(Nearly) Sample-Optimal Sparse Fourier Transform in Any Dimension; RIPless and Filterless

Vasileios Nakos, Zhao Song, Zhengyu Wang
School Of Engineering And Applied Sciences
Harvard University
Cambridge, Massachusetts, USA
{vasileiosnakos,zhaos,zhengyuwang}@g.harvard.edu

***Abstract*—In this paper, we consider the extensively studied problem of computing a k-sparse approximation to the d-dimensional Fourier transform of a length n signal. Our algorithm uses $O(k \log k \log n)$ samples, is dimension-free, operates for any universe size, and achieves the strongest ℓ_∞/ℓ_2 guarantee, while running in a time comparable to the Fast Fourier Transform. In contrast to previous algorithms which proceed either via the Restricted Isometry Property or via filter functions, our approach offers a fresh perspective to the sparse Fourier Transform problem.**

***Keywords*-Fourier transform; sparse recovery; compressed sensing;**

I. Introduction

Initiated in discrete signal processing, compressed sensing/sparse recovery is an extensively studied branch of mathematics and algorithms, which postulates that a small number of linear measurements suffice to approximately reconstruct the best k-sparse approximation of a vector $x \in \mathbb{C}^n$ [CT06], [CRT06b], [Don06]. Besides substantial literature on the subject, compressed sensing has wide real-world applications in imaging, astronomy, seismology, etc. One of the initial papers in the field, Candes, Romberg and Tao [CRT06a], has almost $15,000$ references.

Probably the most important subtopic is sparse Fourier transform, which aims to reconstruct a k-sparse vector from Fourier measurements. In other words, measurements are confined to the so-called Fourier ensemble. In Optics imaging [Goo05], [Voe11] and Magnetic resonance imaging (MRI) [ASSN08], the physics [Rey89] of the underlying device restricts us to the Fourier ensemble, where the sparse Fourier problem becomes highly relevant. In fact, these applications became one of the inspirations for Candes, Romberg and Tao. The number of samples plays a crucial role: they determine the amount of radiation a patient receives in CT scans and taking fewer samples can reduce the amount of time the patient needs to spend in the machine. The framework has found its way in life-changing applications, including Compressed Sensing GRAB-VIBE, CS SPACE, CS SEMAC and CS TOF by Siemens [Sie], and Compressed Sense by Phillips [Phi]. Its incorporation into the MRI technology allows faster acquisition rates, depiction of dynamic processes or moving organs, as well as acceleration of MRI scanning up to a factor of 40. In the words of SIEMENS Healthineers:

This allows bringing the advantages of Compressed Sensing GRASP-VIBE *to daily clinical routine.*

- *Perform push-button, free-breathing liver dynamics.*
- *Overcome timing challenges in dynamic imaging and respiratory artifacts.*
- *Expand the patient population eligible for abdominal* MRI.

On the other hand, Fourier transform is ubiquitous: image processing, audio processing, telecommunications, seismology, polynomial multiplication, Subset Sum and other textbook algorithms are some of the best-known applications of Fast Fourier Transform. The Fast Fourier Transform (FFT) by Cooley and Tukey [CT65] runs in $O(n \log n)$ time and has far-reaching impact on all of the aforementioned cases. We can thus expect that algorithms which exploit sparsity assumptions about the input and can outperform FFT in applications are of high practical value. Generally, the two most important parameters one would like to optimize are sample complexity, i.e. the numbers needed to obtain from the time domain, as well as the time needed to approximate the Fourier Transform.

Two different lines of research exist for the problem: one focuses solely on sample complexity, while the other tries to achieve sublinear time while keeping the sample complexity as low as possible. The first line of research operates via the renowned Restricted Isometry Property (RIP), which proceeds by taking random samples and solving a linear/convex program, or an iterative thresholding procedure [CT06], [DDTS06], [TG07], [BD08], [DM08], [RV08], [BD09b], [BD09a], [NT09], [NV09], [GK09], [BD10], [NV10], [Fou11], [Bou14], [HR16]. Such algorithms are analyzed in two steps as follows: The first step ensures that, after sampling an appropriate number of points from the time domain, the inverse DFT matrix restricted on the rows indexed by those points acts as a near isometry on the space of k-sparse vectors. All of the state-of-the-art results [CT06], [RV08], [Bou14], [HR16] employ chaining arguments to

2575-8454/19/$31.00 ©2019 IEEE
DOI 10.1109/FOCS.2019.00092

make the analysis of this sampling procedure as tight as possible. The second part is to exploit the aforementioned near-isometry property to find the best k-sparse approximation to the signal. There, existing approaches either follow an iterative procedure which gradually denoise the signal [BD08], [NT09], [NV09], or perform ℓ_1 minimization [CT06], a method that promotes sparsity of solutions.

The second line of research tries to implement arbitrary linear measurements via sampling Fourier coefficients [GL89], [Man92], [KM93], [GGI+02], [AGS03], [GMS05], [Iwe08], [Iwe10], [HIKP12a], [HIKP12b], [LWC13], [Iwe13], [PR14], [IKP14], [IK14], [Kap16], [Kap17], [CI17], [BZI17], [MZIC17], [LN19] and use sparse functions (in the time domain) which behave like bandpass filters in the frequency domain. The seminal work of Kapralov [Kap17] achieves $O(k \log n)$ samples and running time that is some log factors away from the sample complexity. This would be the end of the story, if not for the fact that this algorithm does not scale well with dimension, since it has an exponential dependence on d. Indeed, in many applications, one is interested in higher dimensions, rather than the one-dimensional case. The main reason[1] why this curse of dimensionality appears is the lack of dimension-independent ways to construct functions that approximate the ℓ_∞ ball and are sufficiently sparse in the time domain. A very nice work of Kapralov, Velingker and Zandieh [KVZ19] tries to remedy that by combining the standard execution of FFT with careful aliasing, but their algorithm works in a noiseless setting, and has a polynomial, rather than linear, dependence on k; the running time is polynomial in $k, \log n$ and the exponential dependence is avoided. It is an important and challenging question whether a robust and more efficient algorithm can be found.

We note that in many applications, such as MRI or computed tomography (CT), the main focus is the sample complexity; the algorithms that have found their way to industry are, to the best of our knowledge, not concerned with sublinear running time, but with the number of measurements, which determine the acquisition time, or in CT the radiation dose the patient receives. Additionally, it is worth noting some recent works on sparse Fourier transform in the continuous setting, see [Iwe10], [Iwe13], [Iwe13], [BCG+14], [PS15], [CKPS16], [Son17], [AKM+19], [CP19b], [CP19a], [Son19].

Our Contribution.: We give a new algorithm for the sparse Fourier transform problem, which has $O(k \log n \log k)$ sample complexity for any dimension, and achieves the ℓ_∞/ℓ_2 guarantee[2], while running in time $\widetilde{O}(n)$. The previous state-of-the-art algorithm that achieved such a guarantee is the work of Indyk and Kapralov [IK14], which has $2^{O(d \log d)} k \log n$ sample complexity; an exponentially worse dependence on d. The work of [HR16] obtains $O(k \log n \log^2 k)$ samples in any dimension, but has a much weaker error guarantee[3], while their approach requires $\Omega(k \log n \log k)$ samples in high dimensions [Rao19]. Moreover, the algorithm in [IK14] operates when the universe size in each dimension is a power of 2, whereas there is no restriction in our work. To obtain our result, we introduce a set of new techniques, deviating from previous work, which used the Restricted Isometry Property and/or filter functions.

[1]But not the only one: pseudorandom permutations for sparse FT in high dimensions also incur an exponential loss, and it is not known whether this can be avoided.

