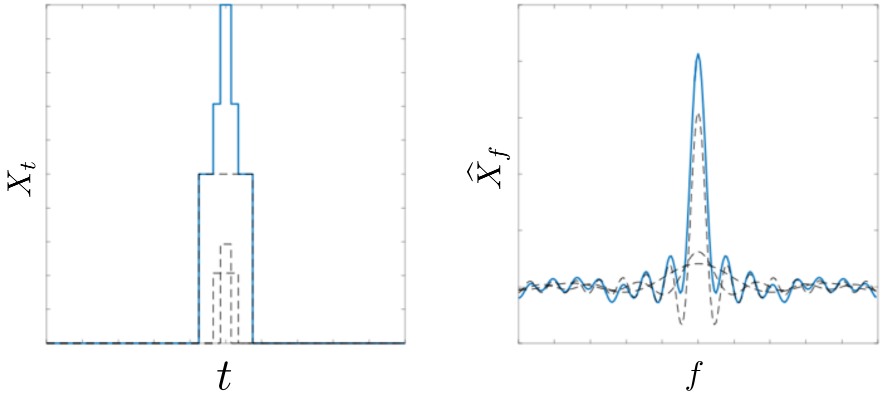

Figure 3: Base signal and its Fourier transform, for constructing a signal where a $\log k_0$ loss is unavoidable with our techniques.

and set

$$W_t = \begin{cases} \sqrt{2^{L-1}} & |t| \le \frac{Cn}{2k_1} \\ \sqrt{2^{L-2}} & \frac{Cn}{2k_1} < |t| \le \frac{3Cn}{2k_1} \\ \vdots & \vdots \\ \sqrt{2^{L-\ell}} & \frac{(2^{\ell-1}-1)Cn}{2k_1} < |t| \le \frac{(2^{\ell}-1)Cn}{2k_1} \\ \vdots & \vdots \\ 1 & \frac{(2^{L-1}-1)Cn}{2k_1} < |t| \le \frac{(2^{L}-1)Cn}{2k_1} \\ 0 & |f| > \frac{(2^{L+1}-1)Cn}{2k_1}. \end{cases} \tag{100}$$

Hence, the signal contains L regions of exponentially increasing width but exponentially decreasing magnitude. See Figure 3 for an illustration ($L = 3$), and observe that we can express this function as a sum of rectangles having geometrically decreasing magnitudes. Hence, we can specify its Fourier transform as a sum of sinc functions.

The narrowest of the rectangles has width $\frac{Cn}{k_1}$, and hence the widest of the sinc pulses has width $\frac{k_1}{C}$. This means that by choosing C to be sufficiently large, we can ensure that an arbitrarily high proportion of the energy lies in a window of length k_1 in frequency domain, meaning W is approximately 1-block sparse.

Constructing a block-spare signal: We construct a k_0-block sparse signal by adding multiple copies of W together, each shifted by a different amount in time domain, and also modulated by a different frequency (i.e., shifted by a different amount in frequency domain). We choose L such that the cases in (100) collectively occupy the whole time domain, yielding $L = \Theta(\log k_1) = \Theta(\log k_0)$.

Then, we set $k_0 = \frac{k_1}{C}$ and let each copy of W be shifted by a multiple of $\frac{Cn}{k_1}$, so that the copies are separated by a distance equal to the length of the thinnest segment of W, and collectively these thin segments cover the whole space $[n]$. As for the modulation, we choose these so that the resulting peaks in frequency domain are separated by $\Omega(k_1^4)$, so that there the tail of the copy of W corresponding to one block has a negligible effect on the other blocks. This is possible within n coefficients, since we have chosen $k_1 = O(n^{0.1})$.

Evaluating the values of $|Z_j^r|^2$ Recall that we are considering G in (99) equaling a rectangle of width $\frac{n}{k_1}$. Because of the above-mentioned separation of the blocks in frequency domain, each copy of W can essentially be treated separately. By construction, within a window of length $\frac{n}{k_1}$, we have one copy of W at magnitude $\sqrt{2^{L-1}}$, two copies at magnitude $\sqrt{2^{L-2}}$, and so on. Upon subsampling by a factor of k_1, the relative magnitudes remain the same; there is no aliasing, since we let G be rectangular. Hence, the dominant coefficients in the spectrum of the subsampled signal exhibit this same structure, having energies of a form such as $(8, 4, 4, 2, 2, 2, 2, 1, 1, 1, 1, 1, 1, 1, 1)$ when sorted and scaled (up to negligible leakage effects). Moreover, the matrix of $|\widehat{Z}_j^r|^2$ values (*cf.*, Figure 1) essentially amounts to circular shifts of a vector of this form – the structure of any given $\widehat{Z}^r$ maintains this geometric structure, but possibly in a different order.

Lower bounding the sum of budgets allocated: We now turn to the allocation problem in (99). Allocating a sparsity budget s to a signal Z^r covers all coefficients j for which $|\widehat{Z}_j^r|^2 \ge \frac{\|\widehat{Z}\|^2}{s}$. For the signal we have constructed, the total energy E is equally spread among the L geometric levels: The ℓ-th level consists of $2^{\ell-1}$ coefficients of energy $2^{1-\ell}\frac{E}{L}$, and hence covering that level requires $s \geq L \cdot 2^{\ell}$.

Hence, setting $s = L \cdot 2^{\ell-1}$ covers the top ℓ levels, for a total of $2^{\ell} - 1$ coefficients. That is, covering some number of coefficients requires letting s be $\Omega(L)$ times that number, and hence covering a constant fraction of the k_0 coefficients requires the sum of sparsity budgets to be $\Omega(Lk_0)$. Moreover, we have designed every block to have the same energy, so accounting for a constant fraction of the energy amounts to covering a constant fraction of the k_0 coefficients.

Since we selected $L = \Theta(\log k_0)$, this means that the sum of sparsity budgets is $\Omega(k_0 \log k_0)$, so that the $\log k_0$ factor must be present in any solution to (99).

F Location of Reduced Signals

In Algorithm 7, we provide a location primitive that, given a sequence of budgets s^r, locates dominant frequencies in the sequence of reduced signals Z^r using $O\big(\sum_{r\in[2k_1]} s^r \log n\big)$ samples. The core of the primitive is a simple k-sparse recovery scheme, where k frequencies are hashed into $B = Ck$ buckets for a large constant $C > 1$, and then each bucket is decoded individually. Specifically, for each bucket that is approximately 1-sparse (i.e., dominated by a single frequency that hashed into it) the algorithm accesses the signal at about a logarithmic number of locations and decodes the bit representation of the dominant frequency bit by bit. More precisely, to achieve the right sample complexity we decode the frequencies in blocks of $O(\log\log n)$ bits. Such schemes or versions thereof have been used in the literature (e.g., [10, 22, 24]).

A novel aspect of our decoding scheme is that it receives access to the input signal X, but must run a basic sparse recovery scheme as above on each reduced signal Z^r. Specifically, for each r it must hash Z^r into s^r buckets (the *budget* computed in MultiblockLocate and passed to LocateSignal as input). This would be trivial since Z^r can be easily accessed given access to X (*cf.*, Lemma 2.2), but the fact that we need to operate on the *residual signal* $X - \chi$ (where $\widehat{\chi}$ is block sparse and given explicitly as input) introduces difficulties.

The difficulty is that we would like to compute χ on the samples that individual invocations of sparse recovery use, for each $r \in [2k_1]$, but computing this directly would be very costly. Our solution consists of ensuring that all invocations of sparse recovery use the same random permutation π, and therefore all need to access $X - \chi$ on a set of shifted intervals after a change of variables given by π (crucially, this change of variables is shared across all r). The lengths of the intervals are different, and given by s^r, but it suffices to compute the values of χ on the shifts of the largest of these intervals, which is done in HashToBinsReduced (see Lemma 4.6). We present the details below in Algorithm 7.

For convenience, throughout this section, we use m to denote the reduced signal length n/k_1.

