Combinatorial Sublinear-Time Fourier Algorithms*

M. A. Iwen†
University of Michigan
markiwen@umich.edu

March 25, 2008

Abstract

We study the problem of estimating the best k term Fourier representation for a given frequency-sparse signal (i.e., vector) $\mathbf{A}$ of length $N \gg k$. More explicitly, we investigate how to deterministically identify k of the largest magnitude frequencies of $\hat{\mathbf{A}}$, and estimate their coefficients, in polynomial$(k, \log N)$ time. Randomized sublinear time algorithms which have a small (controllable) probability of failure for each processed signal exist for solving this problem [23, 24]. In this paper we develop the first known deterministic sublinear time sparse Fourier Transform algorithm. As an added bonus, a simple relaxation of our deterministic Fourier result leads to a new Monte Carlo Fourier algorithm with similar runtime/sampling bounds to the current best randomized Fourier method [24]. Finally, the Fourier algorithm we develop here implies a simpler optimized version of the deterministic compressed sensing method previously developed in [29].

1 Introduction

In many applications only the top few most energetic terms of a signal's Fourier Transform (FT) are of interest. In such applications the Fast Fourier Transform (FFT), which computes all FT terms, is computationally wasteful. Compressed Sensing (CS) methods [13, 6, 45, 42, 33, 12] provide a robust framework for reducing the number of signal samples required to estimate a signal's FT. For this reason CS methods are useful in areas such as MR imaging [36, 37] and analog-to-digital conversion [34, 32] when sampling costs are high. However, despite small sampling requirements, standard CS Fourier methods utilizing Basis Pursuit (BP) [13, 6, 12] and Orthogonal Matching Pursuit (OMP) [45, 42] have runtime requirements which are superlinear in the signal's size/bandwidth. Hence, these methods are inappropriate for applications involving large signal sizes/bandwidths where runtime is of primary importance (e.g., numerical methods for multiscale problems [11]).

A second body of work on algorithmic compressed sensing includes methods which have achieved near-optimal reconstruction runtime bounds [23, 24, 9, 10, 41, 26, 28]. However, with the notable exception of [23, 24], these CS algorithms don't permit sublinear sampling in the Fourier case. Hence, despite their efficient reconstruction algorithms, their total Fourier measurement and reconstruction runtime costs are still superlinear in the signal size/bandwidth. In the Fourier case they generally require more operations than a regular FFT for all nontrivial sparsity levels while utilizing approximately the same number of signal samples.

To date only the randomized Fourier methods [23, 24] have been shown to outperform the FFT in terms of runtime on frequency-sparse broadband superpositions while utilizing only a fraction of the FFT's required samples [30]. However, these algorithms are not deterministic and so can produce incorrect results with some small probability on each input signal. Thus, they aren't appropriate for long-lived failure intolerant applications.

In this paper we construct the first known deterministic sublinear time sparse Fourier algorithm. In order to produce our new Fourier algorithm we introduce a combinatorial object called a k-majority separating collection of sets which can be constructed using number theoretic methods along the lines of [9, 17]. This new combinatorial

*Results herein supersede preliminary Fourier results in [29, 31].

†Supported in part by NSF DMS-0510203.

object yields a simple new CS reconstruction algorithm with better algebraic compressibility results than previous fast deterministic CS methods [9, 10, 41, 28]. Furthermore, the number theoretic nature of our construction allows the sublinear time computation of Fourier measurements via aliasing. As a result, we are able to obtain a deterministic sublinear time Fourier algorithm which behaves well on both algebraically and exponentially compressible signals. Finally, a simple relaxation of our deterministic Fourier method provides a new randomized Fourier algorithm with similar runtime/sampling bounds to [24].

Related work to our results here include all of the aforementioned CS methods (see [1] for many more). Most closely related of these are the deterministic CS methods [9, 10, 41, 28, 12]. The deterministic constructions in [12] require BP or OMP based reconstruction methods [2, 42] with runtimes that are superlinear in the input signal size/bandwidth. On the other hand, our deterministic CS based methods utilize faster recovery procedures along the lines of those first introduced by Cormode and Muthukrishnan (CM) [9, 10, 41]. Indyk's recent work [28] also utilizes similar recovery procedures and achieves theoretically faster reconstruction times on exact superpositions. However, his iterative reconstruction methods don't appear to generalize to algebraically compressible signals. Furthermore, as previously stated, neither Indyk's nor CM's compressed sensing algorithms permit sublinear sampling in the Fourier setting.

Previous randomized Fourier algorithms [23, 24] are similar to our deterministic results in that they obtain both sublinear reconstruction time and sampling (as opposed to other CS Fourier methods). However, they employ random sampling techniques and thus fail to output good approximate answers with non-zero probability. Other related work includes earlier methods for the reconstruction of sparse trigonometric polynomials via random sampling [38, 7]. In turn, these methods can be traced back further to algorithms for learning sparse multivariate polynomials over fields of characteristic zero [18, 39].

Finally, our CS recovery techniques are related group testing methods [15]. In particular, our k-majority separating collection of sets construction is closed related to the number theoretic group testing construction utilized in [17]. This relationship to group testing, in combination with the Fourier transform's natural aliasing behavior, is essentially what allows our sublinear Fourier methods to be constructed. For more on group testing in statistical signal recovery see [25].

The main contributions of this paper are:

1. We present the first known deterministic sublinear time sparse DFT. In the process, we demonstrate the connection between compressed sensing and sublinear time Fourier transform methods.

2. We present a simple randomized Fourier algorithm with runtime superlinear in the input signal's size/bandwidth which exactly recovers k-frequency superpositions with high probability using a near-optimal number of samples. When modified to run in sublinear time, we obtain a Fourier algorithm with runtime/sampling requirements similar to [24].

3. We introduce k-majority strongly selective collections of sets which have potential applications to streaming algorithms along the lines of [40, 21].

The remainder of this paper is organized as follows: In section 2 we introduce relevant definitions, terminology, and background. Then, in Section 3 we define k-majority selective collections of sets and present number theoretic constructions. Section 4 contains simple superlinear-time Fourier algorithms along with analysis of their runtime and sampling requirements. In section 5 we modify Section 4's algorithms to produce sublinear-time Fourier algorithms. Finally, the discrete versions of our algorithms are presented in Section 6. Section 7 contains a short conclusion.

2 Preliminaries

Throughout the remainder of this paper we will be interested in complex valued functions $f : [0, 2\pi] \mapsto \mathbb{C}$ and signals (or arrays) of length N containing f values at various $x \in [0, 2\pi]$. We shall denote such signals by $\mathbf{A}$, where $\mathbf{A}(j) \in \mathbb{C}$ is the signal's j^{th} complex value for all $j \in [0, N)$. Hereafter we will refer to the process of either calculating, measuring, or retrieving the f value associated any $\mathbf{A}(j) \in \mathbb{C}$ from machine memory as *sampling* from f and/or $\mathbf{A}$.

Given a signal $\mathbf{A}$ we define its discrete L^q-norm to be

$$\|\mathbf{A}\|_q = \left(\sum_{j=0}^{N-1} |\mathbf{A}(j)|^q \right)^{\frac{1}{q}}. \tag{1}$$

More specifically, we will refer to $\|\mathbf{A}\|_2^2$ as $\mathbf{A}$'s *energy*. We will say that $\mathbf{A} \in L^q$ if $\|\mathbf{A}\|_q^q$ converges (i.e., we allow $N = \infty$). Finally, j and ω will always denote integers below.

2.1 Compressed Sensing and Compressibility

Given such a signal $\mathbf{A}$, let Ψ be any $N \times N$ change of basis matrix/transform under which $\mathbf{A}$ is sparse (i.e., only $k \ll N$ entries of $\Psi \cdot \mathbf{A}$ are significant/large in magnitude). Algorithmic compressed sensing (CS) methods [23, 24, 9, 10, 41, 26, 28, 29] deal with generating a $K \times N$ measurement matrix, $\mathcal{M}$, with the smallest number of rows possible (i.e., K minimized) so that the k significant entries of $\Psi \cdot \mathbf{A}$ can be well approximated by the K-element vector result of

$$(\mathcal{M} \cdot \Psi) \cdot \mathbf{A}. \tag{2}$$

Note that CS is inherently algorithmic since a procedure for recovering $\Psi \cdot \mathbf{A}$'s largest k-entries from the result of Equation 2 must be specified.

A fast CS recovery algorithm produces output of the form $(\omega_1, C_1), \ldots, (\omega_k, C_k)$ where each $(\omega_j, C_j) \in [0, N) \times \mathbb{C}$. We will refer to any such set of $k < N$ tuples

$$\left\{ (\omega_j, C_j) \in [0, N) \times \mathbb{C} \text{ s.t. } j \in [1, k] \right\}$$

as a **sparse** Ψ **representation** and denote it with a superscript 's'. Note that if we are given a sparse Ψ representation, $\mathbf{R}_\Psi^{\text{s}}$, we may consider $\mathbf{R}_\Psi^{\text{s}}$ to be a length-N signal. We simply view $\mathbf{R}_\Psi^{\text{s}}$ as the N length signal

$$\mathbf{R}_\Psi(j) = \begin{cases} C_j & \text{if } (j, C_j) \in \mathbf{R}_\Psi^{\text{s}} \\ 0 & \text{otherwise} \end{cases}$$

for all $j \in [0, N)$. Using this idea we may reconstruct $\mathbf{R}$ in any desired basis using $\mathbf{R}_\Psi^{\text{s}}$.