[2]This is the strongest guarantee in the sparse recovery literature. See also the caption of the table in Section 1.2

A. Preliminaries

For any positive integer n, we use $[n]$ to denote $\{1, 2, \cdots, n\}$. We assume that the universe size $n = p^d$ for any positive integer p. Our algorithm facilitates $n = \Pi_{j=1}^{d} p_j$ for any positive integers $p_1, \ldots, p_d$, but we decide to present the case $n = p^d$ for ease of exposition; the proof is exactly the same in the more general case. Let $\omega = e^{2\pi \mathbf{i}/p}$ where $\mathbf{i} = \sqrt{-1}$. We will work with the normalized d-dimensional Fourier transform

$$\widehat{x}_f = \frac{1}{\sqrt{n}} \sum_{t \in [p]^d} x_t \cdot \omega^{f^\top t}, \ \forall f \in [p]^d$$

and the inverse Fourier transform is

$$x_t = \frac{1}{\sqrt{n}} \sum_{f \in [p]^d} \widehat{x}_f \cdot \omega^{-f^\top t}, \ \forall t \in [p]^d.$$

For any vector x and integer k, we denote x_{-k} to be the vector obtained by zeroing out the largest (in absolute value) k coordinates from x.

B. Our result

Apart from being dimension-independent and working for any universe size, our algorithm satisfies ℓ_∞/ℓ_2, which is the strongest guarantee out of the standard guarantees considered in compressed sensing tasks. A guarantee G_1 is stronger than guarantee G_2 if for any k-sparse recovery algorithm that satisfies G_1 we can obtain a $O(k)$-sparse recovery algorithm that satisfies G_2. See also below for a comparison between ℓ_∞/ℓ_2 and ℓ_2/ℓ_2, the second stronger guarantee.

Previous work is summarized in Table I. Our result is the following.

Theorem I.1 (main result, informal version)**.** *Let $n = p^d$ where both p and d are positive integers. Let $x \in \mathbb{C}^{[p]^d}$. Let $k \in \{1, \ldots, n\}$. Assume that $R^* \geq \|\widehat{x}\|_\infty / \|\widehat{x}_{-k}\|_2$ where $\log R^* = O(\log n)$ (signal-to-noise ratio). There is an*

[3]They achieve ℓ_2/ℓ_1 instead of ℓ_∞/ℓ_2, see next Section for comparison.

algorithm that takes $O(k \log k \log n)$ samples from x, runs in $\widetilde{O}(n)$ time, and outputs a $O(k)$-sparse vector y such that

$$\|\widehat{x} - y\|_\infty \leq \frac{1}{\sqrt{k}} \|\widehat{x}_{-k}\|_2$$

holds with probability at least $1 - 1/\operatorname{poly}(n)$.

Comparison between ℓ_∞/ℓ_2 and ℓ_2/ℓ_2 (or ℓ_2/ℓ_1).: For the sake of argument, we will consider only the ℓ_2/ℓ_2 guarantee which is stronger than ℓ_2/ℓ_1. The ℓ_2/ℓ_2 guarantee is the following: for $\widehat{x} \in \mathbb{C}^n$ one should output a z such that $\|\widehat{x} - z\|_2 \leq C\|\widehat{x}_{-k}\|_2$, where $C > 1$ is the approximation factor. The ℓ_2/ℓ_2 guarantee can be immediatelly obtained by ℓ_∞/ℓ_2 guarantee by truncating z to its top k coordinates. Consider $C = 1.1$ [4], and think of the following signal: for a set S of size $0.05k$ we have $|\widehat{x}_i| = \frac{2}{\sqrt{k}}\|\widehat{x}_{\overline{S}}\|_2$. Then the all zeros vectors is a valid solution for the ℓ_2/ℓ_2 guarantee, since

$$\begin{aligned}\|\vec{0} - \widehat{x}\|_2^2 &= \|\widehat{x}_S\|_2^2 + \|\widehat{x}_{\overline{S}}\|_2^2 \\ &= 0.05k \cdot \frac{4}{k}\|\widehat{x}_{\overline{S}}\|_2^2 + \|\widehat{x}_{\overline{S}}\|_2^2 \\ &= 1.2\|\widehat{x}_{\overline{S}}\|_2^2 \\ &< 1.1^2\|\widehat{x}_{\overline{S}}\|_2^2.\end{aligned}$$

It is clear that since $\vec{0}$ is a possible output, we may not recover any of the coordinates in S, which is the set of "interesting" coordinates. On the other hand, the ℓ_∞/ℓ_2 guarantee does allow the recovery of every coordinate in S. This is a difference between recovering all $0.05k$ versus 0 coordinates. From the above discussion, one can conclude in the case where there is too much noise, ℓ_2/ℓ_2 becomes much weaker than ℓ_∞/ℓ_2, and can be even meaningless. Thus, ℓ_∞/ℓ_2 is highly desirable, whenever it is possible. The same exact argument holds for ℓ_2/ℓ_1.

Remark I.2. *We note that [CT06], [RV08], [CGV13], [Bou14], [HR16] obtain a uniform guarantee, i.e. with $1 - 1/\operatorname{poly}(n)$ they allow reconstruction of all vectors; ℓ_∞/ℓ_2 and ℓ_2/ℓ_2 are impossible in the uniform case, see [CDD09]. We note that our comparison between the guarantees is in terms of the quality of approximation. With respect to that, ℓ_∞/ℓ_2 is the strongest one.*

C. Summary of previous Filter function based technique

One of the two ways to perform Fourier sparse recovery is by trying to implement arbitrary linear measurements, with algorithms similar to the ubiquitous CountSketch [CCF02]. In the general setting CountSketch hashes every coordinate to one of the $O(k)$ buckets, and repeats $O(\log n)$ times with fresh randomness. Then, it is guaranteed that every heavy coordinate will be isolated, and the contribution from non-heavy elements is small. To implement this in the Fourier setting becomes a highly non-trivial task however: one gets access only to the time-domain but not the frequency domain. One natural way to do this is to exploit the convolution theorem and find a function which is sparse in the time domain and approximates the indicator of an interval (rectangular pulse) in the frequency domain; these functions are called (bandpass) filters. Appropriate filters were designed in [HIKP12a], [HIKP12b]: they were very good approximations of the rectangular pulse, i.e. the contribution from elements outside the passband zone contributed only by $1/\operatorname{poly}(n)$ their mass. These filters had an additional $\log n$ factor (in one dimension) in the sparsity of the time domain and they are sufficient for the purposes of [HIKP12a], but in high dimensions this factor becomes $\log^d n$. Filters based on the Dirichlet kernel give a better dependence in terms of sparsity and dimension (although still an exponential dependence on the latter), but the leak to subsequent buckets, i.e. coordinates outside the passband zone contribute a constant fraction of their mass, in contrast to the filter used in [HIKP12a]. Thus one should perform additional denoising, which is a non-trivial task. The seminal work of Indyk and Kapralov [IK14] was the first that showed how to perform sparse recovery with these filters, and then Kapralov [Kap16], [Kap17] extended this result to run in sublinear time. Note that any filter-based approach with filters which approximate the ℓ_∞ box suffers from the curse of dimensionality. [KVZ19] devised an algorithm which avoids the curse of dimensionality by using careful aliasing, but it works in the noiseless case and has a cubic dependence on k.

D. RIP property-based algorithms: a quick overview

We say the matrix $A \in \mathbb{C}^{m \times n}$ satisfies RIP (Restricted Isometry Property [CT05]) of order k if for all k-sparse vectors $x \in \mathbb{C}^n$ we have $\|Ax\|_2^2 \approx \|x\|_2^2$. A celebrated result of Candes and Tao [CT06] shows that Basis Pursuit (ℓ_1 minimization) suffices for sparse recovery, as long as the samples from the time domain satisfy RIP.

In [CT06] it was also proved using generic chaining that random sampling with oversampling factor $O(\log^6 n)$ gives RIP property for any orthonormal matrix with bounded entries by $1/\sqrt{n}$. Then [RV08] improved the bound to $O(k \cdot \log^2 k \cdot \log(k \log n) \cdot \log n)$ and [CGV13] improved it to $O(k \cdot \log^3 k \cdot \log n)$. Subsequent improvement by Bourgain [Bou14] has lead to $O(k \log k \cdot \log^2 n)$ samples, improved by Haviv and Regev to $O(k \log^2 k \cdot \log n)$[HR16]. The fastest set of algorithms are iterative ones: for example Iterative Hard Thresholding [BD09a] or CoSaMP [NT09] run in $O(\log n)$ iterations[5] and each iteration takes $\widetilde{O}(n)$ time.