Algorithm 7 Location primitive: Given access to the input signal X, a partially recovered signal $\widehat{\chi}$, a budget k and bound on failure probability p, recovers any given $j \in [n/k_1]$ with $|\widehat{Z}_j^r|^2 \geq \|\widehat{Z}^r\|_2^2/k$ for some $r \in [2k_1]$, in the (k_1, δ)-downsampling of $X - \chi$.

```
procedure LocateReducedSignals($X, \widehat{\chi}, n, k_0, k_1, \{s^r\}_{r\in[2k_1]}, \delta, p$)
                                        ▷ Uses large absolute constants $C_1, C_2, C_3 > 0$
    $B^r \leftarrow C_2 s^r$ for each $r$ $in[2k_1]$
    $H^r \leftarrow (m, B, F')$-flat filter for each $r \in [2k_1]$, for sufficiently large $F' \geq 2$
    $B_{\max} \leftarrow B^r$
    $\{Z_X^r\}_{r\in[2k_1]} \leftarrow (k_1, \delta)$-downsampling of $X$          ▷ See Definition 2.2
    $m \leftarrow n/k_1$
    $L \leftarrow \emptyset$
    for $t = \{1, \ldots, C_1 \log(2/p)\}$ do
        $\sigma \leftarrow$ uniformly random odd integer in $[m]$
        $\mathcal{A} \leftarrow C_3 \log\log m$ uniformly random elements in $[m] \times [m]$
        $\Lambda \leftarrow 2^{\lfloor \frac{1}{2} \log_2 \log_2 m \rfloor}$, $N \leftarrow \Lambda^{\lceil \log_\Lambda m \rceil}$ ▷ Implicitly extend $X$ to an $m$-periodic length-$N$ signal
        for $(\alpha, \beta) \in \mathcal{A}$ do
            $\mathbf{w} \leftarrow N\Lambda^{-g}$
            $\Delta \leftarrow \alpha + \mathbf{w} \cdot \beta$
            $\mathbf{H} \leftarrow \{H^r\}_{r\in[2k_1]}$
            $\mathbf{B} \leftarrow \{B^r\}_{r\in[2k_1]}$
            $\widehat{U}^r(\alpha + \mathbf{w} \cdot \beta) \leftarrow$ HashToBinsReduced($\{Z_X^r\}_{r\in[2k_1]}, \widehat{\chi}, \mathbf{H}, n, k_1, \mathbf{B}, \sigma, \Delta$)
        for $r \in [2k_1]$ do
            $B \leftarrow C_2 s^r$                               ▷ Rounded up to a power of two
            for $b \in [B]$ do                                 ▷ Loop over all hash buckets
                $\mathbf{f} \leftarrow \mathbf{0}$
                for $g = \{1, \ldots, \log_\Lambda N\}$ do
                    $\mathbf{w} \leftarrow N\Lambda^{-g}$
                    If there exists a unique $\lambda \in \{0, 1, \ldots, \Lambda - 1\}$ such that
                        $\left|\omega_\Lambda^{-\lambda\cdot\beta} \cdot \omega^{-(N\cdot\Lambda^{-g}\mathbf{f})\cdot\beta} \cdot \frac{\widehat{U}_b^r(\alpha+\mathbf{w}\cdot\beta)}{\widehat{U}_b^r(\alpha)} - 1\right| < \frac{1}{3}$ for at least $\frac{3}{5}$ fraction of $(\alpha, \beta) \in \mathcal{A}$
                    then $\mathbf{f} \leftarrow \mathbf{f} + \Lambda^{g-1} \cdot \lambda$
                $L \leftarrow L \cup \{\sigma^{-1}\mathbf{f} \cdot \frac{m}{N}\}$          ▷ Add recovered element to output list
    return $L$
```

Lemma 2.4 (LocateReducedSignal guarantees – formal version) *Fix (n, k_0, k_1), the signals $X, \widehat{\chi} \in \mathbb{C}^n$ with $\widehat{\chi}_0$ uniformly distributed over an arbitrarily length-$\frac{\|\widehat{\chi}\|^2}{\mathrm{poly}(n)}$ interval, the sparsity budgets $\{s^r\}_{r\in[2k_1]}$ with $s^r = O\big(\frac{k_0}{\delta}\big)$ for all r, and the parameters $\delta \in \big(\frac{1}{n}, \frac{1}{20}\big)$ and $p \in \big(\frac{1}{n^3}, \frac{1}{2}\big)$, and let $\{Z^r\}_{r\in[2k_1]}$ be the (k_1, δ)-downsampling of $X - \chi$.*

If L denotes the output of LocateReducedSignals*$(X, \widehat{\chi}, n, k_0, k_1, \{s^r\}_{r\in[2k_1]}, \delta, p)$, then for any $j \in \big[\frac{n}{k_1}\big]$ such that $|Z_j^r|^2 \geq \|Z^r\|_2^2/s^r$ for some $r \in [2k_1]$, one has $j \in L$ with probability at least $1 - p$. The list size satisfies $|L| = O\big(\sum_{r\in[2k_1]} s^r \log \frac{1}{\delta}\big)$. Moreover, if $\widehat{\chi}$ is $(O(k_0), k_1)$-block sparse, the sample complexity is $O\big(\sum_{r\in[2k_1]} s^r \log \frac{1}{p} \log \frac{1}{\delta} \log n\big)$, and the runtime is $O\big(\sum_{r\in[2k_1]} s^r \log \frac{1}{p} \log \frac{1}{\delta} \log^2 n + \frac{k_0 k_1}{\delta} \log \frac{1}{p} \log^3 n\big)$.*

Proof. We first note that the claim on the list size follows immediately from the fact that $B = O(s^r)$

entries are added to the list for each t and r, and the loop over t is of length $O\big(\log \frac{1}{p}\big)$.

In order to prove the main claim of the lemma, it suffices to show that for any single value of r, if we replace the loop over r by that single value, then L contains any given $j \in \big[\frac{n}{k_1}\big]$ such that $|Z^r_j|^2 \geq \|Z^r\|_2^2/s^r$, with probability at least $1-p$. Since this essentially corresponds to a standard sparse recovery problem, we switch to simpler notation throughout the proof: We let Y denote a generic signal Z^r, we write its length as $m = n/k_1$, we index its entries in frequency domain as $\widehat{Y}_f$, and we define $k = s^r$.

The proof now consists of two steps. First, we show correctness of the location algorithm assuming that the SemiEquiInverseBlockFFT computation in line 11 computes all the required values for the computation of $\widehat{U}$ in line 19 (**step 1** below). We then prove that SemiEquiInverseBlockFFT indeed computes all the required values of χ, and conclude with sample complexity and runtime bounds (**step 2** below).

Proving correctness of the location process We show that each element f with $|\widehat{Y}_f|^2 \geq \|\widehat{Y}\|_2^2/k$ is reported in a given iteration of the outer loop over $t = 1, \ldots, C_1 \log(2/p)$, with probability at least 9/10. Since the loops use independent randomness, the probability of f not being reported in any of the iterations is bounded by $(1/10)^{C_1 \log(2/p)} \leq p/2$ if C_1 is sufficiently large.

Fix an iteration t. We first show that the random set $\mathcal{A}$ chosen in LocateReducedSignals has useful error-correcting properties with high probability. Specifically, we let $\mathcal{E}_{\text{balanced}}$ denote the event that for every $\lambda \in [\Lambda], \lambda \neq 0$ at least a fraction 49/100 of the numbers $\{\omega_\Lambda^{\lambda \cdot \beta}\}_{(\alpha,\beta)\in\mathcal{A}}$ have non-positive real part; in that case, we say that $\mathcal{A}$ is *balanced.* We have for fixed $\lambda \in [\Lambda], \lambda \neq 0$ that since the pair (α, β) was chosen uniformly at random from $[m] \times [m]$, the quantity $\omega_\Lambda^{\lambda\cdot\beta}$ is uniformly distributed on the set of roots of unity of order 2^s for some $s > 0$ (since $\lambda \neq 0$). At least half of these roots have non-positive real part, so for every fixed $\lambda \in [\Lambda], \lambda \neq 0$ one has $\Pr_\beta[\mathrm{Re}(\omega_\Lambda^{\lambda\cdot\beta}) \leq 0] \geq 1/2$. It thus follows by standard concentration inequalities that for every fixed λ at least 49/100 of the numbers $\{\omega_\Lambda^{\lambda\cdot\beta}\}_{(\alpha,\beta)\in\mathcal{A}}$ have non-positive real part with probability at least $1 - e^{-\Omega(|\mathcal{A}|)} = 1 - \exp(-\Omega(C_3 \log\log m))) \geq 1 - 1/(100 \log_2 m)$ as long as C_3 is larger than an absolute constant. A union bound over $\Lambda \leq \log_2 m$ values of λ shows that $\Pr[\mathcal{E}_{\text{balanced}}] \geq 1 - (\log_2 m) \cdot /(100 \log_2 m) = 1 - 1/100$ for sufficiently large m (recall from Section 1 that $\frac{n}{k_1}$ exceeds a large absolute constant by assumption). We henceforth condition on $\mathcal{E}_{\text{balanced}}$.

Fix any f such that $|\widehat{Y}_f|^2 \geq ||\widehat{Y}||_2^2/k$, and let $q = \sigma i$ for convenience. We show by induction on $g = 1, \ldots, \log_\Lambda N$ that before the g-th iteration of lines 24–27 of Algorithm 7, we have that $\mathbf{f}$ coincides with $\mathbf{q}$ on the bottom $g \cdot \log_2 \Lambda$ bits, i.e., $\mathbf{f} - \mathbf{q} = 0 \mod \Lambda^{g-1}$.

The **base** of the induction is trivial and is provided by $g = 1$. We now show the **inductive step**. Assume by the inductive hypothesis that $\mathbf{f} - \mathbf{q} = 0 \mod \Lambda^{g-1}$, so that $\mathbf{q} = \mathbf{f} + \Lambda^{g-1}(\lambda_0 + \Lambda\lambda_1 + \Lambda^2\lambda_2 + \ldots)$. Thus, $(\lambda_0, \lambda_1, \ldots)$ is the expansion of $(\mathbf{q} - \mathbf{f})/\Lambda^{g-1}$ base Λ, and λ_0 is the least significant digit. We now show that λ_0 is the unique value of λ that satisfies the condition of line 25 of Algorithm 7, with high probability

In the following, we use the definitions of $\pi(f)$, $h(f)$, and $o_f(f')$ from Definition 4.2 with $\Delta = 0$.

First, we have for each $a = (\alpha, \beta) \in \mathcal{A}$ and $\mathbf{w} \in \mathbf{W}$ that

$$\begin{aligned}\widehat{H}^{-1}_{o_f(f)}\widehat{U}_{h(f)}(\alpha + \mathbf{w}\cdot\beta) - \widehat{Y}_f\omega_N^{(\alpha+\mathbf{w}\cdot\beta)\mathbf{q}} &= \widehat{H}^{-1}_{o_f(f)}\widehat{U}^*_{h(f)}(\alpha + \mathbf{w}\cdot\beta) - \widehat{Y}_f\omega_N^{(\alpha+\mathbf{w}\cdot\beta)\mathbf{q}} + \mathrm{err}_w \\ &= \widehat{H}^{-1}_{o_f(f)} \sum_{f'\in[m]\setminus\{f\}} \widehat{H}_{o_f(f')}\widehat{Y}_{f'}\omega_N^{\sigma f'\cdot(\alpha+\mathbf{w}\cdot\beta)} + \mathrm{err}_w =: E'(\mathbf{w}),\end{aligned}$$

where $\mathrm{err}_w = \widehat{H}^{-1}_{o_f(f)}(\widehat{U}_{h(f)} - \widehat{U}^*_{h(f)})(\alpha + \mathbf{w}\cdot\beta)$.