A k term/tuple sparse Ψ representation is k-optimal for a signal $\mathbf{A}$ if it contains k of the largest magnitude entries of $\Psi \cdot \mathbf{A}$. More precisely, we'll say that a sparse Ψ representation $\mathbf{R}_\Psi^{\text{s}}$ is k**-optimal** for $\mathbf{A}$ if there exists a valid ordering of $\Psi \cdot \mathbf{A}$ by magnitude

$$\left| (\Psi \cdot \mathbf{A})(\omega_1) \right| \geq \left| (\Psi \cdot \mathbf{A})(\omega_2) \right| \geq \cdots \geq \left| (\Psi \cdot \mathbf{A})(\omega_j) \right| \geq \cdots \geq \left| (\Psi \cdot \mathbf{A})(\omega_N) \right| \tag{3}$$

so that $\left\{ (\omega_l, (\Psi \cdot \mathbf{A})(\omega_l)) \;\middle|\; l \in [1, k] \right\} = \mathbf{R}_\Psi^{\text{s}}$. Note that a signal $\mathbf{A}$ may have several k-optimal Ψ representations if its $\Psi \cdot \mathbf{A}$ entry magnitudes are non-unique. For example, there are two 1-optimal sparse Fourier representations for the signal

$$\mathbf{A}(j) = 2e^{\frac{-2\pi ij}{N}} + 2e^{\frac{2\pi ij}{N}},\ N > 2.$$

However, all k-optimal Ψ representations, ${}_\Psi\mathbf{R}_{\text{opt}}^{\text{s}}$, for any signal $\mathbf{A}$ will always have both the same unique $\|{}_\Psi\mathbf{R}_{\text{opt}}\|_2$ and $\|(\Psi \cdot \mathbf{A}) -_\Psi \mathbf{R}_{\text{opt}}\|_2$ values.

We conclude this subsection with two final definitions: Let ω_b be a b^{th} largest magnitude entry of $\Psi \cdot \mathbf{A}$ as per Equation 3. We will say that a signal $\Psi \cdot \mathbf{A}$ is **(algebraically)** p**-compressible** for some $p > 1$ if $|(\Psi \cdot \mathbf{A})(\omega_b)| = O(b^{-p})$ for all $b \in [1, N]$. If ${}_\Psi\mathbf{R}_{\text{opt}}^{\text{s}}$ is a k-optimal Ψ representation we can see that

$$\|(\Psi \cdot \mathbf{A}) -_\Psi \mathbf{R}_{\text{opt}}\|_2^2 = \sum_{b=k+1}^{N} \left| (\Psi \cdot \mathbf{A})(\omega_b) \right|^2 = O\left(\int_k^\infty b^{-2p} db \right) = O(k^{1-2p}). \tag{4}$$

Hence, any p-compressible signal $\Psi \cdot \mathbf{A}$ (i.e., any signal with a fixed $c \in \mathbb{R}$ so that $\left| (\Psi \cdot \mathbf{A})(\omega_b) \right| \leq c \cdot b^{-p}$ for all $b \in [1, N]$) will have $\|(\Psi \cdot \mathbf{A}) -_\Psi \mathbf{R}_{\text{opt}}\|_2^2 \leq \tilde{c}_p \cdot k^{1-2p}$ for some $\tilde{c}_p \in \mathbb{R}$. For any p-compressible signal class (i.e., for

any choice of p and c) we will refer to the related optimal $O(k^{1-2p})$-size worst case error value (i.e., Equation 4 above) as $\|C_k^{\text{opt}}\|_2^2$. Similarly, we define an **exponentially compressible** (or **exponentially decaying**) signal for a fixed α to be one for which $\left|(\Psi \cdot \mathbf{A})(\omega_b)\right| = O(2^{-\alpha b})$. The optimal worst case error is then

$$\|C_k^{\text{opt}}\|_2^2 = O\left(\int_k^\infty 4^{-\alpha b} db\right) = O(4^{-\alpha k}). \tag{5}$$

2.2 The Fourier Case

For the remainder of this paper we will be interested the special CS case where Ψ is the $N \times N$ Discrete Fourier Transform (DFT) matrix. In this case we have

$$\Psi_{i,j} = \frac{2\pi}{N} \cdot e^{\frac{2\pi \mathbf{i} \cdot i \cdot j}{N}}. \tag{6}$$

Thus, $\mathbf{A}$'s DFT, denoted $\widehat{\mathbf{A}}$, is another signal of length N defined as follows:

$$\widehat{\mathbf{A}}(\omega) = \frac{2\pi}{N} \cdot \sum_{j=0}^{N-1} e^{\frac{-2\pi i \omega j}{N}} \mathbf{A}(j), \quad \forall \omega \in \left(-\left\lceil \frac{N}{2} \right\rceil, \left\lfloor \frac{N}{2} \right\rfloor\right]. \tag{7}$$

We will refer to any index, ω, of $\widehat{\mathbf{A}}$ as a *frequency*. Furthermore, we will refer to $\widehat{\mathbf{A}}(\omega)$ as frequency ω's *coefficient* for each $\omega \in \left(-\left\lceil \frac{N}{2} \right\rceil, \left\lfloor \frac{N}{2} \right\rfloor\right]$.

We may recover $\mathbf{A}$ from its DFT via the Inverse Discrete Fourier Transform (IDFT) as follows:

$$\mathbf{A}(j) = \widehat{\widehat{\mathbf{A}}}^{-1}(j) = \frac{1}{2\pi} \cdot \sum_{\omega=1-\lceil \frac{N}{2} \rceil}^{\lfloor \frac{N}{2} \rfloor} e^{\frac{2\pi i \omega j}{N}} \widehat{\mathbf{A}}(\omega), \quad \forall j \in [0, N). \tag{8}$$

Parseval's equality tells us that $\|\widehat{\mathbf{A}}\|_2 = \sqrt{\frac{2\pi}{N}} \cdot \|\mathbf{A}\|_2$ for any signal. Note that any non-zero coefficient frequency will contribute to $\widehat{\mathbf{A}}$'s energy. Hence, we will also refer to $|\widehat{\mathbf{A}}(\omega)|^2$ as frequency ω's *energy*. If $|\widehat{\mathbf{A}}(\omega)|$ is relatively large we'll say that ω is *energetic*.

Fix δ small (e.g., $\delta = 0.1$). Given an input signal, $\mathbf{A}$, with a compressible Fourier transform, our deterministic Fourier algorithm will identify k of the most energetic frequencies from $\widehat{\mathbf{A}}$ and approximate their coefficients to produce a sparse Fourier representation $\hat{\mathbf{R}}^{\text{s}}$ with $\|\widehat{\mathbf{A}} - \hat{\mathbf{R}}\|_2^2 \leq \|\widehat{\mathbf{A}} - \hat{\mathbf{R}}_{\text{opt}}\|_2^2 + \delta\|C_k^{\text{opt}}\|_2^2$. It should be noted that the Fourier reconstruction algorithms below all extend naturally to the general compressed sensing case presented in Section 2.1 above via work analogous to that presented in [29].

3 Combinatorial Constructions

The following combinatorial structures are motivated by k-strongly separating sets [27, 9]. There properties directly motivate our Fourier reconstruction procedures in Sections 4 and 5.

Definition 1. *A collection, $\mathcal{S}$, of subsets of $\left(-\left\lceil \frac{N}{2} \right\rceil, \left\lfloor \frac{N}{2} \right\rfloor\right]$ is called k-**majority selective** if for all $X \subset \left(-\left\lceil \frac{N}{2} \right\rceil, \left\lfloor \frac{N}{2} \right\rfloor\right]$ with $|X| \leq k$ and all $n \in \left(-\left\lceil \frac{N}{2} \right\rceil, \left\lfloor \frac{N}{2} \right\rfloor\right]$, more than half of the subsets $S \in \mathcal{S}$ containing n are such that $S \cap X = \{n\} \cap X$ (i.e., every $n \in \left(-\left\lceil \frac{N}{2} \right\rceil, \left\lfloor \frac{N}{2} \right\rfloor\right]$ occurs separated from all (other) members of X in more than half of the $\mathcal{S}$ elements containing n).*

Definition 2. *Fix an unknown $X \subset \left(-\left\lceil \frac{N}{2} \right\rceil, \left\lfloor \frac{N}{2} \right\rfloor\right]$ with $|X| \leq k$. A randomly assembled collection of $\left(-\left\lceil \frac{N}{2} \right\rceil, \left\lfloor \frac{N}{2} \right\rfloor\right]$ subsets, $\mathcal{S}$, is called (k, σ)-**majority selective** if the following is true with probability at least σ: For all $n \in \left(-\left\lceil \frac{N}{2} \right\rceil, \left\lfloor \frac{N}{2} \right\rfloor\right]$, more than half of the subsets $S \in \mathcal{S}$ containing n have $S \cap X = \{n\} \cap X$ (i.e., with probability $\geq \sigma$ every $n \in \left(-\left\lceil \frac{N}{2} \right\rceil, \left\lfloor \frac{N}{2} \right\rfloor\right]$ occurs separated from all (other) members of X in more than half of the $\mathcal{S}$ elements containing n).*