[4] This is the case with the RIP based approaches, which obtain ℓ_2/ℓ_1. In fact, many filter-based algorithms facilitate $(1+\epsilon)$ on the right-hand side, with the number of measurements being multiplied by ϵ^{-1}. By enabling the same dependence on ϵ^{-1} our algorithm facilitates a multiplicative ϵ factor on the right-hand side of the ℓ_∞/ℓ_2, which makes it much stronger. Thus, a similar argument can go through.

[5] To be precise, their running time is logarithmic in the signal-to-noise ratio.

Reference	Samples	Time	Filter	RIP	Guarantee
[GMS05]	$k \log^{O(d)} n$	$k \log^{O(d)} n$	Yes	No	ℓ_2/ℓ_2
[CT06]	$k \log^6 n$	$\mathrm{poly}(n)$	No	Yes	ℓ_2/ℓ_1
[RV08]	$k \log^2 k \log(k \log n) \log n$	$\widetilde{O}(n)$	No	Yes	ℓ_2/ℓ_1
[HIKP12a]	$k \log^d n \log(n/k)$	$k \log^d n \log(n/k)$	Yes	No	ℓ_2/ℓ_2
[CGV13]	$k \log^3 k \log n$	$\widetilde{O}(n)$	No	Yes	ℓ_2/ℓ_1
[IK14]	$2^{d \log d} k \log n$	$\widetilde{O}(n)$	Yes	No	ℓ_∞/ℓ_2
[Bou14]	$k \log k \log^2 n$	$\widetilde{O}(n)$	No	Yes	ℓ_2/ℓ_1
[HR16]	$k \log^2 k \log n$	$\widetilde{O}(n)$	No	Yes	ℓ_2/ℓ_1
[Kap16], [Kap17]	$2^{d^2} k \log n$	$2^{d^2} k \log^{d+O(1)} n$	Yes	No	ℓ_2/ℓ_2
[KVZ19]	$k^3 \log^2 k \log^2 n$	$k^3 \log^2 k \log^2 n$	Yes	Yes	Exactly k-sparse
Theorem I.1	$k \log k \log n$	$\widetilde{O}(n)$	No	No	ℓ_∞/ℓ_2

Table I: $n = p^d$. We ignore the O for simplicity. The ℓ_∞/ℓ_2 is the strongest possible guarantee, with ℓ_2/ℓ_2 coming second, ℓ_2/ℓ_1 third and exactly k-sparse being the weaker. We also note that all [RV08], [CGV13], [Bou14], [HR16] obtain improved analyses of the Restricted Isometry property; the algorithm is suggested and analyzed (modulo the RIP property) in [BD08]. The work in [HIKP12a] does not explicitly state the extension to the d-dimensional case, but can easily be inferred from the arguments. [HIKP12a], [IK14], [Kap16], [KVZ19] work when the universe size in each dimension are powers of 2. We also assume that the signal-to-noise ratio is bounded by a polynomial of n, which is a standard assumption in the sparse Fourier transform literature [HIKP12a], [IK14], [Kap16], [Kap17], [LN19].

We note the recent lower bound of [Rao19]: a subsampled Fourier matrix that satisfies the RIP properties should have $\Omega(k \log k \cdot d)$ rows[6]. This bound is particularly useful in high dimensions, since it deteriorates to a trivial bound in low dimensions. We still believe though that a bound of $\Omega(k \log k \log n)$ should hold in all dimensions. Thus, what remains is to obtain the ℓ_2/ℓ_2 guarantee by giving a tighter analysis, and removing the one $\log k$ factor to match the lower bound, but our algorithm already allows Fourier sparse recovery with these number of samples, even with a stronger guarantee.

E. Overview of our technique

Let $x \in \mathbb{C}^{[p]^d}$ denote our input signal in the time domain. In the following we assume the knowledge of $\mu = \frac{1}{\sqrt{k}} \|\widehat{x}_{-k}\|_2$ and R^* which is an upper bound of $\|\widehat{x}\|_\infty/\mu$, and bounded by $\mathrm{poly}(n)$. These are standard assumption [HIKP12a], [IK14], [Kap16], [Kap17], [LN19] in the sparse Fourier transform literature. The bound on R^* is useful for bounding the running time (or the number of measurements in [HIKP12a]) and in any of [HIKP12a], [IK14], [Kap16], [Kap17], [LN19] a $\log n$ can be substituted by $\log R^*$ in the general case, which is also the case for our algorithm. We note that our algorithm will be correct with probability $1 - 1/\mathrm{poly}(n)$ whenever $R^* < 2^{n^{100}}$; this is fine for every reasonable application. It might seem counter-intuitive that we need this upper bound on R^*, since intuitively larger signal to noise ratio should only help. However, this is an artifact of the techniques of Sparse Fourier Transform in general, either they are iterative or not. We assumed the rounding errors in FFT computation to be negligible, similarly to Remark 3.4 in [IK14].

[6][BLLM19] independently gives a similar bound for $d = \log n$.

1) Estimators and random shifts: Consider the simplest scenario: $d = 1$, p is a prime number and a 1-sparse signal $\widehat{x}$ which is 1 on some frequency f^*. From a sample x_t in the time-domain what would be the most reasonable way to find f^*? For every $f \in [p]$ we would compute

$$\sqrt{n}\omega^{ft} x_t = \sqrt{n}\omega^{ft} \cdot \frac{1}{\sqrt{n}} \sum_{f' \in [p]} \omega^{-f't} \widehat{x}_{f'} = \omega^{(f-f^*)t},$$

and keep, for $t \neq 0$, the frequency that gives a real number. Since $(f - f^*)t$ will be zero only for $f = f^*$, we are guaranteed correct recovery. In the noisy and multi-dimensional case or p is an arbitrary integer, however, this argument will not work, because of the presence of contribution from other elements and the fact that $(f - f^*)^\top t$ can be zero modulo p for other frequencies apart from f. However, we can take a number of samples t and average $\sqrt{n}\omega^{f^\top t}$, and hope that this will make the contribution from other frequencies small enough, so that we can infer whether f corresponds to a heavy coordinate or not. More specifically, we pick a set T of frequencies uniformly at random from $[p]^d$ and compute

$$\frac{\sqrt{n}}{|T|} \sum_{t \in T} \omega^{f^\top t} x_t$$

for all frequencies f. We show that if $|T| = O(k)$ our estimator is good on average (and later we will maintain $O(\log n)$ independent instances and take the median to make sure with probability $1 - 1/\mathrm{poly}(n)$ the estimators for all the frequencies are good), and in fact behaves like a crude filter, similarly to the ones used in [IK14], in the sense that every coordinate contributes a non-trivial amount to every other coordinate. However, these estimators do not suffer from the curse of dimensionality and our case is a little bit different, requiring a quite different handling. The main reason is that

in contrast to the filters used in [IK14], there is not an easy way to formulate an isolation argument from heavy elements that would allow easy measurement re-use, like Definition 5.2 and Lemma 5.4 from [IK14]. Buckets induced by filter functions have a property of locality, since they correspond to approximate ℓ_∞ boxes (with a polynomial decay outside of the box) in $[p]^d$: the closer two buckets are the more contribute the elements of one into the other. Our estimators on the other side do not enjoy such a property. Thus, one has to proceed via a different argument.

In what follows, we will discuss how to combine the above estimators with an iterative loop that performs denoising, i.e. removes the contribution of every heavy element to other heavy elements.