And similarly

$$\begin{aligned}\widehat{H}^{-1}_{o_f(f)}\widehat{U}_{h(f)}(\alpha) - \widehat{Y}_f\omega_N^{\alpha\mathbf{q}} &= \widehat{H}^{-1}_{o_f(f)}\widehat{U}^*_{h(f)}(\alpha) - \widehat{Y}_f\omega_N^{\alpha\mathbf{q}} + \mathrm{err} \\ &= \widehat{H}^{-1}_{o_f(f)} \sum_{f'\in[m]\setminus\{f\}} \widehat{H}_{o_f(f')}\widehat{Y}'_{f'}\omega_N^{\sigma f'\cdot\alpha} + \mathrm{err} =: E''.\end{aligned}$$

where $\mathrm{err} = \widehat{H}^{-1}_{o_f(f)}(\widehat{U}_{h(f)} - \widehat{U}^*_{h(f)})(\alpha)$.

We will show that f is recovered from bucket $h(f)$ with high (constant) probability. The bounds above imply that

$$\frac{\widehat{U}_{h(f)}(\alpha + \mathbf{w}\beta))}{\widehat{U}_{h(f)}(\alpha)} = \frac{\widehat{Y}_f\omega_N^{(\alpha+\mathbf{w}\beta)\mathbf{q}} + E'(\mathbf{w})}{\widehat{Y}_f\omega_N^{\alpha\mathbf{q}} + E''}. \tag{101}$$

The rest of the proof consists of two parts. We first show that with high probability over the choice of π, the error terms $E'(\mathbf{w})$ and E'' are small in absolute value for most $a = (\alpha, \beta) \in \mathcal{A}$ with extremely high probability. We then use this assumption to argue that f is recovered.

Bounding the error terms $E'(\mathbf{w})$ and E'' (part (i)). We have by Parseval's theorem that

$$\mathbb{E}_a[|E'(\mathbf{w})|^2] \leq \widehat{H}^{-2}_{o_f(f)} \sum_{f'\in[m]\setminus\{f\}} \widehat{H}^2_{o_f(f')}|Y_{f'}|^2 + |\mathrm{err}_w|^2 + 2|\mathrm{err}_w|\widehat{H}^{-1}_{o_f(f)} \sum_{f'\in[m]\setminus\{f\}} \widehat{H}_{o_f(f')}|Y_{f'}|, \tag{102}$$

and

$$\mathbb{E}_a[|E''|^2] \leq \widehat{H}^{-2}_{o_f(f)} \sum_{f'\in[m]\setminus\{f\}} \widehat{H}^2_{o_f(f')}|\widehat{Y}_{f'}|^2 + |\mathrm{err}|^2 + 2|\mathrm{err}|\widehat{H}^{-1}_{o_f(f)} \sum_{f'\in[m]\setminus\{f\}} \widehat{H}_{o_f(f')}|\widehat{Y}_{f'}|,$$

where we used the fact that $\alpha + \mathbf{w}\beta$ is uniformly random in $[m]$ (due to α being uniformly random in $[m]$ and independent of β by definition of $\mathcal{A}$ in line 6 of Algorithm 7).

Taking the expectation of the term $\widehat{H}^{-2}_{o_f(f)} \sum_{f'\in[m]\setminus\{f\}} \widehat{H}^2_{o_f(f')}|Y_{f'}|^2$ with respect to π, we obtain

$$\mathbb{E}_\pi\left[\widehat{H}^{-2}_{o_f(f)} \sum_{f'\in[m]\setminus\{f\}} \widehat{H}^2_{o_f(f')}|Y_{f'}|^2\right] = O(\|Y\|_2^2/B) = O(||X'||_2^2/(C_2 k))$$

by Lemma 4.3 (note that $F' \geq 2$, so the lemma applies) and the choice $B = C_2 \cdot k$ (line 20 of Algorithm 7). We thus have by Markov's inequality together with the assumption that $|\widehat{Y}_f|^2 \geq ||\widehat{Y}||_2^2/k$ that

$$\mathrm{Pr}_\pi\left[\widehat{H}^{-2}_{o_f(f)} \sum_{f'\in[m]\setminus\{f\}} \widehat{H}^2_{o_f(f')}|\widehat{Y}_{f'}|^2 > |\widehat{Y}_f|^2/1700\right] < O(1/C_2) < 1/40$$

and

$$\mathrm{Pr}_\pi\left[\widehat{H}^{-2}_{o_f(f)} \sum_{f'\in[m]\setminus\{f\}} \widehat{H}^2_{o_f(f')}|\widehat{Y}_{f'}|^2 > |\widehat{Y}_f|^2/1700\right] < O(1/C_2) < 1/40$$

since C_2 is larger than an absolute constant by assumption.

Bounding err **and** err_w **(numerical errors from semi-equispaced FFT computation):** Recall that we have by assumption that $\widehat{\chi}_0$ uniformly distributed over an arbitrarily length-$\frac{\|\widehat{\chi}\|^2}{\mathrm{poly}(n)}$ interval, and that $\widehat{Y} = \widehat{Z}^r$ for some $\widehat{Z}^r$ in the (k_1, δ)-downsampling of $X - \chi$. By decomposing $\widehat{Z}^r_j = ((\widehat{X}^r - \widehat{\chi}^r) \star \widehat{G})_{jk_1}$ into a deterministic part and a random part (in terms of the above-mentioned uniform distribution), we readily obtain for some $c' > 0$ that

$$\|\widehat{Y}\|^2 \geq \frac{\|\widehat{\chi}\|_2^2}{n^{c'}} \tag{103}$$

with probability at least $1 - \frac{1}{n^4}$. Since $p \geq \frac{1}{n^3}$ by assumption, we deduce that this also holds with probability at least $1 - p/2$. By the the accuracy of the $\widehat{\chi}$ values stated in Lemma 4.5, along with the argument used in (70)–(71) its proof in Appendix C.3 to convert (103) to accuracy on hashed values, we know that $|\widehat{U}_{h(f)} - \widehat{U}^*_{h(f)}| \leq \|\widehat{U} - \widehat{U}^*\|_\infty \leq n^{-c+1}\|\widehat{\chi}\|_2$. Hence, by using $|\widehat{H}_{o_f(f)}|^{-2} \leq 2$ and $\alpha, \alpha + \mathbf{w} \cdot \beta \leq m$, we find that

$$|\mathrm{err}| \leq 2n^{-c+1}\|\widehat{\chi}\|_2 \leq 2n^{-c+c'+1}\|\widehat{Y}\|_2$$
$$|\mathrm{err}_w| \leq 2n^{-c+1}\|\widehat{\chi}\|_2 \leq 2n^{-c+c'+1}\|\widehat{Y}\|_2,$$

where the second inequality in each equation holds for some $c' > 0$ by (103).

Note also that $\widehat{H}^{-1}_{o_f(f)} \sum_{f' \in [m] \setminus \{f\}} \widehat{H}_{o_f(f')}|\widehat{Y}_{f'}| \leq 2\|\widehat{Y}\|_1 \leq 2\sqrt{m}\|\widehat{Y}\|_2$, since we have $\widehat{H}^{-1}_{o_f(f)} \leq 2$ and $\widehat{H}_{f'}| \leq 1$ for all f'. We can thus write

$$\begin{aligned} |\mathrm{err}|^2 + 2|\mathrm{err}| \cdot \widehat{H}^{-1}_{o_f(f)} \sum_{f' \in [m] \setminus \{f\}} \widehat{H}_{o_f(f')}|\widehat{Y}_{f'}| &\leq |\mathrm{err}|^2 + 4\sqrt{m}|\mathrm{err}| \cdot \|\widehat{Y}\|_2 \\ &\leq 4n^{-2c+2c'+2}\|\widehat{Y}\|_2^2 + 8n^{-c+c'+3/2}\|\widehat{Y}\|_2^2 \\ &= n^{\Omega(-c+c')}\|\widehat{Y}\|_2^2, \end{aligned} \tag{104}$$

which can thus be made to behave as $\frac{1}{\mathrm{poly}(n)}\|\widehat{Y}\|_2^2$ by a suitable choice of c.

Bounding the error terms $E'(\mathbf{w})$ and E'' (part (ii)). By the union bound, we have $|E'(\mathbf{w})|^2 \leq |\widehat{Y}_f|^2/1600$ and $|E''|^2 \leq |\widehat{Y}_f|^2/1600$ simultaneously with probability at least $1 - 1/20$ – denote the success event by $\mathcal{E}^t_{f,\pi}(\mathbf{w})$. Conditioned on $\mathcal{E}^t_{f,\pi}(\mathbf{w})$, we thus have by (102) and (104), along with the fact that $\mathcal{A}$ is independent of π, that

$$\mathbb{E}_a[|E'(\mathbf{w})|^2|] \leq |Y_f|^2/1600 \quad \text{and} \quad \mathbb{E}_a[|E''|^2] \leq |Y_f|^2/1600.$$

Another application of Markov's inequality gives

$$\mathrm{Pr}_a[|E'(\mathbf{w})|^2 \geq |\widehat{Y}_f|^2/40] \leq 1/40 \quad \text{and} \quad \mathrm{Pr}_a[|E''|^2 \geq |\widehat{Y}_f|^2/40] \leq 1/40.$$

This means that conditioned on $\mathcal{E}^t_{f,\pi}(\mathbf{w})$, with probability at least $1 - e^{-\Omega(|\mathcal{A}|)} \geq 1 - 1/(100\log_2 m)$ over the choice of $\mathcal{A}$, both events occur for all but $2/5$ fraction of $a \in \mathcal{A}$; denote this success event by $\mathcal{E}^t_{f,\mathcal{A}}(\mathbf{w})$. We condition on this event in what follows. Let $\mathcal{A}^*(\mathbf{w}) \subseteq \mathcal{A}$ denote the set of values of $a \in \mathcal{A}$ that satisfy the bounds above.