The existence of such sets is easy to see. For example, the collection of subsets

$$\mathcal{S} = \left\{ \{n\} \middle| n \in \left(-\left\lceil \frac{N}{2} \right\rceil, \left\lfloor \frac{N}{2} \right\rfloor \right] \right\}$$

consisting of all the singleton subsets of $\left(-\left\lceil \frac{N}{2} \right\rceil, \left\lfloor \frac{N}{2} \right\rfloor\right]$ is k-majority selective for all $k \leq N$. Generally, however, we are interested in creating k-majority selective collections which contain as few subsets as possible (i.e., much fewer than N subsets). We next give a construction for a k-majority selective collection of subsets for any $k, N \in \mathbb{N}$ with $k \leq N$. Our construction is motivated by the prime groupings techniques first employed in [40]. We begin as follows:

Define $p_0 = 1$ and let p_l be the l^{th} prime natural number. Thus, we have

$$p_0 = 1, p_1 = 2, p_2 = 3, p_3 = 5, p_4 = 7, \ldots$$

Choose $q, K \in \mathbb{N}$ (to be specified later). We are now ready to build a collection of subsets, $\mathcal{S}$. We begin by letting $S_{j,h}$ for all $0 \leq j \leq K$ and $0 \leq h \leq p_j - 1$ be

$$S_{j,h} = \left\{ n \in \left(-\left\lceil \frac{N}{2} \right\rceil, \left\lfloor \frac{N}{2} \right\rfloor \right] \middle| \, n \equiv h \bmod p_{q+j} \right\}. \tag{9}$$

Next, we progressively define S_j to be all integer residues mod p_{q+j}, i.e.,

$$S_j = \{S_{j,h} \mid h \in [0, p_{q+j})\}, \tag{10}$$

and conclude by setting $\mathcal{S}$ equal to all K such p_{q+j} residue groups:

$$\mathcal{S} = \cup_{j=0}^{K} S_j. \tag{11}$$

We now prove that $\mathcal{S}$ is indeed k-majority selective if K is chosen appropriately.

Lemma 1. *Fix k. If we set $K \geq 2k\lfloor \log_{p_q} N \rfloor$ then $\mathcal{S}$ as constructed above will be a k-majority selective collection of sets.*

Proof:

Let $X \subset \left(-\left\lceil \frac{N}{2} \right\rceil, \left\lfloor \frac{N}{2} \right\rfloor\right]$ be such that $|X| \leq k$. Furthermore, choose $n \in \left(-\left\lceil \frac{N}{2} \right\rceil, \left\lfloor \frac{N}{2} \right\rfloor\right]$ and let $x \in X$ be such that $x \neq n$. By the Chinese Remainder Theorem we know that x and n may only collide modulo at most $\lfloor \log_{p_q} N \rfloor$ of the $K + 1$ primes $p_{q+K} \geq \cdots \geq p_q$. Hence, n may collide with all the (other) elements of X (i.e., with $X - \{n\}$) modulo at most $k\lfloor \log_{p_q} N \rfloor$ $\mathcal{S}_j$-primes. We can now see that n will be isolated from all the (other) elements of X modulo at least $K + 1 - k\lfloor \log_{p_q} N \rfloor \geq k\lfloor \log_{p_q} N \rfloor + 1 > \frac{K+1}{2}$ $\mathcal{S}_j$-primes. Furthermore, n will appear in at most $K + 1$ of $\mathcal{S}$'s subsets. This leads us to the conclusion that $\mathcal{S}$ is indeed k-majority selective. □

Note that at least $\Omega(k)$ coprime integers are required in order to create a k-majority separating collection of subsets in this fashion. Given any $n \in \left(-\left\lceil \frac{N}{2} \right\rceil, \left\lfloor \frac{N}{2} \right\rfloor\right]$ a k element subset X can be created via the Chinese Remainder Theorem and n moduli so that every element of X collides with n in any desired $\Omega(1)$ $\mathcal{S}_j$-coprime numbers $\leq \frac{N}{2}$. Thus, it isn't possible to significantly decrease the number of relatively prime values required to construct k-majority separating collections using these arguments.

The number of coprime integers required to construct each k-majority separating collection is directly related to the number of signal samples required by our subsequent Fourier algorithms. Given that we depend on the number theoretic nature of our constructions in order to take advantage of aliasing phenomena, it is unclear how to reduce the sampling complexity for our deterministic Fourier methods below. However, this does not stop us from appealing to randomized number theoretic constructions in order to decrease the number of required coprime values (and, therefore, samples). We next present a construction for (k, σ)-majority selective collections which motives our subsequent Monte Carlo Fourier algorithms.

Lemma 2. *Fix k and an unknown $X \subset \left(-\left\lceil \frac{N}{2} \right\rceil, \left\lfloor \frac{N}{2} \right\rfloor\right]$ with $|X| \leq k$. We may form a (k, σ)-majority selective collection of subsets, $\mathcal{S}$, as follows: Set $K \geq 3k\lfloor \log_{p_q} N \rfloor$ and create $J \subset [q, q+K]$ by choosing $O\left(\log\left(\frac{N}{1-\sigma}\right)\right)$ elements from $[q, q+K]$ uniformly at random. Set $\mathcal{S} = \cup_{j \in J} S_j$ (see Equation 10).*

Proof:

Choose any $n \in \left(-\left\lceil \frac{N}{2} \right\rceil, \left\lfloor \frac{N}{2} \right\rfloor\right]$. A prime chosen uniformly at random from $\{p_q, \ldots, p_{q+K}\}$ will separate n from all (other) elements of X with probability at least $\frac{2}{3}$ (see proof of Lemma 1). Using the Chernoff bound we can see that choosing $O\left(\log\left(\frac{N}{1-\sigma}\right)\right)$ primes for J is sufficient to guarantee that the probability of n being congruent to any element of X modulo more than half of J's primes is less than $\frac{1-\sigma}{N}$. The union bound can now be employed to show that J's primes separate every element of $\left(-\left\lceil \frac{N}{2} \right\rceil, \left\lfloor \frac{N}{2} \right\rfloor\right]$ from the (other) elements of X with probability at least σ. $\square$

We conclude this section by bounding the number of subsets contained in our k-majority and (k, σ)-majority selective collections. These subset bounds will ultimately provide us with sampling and runtime bounds for our Fourier algorithms. The following theorem is easily proved using results from [31].

Lemma 3. *Choose q so that p_q is the smallest prime $\geq k$. If $\mathcal{S}$ is a k-majority selective collection of subsets created as per Lemma 1, then $|\mathcal{S}|$ is $\Theta\left(k^2 \cdot \log_k^2 N \cdot \log(k \log N)\right)$. If $\mathcal{S}$ is a $\left(k, 1 - \frac{1}{N^{O(1)}}\right)$-majority selective collection of subsets created as per Lemma 2, then $|\mathcal{S}|$ is $O\left(k \cdot \log_k N \cdot \log(k \log N) \cdot \log N\right)$.*

Let $\alpha \in (0, 1)$ be a constant, and suppose that $k = \Theta(N^\alpha)$. In this case, we have a construction for k-majority selective collections, $\mathcal{S}$, with $|\mathcal{S}| = \Theta\left(k^2 \cdot \log N\right)$. Furthermore, we have a construction for $\left(k, 1 - \frac{1}{N^{O(1)}}\right)$-majority selective collections, $\mathcal{S}$, with $|\mathcal{S}| = O\left(k \cdot \log^2 N\right)$.

4 Superlinear-Time Fourier Algorithms

For the remainder of the paper we will assume that $f : [0, 2\pi] \mapsto \mathbb{C}$ has the property that $\hat{f} \in L^1$. Our goal is to identify k of the most energetic frequencies in $\hat{f}$ (i.e., the first k entries in a valid ordering of $\hat{f}$ as in Equation 3) and then estimate their Fourier coefficients. Intuitively, we want f to be a continuous multiscale function. In this scenario our algorithms will allow us to ignore f's separation of scales and sample at a rate primarily dependent on the number of energetic frequencies present in f's Fourier spectrum.

Let $C \geq 1$ is a constant (to be specified later) and set

$$\epsilon = \frac{|\hat{f}(\omega_k)|}{C} \tag{12}$$

where B is the smallest integer such that

$$\sum_{b=B+1}^{\infty} |\hat{f}(\omega_b)| \leq \frac{\epsilon}{2}. \tag{13}$$

Note that B is defined to be the last possible significant frequency $\left(\text{i.e., with energy } > \text{ a fraction of } |\hat{f}(\omega_k)|\right)$. We will assume below that N is chosen large enough so that

$$\Omega = \{\omega_1, \ldots, \omega_B\} \subset \left(-\left\lceil \frac{N}{2} \right\rceil, \left\lfloor \frac{N}{2} \right\rfloor\right]. \tag{14}$$

We expect to work with multiscale signals so that $k \leq B \ll N$. Later we will give specific values for C and B depending on k, the desired approximation error, and $\hat{f}$'s compressibility characteristics. For now we show that we can identify/approximate k of $\hat{f}$'s largest magnitude entries each to within ϵ-precision via Algorithm 1.