Random shifts.: Our approach for performing denoising is quite general, and is clean on a high-level. Let S be the set of the large coordinates of $\widehat{x}$, i.e. those with magnitude at least $(1/\sqrt{k})\|\widehat{x}_{-k}\|_2$. We are going to estimate $\widehat{x}_f$ for $f \in [p]^d$ using the estimators introduced in the previous paragraphs. Then, for those frequencies f for which the values obtained are sufficiently large (larger than $\|x\|_\infty \cdot 2^{-\ell}$) , we are going to implicitly subtract that value from $\widehat{x}_f$; this corresponds to updating the signal, a common trick in the Sparse Fourier literature. Then we shall iterate, re-using the same samples again till the signal to noise ratio becomes small enough. It can be shown that only coordinates in S are ever updated. The trick for sample reuse in our case is the following: in the ℓth iteration we approximate $\mathbb{C}^{[p]^d}$ by an appropriate grid of side length $\beta\|x\|_\infty \cdot 2^{-\ell}$, where β is an absolute constant, and then keep $O(\log n)$ random shifts of it. Keeping $O(\log n)$ randomly shifted grids one can show that in every iteration a nice property is satisfied: we can guarantee that there exists a grid such that for all $f \in S$, $\widehat{x}_f$ and its estimator round to the same point. Projecting onto that grid, what our algorithm shows is that the signal under update follows a predictable trajectory, something that allows us to argue about the correctness of the algorithm without taking an intractable union-bound over all possible trajectories the algorithm could evolve. In essence, our algorithm shows that we can at ℓth step compute every $\widehat{x}_f$ up to $\|\widehat{x}\|_\infty 2^{-\ell}$ error.

A high-level explanation.: Let us try to combine the previous two ideas. Assume that at iteration ℓ we have an approximation of $\widehat{x}_f$ for all $f \in S$ up to $\|x\|_\infty \cdot 2^{-\ell}$. If we were to pick $O(k \log n \log k)$ fresh samples, we could, using our estimators, approximate $\widehat{x}_f$ up to $(1/k)\|x\|_\infty \cdot 2^{-\ell}$. Let that approximation be y. We then round y to $O(\log n)$ randomly shifted grids of diameter $\|x\|_\infty \cdot 2^{-\ell}$. A probabilistic argument shows that, due to our choice of parameters, with high probability there exists a grid such that $\widehat{x}_f$ and y_f are rounded to the same grid point (the additional $O(\log k)$ factor in the sample complexity is what makes this argument go through); and we can also decide which grid this is! Thus, we safely project y and be sure that what we now have at our hands is $\widehat{x}$ projected onto that grid. Thus, in the next iteration $\ell+1$ we only need to argue correctness for at most $O(\log n)$ vectors, that is, vectors $\widehat{x}$ rounded on one of the aforementioned $O(\log n)$ grids. This dramatically decreases the number of events we have to analyze, and we can set up an inductive argument that guarantees the correctness of the algorithm. Note that there is no independence issue, since the randomness between the samples taken (used for the estimators) and the projection onto the randomly shifted grids is not shared.

We first implement a procedure which takes $O(k \log n)$ uniform random measurements from x and has the guarantee that for any $\nu \geq \mu$ any $y \in \mathbb{C}^{[p]^d}$ where $\|\widehat{x} - y\|_\infty \leq 2\nu$ and y is independent from the randomness of the measurements, the procedure outputs a $O(k)$-sparse $z \in \mathbb{C}^{[p]^d}$ such that $\|\widehat{x} - y - z\|_\infty \leq \nu$ with probability $1 - 1/\operatorname{poly}(n)$.

Lemma I.3 (LINFINITYREDUCE procedure, informal)**.** *Let $\mu = \frac{1}{\sqrt{k}}\|\widehat{x}_{-k}\|_2$, and $\nu \geq \mu$. Let $\mathcal{T}^{(0)}$ be a list of $O(k \log n)$ i.i.d. elements in $[p]^d$. Let S be top $O(k)$ coordinates in $\widehat{x}$. There is a procedure that takes $\{x_t\}_{t \in \mathcal{T}}$, $y \in \mathbb{C}^{[p]^d}$ and ν as input, runs in $\widetilde{O}(n)$ time, and outputs $z \in \mathbb{C}^{[p]^d}$ so that if $\|\widehat{x} - y\|_\infty \leq 2\nu$, $\operatorname{supp}(y) \subseteq S$ and y is independent from the randomness of $\mathcal{T}^{(0)}$, then $\|\widehat{x} - y - z\|_\infty \leq \nu$ and $\operatorname{supp}(z) \subseteq S$ with probability $1 - 1/\operatorname{poly}(n)$ under the randomness of $\mathcal{T}^{(0)}$.*

Namely, we can take $O(k \log n)$ measurements and run the procedure in Lemma I.3 to reduce (the upper bound of) the ℓ_∞ norm of the residual signal by half. We call the procedure in Lemma I.3 LINFINITYREDUCE procedure. More generally, we can take $O(H \cdot k \log n)$ measurements and run the LINFINITYREDUCE procedure H times to reduce the ℓ_∞ norm of the residual signal to $1/2^H$ of its original magnitude, with failure probability at most $1/\operatorname{poly}(n)$. Note that if we set $H = \log R^*$, we have already obtained a m taking $O(k \log n \log R^*)$ measurements, because we can drive down (the upper bound of) the ℓ_∞ norm of the residual signal from $\|\widehat{x}\|_\infty$ to μ in $\log R^*$ iterations.

2) $O(k \log n)$ samples for $k = O(\log n)$: We first discuss a measurement reuse idea that leads us to a sparse recovery algorithm (Algorithm 1) taking $O(k \log n)$ measurements for $k = O(\log n)$. We set $H = 5$, and let $\mathcal{T} = \{\mathcal{T}^{(1)}, \ldots, \mathcal{T}^{(H)}\}$, where each $\mathcal{T}^{(h)}$ is a list of $O(k \log n)$ i.i.d. elements in $[p]^d$. Note that $\mathcal{T}^{(1)}, \ldots, \mathcal{T}^{(H)}$ are independent. In our sparse Fourier recovery algorithm, we will measure x_t for all $t \in \mathcal{T}$.

In a nutshell, our approach finely discretizes the space of possible trajectories the algorithm could evolve, and carefully argues about the correctness of the algorithm by avoiding the intractable union-bound over all trajectories.

Recovery algorithm.: The recovery algorithm proceeds in $\log R^* - H + 1$ iterations, where each iteration (except

the last iteration) the goal is to reduce the upper bound of ℓ_∞ norm of the residual signal by half. Initially, the upper bound is R^*. It is important to note that we use the same measurements $\mathcal{T} = \{\mathcal{T}^{(1)}, \ldots, \mathcal{T}^{(H)}\}$ in all of these $\log R^* - H + 1$ iterations.

In the following, we will describe one iteration of the recovery algorithm. Let $y \in \mathbb{C}^{[p]^d}$ denote the sparse vector recovered so far, and let the upper bound of $\|\widehat{x}-y\|_\infty$ be 2ν. Running the LINFINITYREDUCE procedure H times where in the h-th time we use measurements in $\mathcal{T}^{(h)}$, we obtain a $O(k)$-sparse z such that with probability $1 - 1/\operatorname{poly}(n)$, $\|\widehat{x} - y - z\|_\infty \leq 2^{1-H}\nu \leq 0.1\nu$ (we call such z a desirable output by the LINFINITYREDUCE procedure). Instead of taking $y + z$ as our newly recovered sparse signal, for each $f \in \operatorname{supp}(y + z)$, we project $y_f + z_f$ to the nearest points in $\mathcal{G}_{0.6\nu} := \{0.6\nu(x + y\mathbf{i}) : x, y \in \mathbb{Z}\}$ and assign to y'_f, where y' denotes our newly recovered sparse signal. For all $f \notin \operatorname{supp}(y + z)$, we let $y'_f = 0$.