In particular, we can rewrite (101) as

$$\begin{aligned}\frac{\widehat{U}_{h(f)}(\alpha+\mathbf{w}\beta)}{\widehat{U}_{h(f)}(\alpha)} &= \frac{\widehat{Y}_f\omega_N^{(\alpha+\mathbf{w}\beta)\mathbf{q}}+E'(\mathbf{w})}{\widehat{Y}_f\omega_N^{\alpha\mathbf{q}}+E''} \\ &= \frac{\omega_N^{(\alpha+\mathbf{w}\beta)\mathbf{q}}}{\omega_N^{\alpha\mathbf{q}}}\cdot\xi \quad \left(\text{where } \xi = \frac{1+\omega^{-(\alpha+\mathbf{w}\beta)\mathbf{q}}E'(\mathbf{w})/\widehat{Y}'_f}{1+\omega_N^{-\alpha\mathbf{q}}E''/\widehat{Y}'_f}\right) \\ &= \omega_N^{(\alpha+\mathbf{w}\beta)\mathbf{q}-\alpha\mathbf{q}}\cdot\xi \\ &= \omega_N^{\mathbf{w}\beta\mathbf{q}}\cdot\xi.\end{aligned}$$

We thus have for $a \in \mathcal{A}^*(\mathbf{w})$ that

$$|E'(\mathbf{w})|/|\widehat{Y}'_f| \leq 1/40 \quad \text{and} \quad |E''|/|\widehat{Y}'_f| \leq 1/40. \tag{105}$$

Showing that $\mathcal{A}^*(\mathbf{w}) \subseteq \mathcal{A}$ suffices for recovery. By the above calculations, we get

$$\frac{\widehat{U}_{h(f)}(\alpha+\mathbf{w}\beta)}{\widehat{U}_{h(f)}(\alpha)} = \omega_N^{\mathbf{w}\beta\mathbf{q}}\cdot\xi = \omega_N^{N\Lambda^{-g}\beta\mathbf{q}}\cdot\xi = \omega_N^{N\Lambda^{-g}\beta\mathbf{q}} + \omega_N^{N\Lambda^{-g}\beta\mathbf{q}}(\xi-1).$$

We proceed by analyzing the first term, and we will later show that the second term is small. Since $\mathbf{q} = \mathbf{f} + \Lambda^{g-1}(\lambda_0 + \Lambda\lambda_1 + \Lambda^2\lambda_2 + \ldots)$, by the inductive hypothesis, we have

$$\begin{aligned}\omega_\Lambda^{-\lambda\cdot\beta}\cdot\omega_N^{-N\Lambda^{-g}\mathbf{f}\cdot\beta}\cdot\omega_N^{N\Lambda^{-g}\beta\mathbf{q}} &= \omega_\Lambda^{-\lambda\cdot\beta}\cdot\omega_N^{N\Lambda^{-g}(\mathbf{q}-\mathbf{f})\cdot\beta} \\ &= \omega_\Lambda^{-\lambda\cdot\beta}\cdot\omega_N^{N\Lambda^{-g}(\Lambda^{g-1}(\lambda_0+\Lambda\lambda_1+\Lambda^2\lambda_2+\ldots))\cdot\beta} \\ &= \omega_\Lambda^{-\lambda\cdot\beta}\cdot\omega_N^{(N/\Lambda)\cdot(\lambda_0+\Lambda\lambda_1+\Lambda^2\lambda_2+\ldots)\cdot\beta} \\ &= \omega_\Lambda^{-\lambda\cdot\beta}\cdot\omega_\Lambda^{\lambda_0\cdot\beta} \\ &= \omega_\Lambda^{(-\lambda+\lambda_0)\cdot\beta},\end{aligned}$$

where we used the fact that $\omega_N^{N/\Lambda} = e^{2\pi f(N/\Lambda)/N} = e^{2\pi f/\Lambda} = \omega_\Lambda$. Thus, we have

$$\omega_\Lambda^{-\lambda\cdot\beta}\omega^{-(N\Lambda^{-g}\mathbf{f})\cdot\beta}\frac{\widehat{U}_{h(f)}(\alpha+\mathbf{w}\beta)}{\widehat{U}_{h(f)}(\alpha)} = \omega_\Lambda^{(-\lambda+\lambda_0)\cdot\beta}\xi.$$

We now consider two cases. First suppose that $\lambda = \lambda_0$. Then $\omega_\Lambda^{(-\lambda+\lambda_0)\cdot\beta} = 1$, and it remains to note that by (105) we have $|\xi - 1| \leq \frac{1+1/40}{1-1/40} - 1 < 1/3$. Thus, every $a \in \mathcal{A}^*(\mathbf{w})$ passes the test in line 25 of Algorithm 7. Since $|\mathcal{A}^*(\mathbf{w})| \geq (3/5)|\mathcal{A}|$ by the argument above, we have that λ_0 passes the test in line 25. It remains to show that λ_0 is the unique element in $0, \ldots, \Lambda - 1$ that passes this test.

Suppose that $\lambda \neq \lambda_0$. Then, by conditioning on $\mathcal{E}_{\text{balanced}}$, at least a 49/100 fraction of $\omega_\Lambda^{(-\lambda+\lambda_0)\cdot\beta}$ have negative real part. This means that for at least 49/100 of $a \in \mathcal{A}$, we have

$$|\omega_\Lambda^{(-\lambda+\lambda_0)\cdot\beta}\xi - 1| \geq |\mathbf{i}\cdot|\xi| - 1| \geq |(7/9)\mathbf{i} - 1| > 1/3,$$

and hence the condition in line 16 of Algorithm 7 is not satisfied for any $\lambda \neq \lambda_0$.

We thus get that conditioned on $\mathcal{E}_{\text{balanced}}$ and the intersection of $\mathcal{E}^t_{f,\pi}(\mathbf{w})$ for all $\mathbf{w} \in \mathbf{W}$ and $\mathcal{E}^t_{f,\mathcal{A}}$, recovery succeeds for all values of $g = 1, \ldots, \log_\Lambda N$. By a union bound over the failure events, we get that

$$\Pr\left[\mathcal{E}_{\text{balanced}} \cap \mathcal{E}_{f,\mathcal{A}}^t \cap \bigcap_{\mathbf{w}\in\mathbf{W}} \mathcal{E}_{f,\pi}^t(\mathbf{w})\right] \geq 1 - 1/100 - (\log_\Lambda N) \cdot \frac{1}{100 \log_2 m} \geq 98/100.$$

This shows that location is successful for f in a single iteration t with probability at least $98/100 \geq 9/10$, as required.

Sample complexity and runtime. We first consider the calls to HashToBinsReduced. This is called for $\log\log m$ values of (α,β) and $\log_\Lambda N = O\big(\frac{\log m}{\log\log m}\big)$ values of g in each iteration, the product of which is $O(\log m) = O(\log n)$. Moreover, the number of iterations is $O\big(\log\frac{1}{p}\big)$. Hence, using Lemma 4.6, we find that the combination of all of these calls costs $O\big(F\sum_{r\in[2k_1]} B^r \log\frac{1}{\delta}\log n\big)$ samples, with a runtime of $O\big((B_{\max}F + k_0)k_1\log^3 n\big)$, where $B_{\max} = O(\max_r s^r)$, and k_0 is such that $\widehat{\chi}$ is $(O(k_0),k_1)$-block sparse. By the assumption $\max_r s^r = O\big(\frac{k_0}{\delta}\big)$, the runtime simplifies to $O\big(\frac{k_0k_1}{\delta}\log^3 n\big)$

□

G Pruning the Location List

Algorithm 8 Prune a location list via hashing and thresholding techniques.

```
procedure PruneLocation($X, \widehat{\chi}, L, n, k_0, k_1, \delta, p, \theta$)
    $B \leftarrow 160\frac{k_0k_1}{\delta}$
    $F \leftarrow 10\log\frac{1}{\delta}$
    $G \leftarrow (n,B,F)$-flat filter
    $T \leftarrow 10\log\frac{1}{\delta p}$
    for $t \in \{1,\dots,T\}$ do
        $\Delta \leftarrow$ uniform random sample from $[\frac{n}{k_1}]$
        $\sigma \leftarrow$ uniform random sample from odd numbers in $[\frac{n}{k_1}]$
        $\widehat{U} \leftarrow$ HashToBins($X, \widehat{\chi}, G, n, B, \sigma, \Delta$)
        $W_j^{(t)} \leftarrow \sum_{f\in I_j} \big|\widehat{G}_{o_f(f)}^{-1}\widehat{U}_{h(f)}\omega_n^{-\sigma\Delta f}\big|^2$ for all $j \in L$        ▷ $h(f), o_f(f)$ in Definition 4.2
    $W_j \leftarrow \text{Median}_t(W_j^{(t)})$ for all $j \in L$
    $L' \leftarrow \{j \in L : W_j \geq \theta\}$
    return $L'$
```

The pruning procedure is given in Algorithm 8. Its goal is essentially to reduce the size of the list returned by MultiBlockLocate (*cf.*, Algorithm 1) from $O(k_0\log(1+k_0))$ to $O(k_0)$. More formally, the following lemma shows that with high probability, the pruning algorithm retains most of the energy in the head elements, while removing most tail elements.