Algorithm 1 SUPERLINEAR APPROXIMATE

1: **Input: Signal pointer** f**, integers** $k \leq B \leq N$
2: **Output:** $\hat{\mathbf{R}}^s$**, a sparse representation for** $\hat{f}$
3: Initialize $\hat{\mathbf{R}}^s \leftarrow \emptyset$
4: Set $K = 2B\lfloor \log_B N \rfloor$, q so that $p_{q-1} < B \leq p_q$
5: **for** j from 0 to K **do**
6: $\quad \mathbf{A}_{p_{q+j}} \leftarrow f(0), f\left(\frac{2\pi}{p_{q+j}}\right), \ldots, f\left(\frac{2\pi(p_{q+j}-1)}{p_{q+j}}\right)$
7: $\quad \widehat{\mathbf{A}_{p_{q+j}}} \leftarrow \mathbf{DFT}[\mathbf{A}_{p_{q+j}}]$
8: **end for**
9: **for** ω from $1 - \left\lceil \frac{N}{2} \right\rceil$ to $\left\lfloor \frac{N}{2} \right\rfloor$ **do**
10: $\quad \mathbb{Re}\{C_\omega\} \leftarrow$ median of multiset $\left\{ \mathbb{Re}\left\{ \widehat{\mathbf{A}_{p_{q+j}}}(\omega \bmod p_{q+j}) \right\} \;\middle|\; 0 \leq j \leq K \right\}$
11: $\quad \mathbb{Im}\{C_\omega\} \leftarrow$ median of multiset $\left\{ \mathbb{Im}\left\{ \widehat{\mathbf{A}_{p_{q+j}}}(\omega \bmod p_{q+j}) \right\} \;\middle|\; 0 \leq j \leq K \right\}$
12: **end for**
13: $\hat{\mathbf{R}}^s \leftarrow (\omega, C_\omega)$ entries for k largest magnitude C_ω's

Algorithm 1 works by using the k-majority separating structure created by the aliased DFTs in line 7 to isolate $\hat{f}$'s significantly energetic frequencies. Every DFT which successfully separates a frequency ω_j from all the (other) members of Ω will provide a good $\left(\text{i.e., within } \frac{\epsilon}{2} \leq \frac{|\hat{\mathbf{A}}(\omega_k)|}{2} \right)$ coefficient estimate for ω_j. Frequency separation occurs because more than $\frac{1}{2}$ of our aliased DFT's won't collide any $n \in \left(-\left\lceil \frac{N}{2} \right\rceil, \left\lfloor \frac{N}{2} \right\rfloor\right]$ with any (other) member of Ω (see Lemma 1). At most $B \log_B N$ of the DFT calculations for any particular frequency can be significantly contaminated via collisions with Ω members. Therefore, we can take medians of each frequency's associated $2B \log_B N + 1$ DFT residue's real/imaginary parts as a good estimate of that frequency coefficient's real/imaginary parts. Since more than half of these measurements must be accurate, the medians will be accurate. In order to formalize this argument we need the following lemma.

Lemma 4. *Every C_ω calculated in lines 10 and 11 is such that $|\hat{f}(\omega) - C_\omega| \leq \epsilon$.*

Proof:

Suppose that C_ω is calculated by lines 10 and 11. Then, it's real/imaginary part is given the median of K estimates of $\hat{f}(\omega)$'s real/imaginary parts. Each of these estimates is calculated by

$$\widehat{\mathbf{A}_{p_{q+j}}}(h) = \frac{2\pi}{p_{q+j}} \sum_{k=0}^{p_{q+j}-1} f\left(\frac{2\pi k}{p_{q+j}}\right) e^{\frac{-2\pi i h k}{p_{q+j}}} \tag{15}$$

for some $0 \leq j \leq K$, $0 \leq h < p_{q+j}$. Via aliasing each estimate reduces to

$$\widehat{\mathbf{A}_{p_{q+j}}}(h) = \frac{2\pi}{p_{q+j}} \sum_{k=0}^{p_{q+j}-1} f\left(\frac{2\pi k}{p_{q+j}}\right) e^{\frac{-2\pi i h k}{p_{q+j}}} = \frac{2\pi}{p_{q+j}} \sum_{k=0}^{p_{q+j}-1} \left(\frac{1}{2\pi} \sum_{\rho=-\infty}^{\infty} \hat{f}(\rho) e^{\frac{2\pi i \rho k}{p_{q+j}}} \right) e^{\frac{-2\pi i h k}{p_{q+j}}} \tag{16}$$

$$= \sum_{\omega=-\infty}^{\infty} \hat{f}(\rho) \left(\frac{1}{p_{q+j}} \sum_{k=0}^{p_{q+j}-1} e^{\frac{2\pi i (\rho - h) k}{p_{q+j}}} \right) = \sum_{\rho \equiv h \bmod p_{q+j}} \hat{f}(\rho) \tag{17}$$

$$= \left\langle \chi_{S_{j,h}}, \hat{f} \cdot \chi_{\left(-\lceil \frac{N}{2} \rceil, \lfloor \frac{N}{2} \rfloor\right]} \right\rangle + \sum_{\rho \equiv h \bmod p_{q+j}, \rho \notin \left(-\lceil \frac{N}{2} \rceil, \lfloor \frac{N}{2} \rfloor\right]} \hat{f}(\rho). \tag{18}$$

Thus, by Lemma 1 and Equations 13 and 14, more than half of our $\hat{f}(\omega)$ estimates will have

$$\left|\hat{f}(\omega) - \widehat{\mathbf{A}_{p_{q+j}}}(\omega \bmod p_{q+j})\right| \;\leq\; \sum_{\rho \notin \Omega} \left|\hat{f}(\rho)\right| \leq \frac{\epsilon}{2}.$$

It follows that taking medians as per lines 10 and 11 will result in the desired ϵ-accurate estimate for $\hat{f}(\omega)$. □

The following Theorem presents itself.

Theorem 1. *Let $\hat{\boldsymbol{R}}_{\text{opt}}$ be a k-optimal Fourier representation for our input function f's Fourier transform. Then, the k-term representation $\hat{\boldsymbol{R}}^{\text{s}}$ returned from Algorithm 1 is such that $\|\hat{f} - \hat{\boldsymbol{R}}\|_2^2 \leq \|\hat{f} - \hat{\boldsymbol{R}}_{\text{opt}}\|_2^2 + \frac{9k \cdot |\hat{f}(\omega_k)|^2}{C}$. Furthermore, Algorithm 1's runtime is $O\left(N \cdot B \cdot \frac{\log^2 N \cdot \log^2(B \log N)}{\log^2 B}\right)$. The number of f samples used is $\Theta\left(B^2 \cdot \log_B^2 N \cdot \log(B \log N)\right)$.*

Proof:

Choose any $b \in (0, k]$. Using Lemma 4 we can see that only way some $\omega_b \notin \hat{\mathbf{R}}_B^{\text{s}}$ is if there exists some associated $b' \in (k, N]$ so that $\omega_{b'} \in \hat{\mathbf{R}}^{\text{s}}$ and

$$|\hat{f}(\omega_k)| + \epsilon \geq |\hat{f}(\omega_{b'})| + \epsilon \geq |C_{\omega_{b'}}| \geq |C_{\omega_b}| \geq |\hat{f}(\omega_b)| - \epsilon \geq |\hat{f}(\omega_k)| - \epsilon.$$

In this case we'll have $2\epsilon > |\hat{f}(\omega_b)| - |\hat{f}(\omega_{b'})| \geq 0$ so that

$$|\hat{f}(\omega_{b'})|^2 + 4\epsilon\left(\epsilon + |\hat{f}(\omega_k)|\right) \geq |\hat{f}(\omega_{b'})|^2 + 4\epsilon\left(\epsilon + |\hat{f}(\omega_{b'})|\right) \geq |\hat{f}(\omega_b)|^2. \tag{19}$$

Now using Lemma 4 we can see that

$$\|\hat{f} - \hat{\mathbf{R}}\|^2 = \sum_{(\omega,\cdot) \notin \hat{\mathbf{R}}^{\text{s}}} |\hat{f}(\omega)|^2 + \sum_{(\omega, C_\omega) \in \hat{\mathbf{R}}^{\text{s}}} |\hat{f}(\omega) - C_\omega|^2 \leq \sum_{(\omega,\cdot) \notin \hat{\mathbf{R}}^{\text{s}}} |\hat{\mathbf{A}}(\omega)|^2 + k \cdot \epsilon^2.$$

Furthermore, we have

$$k \cdot \epsilon^2 + \sum_{(\omega,\cdot) \notin \hat{\mathbf{R}}^{\text{s}}} |\hat{f}(\omega)|^2 = k \cdot \epsilon^2 + \sum_{b \in (0,k],\ \omega_b \notin \hat{\mathbf{R}}^{\text{s}}} |\hat{f}(\omega_b)|^2 + \sum_{b' \in (k,N],\ \omega_{b'} \notin \hat{\mathbf{R}}^{\text{s}}} |\hat{f}(\omega_{b'})|^2.$$

Using observation 19 above we can see that this last expression is bounded above by

$$k \cdot (5\epsilon^2 + 4\epsilon|\hat{f}(\omega_k)|) + \sum_{b' \in (k,N],\ \omega_{b'} \in \hat{\mathbf{R}}^{\text{s}}} |\hat{f}(\omega_{b'})|^2 + \sum_{b' \in (k,N],\ \omega_{b'} \notin \hat{\mathbf{R}}^{\text{s}}} |\hat{f}(\omega_{b'})|^2 \leq \|\hat{f} - \hat{\mathbf{R}}_{\text{opt}}\|_2^2 + k \cdot (5\epsilon^2 + 4\epsilon|\hat{f}(\omega_k)|).$$

Substituting for ϵ (see Equation 12) gives us our result. Mainly,

$$k \cdot (5\epsilon^2 + 4\epsilon|\hat{f}(\omega_k)|) = \frac{k|\hat{f}(\omega_k)|^2}{C}\left(\frac{5}{C} + 4\right) \leq \frac{9k|\hat{f}(\omega_B)|^2}{C}.$$

To finish, we provide sampling/runtime bounds. Algorithm 1's lines 5 through 8 take $\Theta\left(B^2 \cdot \frac{\log^2 N \cdot \log^2(B \log N)}{\log^2 B}\right)$ time using the Chirp z-Transform [4, 43] (see [31] for details). Lines 9 through 13 can be accomplished in $O\left(N \cdot B \log_B N \cdot \log(B \log N)\right)$ time. Algorithm 1's sampling complexity follows directly from Lemma 3. □

It's not difficult to see that the proofs of Lemma 4 and Theorem 1 still hold using the (k, σ)-majority selective properties of randomly chosen primes. In particular, if we run Algorithm 1 using randomly chosen primes along the lines of Lemma 2 then Theorem 1 will still hold whenever the primes behave in a majority selective fashion. The only change required to Algorithm 1 is that we compute only a random subset of the DFTs in lines 5 through 8. We have the following corollary.