To simplify our exposition, here we introduce some notations. We call $\mathcal{G}_{0.6\nu}$ a grid of side length 0.6ν, and we generalize the definition to any side length. Namely, for any $r_g > 0$, let grid $\mathcal{G}_{r_g} := \{r_g(x + y\mathbf{i}) : x, y \in \mathbb{Z}\}$. Moreover, we define $\Pi_{r_g} : \mathbb{C} \to \mathcal{G}_{r_g}$ to be the mapping that maps any element in $\mathbb{C}$ to the nearest element in $\mathcal{G}_{r_g}$. Now we can write y' as

$$y'_f = \begin{cases} \Pi_{0.6\nu}(y_f + z_f), & \text{if } f \in \operatorname{supp}(y + z); \\ 0, & \text{if } f \notin \operatorname{supp}(y + z). \end{cases}$$

At the end of each iteration, we assign y' to y, and shrink ν by half. In the last iteration, we will not compute y', instead we output $y + z$. We present the algorithm in Algorithm 1.

Analysis.: We analyze y' conditioned on the event that $\|\widehat{x} - y - z\|_\infty \leq 0.1\nu$ (i.e. z is a desirable output by the LINFINITYREDUCE procedure, which happens with probability $1 - 1/\operatorname{poly}(n)$). We will prove that y' has two desirable properties: (1) $\|\widehat{x} - y'\|_\infty \leq \nu$; (2) the dependence between y' and our measurements $\mathcal{T}$ is under control so that after taking y' as newly recovered sparse signal, subsequent executions of the LINFINITYREDUCE procedure with measurements $\mathcal{T}$ still work with good probability. Property (1) follows from triangle inequality and the fact that $\|\widehat{x} - (y + z)\|_\infty \leq 0.1\nu$ and $\|(y + z) - y'\|_\infty \leq 0.6\nu$. We now elaborate on property (2). We can prove that for any $f \in [p]^d$,

$$y'_f \in \left\{\Pi_{0.6\nu}(\widehat{x}_f + 0.1\nu(\alpha + \beta\mathbf{i})) : \alpha, \beta \in \{-1, 1\}\right\}.$$

Let S denote top $26k$ coordinates (in absolute value) of $\widehat{x}$. We can further prove that for any $f \in \overline{S}$, $y'_f = 0$. Therefore, the total number of possible y' is upper bounded by $4^{|S|} = 4^{O(k)}$. If $k = O(\log n)$, we can afford union bounding all $4^{O(k)} = \operatorname{poly}(n)$ possible y', and prove that with probability $1 - 1/\operatorname{poly}(n)$ for all possible value of y' if we take y' as our newly recovered sparse signal then in the next iteration the LINFINITYREDUCE procedure with measurements $\mathcal{T}$ gives us a desirable output.

Algorithm 1 Fourier sparse recovery by projection, $O(k \log n)$ measurements when $k = O(\log n)$

procedure FOURIERSPARSERECOVERYBYPROJECTION(x, n, k, μ, R^*) ▷ Section I-E2
  Require that $\mu = \frac{1}{\sqrt{k}}\|\widehat{x}_{-k}\|_2$ and $R^* \geq \|\widehat{x}\|_\infty$ /μ
  $H \leftarrow 5$, $\nu \leftarrow \mu R^*/2$, $y \leftarrow \vec{0}$ ▷ $y \in \mathbb{C}^{[p]^d}$ refers to the sparse vector recovered so far
  Let $\mathcal{T} = \{\mathcal{T}^{(1)}, \cdots, \mathcal{T}^{(H)}\}$ where each $\mathcal{T}^{(h)}$ is a list of i.i.d. uniform samples in $[p]^d$
  while true do
    $\nu' \leftarrow 2^{1-H}\nu$
    Use $\{x_t\}_{t\in\mathcal{T}}$ to run the LINFINITYREDUCE procedure (in Lemma I.3) H times (use samples in $\mathcal{T}^{(h)}$ for each $h \in [H]$), and finally it finds z so that $\|\widehat{x} - y - z\|_\infty \leq \nu'$
    if $\nu' \leq \mu$ **then return** $y + z$ ▷ We found the solution
    $y' \leftarrow \vec{0}$
    for $f \in \operatorname{supp}(y + z)$ **do**
      $y'_f \leftarrow \Pi_{0.6\nu}(y_f + z_f)$ ▷ We want $\|\widehat{x} - y'\|_\infty \leq \nu$ and the dependence between y' and $\mathcal{T}$ is under control
    end for
    $y \leftarrow y'$, $\nu \leftarrow \nu/2$
  end while
end procedure

Sufficient event.: More rigorously, we formulate the event that guarantees successful execution of Algorithm 1. Let $\mathcal{E}_1$ be the event that for all $O(\log R^*)$ possible values of $\nu \in \{\mu\frac{R^*}{2}, \mu\frac{R^*}{4}, \ldots, \mu 2^{H-1}\}$, for all possible vector y where $y_f = 0$ for $f \in \overline{S}$ and $y_f \in \{\Pi_{0.6\nu}(\widehat{x}_f + 0.1\nu(\alpha + \beta\mathbf{i})) : \alpha, \beta \in \{-1, 1\}\}$ for $f \in S$ (we also need to include the case that $y = \vec{0}$ for the success of the first iteration), running the LINFINITYREDUCE procedure (in Lemma I.3) H times (where in the h-th time measurements $\{x_t\}_{t\in\mathcal{T}^{(h)}}$ are used to reduce the error from $2^{2-h}\nu$ to $2^{1-h}\nu$) finally gives z so that $\|\widehat{x} - y - z\|_\infty \leq 2^{1-H}\nu$. The randomness of $\mathcal{E}_1$ comes from $\mathcal{T} = \{\mathcal{T}^{(1)}, \ldots, \mathcal{T}^{(H)}\}$.

First, event $\mathcal{E}_1$ happens with probability $1 - 1/\operatorname{poly}(n)$. This is because there are $4^{O(k)} \log R^*$ possible combinations of ν and y to union bound, and each has failure probability at most $1/\operatorname{poly}(n)$. For $k = O(\log n)$, and any $R^* < 2^{n^{100}}$ this gives the desired result. Second, conditioned on event $\mathcal{E}_1$ happens, Algorithm 1 gives correct output. This can be proved by a mathematical induction that in the t-th iteration of the while-true loop in Algorithm 1, $\|\widehat{x} - y\|_\infty \leq 2^{-t}\mu R^*$.

3) $O(k \log k \log n)$ samples suffice: We first introduce some notations. For any $r_g > 0$, define the grid $\mathcal{G}_{r_g} := \{r_g(x+y\mathbf{i}) : x, y \in \mathbb{Z}\}$. Moreover, we define $\Pi_{r_g} : \mathbb{C} \to \mathcal{G}_{r_g}$ to be the mapping that maps any element in $\mathbb{C}$ to the nearest element in $\mathcal{G}_{r_g}$.

Using random shift to reduce projection size.: We introduce the random shift trick, the property of which is captured by Lemma I.4. To simplify notation, for any $r_b > 0$ and $c \in \mathbb{C}$ we define box $\mathcal{B}_\infty(c, r_b) := \{c + r_b(x + y\mathbf{i}) : x, y \in [-1, 1]\}$. For any $S \subseteq \mathbb{C}$, let $\Pi_{r_g}(S) = \{\Pi_{r_g}(c) : c \in S\}$.

Lemma I.4 (property of a randomly shifted box, informal)**.** *Take a box of side length $2r_b$ and shift it randomly by an offset in $\mathcal{B}_\infty(0, r_s)$ (or equivalently, $[-r_s, r_s] \times [-r_s, r_s]$) where $r_s \geq r_b$. Next round every point inside that shifted box to the closest point in G_{r_g} where $r_g \geq 2r_s$. Then, with probability at least $(1 - r_b/r_s)^2$ everyone will be rounded to the same point.*

In the following, we present a sparse Fourier recovery algorithm that incorporates the random shift idea. The algorithm takes $O(k \log k \log n)$ measurements. We set $H = O(\log k)$ and take measurements of $\mathcal{T} = \{\mathcal{T}^{(1)}, \ldots, \mathcal{T}^{(H)}\}$, where $\mathcal{T}^{(h)}$ is a list of $O(k \log n)$ i.i.d elements in $[p]^d$.