Lemma 5.1 (PruneLocation guarantees – re-stated from Section 5.1) *Given (n,k_0,k_1), the parameters $\theta > 0$, $\delta \in \big(\frac{1}{n},\frac{1}{20}\big)$ and $p \in (0,1)$, and the signals $X \in \mathbb{C}^n$ and $\widehat{\chi} \in \mathbb{C}^n$ with $\|\widehat{X} - \widehat{\chi}\|_2 \geq \frac{1}{\text{poly}(n)}\|\widehat{\chi}\|_2$, the output L' of* PruneLocation$(X, \widehat{\chi}, L, k_0, k_1, \delta, p, n, \theta)$ *has the following properties:*

a. *Let S_{tail} denote the tail elements in signal $\widehat{X} - \widehat{\chi}$, defined as*

$$S_{\text{tail}} = \Big\{ j \in \Big[\frac{n}{k_1}\Big] \;:\; \|(\widehat{X} - \widehat{\chi})_{I_j}\|_2 \le \sqrt{\theta} - \sqrt{\frac{\delta}{k_0}}\|\widehat{X} - \widehat{\chi}\|_2 \Big\},$$

where I_j is defined in Definition 1.1. Then, we have

$$\mathbb{E}\Big[\big|L' \cap S_{\text{tail}}\big|\Big] \le \delta p \cdot |L \cap S_{\text{tail}}|.$$

b. *Let S_{head} denote the head elements in signal $\widehat{X} - \widehat{\chi}$, defined as*

$$S_{\text{head}} = \Big\{ j \in \Big[\frac{n}{k_1}\Big] \;:\; \|(\widehat{X} - \widehat{\chi})_{I_j}\|_2 \ge \sqrt{\theta} + \sqrt{\frac{\delta}{k_0}}\|\widehat{X} - \widehat{\chi}\|_2 \Big\}.$$

Then, we have

$$\mathbb{E}\Big[\sum_{j \in (L \cap S_{\text{head}}) \setminus L'} \|(\widehat{X} - \widehat{\chi})_{I_j}\|_2^2 \Big] \le \delta p \sum_{j \in L \cap S_{\text{head}}} \|(\widehat{X} - \widehat{\chi})_{I_j}\|_2^2.$$

Moreover, provided that $\|\widehat{\chi}\|_0 = O(k_0 k_1)$, the sample complexity is $O(\frac{k_0 k_1}{\delta} \log \frac{1}{\delta p} \log \frac{1}{\delta})$, and the runtime is $O(\frac{k_0 k_1}{\delta} \log \frac{1}{\delta p} \log \frac{1}{\delta} \log n + k_1 \cdot |L| \log \frac{1}{\delta p})$.

Proof. We begin by analyzing the properties of the random variables W_j used in the threshold test. We define $X' = X - \chi$, let $\widehat{U}$ be the output of HashToBins, and let $\widehat{U}^*$ be its exact counterpart as defined in Lemma 4.6. It follows that we can write the random variable $W_j^{(t)}$ (*cf.*, Algorithm 8) as

$$\begin{aligned} W_j^{(t)} &= \sum_{f \in I_j} \Big| \widehat{G}^{-1}_{o_f(f)} \widehat{U}_{h(f)} \omega_n^{-\sigma \Delta f} \Big|^2 \\ &= \sum_{f \in I_j} \Big| \widehat{G}^{-1}_{o_f(f)} \widehat{U}^*_{h(f)} \omega_n^{-\sigma \Delta f} + \widehat{G}^{-1}_{o_f(f)} (\widehat{U}_{h(f)} - \widehat{U}^*_{h(f)}) \omega_n^{-\sigma \Delta f} \Big|^2 \\ &= \sum_{f \in I_j} \Big| \widehat{X}'_f + \mathrm{err}_f^{(t)} + \widetilde{\mathrm{err}}_f^{(t)} \Big|^2, \end{aligned} \tag{106}$$

where (i) $\mathrm{err}_f^{(t)} = \widehat{G}^{-1}_{o_f(f)} \sum_{f' \in [n] \setminus \{f\}} \widehat{X}'_{f'} \widehat{G}_{o_f(f')} \omega_n^{\sigma \Delta (f' - f)}$, with (σ, Δ) implicitly depending on t; this follows directly from Lemma 4.2, along with the definitions $\pi(f) = \sigma f$ and $o_f(f') = \pi(f') - \frac{n}{B} h(f)$. (ii) $\widetilde{\mathrm{err}}_f^{(t)} = \widehat{G}^{-1}_{o_f(f)} (\widehat{U}_{h(f)} - \widehat{U}^*_{h(f)}) \omega^{-\sigma \Delta f}$, a polynomially small error term (*cf.*, Lemma 4.6).

Bounding $\mathrm{err}_f^{(t)}$ and $\widetilde{\mathrm{err}}_f^{(t)}$: In Lemma G.1 below, we show that

$$\mathbb{E}_{\Delta, \pi}\Big[|\mathrm{err}_f^{(t)}|^2 \Big] \le \frac{20}{B} \|\widehat{X}'\|_2^2 \tag{107}$$

$$|\widetilde{\mathrm{err}}_f^{(t)}| \le 2 n^{-c + c'} \|\widehat{X}'\|_2, \tag{108}$$

where c is used in HashToBins, and c' is value such that $\|\widehat{X} - \widehat{\chi}\|_2 \ge \frac{1}{n^{c'}} \|\widehat{\chi}\|_2$. For (108), we upper bound the ℓ_2 norm by the square root of the vector length times the ℓ_∞ norm, yielding

$$\sqrt{\sum_{f \in [n]} |\widetilde{\mathrm{err}}_f^{(t)}|^2} \le \sqrt{n} \max_{f \in [n]} |\widetilde{\mathrm{err}}_f^{(t)}| \le 2 n^{-c + c' + 1/2} \|\widehat{X}'\|_2. \tag{109}$$

We now calculate the probability of a given block j passing the threshold test, considering two separate cases.

If j is in the tail: The probability for j to pass the threshold is closely related to $\Pr[W_j^{(t)} \ge \theta] =$

$\Pr\big[\sqrt{W_j^{(t)}} \geq \sqrt{\theta}\big]$. From (106), $\sqrt{W_j^{(t)}}$ is the ℓ_2-norm of a sum of three signals, and hence we can apply the triangle inequality to obtain

$$\Pr\big[W_j^{(t)} \geq \theta\big] \leq \Pr\bigg[\sum_{f\in I_j} |\mathrm{err}_f^{(t)}|^2 \geq \Big(\sqrt{\theta} - \sqrt{\sum_{f\in I_j} |\widehat{X}'_f|^2} - 2n^{-c+c'+1/2}\|\widehat{X}'\|_2\Big)^2\bigg],$$

where we have applied (109).

By definition, for any $j \in S_{\mathrm{tail}}$, we have $\sqrt{\theta} - \|\widehat{X}'_{I_j}\|_2 \geq \sqrt{\frac{\delta}{k_0}}\|\widehat{X}'\|_2$. Hence, and recalling that $\delta \geq \frac{1}{n}$, if c if sufficiently large so that $\sqrt{\frac{\delta}{k_0}} - 2n^{-c+c'+1/2} \geq \sqrt{\frac{0.9\delta}{k_0}}$, then Markov's inequality yields

$$\begin{aligned}\Pr\big[W_j^{(t)} \geq \theta\big] &\leq \Pr\bigg[\sum_{f\in I_j} |\mathrm{err}_f^{(t)}|^2 \geq \frac{0.9\delta}{k_0}\|\widehat{X}'\|_2^2\bigg] \\ &\leq \frac{\mathbb{E}_{\Delta,\pi}\Big[\sum_{f\in I_j} |\mathrm{err}_f^{(t)}|^2\Big]}{\frac{0.9\delta}{k_0}\|\widehat{X}'\|_2^2} \\ &\leq \frac{\frac{20k_1}{B}\|\widehat{X}'\|_2^2}{\frac{0.9\delta}{k_0}\|\widehat{X}'\|_2^2} \\ &\leq \frac{1}{6}\end{aligned}$$

where the third line follows form (107) and $|I_j| = k_1$, and the final line follows from the choice $B = 160\frac{k_0k_1}{\delta}$. Since W_j is the median of T independent such random variables, it can only exceed θ if there exists a subset of t values of size $\frac{T}{2}$ with $W_j^{(t)} \geq \theta$. Hence,

$$\Pr[W_j \geq \theta] \leq \binom{T}{T/2}\Big(\frac{1}{6}\Big)^{T/2} \leq \Big(\frac{2}{3}\Big)^{T/2} \leq \delta p,$$

where we applied $\binom{T}{T/2} \leq 2^T$, followed by $T = 10\log\frac{1}{\delta p}$ (*cf.*, Algorithm 8).

If j is in the head: We proceed similarly to the tail case, but instead use the triangle inequality in the form of a lower bound (i.e., $\|a+b\|_2 \geq \|a\|_2 - \|b\|_2$), yielding

$$\Pr\big[W_j^{(t)} \leq \theta\big] \leq \Pr\bigg[\sum_{f\in I_j} |\mathrm{err}_f^{(t)}|^2 \geq \Big(\sqrt{\sum_{f\in I_j} |\widehat{X}'_f|^2} - \sqrt{\theta} - 2n^{-c+c'+1/2}\|\widehat{X}'\|_2\Big)^2\bigg].$$

By definition, for any $j \in S_{\mathrm{head}}$, we have $\|\widehat{X}'_{I_j}\|_2 - \sqrt{\theta} \geq \sqrt{\frac{\delta}{k_0}}\|\widehat{X}'\|_2$. Hence, if c if sufficiently large so that $\sqrt{\frac{\delta}{k_0}} - 2n^{-c+c'+1/2} \geq \sqrt{\frac{0.9\delta}{k_0}}$, then analogously to the tail case above, we have

$$\Pr\big[W_j^{(t)} \leq \theta\big] \leq \Pr\bigg[\sum_{f\in I_j} |\mathrm{err}_f^{(t)}|^2 \geq \frac{0.9\delta}{k_0}\|\widehat{X}'\|_2^2\bigg] \leq \frac{1}{6},$$

and consequently $\Pr[W_j \leq \theta] \leq \delta p$.