Corollary 1. *Let $\hat{\boldsymbol{R}}_{\text{opt}}$ be a k-optimal Fourier representation for our input function f's Fourier transform. If we run Algorithm 1 using $O\left(\log\left(\frac{N}{1-\sigma}\right)\right)$ randomly selected primes along the lines of Lemma 2, then with probability at least σ we will obtain a k-term representation $\hat{\boldsymbol{R}}^{\text{s}}$ having $\|\hat{f} - \hat{\boldsymbol{R}}\|_2^2 \leq \|\hat{f} - \hat{\boldsymbol{R}}_{\text{opt}}\|_2^2 + \frac{9k\cdot|\hat{f}(\omega_k)|^2}{C}$. The runtime will be $O\left(N \cdot \log_B N \cdot \log\left(\frac{N}{1-\sigma}\right) \cdot \log^2\left(B\log\left(\frac{N}{1-\sigma}\right)\right)\right)$. The number of f samples will be $O\left(B \cdot \log_B N \cdot \log(B\log N) \cdot \log\left(\frac{N}{1-\sigma}\right)\right)$.*

It has been popular in the compressed sensing literature to consider the recovery of k-frequency superpositions (see [33] and references therein). Suppose we have

$$f(x) = \sum_{b=1}^{k} C_b \cdot e^{\mathbf{i}\omega_b x} \text{ for all } x \in [0, 2\pi],\ \Omega = \{\omega_1, \ldots, \omega_k\} \subset \left(-\left\lceil\frac{N}{2}\right\rceil, \left\lfloor\frac{N}{2}\right\rfloor\right]. \tag{20}$$

Setting $B = k$ and $C = 1$ is then sufficient to guarantee that $\sum_{b=B+1}^{\infty} |\hat{f}(\omega_b)| = 0$. Theorem 1 now tells us that Algorithm 1 will perfectly reconstruct f. We quickly obtain the final result of this section.

Corollary 2. *Suppose f is a k-frequency superposition. Then, Algorithm 1 can exactly recover f in $O\left(N \cdot k \cdot \frac{\log^2 N \cdot \log^2(k\log N)}{\log^2 k}\right)$ time. The number of f samples used is $\Theta\left(k^2 \cdot \log_k^2 N \cdot \log(k\log N)\right)$. If we run Algorithm 1 using $O\left(\log\left(\frac{N}{1-\sigma}\right)\right)$ randomly selected primes along the lines of Lemma 2, then we will exactly recover f with probability at least σ. In this case the runtime will be $O\left(N \cdot \log_k N \cdot \log\left(\frac{N}{1-\sigma}\right) \cdot \log^2\left(k\log\left(\frac{N}{1-\sigma}\right)\right)\right)$. The number of f samples will be $O\left(k \cdot \log_k N \cdot \log(k\log N) \cdot \log\left(\frac{N}{1-\sigma}\right)\right)$.*

As before, let $\alpha \in (0, 1)$ be a constant and suppose that $k = \Theta(N^{\alpha})$. Furthermore, let $\sigma = 1 - \frac{1}{N^{O(1)}}$. Corollary 2 implies that our deterministic Algorithm 1 exactly recovers k-frequency superpositions using $O(k^2 \log N)$ samples. If randomly selected primes are used then Algorithm 1 can exactly reconstruct k-frequency superpositions with probability $1 - \frac{1}{N^{O(1)}}$ using $O(k \log^2 N)$ samples. In this case our randomized Algorithm 1's sampling complexity is within a logarithmic factor of the best known Fourier sampling bounds concerning high probability exact recovery of superpositions [6, 33]. This is encouraging given Algorithm 1's simplicity. Of greater interest for our purposes here, however, is that Algorithm 1 can be easily modified to run in sublinear time.

5 Sublinear-Time Fourier Algorithms

In order to reduce Algorithm 1's runtime we will once again utilize the combinatorial properties of line 7's aliased DFTs. If we can correctly identify any energetic frequencies that are isolated from the other elements of Ω by any given line 7 DFT, we will be guaranteed to recover all energetic frequencies more than $\frac{K}{2}$ times. Thus, collecting all frequencies recovered from more than half of line 7's DFTs will give us the k most energetic Ω frequencies (along with some possibly 'junk frequencies'). The 'junk' can be discarded, however, by using our existing coefficient estimation method (lines 9 - 13) on the collected potentially energetic frequencies. Only truly energetic frequencies will yield large magnitude coefficient estimates by Lemma 4. Finally, note that only $O(K \log K)$ potentially energetic frequencies may be recovered more than $\frac{K}{2}$ times via line 7's DFTs. Thus, our formally superlinear-time loop (lines 9 - 12) will be sublinearized.

In order to correctly identify energetic frequencies isolated by any Algorithm 1 DFT we will utilize a procedure along the lines of Cormode and Muthukrishnan's CS reconstruction method [41, 9, 10]. However, in order to take advantage of aliasing, we will utilize a Chinese Remainder Theorem based identification procedure instead of CM's Hamming code based bit testing. For a simple illustration of how our method works in the single frequency case see [29, 31]. Algorithm 2 is the sublinear-time algorithm obtained by modifying Algorithm 1 as outlined above.

Let m be the smallest integer such that

$$\prod_{l=0}^{m} p_l \geq \frac{N}{B}. \tag{21}$$

The following lemma establishes the correctness of Algorithm 2's energetic frequency identification procedure.

Algorithm 2 SUBLINEAR APPROXIMATE

1: **Input: Signal pointer** f**, integers** $m, k \leq B \leq N$
2: **Output:** $\hat{\mathbf{R}}^{s}$**, a sparse representation for** $\hat{f}$
3: Initialize $\hat{\mathbf{R}}^{s} \leftarrow \emptyset$
4: Set $K = 2B\lfloor \log_B N \rfloor$, q so that $p_{q-1} \leq \max(B, p_m) \leq p_q$
5: **for** j from 0 to K **do**
6: **for** l from 0 to m **do**
7: $\mathbf{A}_{p_l \cdot p_{q+j}} \leftarrow f(0), f\left(\frac{2\pi}{p_l \cdot p_{q+j}}\right), \ldots, f\left(\frac{2\pi(p_l \cdot p_{q+j}-1)}{p_l \cdot p_{q+j}}\right)$
8: $\widehat{\mathbf{A}_{p_l \cdot p_{q+j}}} \leftarrow \mathbf{DFT}[\mathbf{A}_{p_l \cdot p_{q+j}}]$
9: **end for**
10: **end for**

ENERGETIC FREQUENCY IDENTIFICATION

11: **for** j from 0 to K **do**
12: $\hat{A}_{\text{sort}} \leftarrow$ Sort $\widehat{\mathbf{A}_{p_0 \cdot p_{q+j}}}$ by magnitude (i.e., b^{th} largest magnitude entry in $\hat{A}_{\text{sort}}(b)$)
13: **for** b from 1 to B **do**
14: $r_{0,b} \leftarrow$ index of $\widehat{\mathbf{A}_{p_0 \cdot p_{q+j}}}$'s b^{th} largest magnitude entry $\left(\text{i.e., } \hat{A}_{\text{sort}}(b)\text{'s associated residue mod } p_{q+j}\right)$
15: **for** l from 1 to m **do**
16: $t_{\min} \leftarrow \min_{t \in [0, p_l)} \left| \hat{A}_{\text{sort}}(b) - \widehat{\mathbf{A}_{p_l \cdot p_{q+j}}}(t \cdot p_{q+j} + r_{0,b}) \right|$
17: $r_{l,b} \leftarrow \left(r_{0,b} + t_{\min} \cdot p_{q+j}\right) \bmod p_l$
18: **end for**
19: Construct $\omega_{j,b}$ from $r_{0,b}, \ldots, r_{m,b}$ via modular arithmetic
20: **end for**
21: **end for**
22: Sort $\omega_{j,b}$'s maintaining duplicates and set $C(\omega_{j,b})$ = the number of times $\omega_{j,b}$ was constructed via line 19