In a nutshell, our approach finely discretizes the space of possible trajectories the algorithm could evolve. After we find estimates for $\widehat{x}_f$ we shift them randomly and project them onto a coarse grid (which is the same as projecting onto one of randomly shifted grids). We shall show that then the number of trajectories is pruned, and we need to argue for a much smaller collection of events. We note that we make the decoding algorithm randomized: the randomness in previous algorithms was present only when taking samples, and the rest of the algorithm was deterministic. However, here we need randomness in both cases, and that helps us prune the number of possible trajectories. To the best of our knowledge, this is a novel argument and approach, and might be helpful for future progress in the field.

Recovery algorithm.: We assume that we have already obtained a $O(k)$-sparse $y \in \mathbb{C}^{[p]^d}$ such that $\|\widehat{x} - y\|_\infty \leq 2\nu$ and y is "almost" independent from $\mathcal{T}$. We show how to obtain $y' \in \mathbb{C}^{[p]^d}$ such that $\|\widehat{x} - y'\|_\infty \leq \nu$ with probability $1 - 1/\operatorname{poly}(n)$ and y' is "almost" independent from $\mathcal{T}$. The idea is the following. We first run the LINFINITYREDUCE procedure $H = O(\log k)$ times to get an $O(k)$-sparse $z \in \mathbb{C}^{[p]^d}$ such that $\|\widehat{x} - y - z\|_\infty \leq \frac{1}{2^{20}k}\nu$. Then we repeatedly sample a uniform random shift $s \in [-10^{-3}\nu, 10^{-3}\nu] + \mathbf{i}[-10^{-3}\nu, 10^{-3}\nu]$ until for every $f \in \operatorname{supp}(y + z)$, all the points (or complex numbers) of the form $y_f + z_f + s + a + b\mathbf{i}$ with $a, b \in [-\frac{\nu}{2^{20}k}, \frac{\nu}{2^{20}k}]$ round to the same grid point in $\mathcal{G}_{0.04\nu}$. Finally, for every $f \in \operatorname{supp}(y + z)$, we assign $\Pi_{0.04\nu}(y_f + z_f + s)$ to y'_f; all remaining coordinates in y' will be assigned 0. We present an informal version of our algorithm in Algorithm 2, and defer its formal version to the full version [NSW19].

Algorithm 2 Fourier sparse recovery by random shift and projection (informal version)

1: **procedure** FOURIERSPARSERECOVERY(x, n, k, μ, R^*) ▷ Theorem I.1, $n = p^d$
2: **Require** that $\mu = \frac{1}{\sqrt{k}}\|\widehat{x}_{-k}\|_2$ and $R^* \geq \|\widehat{x}\|_\infty / \mu$
3: $H \leftarrow O(\log k)$, $\nu \leftarrow \mu R^*/2$, $y \leftarrow \vec{0}$ ▷ $y \in \mathbb{C}^{[p]^d}$ refers to the sparse vector recovered so far
4: Let $\mathcal{T} = \{\mathcal{T}^{(1)}, \cdots, \mathcal{T}^{(H)}\}$ where each $\mathcal{T}^{(h)}$ is a list of i.i.d. uniform samples in $[p]^d$
5: **while true do**
6: $\nu' \leftarrow \frac{1}{2^{20}k}\nu$
7: Use $\{x_t\}_{t\in\mathcal{T}}$ to run the LINFINITYREDUCE procedure (in Lemma I.3) H times (use samples in $\mathcal{T}^h$ for each $h \in [H]$), and finally it finds z so that $\|\widehat{x} - y - z\|_\infty \leq \nu'$
8: **if** $\nu' \leq \mu$ **then return** $y + z$ ▷ We found the solution
9: **repeat**
10: Pick $s \in \mathcal{B}_\infty(0, 10^{-3}\nu)$ uniformly at random
11: **until** $\forall f \in \operatorname{supp}(y + z), |\Pi_{0.04\nu}(\mathcal{B}_\infty(y_f + z_f + s, \nu'))| = 1$
12: $y' \leftarrow \vec{0}$
13: **for** $f \in \operatorname{supp}(y + z)$ **do**
14: $y'_f \leftarrow \Pi_{0.04\nu}(y_f + z_f + s)$ ▷ We want $\|\widehat{x} - y'\|_\infty \leq \nu$ and the depend-
15: **end for** ▷ ence between y' and $\mathcal{T}$ is under control
16: $y \leftarrow y'$, $\nu \leftarrow \nu/2$
17: **end while**
18: **end procedure**

Informal Analysis.: We analyze the above approach.

At every iteration, our algorithm holds a vector y, and computes a vector z. Instead of setting y to $y + z$ and iterating, as would be the natural thing to do, we set y to y' (lines 12 to 16) where

$$y'_f = \begin{cases} \Pi_{0.6\nu}(y_f + z_f), & \text{if } f \in \operatorname{supp}(y + z); \\ 0, & \text{if } f \notin \operatorname{supp}(y + z). \end{cases}$$

First, we have the guarantee that $\|\widehat{x} - y'\|_\infty \leq \nu$. Moreover, by our choice of s, for every $f \in \operatorname{supp}(y + z)$, $y_f + z_f + s$ and $\widehat{x}_f + s$ round to the same grid point in $\mathcal{G}_{0.04\nu}$. Therefore, for the new vector y' we have recovered, we "hide" the randomness in $\mathcal{T}$, and the randomness only leaks from failed attempts of the shifts. In the following, we show that each attempt of shift succeeds with probability $\frac{1}{2}$.

We can restate the procedure of choosing s to be:

repeatedly sample $s \sim \mathcal{B}_\infty(0, 10^{-3}\nu)$,

until for all $f \in \mathrm{supp}(y+z)$,

$$\left|\Pi_{0.04\nu}\left(\mathcal{B}_\infty\left(y_f + z_f + s, \frac{\nu}{2^{20}k}\right)\right)\right| = 1.$$

Note that $|\mathrm{supp}(y+z)| = O(k)$. Let us say that we can always guarantee that $|\mathrm{supp}(y+z)| \le 50k$. By Lemma I.4 where we let $r_b = \frac{\nu}{2^{20}k}$, $r_s = 10^{-3}\nu$ and $r_g = 0.04\nu$, for $f \in \mathrm{supp}(y+z)$,

$$\Pr\left[\left|\Pi_{0.04\nu}\left(\mathcal{B}_\infty\left(y_f + z_f + s, \frac{\nu}{2^{20}k}\right)\right)\right| = 1\right] \ge \left(1 - \frac{r_b}{r_s}\right)^2 \ge 1 - \frac{1}{100k}.$$

By a union bound over $f \in \mathrm{supp}(y+z)$, the probability is at least $\frac{1}{2}$ that for all $f \in \mathrm{supp}(y+z)$, $|\Pi_{0.04\nu}(\mathcal{B}_\infty(y_f + z_f + s, \frac{\nu}{2^{20}k}))| = 1$.

Therefore, with probability $1 - 1/\mathrm{poly}(n)$, we will only try $O(\log n)$ shifts. We can apply a union bound over $O(\log n)$ possible shifts, and prove that with probability $1 - 1/\mathrm{poly}(n)$ if taking y' as our new y, and shrinking ν by half, the LINFINITYREDUCE procedure will work as desired as if there is no dependence issue.