First claim of lemma: Since $L' \subset L$, we have

$$\mathbb{E}\Big[\big|L' \cap S_{\mathrm{tail}}\big|\Big] = \sum_{j\in L\cap S_{\mathrm{tail}}} \Pr\big[j \in L'\big] = \sum_{j\in L\cap S_{\mathrm{tail}}} \Pr\big[W_j \geq \theta\big].$$

Since we established that $\mathbb{P}[W_j \geq \theta]$ is at most δp, we obtain

$$\mathbb{E}\Big[\big|L' \cap S_{\text{tail}}\big|\Big] \leq \sum_{j \in L \cap S_{\text{tail}}} \delta p = \delta p \cdot |L \cap S_{\text{tail}}|.$$

Second claim of lemma: In order to upper bound $\sum_{j\in(L\cap S_{\text{head}})\setminus L'} \|\widehat{X}'_{I_j}\|_2^2$, we first calculate its expected value as follows:

$$\begin{aligned}\mathbb{E}\Big[\sum_{j\in(L\cap S_{\text{head}})\setminus L'} \|\widehat{X}'_{I_j}\|_2^2\Big] &= \mathbb{E}\Big[\sum_{j\in L\cap S_{\text{head}}} \|\widehat{X}'_{I_j}\|_2^2 \mathcal{I}\big[j \notin L'\big]\Big] \\ &= \sum_{j\in L\cap S_{\text{head}}} \|\widehat{X}'_{I_j}\|_2^2 \Pr\big[j \notin L'\big] \\ &= \sum_{j\in L\cap S_{\text{head}}} \|\widehat{X}'_{I_j}\|_2^2 \Pr\big[W_j \leq \theta\big].\end{aligned}$$

The probability $\Pr\big[W_j \leq \theta\big]$ for $j \in L \cap S_{\text{head}}$ is at most δp, and hence

$$\mathbb{E}\Big[\sum_{j\in(L\cap S_{\text{head}})\setminus L'} \|\widehat{X}'_{I_j}\|_2^2\Big] \leq \delta p \sum_{j\in L\cap S_{\text{head}}} \|\widehat{X}'_{I_j}\|_2^2.$$

Sample complexity and runtime For the sample complexity, note that the algorithm only uses samples via its call to HashToBins. By part (i) of Lemma 4.6 and the choices $B = 160\frac{k_0k_1}{\delta}$ and $F = 10\log\frac{1}{\delta}$, the sample complexity is $O(FB) = O(\frac{k_0k_1}{\delta}\log\frac{1}{\delta})$ per hashing operation. Since we run the hashing in a loop $10\log\frac{1}{\delta p}$ times, the sample complexity is $O(\frac{k_0k_1}{\delta}\log\frac{1}{\delta}\log\frac{1}{\delta p})$.

The runtime depends on three operations. The first is calling HashToBins, for which an analogous argument as that for the sample complexity holds, with the extra $\log n$ factor arising from Lemma 4.6. The second operation is the computation of $W_j^{(t)}$, which takes $|I_j| = O(k_1)$ time for each $j \in L$. Hence, the total contribution from the loop is $O(k_1 \cdot |L|\log\frac{1}{\delta p})$. Finally, since the median can be computed in linear time, computing the medians for every $j \in L$ costs $O(|L|\log\frac{1}{\delta p})$ time, which is dominated by the computation of $W_j^{(t)}$.

□

In the preceding proof, we made use of the following.

Lemma G.1. (EstimateValues guarantees – re-stated from Section 5.1) *Fix (n, k_0, k_1, B), the signals $X \in \mathbb{C}^n$ and $\widehat{\chi} \in \mathbb{C}^n$ with $\|\widehat{X}-\widehat{\chi}\|_2 \geq \frac{1}{n^{c'}}\|\widehat{\chi}\|_2$, and the uniformly random parameters $\sigma, \Delta \in [n]$ with σ odd, and let $\widehat{U}$ be the output of* HashToBins$(X, \widehat{\chi}, G, n, B, \sigma, \Delta)$ *and $\widehat{U}^*$ its exact counterpart. Then defining $\text{err}_f = \widehat{G}^{-1}_{o_f(f)} \sum_{f'\in[n]\setminus\{f\}} \widehat{X}'_{f'}\widehat{G}_{o_f(f')}\omega_n^{\sigma\Delta(f'-f)}$ and $\widetilde{\text{err}}_f = \widehat{G}^{-1}_{o_f(f)}(\widehat{U}_{h(f)} - \widehat{U}^*_{h(f)})\omega^{-\sigma\Delta f}$ (for h and σ_f in Definition 4.2), we have*

$$\mathbb{E}_{\Delta,\pi}\Big[|\text{err}_f|^2\Big] \leq \frac{20}{B}\|\widehat{X}'\|_2^2 \tag{110}$$

$$|\widetilde{\text{err}}_f| \leq 2n^{-c+c'}\|\widehat{X}'\|_2 \tag{111}$$

for c used in HashToBins.

Proof. We take the expectation of $|\text{err}_f|^2$, first over Δ:

$$\mathbb{E}_\Delta\Big[|\text{err}_f|^2\Big] = |\widehat{G}_{o_f(f)}|^{-2} \sum_{f'\in[n]\setminus\{f\}} |\widehat{X}'_{f'}|^2 |\widehat{G}_{o_f(f')}|^2$$

by Parseval. By Definition 2.1 and the definition of $o_f(\cdot)$, we can upper bound $|\widehat{G}_{o_f(f)}|^{-2} \leq 2$. Continuing, we take the expectation with respect to the random permutation π:

$$\begin{aligned} \mathbb{E}_{\Delta,\pi}\Big[|\mathrm{err}_f|^2\Big] &\leq \mathbb{E}_\pi\Big[2 \sum_{f' \in [n] \setminus \{f\}} |\widehat{X}'_{f'}|^2 |\widehat{G}_{o_f(f')}|^2\Big] \\ &= 2 \sum_{f' \in [n] \setminus \{f\}} |\widehat{X}'_{f'}|^2 \mathbb{E}_\pi\Big[|\widehat{G}_{o_f(f')}|^2\Big] \leq \frac{20}{B} \|\widehat{X}'\|_2^2. \end{aligned} \tag{112}$$

by Lemma 4.3.

We now turn to $\widetilde{\mathrm{err}}_f$. We know from Lemma 4.6 that $|\widehat{U}_{h(f)} - \widehat{U}^*_{h(f)}| \leq \|\widehat{U} - \widehat{U}^*\|_\infty \leq n^{-c} \|\widehat{\chi}\|_2$. Hence, and again using $|\widehat{G}_{o_f(f)}|^{-2} \leq 2$, we find that

$$|\widetilde{\mathrm{err}}_f| \leq 2n^{-c} \|\widehat{\chi}\|_2 \leq 2n^{-c+c'} \|\widehat{X}'\|_2, \tag{113}$$

where the second inequality follows since $\|\widehat{\chi}\|_2 \leq n^{c'} \|\widehat{X}'\|_2$ for some $c' > 0$ by assumption. □

H Estimating Individual Frequency Values

Algorithm 9 Energy estimation procedure for individual frequencies

```
procedure EstimateValues($X, \widehat{\chi}, L, n, k_0, k_1, \delta, p$)
    $B \leftarrow \frac{1200}{\delta} k_0 k_1$
    $F \leftarrow 10 \log \frac{1}{\delta}$
    $G \leftarrow (n, B, F)$-flat filter                                   ▷ See Definition 2.1
    $\mathcal{F} \leftarrow \{f \in [n] : \mathrm{round}(\frac{f}{k_1}) \in L\}$
    $T \leftarrow 10 \log \frac{2}{p}$
    for $t \in \{1, \dots, T\}$ do
        $\Delta \leftarrow$ uniform random sample from $[\frac{n}{k_1}]$
        $\sigma \leftarrow$ uniform random sample from odd numbers in $[\frac{n}{k_1}]$
        $\widehat{U} \leftarrow$ HashToBins($X, \widehat{\chi}, G, n, B, \sigma, \Delta$)          ▷ $o_f(f), h(f)$ in Definition 4.2
        $W_f^{(t)} \leftarrow \widehat{G}^{-1}_{o_f(f)} \widehat{U}_{h(f)} \omega^{-\sigma \Delta f}$ for each $f \in \mathcal{F}$
    $W_f \leftarrow \mathrm{Median}_t(x_f^{(t)})$ for each $f \in \mathcal{F}$        ▷ Separately for the real and imaginary parts
    return $W$
```

Once we have located the blocks, we need to estimate the frequency values with them. The function EstimateValues in Algorithm 9 performs this task for us via basic hashing techniques. The following lemma characterizes the guarantee on the output.