COEFFICIENT ESTIMATION

23: **for** j from 1 to K **do**
24: **for** b from 1 to B **do**
25: **if** $C(\omega_{j,b}) > \frac{K}{2}$ **then**
26: $\mathbb{Re}\left\{C_{\omega_{j,b}}\right\} \leftarrow \text{median of multiset}\left\{\mathbb{Re}\left\{\widehat{\mathbf{A}_{p_m \cdot p_{q+h}}}(\omega_{j,b} \bmod p_m \cdot p_{q+h})\right\} \;\middle|\; 0 \leq h \leq K\right\}$
27: $\mathbb{Im}\left\{C_{\omega_{j,b}}\right\} \leftarrow \text{median of multiset}\left\{\mathbb{Im}\left\{\widehat{\mathbf{A}_{p_m \cdot p_{q+h}}}(\omega_{j,b} \bmod p_m \cdot p_{q+h})\right\} \;\middle|\; 0 \leq h \leq K\right\}$
28: **end if**
29: **end for**
30: **end for**
31: $\hat{\mathbf{R}}^{s} \leftarrow (\omega_{j,b}, C_{\omega_{j,b}})$ entries for k largest magnitude $C_{\omega_{j,b}}$'s

Lemma 5. *Lines 11 through 22 of Algorithm 2 are guaranteed to recover all valid* $\omega_1, \ldots, \omega_k$ *(i.e., all* ω *with* $|\hat{\mathbf{A}}(\omega)|_2 \geq |\hat{\mathbf{A}}(\omega_k)|_2$ *- there may be* $> k$ *such entries) more then* $\frac{K}{2}$ *times. Hence, despite line 25, an entry for all such* $\omega_b, 1 \leq b \leq k$*, will be added to* $\hat{\mathbf{R}}^{s}$ *in line 31.*

Proof:

Fix $b \in [1, k]$. By Lemma 1 we know that there exist more than $\frac{K}{2}$ p_{q+j}-primes that isolate ω_b from all of $\Omega - \{\omega_b\}$. Denote these primes by

$$p_{j_1}, p_{j_2}, \ldots, p_{j_{K'}}, \quad \frac{K}{2} < K' \leq K.$$

We next show, for each $k' \in [1, K']$, that we get $\widehat{\mathbf{A}_{p_0 \cdot p_{j_{k'}}}}(\omega_b \bmod p_{j_{k'}})$ as one of the B largest magnitude entries found

in line 12. Choose any $k' \in [1, K']$. Using Equations 12 and 13 we can see that

$$\frac{\epsilon}{2} \;\leq\; |\hat{f}(\omega_k)| - \sum_{b'=B+1}^{\infty} |\hat{f}(\omega_{b'})| \;\leq\; |\hat{f}(\omega_b)| - \left| \sum_{b' \notin \Omega,\ \omega_{b'} \equiv \omega_b} \hat{f}(\omega_{b'}) \right| \;\leq\; \left| \widehat{\mathbf{A}_{p_0 \cdot p_{j_{k'}}}}(\omega_b \bmod p_{j_{k'}}) \right|.$$

We also know that the $(B+1)^{\text{st}}$ largest magnitude entry of $\widehat{\mathbf{A}_{p_0 \cdot p_{j_{k'}}}}$ must be $\leq \frac{\epsilon}{2}$. Hence, we are guaranteed to execute lines 13-20 with an $r_{0,\cdot} = \omega_b \bmod p_{j_{k'}}$.

Next, choose any $l \in [1, m]$ and set

$$\bar{\Omega}' = \left\{ \omega_{b'} \;\middle|\; \omega_{b'} \notin \Omega,\ \omega_{b'} \equiv \omega_b \bmod p_{j_{k'}},\ \omega_{b'} \not\equiv \omega_b \bmod p_l p_{j_{k'}} \right\}.$$

Line 16 inspects all the necessary residues of $\omega_b \bmod p_l p_{j_{k'}}$ since

$$\omega_b \equiv h \bmod p_{j_{k'}} \longrightarrow \omega_b \equiv h + t \cdot p_{j_{k'}} \bmod p_l p_{j_{k'}}$$

for some $t \in [0, p_l)$. To see that t_{min} will be chosen correctly we note first that

$$\left| \widehat{\mathbf{A}_{p_0 \cdot p_{j_{k'}}}}(\omega_b \bmod p_{j_{k'}}) - \widehat{\mathbf{A}_{p_l \cdot p_{j_{k'}}}}(\omega_b \bmod p_l p_{j_{k'}}) \right| \;\leq\; \sum_{\omega_{b'} \in \bar{\Omega}'} |\hat{f}(\omega_{b'})| \leq \frac{\epsilon}{2} \leq |\hat{f}(\omega_k)| - \sum_{b'=B+1}^{\infty} |\hat{f}(\omega_{b'})|.$$

Furthermore, setting $r_{0,\cdot} = \omega_b \bmod p_{j_{k'}}$ and

$$\tilde{\Omega}' = \left\{ \omega_{b'} \;\middle|\; \omega_{b'} \notin \Omega,\ \omega_{b'} \equiv \omega_b \bmod p_{j_{k'}},\ \omega_{b'} \not\equiv (r_{0,\cdot} + t p_{j_{k'}}) \bmod p_{j_{k'}} p_l \text{ for some } t \text{ with } (r_{0,\cdot} + t p_{j_{k'}}) \not\equiv \omega_b \bmod p_l p_{j_{k'}} \right\},$$

we have

$$|\hat{f}(\omega_k)| - \sum_{b'=B+1}^{\infty} |\hat{f}(\omega_{b'})| \;\leq\; |\hat{f}(\omega_b)| - \left| \sum_{\omega_{b'} \in \tilde{\Omega}'} \hat{f}(\omega_{b'}) \right| \;\leq\; \left| \widehat{\mathbf{A}_{p_0 \cdot p_{j_{k'}}}}(\omega_b \bmod p_{j_{k'}}) - \widehat{\mathbf{A}_{p_l \cdot p_{j_{k'}}}}\left((r_{0,\cdot} + t q_{j_{k'}}) \not\equiv \omega_b \bmod p_l q_{j_{k'}} \right) \right|.$$

Hence, lines 16 and 17 will indeed select the correct residue for ω_b modulo p_l. And, line 19 will correctly reconstruct ω_b at least $K' > \frac{K}{2}$ times. □

Using Lemma 5 along with Lemma 4 and Theorem 1 we obtain the following Theorem concerning Algorithm 2. The sampling and runtime bounds are computed in [29, 31].

Theorem 2. *Let $\hat{\boldsymbol{R}}_{\text{opt}}$ be a k-optimal Fourier representation for our input function f's Fourier transform. Then, the k-term representation $\hat{\boldsymbol{R}}^{\text{s}}$ returned from Algorithm 2 is such that $\|\hat{f} - \hat{\boldsymbol{R}}\|_2^2 \leq \|\hat{f} - \hat{\boldsymbol{R}}_{\text{opt}}\|_2^2 + \frac{9k \cdot |\hat{f}(\omega_k)|^2}{C}$. Furthermore, Algorithm 2's runtime is $O\left(B^2 \cdot \frac{\log^2 N \cdot \log^2(B \log N) \cdot \log^2 \frac{N}{B}}{\log^2 B \cdot \log\log \frac{N}{B}}\right)$. The number of f samples used is $O\left(B^2 \cdot \frac{\log^2 N \cdot \log(B \log N) \cdot \log^2 \frac{N}{B}}{\log^2 B \cdot \log\log \frac{N}{B}}\right)$.*

Also, as above, if we run Algorithm 2 using randomly chosen p_{q+j}-primes along the lines of Lemma 2 then Theorem 2 will still hold whenever the p_{q+j}-primes behave in a majority selective fashion. We have the following corollary.

Corollary 3. *Let $\hat{\boldsymbol{R}}_{\text{opt}}$ be a k-optimal Fourier representation for our input function f's Fourier transform. If we run Algorithm 2 using $O\left(\log\left(\frac{N}{1-\sigma}\right)\right)$ randomly selected p_{q+j}-primes along the lines of Lemma 2, then with probability at least σ we will obtain a k-term representation $\hat{\boldsymbol{R}}^{\text{s}}$ having $\|\hat{f} - \hat{\boldsymbol{R}}\|_2^2 \leq \|\hat{f} - \hat{\boldsymbol{R}}_{\text{opt}}\|_2^2 + \frac{9k \cdot |\hat{f}(\omega_k)|^2}{C}$. The runtime will be $O\left(B \cdot \frac{\log N \cdot \log\left(\frac{N}{1-\sigma}\right) \cdot \log^2\left(B \log\left(\frac{N}{1-\sigma}\right)\right) \cdot \log^2 \frac{N}{B}}{\log B \cdot \log\log \frac{N}{B}}\right)$. The number of f samples will be $O\left(B \cdot \frac{\log^2\left(\frac{N}{1-\sigma}\right) \cdot \log(B \log N) \cdot \log^2 \frac{N}{B}}{\log B \cdot \log\log \frac{N}{B}}\right)$.*

Let $\alpha \in (0,1)$ be a constant and suppose that $k = \Theta(N^{\alpha})$. Furthermore, suppose that $\sigma = 1 - \frac{1}{N^{O(1)}}$. Theorem 2 tells us that our sublinear-time deterministic Algorithm 2 exactly recovers k-frequency superpositions in $O\left(k^2 \cdot \frac{\log^4 N}{\log\log N}\right)$ time using $O\left(k^2 \cdot \frac{\log^3 N}{\log\log N}\right)$ samples. If randomly selected p_{q+j}-primes are used then Algorithm 2 can exactly reconstruct k-frequency superpositions with probability $1 - \frac{1}{N^{O(1)}}$ in $O\left(k \cdot \frac{\log^5 N}{\log\log N}\right)$ time using $O\left(k \cdot \frac{\log^4 N}{\log\log N}\right)$ samples. It is worth noting here that previous randomized sublinear-time Fourier results [23, 24] do not yield exact reconstructions of sparse superpositions in this manner. They iteratively produce approximate solutions which converge to the true superposition in the limit.