Sufficient event.: Let S be top $O(k)$ coordinates in $\widehat{x}$ which are also larger than $(1/\sqrt{k})\|x_{-k}\|_2$ in magnitude. Let $L = O(\log R^*)$ denote the number of iterations in Algorithm 2. For $\ell \in [L]$, let $\nu_\ell = 2^{-\ell}\mu R^*$. For $\ell \in [L-1]$, let $s_\ell^{(a)}$ be the a-th sample from $\mathcal{B}_\infty(0, 10^{-3}\nu_\ell)$ as appeared on Line 10 in Algorithm 2. For the sake of analysis, we assume that Algorithm 2 actually produces an infinite sequence of shifts $s_\ell^{(1)}, s_\ell^{(2)}, \ldots$. We formulate the event that guarantees successful execution of Algorithm 2. We define event $\mathcal{E}_2$ to be the union of all the following events.

1. For all $\ell \in [L-1]$, there exists $a \in [10\log n]$ so that for all $f \in S$,

$$\left|\Pi_{0.04\nu_\ell}\left(\mathcal{B}_\infty(\widehat{x}_f + s_\ell^{(a)}, \frac{1}{100k}\nu_\ell)\right)\right| = 1.$$

2. For $\ell = 1$, if we run the LINFINITYREDUCE procedure H times with $y = \vec{0}$ and measurements in $\mathcal{T}$, we get z such that $\|\widehat{x} - z\|_\infty \le 2^{1-H}\nu_1$ and $\mathrm{supp}(z) \subseteq S$.

3. For all $\ell \in \{2, \ldots, L\}$, for all $a \in [10\log n]$, if we run the LINFINITYREDUCE procedure H times with $y = \xi$ where

$$\xi_f = \begin{cases} \Pi_{0.04\nu_\ell}(\widehat{x}_f + s_{\ell-1}^{(a)}), & \text{if } f \in S; \\ 0, & \text{if } f \in \overline{S}. \end{cases}$$

then we get z such that $\|\widehat{x} - y - z\|_\infty \le 2^{1-H}\nu_\ell$ and $\mathrm{supp}(y+z) \subseteq S$.

We can prove that event $\mathcal{E}_2$ happens with probability $1 - 1/\mathrm{poly}(n)$. Moreover, we can prove that conditioned on event $\mathcal{E}_2$ Algorithm 2 gives correct output. We defer both proofs in the full version [NSW19].

REFERENCES

[AGS03] Adi Akavia, Shafi Goldwasser, and Shmuel Safra. Proving hard-core predicates using list decoding. In *FOCS*, volume 44, pages 146–159, 2003.

[AKM+19] Haim Avron, Michael Kapralov, Cameron Musco, Christopher Musco, Ameya Velingker, and Amir Zandieh. A universal sampling method for reconstructing signals with simple fourier transforms. In *STOC*. arXiv preprint arXiv:1812.08723, 2019.

[ASSN08] Abiodun M Aibinu, Momoh JE Salami, Amir A Shafie, and Athaur Rahman Najeeb. MRI reconstruction using discrete Fourier transform: a tutorial. *World Academy of Science, Engineering and Technology*, 42:179, 2008.

[BCG+14] Petros Boufounos, Volkan Cevher, Anna C Gilbert, Yi Li, and Martin J Strauss. What's the frequency, Kenneth?: Sublinear Fourier sampling off the grid. In *Algorithmica(A preliminary version of this paper appeared in the Proceedings of RANDOM/APPROX 2012, LNCS 7408, pp.61–72)*, pages 1–28. Springer, 2014.

[BD08] Thomas Blumensath and Mike E Davies. Iterative thresholding for sparse approximations. *Journal of Fourier analysis and Applications*, 14(5-6):629–654, 2008.

[BD09a] Thomas Blumensath and Mike E Davies. Iterative hard thresholding for compressed sensing. *Applied and computational harmonic analysis*, 27(3):265–274, 2009.

[BD09b] Thomas Blumensath and Mike E Davies. A simple, efficient and near optimal algorithm for compressed sensing. *2009 IEEE International Conference on Acoustics, Speech and Signal Processing*, 2009.

[BD10] Thomas Blumensath and Mike E Davies. Normalized iterative hard thresholding: Guaranteed stability and performance. *IEEE Journal of selected topics in signal processing*, 4(2):298–309, 2010.

[BLLM19] Jaroslaw Blasiok, Patrick Lopatto, Kyle Luh, and Jake Marcinek. Sparse reconstruction from hadamard matrices: A lower bound. In *60th Annual IEEE Symposium on Foundations of Computer Science (FOCS)*. https://arxiv.org/pdf/1903.12135.pdf, 2019.

[Bou14] Jean Bourgain. An improved estimate in the restricted isometry problem. In *Geometric Aspects of Functional Analysis*, pages 65–70. Springer, 2014.

[BZI17] Sina Bittens, Ruochuan Zhang, and Mark A Iwen. A deterministic sparse FFT for functions with structured Fourier sparsity. *Advances in Computational Mathematics, to appear.*, 2017.

[CCF02] Moses Charikar, Kevin Chen, and Martin Farach-Colton. Finding frequent items in data streams. In *Automata, Languages and Programming*, pages 693–703. Springer, 2002.

[CDD09] Albert Cohen, Wolfgang Dahmen, and Ronald DeVore. Compressed sensing and best k-term approximation. *Journal of the American mathematical society*, 22(1):211–231, 2009.

[CGV13] Mahdi Cheraghchi, Venkatesan Guruswami, and Ameya Velingker. Restricted isometry of Fourier matrices and list decodability of random linear codes. *SIAM Journal on Computing*, 42(5):1888–1914, 2013.

[CI17] Mahdi Cheraghchi and Piotr Indyk. Nearly optimal deterministic algorithm for sparse walsh-hadamard transform. *ACM Transactions on Algorithms (TALG)*, 13(3):34, 2017.

[CKPS16] Xue Chen, Daniel M Kane, Eric Price, and Zhao Song. Fourier-sparse interpolation without a frequency gap. In *Foundations of Computer Science (FOCS), 2016 IEEE 57th Annual Symposium on*, pages 741–750. IEEE, 2016.

[CP19a] Xue Chen and Eric Price. Active regression via linear-sample sparsification. In *Conference on Learning Theory, COLT 2019, 25-28 June 2019, Phoenix, AZ, USA*, pages 663–695, 2019.

[CP19b] Xue Chen and Eric Price. Estimating the frequency of a clustered signal. In *46th International Colloquium on Automata, Languages, and Programming, ICALP 2019, July 9-12, 2019, Patras, Greece.*, pages 36:1–36:13, 2019.

[CRT06a] Emmanuel J Candès, Justin Romberg, and Terence Tao. Robust uncertainty principles: Exact signal reconstruction from highly incomplete frequency information. *IEEE Transactions on information theory*, 52(2):489–509, 2006.

[CRT06b] Emmanuel J Candes, Justin K Romberg, and Terence Tao. Stable signal recovery from incomplete and inaccurate measurements. *Communications on pure and applied mathematics*, 59(8):1207–1223, 2006.

[CT65] James W Cooley and John W Tukey. An algorithm for the machine calculation of complex Fourier series. *Mathematics of computation*, 19(90):297–301, 1965.

[CT05] Emmanuel Candes and Terence Tao. Decoding by linear programming. *arXiv preprint math/0502327*, 2005.

[CT06] Emmanuel J Candes and Terence Tao. Near-optimal signal recovery from random projections: Universal encoding strategies? *IEEE transactions on information theory*, 52(12):5406–5425, 2006.

[DDTS06] David Leigh Donoho, Iddo Drori, Yaakov Tsaig, and Jean-Luc Starck. *Sparse solution of underdetermined linear equations by stagewise orthogonal matching pursuit*. Department of Statistics, Stanford University, 2006.

[DM08] Wei Dai and Olgica Milenkovic. Subspace pursuit for compressive sensing: Closing the gap between performance and complexity. Technical report, ILLINOIS UNIV AT URBANA-CHAMAPAIGN, 2008.

[Don06] David L. Donoho. Compressed sensing. *IEEE Trans. Information Theory*, 52(4):1289–1306, 2006.

[Fou11] Simon Foucart. Hard thresholding pursuit: an algorithm for compressive sensing. *SIAM Journal on Numerical Analysis*, 49(6):2543–2563, 2011.