Lemma 5.2 (Re-stated from Section 5.1) *For any integers (n, k_0, k_1), list of block indices L, parameters $\delta \in \left(\frac{1}{n}, \frac{1}{20}\right)$ and $p \in (0, 1/2)$, and signals $X \in \mathbb{C}^n$ and $\widehat{\chi} \in \mathbb{C}^n$ with $\|\widehat{X} - \widehat{\chi}\|_2 \geq \frac{1}{\mathrm{poly}(n)} \|\widehat{\chi}\|_2$, the output W of the function* EstimateValues$(X, \widehat{\chi}, L, n, k_0, k_1, \delta, p)$ *has the following property:*

$$\sum_{f \in \bigcup_{j \in L} I_j} |W_f - (\widehat{X} - \widehat{\chi})_f|^2 \leq \delta \frac{|L|}{3k_0} \|\widehat{X} - \widehat{\chi}\|_2^2$$

with probability at least $1 - p$, where I_j is the j-th block. Moreover, the sample complexity is $O(\frac{k_0 k_1}{\delta} \log \frac{1}{p} \log \frac{1}{\delta})$, and if $\|\widehat{\chi}\|_0 = O(k_0 k_1)$, then the runtime is $O(\frac{k_0 k_1}{\delta} \log \frac{1}{p} \log \frac{1}{\delta} \log n + k_1 \cdot |L| \log \frac{1}{p})$.

Proof. Let $X' = X - \chi$, and let $\widehat{U}$ be the output of HashToBins and $\widehat{U}^*$ its exact counterpart. We start by calculating $W_f^{(t)}$ for an arbitrary $f \in \mathcal{F}$:

$$\begin{aligned} W_f^{(t)} &= \widehat{G}^{-1}_{o_f(f)} \widehat{U}_{h(f)} \omega^{-\sigma \Delta f} \\ &= \widehat{G}^{-1}_{o_f(f)} \widehat{U}^*_{h(f)} \omega^{-\sigma \Delta f} + \widehat{G}^{-1}_{o_f(f)} (\widehat{U}_{h(f)} - \widehat{U}^*_{h(f)}) \omega^{-\sigma \Delta f} \\ &= \widehat{X}'_f + \mathrm{err}_f^{(t)} + \widetilde{\mathrm{err}}_f^{(t)} \quad \text{(by Lemma 4.2)}, \end{aligned} \tag{114}$$

where $\mathrm{err}_f^{(t)} = \widehat{G}^{-1}_{o_f(f)} \sum_{f' \in [n] \setminus \{f\}} \widehat{X}'_{f'} \widehat{G}_{o_f(f')} \omega^{\sigma \Delta (f' - f)}$, and $\widetilde{\mathrm{err}}_f^{(t)} = \widehat{G}^{-1}_{o_f(f)} (\widehat{U}_{h(f)} - \widehat{U}^*_{h(f)}) \omega^{-\sigma \Delta f}$, for (σ, Δ) implicitly depending on t.

Bounding $\mathrm{err}_f^{(t)}$ and $\widetilde{\mathrm{err}}_f^{(t)}$: Using Lemma G.1 in Appendix G, we have

$$\mathbb{E}_{\Delta, \pi}\Big[|\mathrm{err}_f^{(t)}|^2\Big] \le \frac{20}{B} \|\widehat{X}'\|_2^2 \tag{115}$$

$$|\widetilde{\mathrm{err}}_f^{(t)}| \le 2n^{-c+c'} \|\widehat{X}'\|_2, \tag{116}$$

where c is used in HashToBins, and c' is the exponent in the poly(n) notation of the assumption $\|\widehat{X} - \widehat{\chi}\|_2 \ge \frac{1}{\mathrm{poly}(n)} \|\widehat{\chi}\|_2$.

In order to characterize $|W_f^{(t)} - \widehat{X}'_f|^2$, we use the following:

$$|\widetilde{\mathrm{err}}_f|^2 + 2|\mathrm{err}_f^{(t)}| \cdot |\widetilde{\mathrm{err}}_f^{(t)}| \le 4n^{2(-c+c')} \|\widehat{X}'\|_2^2 + 4n^{-c+c'} \|\widehat{X}'\|_2 \cdot |\mathrm{err}_f^{(t)}|. \tag{117}$$

which follows directly from (116).

Characterizing $|W_f^{(t)} - \widehat{X}'_f|^2$: We have from (114), (115), and (117) that

$$\begin{aligned} \mathbb{E}[|W_f^{(t)} - \widehat{X}'_f|^2] &\le \mathbb{E}\big[|\mathrm{err}_f^{(t)}|^2 + 2|\mathrm{err}_f^{(t)}| \cdot |\widetilde{\mathrm{err}}_f^{(t)}| + |\widetilde{\mathrm{err}}_f^{(t)}|^2\big] \\ &\le \frac{20}{B} \|\widehat{X}'\|_2^2 + 4n^{2(-c+c')} \|\widehat{X}'\|_2^2 + 4n^{-c+c'} \|\widehat{X}'\|_2 \mathbb{E}\big[|\mathrm{err}_f^{(t)}|\big] \\ &\le \frac{20}{B} \|\widehat{X}'\|_2^2 + 4n^{2(-c+c')} \|\widehat{X}'\|_2^2 + 4\sqrt{\frac{20}{B}} n^{-c+c'} \|\widehat{X}'\|_2^2, \end{aligned} \tag{118}$$

where the last line follows by writing $\mathbb{E}\big[|\mathrm{err}_f^{(t)}|\big] \le \sqrt{\mathbb{E}\big[|\mathrm{err}_f^{(t)}|^2\big]}$ via Jensen's inequality, and then applying (115).

Since $B = \frac{1200 k_0 k_1}{\delta}$ and we have assumed $\delta \ge \frac{1}{n}$, we have $B \le 1200n^3$, and hence we have for sufficiently large c that (118) simplifies to $\mathbb{E}[|W_f^{(t)} - \widehat{X}'_f|^2] \le \frac{25}{B} \|\widehat{X}'\|_2^2$. This means that

$$\mathrm{Pr}_{\Delta, \pi}\Big[|W_f^{(t)} - \widehat{X}'_f|^2 \ge \frac{160t}{B} \|\widehat{X}'\|_2^2\Big] \le \frac{\mathbb{E}_{\Delta, \pi}\Big[|W_f^{(t)} - \widehat{X}'_f|^2\Big]}{\frac{160t}{B} \|\widehat{X}'\|_2^2} \le \frac{1}{6t}. \tag{119}$$

by Markov's inequality.

Taking the median: Recall that W_f is the median of T independent random variables, with the median taken separately for the real and imaginary parts. Since $|W|^2 = |\mathrm{Re}(W)|^2 + |\mathrm{Im}(W)|^2$, we find that (119) holds true when $W_f^{(t)} - \widehat{X}'_f$ is replaced by its real or imaginary part. Hence, with probability at least $1 - \binom{T}{T/2} \big(\frac{1}{6t}\big)^{T/2}$, we have $|\mathrm{Re}(W_f^{(t)} - \widehat{X}'_f)|^2 < \frac{160t}{B} \|\widehat{X}'\|_2^2$, and analogously for the imaginary part. Combining these and applying the union bound, we obtain

$$\mathbb{P}\bigg[|W_f - \widehat{X}'_f|^2 \ge \frac{320t}{B} \|\widehat{X}'\|_2^2\bigg] \le 2\binom{T}{T/2} \Big(\frac{1}{6t}\Big)^{T/2} \le 2\Big(\frac{2}{3t}\Big)^{T/2} \le \frac{p}{t^{T/2}}, \tag{120}$$

where we first applied $\binom{T}{T/2} \leq 2^T$, and then the choice $T = 10 \log \frac{2}{p}$ from Algorithm 9 and the choice of $p \leq 1/2$.

We now bound the error as follows:

$$|W_f - \widehat{X}'_f|^2 \leq \frac{320}{B} \|\widehat{X}'\|_2^2 + \left| |W_f - \widehat{X}'_f|^2 - \frac{320}{B} \|\widehat{X}'\|_2^2 \right|_+. \tag{121}$$

We write the expected value of the second term as

$$\begin{aligned}
\mathbb{E}\Big[\Big| |W_f - \widehat{X}'_f|^2 - \frac{320}{B}\|\widehat{X}'\|_2^2 \Big|_+\Big] &= \int_0^\infty \Pr\Big[\Big| |W_f - \widehat{X}'_f|^2 - \frac{320}{B}\|\widehat{X}'\|_2^2 \Big|_+ \geq u\Big] du \\
&= \int_0^\infty \Pr\Big[|W_f - \widehat{X}'_f|^2 \geq \frac{320}{B}\|\widehat{X}'\|_2^2 + u\Big] du \\
&= \int_1^\infty \frac{320}{B}\|\widehat{X}'\|_2^2 \Pr\Big[|W_f - \widehat{X}'_f|^2 \geq \frac{320v}{B}\|\widehat{X}'\|_2^2 \Big] dv
\end{aligned}$$

where we applied the change of variable $v = 1 + \frac{u}{\frac{320}{B}\|\widehat{X}'\|_2^2}$. By incorporating (120) into this integral, we obtain

$$\begin{aligned}
\mathbb{E}\Big[\Big| |W_f - \widehat{X}'_f|^2 - \frac{320}{B}\|\widehat{X}'\|_2^2 \Big|_+\Big] &\leq \frac{320}{B}\|\widehat{X}'\|_2^2 \int_1^\infty \frac{p}{v^{T/2}} dv \\
&\leq \frac{320}{B}\|\widehat{X}'\|_2^2 \cdot \frac{p}{T/2 - 1} \\
&\leq \frac{80}{B}\|\widehat{X}'\|_2^2 \cdot p,
\end{aligned} \tag{122}$$

where the second line is by explicitly evaluating the integral (with $T > 2$), and the third by $T/2 - 1 \geq 4$ (*cf.*, Algorithm 9).