We are now ready to give sublinear-time results concerning functions with compressible Fourier coefficients. For the remainder of this paper we will assume that our input function $f : [0, 2\pi] \mapsto \mathbb{C}$ has both (i) an integrable p^{th} derivative, and (ii) $f(0) = f(2\pi), f'(0) = f'(2\pi), \ldots, f^{(p-2)}(0) = f^{(p-2)}(2\pi)$ for some $p > 1$. Standard Fourier coefficient bounds then imply that $\hat{f}$ is a p-compressible ∞-length signal [20, 5]. Before applying Theorem 2 we will determine Algorithm 2's B and Equation 12's C variables based on the desired Fourier representation's size and accuracy. Moving toward that goal, we note that since $\hat{f}$ is algebraically compressible we have

$$\frac{9k \cdot |\hat{f}(\omega_k)|^2}{C} = \frac{1}{C} O\left(k^{-2p+1}\right) = O\left(\frac{1}{C}\right) \|C_k^{\text{opt}}\|_2^2. \tag{22}$$

Thus, we should use $C = O\left(\frac{1}{\delta}\right)$ and a B so that

$$\sum_{b=B+1}^{\infty} |\hat{f}(\omega_b)| = O(B^{1-p}) = O(\delta \cdot |\hat{f}(\omega_k)|) = O(\delta \cdot k^{-p}). \tag{23}$$

Solving, we get that $B = O\left(\delta^{\frac{1}{1-p}} k^{\frac{p}{p-1}}\right)$. Applying Theorem 2 gives us Algorithm 2's runtime and number of required measurements. We obtain the following Corollary.

Corollary 4. *Let $f : [0, 2\pi] \mapsto \mathbb{C}$ have (i) an integrable p^{th} derivative, and (ii) $f(0) = f(2\pi), \ldots, f^{(p-2)}(0) = f^{(p-2)}(2\pi)$ for some $p > 1$. Furthermore, assume that $\hat{f}$'s $B = O\left(\delta^{\frac{1}{1-p}} k^{\frac{p}{p-1}}\right)$ largest magnitude frequencies all belong to $\left(-\left\lceil \frac{N}{2} \right\rceil, \left\lfloor \frac{N}{2} \right\rfloor\right]$. Then, we may use Algorithm 2 to return a k-term sparse Fourier representation, $\hat{\boldsymbol{R}}^{\text{s}}$, for $\hat{f}$ with $\|\hat{f} - \hat{\boldsymbol{R}}\|_2^2 \leq \|\hat{f} - \hat{\boldsymbol{R}}_{\text{opt}}\|_2^2 + \delta \|C_k^{\text{opt}}\|_2^2$ in $O\left(\delta^{\frac{2}{1-p}} k^{\frac{2p}{p-1}} \cdot \frac{\log^6 N}{\log^2 \frac{kp}{\delta}}\right)$ time. The number of f samples used is $O\left(\delta^{\frac{2}{1-p}} k^{\frac{2p}{p-1}} \cdot \frac{\log^5 N}{\log^2 \frac{kp}{\delta}}\right)$. If we run Algorithm 2 using $O\left(\log\left(\frac{N}{1-\sigma}\right)\right)$ randomly selected p_{q+j}-primes along the lines of Lemma 2, then with probability at least σ we will obtain a k-term representation $\hat{\boldsymbol{R}}^{\text{s}}$ having $\|\hat{f} - \hat{\boldsymbol{R}}\|_2^2 \leq \|\hat{f} - \hat{\boldsymbol{R}}_{\text{opt}}\|_2^2 + \delta \|C_k^{\text{opt}}\|_2^2$ in $O\left(\delta^{\frac{1}{1-p}} k^{\frac{p}{p-1}} \cdot \frac{\log^6 N}{\log \frac{kp}{\delta}}\right)$ time. The number of f samples used is $O\left(\delta^{\frac{1}{1-p}} k^{\frac{p}{p-1}} \cdot \frac{\log^5 N}{\log \frac{kp}{\delta}}\right)$.*

If $f : [0, 2\pi] \to \mathbb{C}$ is smooth (i.e., has infinitely many continuous derivatives on the unit circle where 0 is identified with 2π) it follows from Corollary 4 that Algorithm 2 can be used to find an δ-accurate, with $\delta = O\left(\frac{1}{N}\right)$, sparse k-term Fourier representation for $\hat{f}$ in $O(k^2 \log^6 N)$ time using $O(k^2 \log^5 N)$ measurements. If randomly selected p_{q+j}-primes are utilized then Algorithm 2 can obtain a $O\left(\frac{1}{N}\right)$-accurate k-term Fourier representation for $\hat{f}$ with high probability in $O(k \log^6 N)$ time using $O(k \log^5 N)$ measurements. Similarly, standard results concerning the exponential decay of Fourier coefficients for functions with analytic extensions can be used to generate exponentially compressible Fourier results.

6 Discrete Fourier Results

Suppose we are provided with an array $\mathbf{A}$ containing N equally spaced samples from an unknown smooth function $f : [0, 2\pi] \to \mathbb{C}$ (i.e., $\mathbf{A}$'s band-limited interpolent). Hence,

$$\mathbf{A}(j) = f\left(\frac{2\pi j}{N}\right),\ j \in [0, N). \tag{24}$$

We would like to use Algorithm 2 to find a sparse Fourier representation for $\hat{\mathbf{A}}$. Not having access to f directly, and restricting ourselves to sublinear time approaches only, we have little recourse but to locally interpolate f around Algorithm 2's required samples.

For each required Algorithm 2 f-sample at $t = \frac{2\pi h}{p_{q+j}p_l}, h \in [0, p_{q+j}p_l)$, we may approximate $f(t)$ to within $O\left(N^{-2\kappa}\right)$ error by constructing 2 local interpolents (one real, one imaginary) around t using $\mathbf{A}$'s nearest 2κ entries [22]. These errors in f-samples can lead to errors of size $O\left(N^{-2\kappa} \cdot p_m p_{q+K} \log p_{q+K}\right)$ in each of Algorithm 2 line 8's DFT entries. However, as long as these errors are small enough (i.e., of size $O(\delta \cdot k^{-p})$ in the p-compressible case) Theorem 2 and all related Section 5 results and will still hold. Hence, using $2\kappa = O\left(\log\left(\delta^{-1} \cdot k^p\right)\right)$ interpolation points per f-sample will be sufficient. We have the following result.

Corollary 5. *Let $\boldsymbol{A}$ be an N-length complex valued array and suppose that $\hat{\boldsymbol{A}}$ is p-compressible. Then, we may use Algorithm 2 to return a k-term sparse Fourier representation, $\hat{\boldsymbol{R}}^{\text{s}}$, for $\hat{\boldsymbol{A}}$ with $\|\hat{\boldsymbol{A}} - \hat{\boldsymbol{R}}\|_2^2 \leq \|\hat{\boldsymbol{A}} - \hat{\boldsymbol{R}}_{\text{opt}}\|_2^2 + \delta\|C_k^{\text{opt}}\|_2^2$ in $O\left(\delta^{\frac{2}{1-p}} k^{\frac{2p}{p-1}} \cdot \frac{\log^6 N}{\log \frac{k^p}{\delta}}\right)$ time. The number of samples used is $O\left(\delta^{\frac{2}{1-p}} k^{\frac{2p}{p-1}} \cdot \frac{\log^5 N}{\log \frac{k^p}{\delta}}\right)$. If we run Algorithm 2 using $O\left(\log\left(\frac{N}{1-\sigma}\right)\right)$ randomly selected p_{q+j}-primes along the lines of Lemma 2, then with probability at least σ we will obtain a k-term representation $\hat{\boldsymbol{R}}^{\text{s}}$ having $\|\hat{\boldsymbol{A}} - \hat{\boldsymbol{R}}\|_2^2 \leq \|\hat{\boldsymbol{A}} - \hat{\boldsymbol{R}}_{\text{opt}}\|_2^2 + \delta\|C_k^{\text{opt}}\|_2^2$ in $O\left(\delta^{\frac{1}{1-p}} k^{\frac{p}{p-1}} \cdot \log^6 N\right)$ time. The number of $\boldsymbol{A}$ samples used is $O\left(\delta^{\frac{1}{1-p}} k^{\frac{p}{p-1}} \cdot \log^5 N\right)$.*

Notice that Corollary 5 doesn't guarantee the exact recovery of k-frequency superpositions in the discrete setting. Generally, the sparse Fourier representations produced by Algorithm 2 on discrete data will always contain interpolation errors. However, for $\delta = \Theta\left(N^{-1}\right)$, we can still consider smooth data $\mathbf{A}$ to be $\Theta(\log N)$-compressible and so achieve an accurate $\tilde{O}(k^2)$-time DFT algorithm for large N.

7 Conclusion

In this paper the first known deterministic Fourier algorithm with both sublinear-time sampling and runtime complexity was developed. Hence, we have established the first deterministic algorithm which can exactly reconstruct a k-frequency superposition using time polynomial in the superposition's *information content*. When viewed from this perspective the following open problem presents itself.