[GGI+02] Anna C Gilbert, Sudipto Guha, Piotr Indyk, S Muthukrishnan, and Martin Strauss. Near-optimal sparse Fourier representations via sampling. In *Proceedings of the thiry-fourth annual ACM symposium on Theory of computing*, pages 152–161. ACM, 2002.

[GK09] Rahul Garg and Rohit Khandekar. Gradient descent with sparsification: an iterative algorithm for sparse recovery with restricted isometry property. In *ICML*, volume 9, pages 337–344, 2009.

[GL89] Oded Goldreich and Leonid A Levin. A hard-core predicate for all one-way functions. In *Proceedings of the twenty-first annual ACM symposium on Theory of computing*, pages 25–32. ACM, 1989.

[GMS05] Anna C Gilbert, S Muthukrishnan, and Martin Strauss. Improved time bounds for near-optimal sparse Fourier representations. In *Optics & Photonics 2005*, pages 59141A–59141A. International Society for Optics and Photonics, 2005.

[Goo05] Joseph W Goodman. *Introduction to Fourier optics*. Roberts and Company Publishers, 2005.

[HIKP12a] Haitham Hassanieh, Piotr Indyk, Dina Katabi, and Eric Price. Nearly optimal sparse Fourier transform. In *Proceedings of the forty-fourth annual ACM symposium on Theory of computing*, pages 563–578. ACM, 2012.

[HIKP12b] Haitham Hassanieh, Piotr Indyk, Dina Katabi, and Eric Price. Simple and practical algorithm for sparse Fourier transform. In *Proceedings of the twenty-third annual ACM-SIAM symposium on Discrete Algorithms*, pages 1183–1194. SIAM, 2012.

[HR16] Ishay Haviv and Oded Regev. The restricted isometry property of subsampled Fourier matrices. In *SODA*, pages 288–297. arXiv preprint arXiv:1507.01768, 2016.

[IK14] Piotr Indyk and Michael Kapralov. Sample-optimal Fourier sampling in any constant dimension. In *Foundations of Computer Science (FOCS), 2014 IEEE 55th Annual Symposium on*, pages 514–523. IEEE, 2014.

[IKP14] Piotr Indyk, Michael Kapralov, and Eric Price. (Nearly) Sample-optimal sparse Fourier transform. In *Proceedings of the Twenty-Fifth Annual ACM-SIAM Symposium on Discrete Algorithms*, pages 480–499. SIAM, 2014.

[Iwe08] Mark A Iwen. A deterministic sub-linear time sparse Fourier algorithm via non-adaptive compressed sensing methods. In *Proceedings of the nineteenth annual ACM-SIAM symposium on Discrete algorithms*, pages 20–29. Society for Industrial and Applied Mathematics, 2008.

[Iwe10] Mark A Iwen. Combinatorial sublinear-time Fourier algorithms. *Foundations of Computational Mathematics*, 10(3):303–338, 2010.

[Iwe13] Mark A Iwen. Improved approximation guarantees for sublinear-time Fourier algorithms. *Applied And Computational Harmonic Analysis*, 34(1):57–82, 2013.

[Kap16] Michael Kapralov. Sparse Fourier transform in any constant dimension with nearly-optimal sample complexity in sublinear time. In *Symposium on Theory of Computing Conference, STOC'16, Cambridge, MA, USA, June 19-21, 2016*, 2016.

[Kap17] Michael Kapralov. Sample efficient estimation and recovery in sparse FFT via isolation on average. In *Foundations of Computer Science (FOCS), 2017 IEEE 58th Annual Symposium on*, pages 651–662. Ieee, 2017.

[KM93] Eyal Kushilevitz and Yishay Mansour. Learning decision trees using the Fourier spectrum. *SIAM Journal on Computing*, 22(6):1331–1348, 1993.

[KVZ19] Michael Kapralov, Ameya Velingker, and Amir Zandieh. Dimension-independent sparse Fourier transform. In *Proceedings of the Thirtieth Annual ACM-SIAM Symposium on Discrete Algorithms*, pages 2709–2728. SIAM, 2019.

[LN19] Yi Li and Vasileios Nakos. Deterministic sparse Fourier transform with an ℓ_∞ guarantee. *arXiv preprint arXiv:1903.00995*, 2019.

[LWC13] David Lawlor, Yang Wang, and Andrew Christlieb. Adaptive sub-linear time Fourier algorithms. *Advances in Adaptive Data Analysis*, 5(01):1350003, 2013.

[Man92] Yishay Mansour. Randomized interpolation and approximation of sparse polynomials. In *International Colloquium on Automata, Languages, and Programming*, pages 261–272. Springer, 1992.

[MZIC17] Sami Merhi, Ruochuan Zhang, Mark A Iwen, and Andrew Christlieb. A new class of fully discrete sparse Fourier transforms: Faster stable implementations with guarantees. *Journal of Fourier Analysis and Applications*, pages 1–34, 2017.

[NSW19] Vasileios Nakos, Zhao Song, and Zhengyu Wang. (Nearly) Sample-optimal sparse Fourier transform in any dimension; RIPless and Filterless. In *FOCS*. https://arxiv.org/pdf/1909.11123.pdf, 2019.

[NT09] Deanna Needell and Joel A Tropp. CoSaMP: Iterative signal recovery from incomplete and inaccurate samples. *Applied and computational harmonic analysis*, 26(3):301–321, 2009.

[NV09] Deanna Needell and Roman Vershynin. Uniform uncertainty principle and signal recovery via regularized orthogonal matching pursuit. *Foundations of computational mathematics*, 9(3):317–334, 2009.

[NV10] D Needel and R Vershynin. Signal recovery from inaccurate and incomplete measurements via regularized orthogonal matching pursuit. *IEEE Journal of Selected Topics in Signal Processing*, pages 310–316, 2010.

[Phi] Phillips. Compressed sense. https://www.philips.com/healthcare/resources/landing/compressed-sense.

[PR14] Sameer Pawar and Kannan Ramchandran. A robust R-FFAST framework for computing a k-sparse n-length DFT in $O(k \log n)$ sample complexity using sparse-graph codes. In *Information Theory (ISIT), 2014 IEEE International Symposium on*, pages 1852–1856. IEEE, 2014.

[PS15] Eric Price and Zhao Song. A robust sparse Fourier transform in the continuous setting. In *Foundations of Computer Science (FOCS), 2015 IEEE 56th Annual Symposium on*, pages 583–600. IEEE, 2015.

[Rao19] Sharavas Rao. Improved lower bounds for the restricted isometry property of subsampled fourier matrices. In *arXiv preprint*. https://arxiv.org/pdf/1903.12146.pdf, 2019.

[Rey89] George O Reynolds. *The New Physical Optics Notebook: Tutorials in Fourier Optics.* ERIC, 1989.

[RV08] Mark Rudelson and Roman Vershynin. On sparse reconstruction from Fourier and Gaussian measurements. *Communications on Pure and Applied Mathematics: A Journal Issued by the Courant Institute of Mathematical Sciences*, 61(8):1025–1045, 2008.

[Sie] Siemens. Compressed sensing beyond speed. https://www.healthcare.siemens.com/magnetic-resonance-imaging/clinical-specialities/compressed-sensing.

[Son17] Zhao Song. High dimensional Fourier transform in the continuous setting. In *Manuscript*, 2017.

[Son19] Zhao Song. *Matrix Theory : Optimization, Concentration and Algorithms*. PhD thesis, The University of Texas at Austin, 2019.

[TG07] Joel Tropp and Anna C Gilbert. Signal recovery from partial information via orthogonal matching pursuit. *IEEE Trans. Inform. Theory*, 53(12):4655–4666, 2007.

[Voe11] David G Voelz. *Computational Fourier Optics: A MATLAB Tutorial (SPIE Tutorial Texts Vol. TT89)*. SPIE press, 2011.