Summing (121) over over $\mathcal{F} = \cup_{j \in L} I_j$, we find that the total error is upper bounded as follows:

$$\sum_{f \in \mathcal{F}} |W_f - \widehat{X}'_f|^2 \leq \frac{320|\mathcal{F}|}{B}\|\widehat{X}'\|_2^2 + \sum_{f \in \mathcal{F}} \left| |W_f - \widehat{X}'_f|^2 - \frac{320}{B}\|\widehat{X}'\|_2^2 \right|_+.$$

From (122), the expected value of the second term is at most $p \cdot \frac{80|\mathcal{F}|}{B}\|\widehat{X}'\|_2^2$, and hence

$$\sum_{f \in \mathcal{F}} |W_f - \widehat{X}'_f|^2 \leq \frac{400|\mathcal{F}|}{B}\|\widehat{X}'\|_2^2$$

with probability at least $1 - p$, by Markov. The lemma follows since $B = \frac{1200}{\delta} k_0 k_1$ in Algorithm 9.

Sample complexity and runtime: To calculate the sample complexity, note that the only operation in the algorithm that takes samples is the call to HashToBins. By Lemma 4.6, and the choices $B = 1200\frac{k_0 k_1}{\delta}$ and $F = 10 \log \frac{1}{\delta}$, the sample complexity is $O(\frac{k_0 k_1}{\delta} \log \frac{1}{\delta})$ per hashing performed. Since we run the hashing in a loop $10 \log \frac{2}{p}$ times, this amounts to a total of $O(\frac{k_0 k_1}{\delta} \log \frac{1}{\delta} \log \frac{1}{p})$.

The runtime depends on two operations. The first one is calling HashToBins, whose analysis follows similarly to the aforementioned sample complexity analysis using the assumption $\|\widehat{\chi}\|_0 = O(k_0 k_1)$, but with an extra $\log n$ factor compared to the sample complexity, as per Lemma 4.6.

The other operation is computation of $W_f^{(t)}$, which takes unit time for each $f \in \mathcal{F}$. Since the size of $|\mathcal{F}| = k_1 \cdot |L|$, running it in a loop costs $O(k_1 \cdot |L| \log \frac{1}{p})$. Computing the median is done in linear time which consequently results in $|\mathcal{F}|T = O(k_1 \cdot |L| \log \frac{1}{p})$. □

References

[1] A. C. Gilbert, P. Indyk, M. Iwen, and L. Schmidt, "Recent developments in the sparse fourier transform: a compressed fourier transform for big data," *IEEE Signal Processing Magazine*, vol. 31, no. 5, pp. 91–100, 2014.

[2] A. C. Gilbert, M. J. Strauss, and J. A. Tropp, "A tutorial on fast fourier sampling," *IEEE Signal Processing Magazine*, vol. 25, no. 2, pp. 57–66, 2008.

[3] E. Candes and T. Tao, "Near optimal signal recovery from random projections: Universal encoding strategies," *IEEE Trans. on Info.Theory*, 2006.

[4] M. Rudelson and R. Vershynin, "On sparse reconstruction from Fourier and Gaussian measurements," *CPAM*, vol. 61, no. 8, pp. 1025–1171, 2008.

[5] O. Goldreich and L. Levin, "A hard-corepredicate for allone-way functions," *STOC*, pp. 25–32, 1989.

[6] E. Kushilevitz and Y. Mansour, "Learning decision trees using the Fourier spectrum," *STOC*, 1991.

[7] Y. Mansour, "Randomized interpolation and approximation of sparse polynomials," *ICALP*, 1992.

[8] A. Gilbert, S. Guha, P. Indyk, M. Muthukrishnan, and M. Strauss, "Near-optimal sparse Fourier representations via sampling," *STOC*, 2002.

[9] A. Akavia, S. Goldwasser, and S. Safra, "Proving hard-core predicates using list decoding," *FOCS*, vol. 44, pp. 146–159, 2003.

[10] A. Gilbert, M. Muthukrishnan, and M. Strauss, "Improved time bounds for near-optimal space Fourier representations," *SPIE Conference, Wavelets*, 2005.

[11] M. A. Iwen, "Combinatorial sublinear-time Fourier algorithms," *Foundations of Computational Mathematics*, vol. 10, pp. 303–338, 2010.

[12] A. Akavia, "Deterministic sparse Fourier approximation via fooling arithmetic progressions," *COLT*, pp. 381–393, 2010.

[13] H. Hassanieh, P. Indyk, D. Katabi, and E. Price, "Simple and practical algorithm for sparse Fourier transform," *SODA*, 2012.

[14] ——, "Near-optimal algorithm for sparse Fourier transform," *STOC*, 2012.

[15] D. Lawlor, Y. Wang, and A. Christlieb, "Adaptive sub-linear time fourier algorithms," *arXiv:1207.6368*, 2012.

[16] P. Boufounos, V. Cevher, A. C. Gilbert, Y. Li, and M. J. Strauss, "What's the frequency, kenneth?: Sublinear fourier sampling off the grid," *RANDOM/APPROX*, 2012.

[17] H. Hassanieh, F. Adib, D. Katabi, and P. Indyk, "Faster gps via the sparse fourier transform," *MOBICOM*, 2012.

[18] S. Pawar and K. Ramchandran, "Computing a k-sparse n-length discrete fourier transform using at most 4k samples and o (k log k) complexity," *ISIT*, 2013.

[19] S. Heider, S. Kunis, D. Potts, and M. Veit, "A sparse prony fft," *SAMPTA*, 2013.

[20] P. Indyk, M. Kapralov, and E. Price, "(Nearly) sample-optimal sparse fourier transform," *SODA*, 2014.

[21] P. Indyk and M. Kapralov, "Sample-optimal fourier sampling in any fixed dimension," *FOCS*, 2014.

[22] M. Kapralov, "Sparse fourier transform in any constant dimension with nearly-optimal sample complexity in sublinear time (available as an arxiv report at http://arxiv.org/abs/1604.00845)," *STOC*, 2016.

[23] E. Price and Z. Song, "A robust sparse fourier transform in the continuous setting," *FOCS*, 2015.

[24] H. Hassanieh, P. Indyk, D. Katabi, and E. Price, "Nearly optimal sparse Fourier transform," in *Proc. ACM Symp. Theory Comp. (STOC).* ACM, 2012, pp. 563–578.

[25] P. Indyk, M. Kapralov, and E. Price, "(nearly) sample-optimal sparse fourier transform," in *ACM-SIAM Symp. Disc. Alg. (SODA).* Society for Industrial and Applied Mathematics, 2014, pp. 480–499.

[26] R. Baraniuk, V. Cevher, M. Duarte, and C. Hegde, "Model-based compressive sensing," *IEEE Trans. Inf. Theory*, vol. 56, no. 4, pp. 1982–2001, April 2010.

[27] L. Baldassarre, N. Bhan, V. Cevher, A. Kyrillidis, and S. Satpathi, "Group-sparse model selection: Hardness and relaxations," *IEEE Transactions on Information Theory*, vol. 62, no. 11, pp. 6508–6534, November 2016.

[28] M. E. Halabi and V. Cevher, "A totally unimodular view of structured sparsity," in *International Conference on Artificial Intelligence and Statistics (AISTATS)*, San Diego, CA, May 2015.

[29] F. R. Bach, "Structured sparsity-inducing norms through submodular functions," in *Advances in Neural Information Processing Systems*, 2010, pp. 118–126.

[30] B. Bah, L. Baldassarre, and V. Cevher, "Model-based sketching and recovery with expanders," in *Proceedings of the Twenty-Fifth Annual ACM-SIAM Symposium on Discrete Algorithms.* Society for Industrial and Applied Mathematics, 2014, pp. 1529–1543.

[31] P. Indyk and I. Razenshteyn, "On model-based rip-1 matrices," in *International Colloquium on Automata, Languages, and Programming.* Springer, 2013, pp. 564–575.

[32] R. G. Baraniuk, V. Cevher, and M. B. Wakin, "Low-dimensional models for dimensionality reduction and signal recovery: A geometric perspective," *Proceedings of the IEEE*, vol. 98, no. 6, pp. 959–971, 2010.

[33] H. Hassanieh, P. Indyk, D. Katabi, and E. Price, "Simple and practical algorithm for sparse Fourier transform," in *Proc. ACM-SIAM Symp. Disc. Alg. (SODA).* SIAM, 2012, pp. 1183–1194.

[34] K. Do Ba, P. Indyk, E. Price, and D. P. Woodruff, "Lower Bounds for Sparse Recovery," *SODA*, 2010.

[35] E. Price and D. P. Woodruff, "$(1+\epsilon)$-approximate sparse recovery," *FOCS*, 2011.

[36] S. Foucart and H. Rauhut, *A Mathematical Introduction to Compressive Sensing.* Springer New York, 2013.

[37] E. Price and D. P. Woodruff, "$(1+\epsilon)$-approximate sparse recovery," in *IEEE Symp. Found. Comp. Sci. (FOCS)*, 2011.

[38] T. Hagerup, K. Mehlhorn, and J. I. Munro, "Maintaining discrete probability distributions optimally," in *Proc. Int. Colloq. Automata, Lang. and Prog.*, 1993, pp. 253–264. [Online]. Available: http://dl.acm.org/citation.cfm?id=646247.685022

[39] P. Indyk and M. Kapralov, "Sample-optimal Fourier sampling in any constant dimension," in *Ann. Symp. Found. Comp. Sci. (FOCS).* IEEE, 2014, pp. 514–523.

[40] T. M. Cover and J. A. Thomas, *Elements of Information Theory.* John Wiley & Sons, Inc., 2006.

[41] B. Ghazi, H. Hassanieh, P. Indyk, D. Katabi, E. Price, and L. Shi, "Sample-optimal average-case sparse fourier transform in two dimensions," in *51st Annual Allerton Conference on Communication, Control, and Computing, Allerton 2013, Allerton Park & Retreat Center, Monticello, IL, USA, October 2-4, 2013.* IEEE, 2013, pp. 1258–1265. [Online]. Available: http://dx.doi.org/10.1109/Allerton.2013.6736670