Open Problem 1. *Construct (or show the impossibility of constructing) a deterministic Fourier algorithm guaranteed to exactly recover k-frequency superpositions in $k \cdot \log^{O(1)} N$ time.*

The status of current methods with respect to Problem 1 is as follows: Gilbert, Muthukrishnan, and Strauss' randomized Fourier algorithm [24] achieves a near optimal runtime, but is neither deterministic nor exact. Similarly, our Section 5 Monte Carlo algorithm achieves exact reconstruction and a near optimal runtime, but isn't deterministic. Linear programming [13, 6] and OMP-based [42] methods achieve deterministic sampling sets of acceptable size [44, 14], but both the verification of the sampling sets deterministic properties and the associated reconstruction procedures are computationally taxing. Finally, Indyk's fast deterministic CS procedure [28] obtains a promising reconstruction runtime, but doesn't allow fast Fourier measurement acquisition.

In terms of applications, there are two compelling motivations for developing fast sparse Fourier transform methods along the lines of [23, 24] and Algorithm 2: runtime and sample usage. In numerical applications such as [11] where runtime is the dominant concern we must assume that our input function f exhibits some multiscale behavior. If $\hat{f}$

contains no unpredictably energetic and large (relative to the number of desired Fourier coefficients) frequencies then it is more computationally efficient to simply use standard FFT/NUFFT methods [8, 35, 3, 16, 19]. In other applications [34, 32, 36, 37] where sampling costs are of greater concern than reconstruction runtime, even mild oversampling for the sake of faster reconstruction may be unacceptable. In such cases the runtime/sampling tradeoff must be carefully weighed.

References

[1] Compressed sensing resources. http://www.dsp.ece.rice.edu/cs/.

[2] l_1-magic website. http://www.acm.caltech.edu/l1magic/.

[3] C. Anderson and M. D. Dahleh. Rapid computation of the discrete Fourier transform. *SIAM J. Sci. Comput.*, 17:913–919, 1996.

[4] L. I. Bluestein. A Linear Filtering Approach to the Computation of Discrete Fourier Transform. *IEEE Transactions on Audio and Electroacoustics*, 18:451–455, 1970.

[5] J. P. Boyd. *Chebyshev and Fourier Spectral Methods*. Dover Publications, Inc., 2001.

[6] E. Candes, J. Romberg, and T. Tao. Robust uncertainty principles: Exact signal reconstruction from highly incomplete frequency information. *IEEE Trans. Inform. Theory*, 52:489–509, 2006.

[7] B. Chazelle. *The Discrepancy Method: Randomness and Complexity*. Brooks/Cole Publishing Company, 1992.

[8] J. Cooley and J. Tukey. An algorithm for the machine calculation of complex Fourier series. *Math. Comput.*, 19:297–301, 1965.

[9] G. Cormode and S. Muthukrishnan. Combinatorial Algorithms for Compressed Sensing. *Technical Report DIMACS TR 2005-40*, 2005.

[10] G. Cormode and S. Muthukrishnan. Combinatorial Algorithms for Compressed Sensing. *Conference on Information Sciences and Systems*, March 2006.

[11] I. Daubechies, O. Runborg, and J. Zou. A sparse spectral method for homogenization multiscale problems. *Multiscale Model. Sim.*, 2007.

[12] R. A. DeVore. Deterministic constructions of compressed sensing matrices. *http://www.ima.umn.edu/2006-2007/ND6.4-15.07/activities/DeVore-Ronald/Henrykfinal.pdf*, 2007.

[13] D. Donoho. Compressed Sensing. *IEEE Trans. on Information Theory*, 52:1289–1306, 2006.

[14] D. L. Donoho and J. Tanner. Thresholds for the recovery of sparse solutions via l1 minimization. In *40th Annual Conference on Information Sciences and Systems (CISS)*, 2006.

[15] D. Z. Du and F. K. Hwang. *Combinatorial Group Testing and Its Applications*. World Scientific, 1993.

[16] A. Dutt and V. Rokhlin. Fast Fourier transforms for nonequispaced data. *SIAM J. Sci. Comput.*, 14:1368–1383, 1993.

[17] D. Eppstein, M. T. Goodrich, and D. S. Hirschberg. Improved combinatorial group testing algorithms for real-world problem sizes, May 2005.

[18] L. Y. Erich Kaltofen. Improved sparse multivariate polynomial interpolation algorithms. *International Symposium on Symbolic and Algebraic Computation*, 1988.

[19] J. A. Fessler and B. P. Sutton. Nonuniform Fast fourier transforms using min-max interpolation. *IEEE Trans. Signal Proc.*, 51:560–574, 2003.

[20] G. B. Folland. *Fourier Analysis and Its Applications*. Brooks/Cole Publishing Company, 1992.

[21] S. Ganguly and A. Majumder. CR-precis: A deterministic summary structure for update data streams. *ArXiv Computer Science e-prints*, Sept. 2006.

[22] C. F. Gerald and P. O. Wheatley. *Applied Numerical Analysis*. Addison-Wesley Publishing Company, 1994.

[23] A. Gilbert, S. Guha, P. Indyk, S. Muthukrishnan, and M. Strauss. Near-optimal sparse Fourier estimation via sampling. *ACM STOC*, pages 152–161, 2002.

[24] A. Gilbert, S. Muthukrishnan, and M. Strauss. Improved time bounds for near-optimal sparse Fourier representations. *SPIE*, 2005.

[25] A. C. Gilbert and M. J. Strauss. Group testing in statistical signal recovery. *submitted*, 2006.

[26] A. C. Gilbert, M. J. Strauss, J. A. Tropp, and R. Vershynin. Algorithmic linear dimension reduction in the l_1 norm for sparse vectors. *submitted*, 2006.

[27] P. Indyk. Explicit constructions of selectors and related combinatorial structures, with applications. In *SODA '02: Proceedings of the thirteenth annual ACM-SIAM symposium on Discrete algorithms*, pages 697–704, Philadelphia, PA, USA, 2002. Society for Industrial and Applied Mathematics.

[28] P. Indyk. Explicit constructions for compressed sensing of sparse signals. In *Proc. of ACM-SIAM symposium on Discrete algorithms (SODA'08)*, 2008.

[29] M. A. Iwen. A deterministic sub-linear time sparse fourier algorithm via non-adaptive compressed sensing methods. In *Proc. of ACM-SIAM symposium on Discrete algorithms (SODA'08)*, 2008.

[30] M. A. Iwen, A. C. Gilbert, and M. J. Strauss. Empirical evaluation of a sub-linear time sparse DFT algorithm. *Submitted for Publication*, 2007.

[31] M. A. Iwen and C. V. Spencer. Improved bounds for a deterministic sublinear-time sparse fourier algorithm. In *Conference on Information Sciences and Systems (CISS)*, 2008.

[32] S. Kirolos, J. Laska, M. Wakin, M. Duarte, D. Baron, T. Ragheb, Y. Massoud, and R. Baraniuk. Analog-to-information conversion via random demodulation. *Proc. IEEE Dallas Circuits and Systems Conference*, 2006.

[33] S. Kunis and H. Rauhut. Random Sampling of Sparse Trigonometric Polynomials II - Orthogonal Matching Pursuit versus Basis Pursuit. *Foundations of Computational Mathematics*, to appear.

[34] J. Laska, S. Kirolos, Y. Massoud, R. Baraniuk, A. Gilbert, M. Iwen, and M. Strauss. Random sampling for analog-to-information conversion of wideband signals. *Proc. IEEE Dallas Circuits and Systems Conference*, 2006.

[35] J.-Y. Lee and L. Greengard. The type 3 nonuniform FFT and its applications. *J Comput. Phys.*, 206:1–5, 2005.

[36] M. Lustig, D. Donoho, and J. Pauly. Sparse MRI: The application of compressed sensing for rapid MR imaging. *Submitted for publication*, 2007.

[37] R. Maleh, A. C. Gilbert, and M. J. Strauss. Signal recovery from partial information via orthogonal matching pursuit. *IEEE Int. Conf. on Image Processing*, 2007.

[38] Y. Mansour. Learning boolean functions via the fourier transform. *Theoretical Advances in Neural Computation and Learning*, pages 391–424, 1994.

[39] Y. Mansour. Randomized approxmation and interpolation of sparse polynomials. *SIAM Journal on Computing*, 24:2, 1995.

[40] S. Muthukrishnan. Data Streams: Algorithms and Applications. *Foundations and Trends in Theoretical Computer Science*, 1, 2005.

[41] S. Muthukrishnan. Some Algorithmic Problems and Results in Compressed Sensing. *Allerton Conference*, 2006.

[42] D. Needell and R. Vershynin. Uniform uncertainty principle and signal recovery via regularized orthogonal matching pursuit. *submitted*, 2007.

[43] L. Rabiner, R. Schafer, and C. Rader. The Chirp z-Transform Algorithm. *IEEE Transactions on Audio and Electroacoustics*, AU-17(2):86–92, June 1969.

[44] M. Rudelson and R. Vershynin. Sparse reconstruction by convex relaxation: Fourier and gaussian measurements. In *40th Annual Conference on Information Sciences and Systems (CISS)*, 2006.

[45] J. Tropp and A. Gilbert. Signal recovery from partial information via orthogonal matching pursuit. *Submitted for Publication*, 2005.