

# FAST SPARSE 2-D DFT COMPUTATION USING SPARSE-GRAPH ALIAS CODES

Frank Ong, Sameer Pawar and Kannan Ramchandran

Dept. of Electrical Engineering and Computer Sciences, University of California, Berkeley  
{ frankong, spawar, kannanr } @eecs.berkeley.edu

## ABSTRACT

We present a novel algorithm, named the 2D-FFAST (Two-dimensional Fast Fourier Aliasing-based Sparse Transform), to compute a sparse 2D-Discrete Fourier Transform (2D-DFT) featuring both low sample and computational complexity. The proposed algorithm is based on diverse concepts from signal processing (sub-sampling and aliasing), coding theory (sparse-graph codes) and number theory (Chinese-remainder-theorem) and generalizes the 1D-FFAST algorithm recently proposed by Pawar and Ramchandran [1, 2] to the 2D setting. Concretely, our proposed 2D-FFAST algorithm computes a  $k$ -sparse 2D-DFT, with a uniformly random support, of size  $N = N_x \times N_y$  using  $O(k)$  noiseless spatial-domain measurements in  $O(k \log k)$  computational time. Our results are attractive when the sparsity is sub-linear with respect to the signal dimension, that is, when  $k \rightarrow \infty$  and  $k/N \rightarrow 0$ . For the case when the spatial-domain measurements are corrupted by additive noise, our 2D-FFAST framework extends to a noise-robust version of computing a 2D-DFT using  $O(k \log^3 N)$  measurements in sub-linear time of  $O(k \log^4 N)$ . Empirically, we show that the 2D-FFAST can compute a  $k = 3509$  sparse 2D-DFT of a  $508 \times 508$ -size phantom image using only  $4.75k$  measurements. We also empirically evaluate the 2D-FFAST algorithm on a real-world magnetic resonance brain image using a total of 60.18% of Fourier measurements to provide an almost instant reconstruction with SNR=4.5 dB. This provides empirical evidence that the 2D-FFAST architecture is applicable to a wider class of input signals than analyzed theoretically in the paper.

**Index Terms**— Sparse graph code, Compressed sensing, Multi-dimensional Signal Processing, Fast Fourier Transform

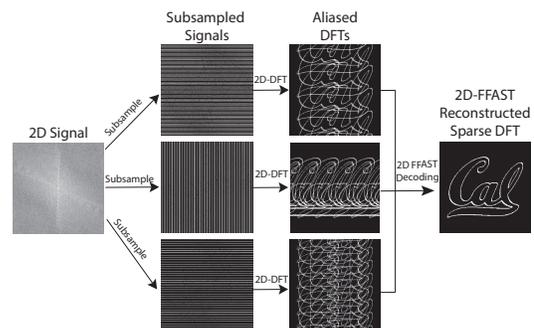
## 1. INTRODUCTION

In many imaging applications, such as magnetic resonance angiography, computed tomography, and astronomical imaging, the image of interest has a sparse representation in the Fourier domain. Recent results in compressed sensing [3, 4] exploit this sparse structure to acquire and reconstruct signals from far fewer measurements than required by the Shannon-Nyquist theorem. However, the most popular class of com-

pressed sensing based reconstruction algorithms involves iteratively alternating between the spatial domain representation and the Fourier domain representation of the signals, and consequently are computationally expensive. Hence, such algorithms have limited scope in devices and acquisition devices and systems demanding inexpensive, low-power or real-time signal analysis.

While many algorithms with low computational complexity [5, 6, 7, 8, 9, 10, 11, 12] have been proposed to compute a sparse 1D discrete-Fourier-transform (DFT), extensions of these algorithms to 2D sparse signals, with recovery guarantees similar to 1D, can be non-trivial, and very few 2D algorithms [12, 13] were proposed. Yet multidimensional signals, such as images and videos, often have much sparser representations than 1D signals, and can arguably be found in a wider range of signal processing applications. Hence, a practical algorithm with *both low computational and sample complexity* for computing a sparse 2D-DFT is of great interest.

In this work, we present an algorithm, named the 2D-FFAST (Two-dimensional Fast Fourier Aliasing-based Sparse Transform), to compute a sparse 2D-DFT featuring both low sample and computational complexity. While a 2D-DFT can be uniquely mapped to a 1D-DFT when the dimensions are co-prime, our main contribution is the design of the 2D-FFAST algorithm for a wide class of 2D signals as described in Section 2, whose 2D-DFT computation *cannot* be mapped to 1D-DFT. In particular, we show that the 1D-FFAST architecture proposed in Pawar and Ramchandran [1] can be lifted to the 2D setting as shown in Section 4, and show how the recovery guarantees can be preserved.



**Fig. 1.** A simplified visual illustration of how the 2D-FFAST algorithm computes a 2D sparse DFT.

Thanks to MURI CHASE Grant No. 556016, NSF CCF Eager 1439725 and the NSF fellowship for funding.

At a high level, our 2D FFAST algorithm induces sparse graph codes in the 2D-DFT domain via a Chinese-Remainder-Theorem (CRT)-guided 2D sub-sampling operation in the spatial-domain. This insight can then be exploited to devise a fast greedy onion-peeling style algorithm that computes the 2D-DFT. A simplified visual illustration of the 2D-FFAST architecture is provided in Figure 1.

Before diving into the main results, we emphasize the following caveats of our algorithm and analysis: First, our proposed 2D-FFAST algorithm does not apply to 2D signals with arbitrary dimensions but applies to a large set of 2D signals, whose dimensions satisfy certain conditions, as described in Section 2. Secondly, our analytical results are probabilistic and hold for asymptotic values of  $k$  and signal dimensions  $N_x \times N_y$ , with a success probability that approaches 1 asymptotically. Thirdly, our analytical results assume a uniformly random model for the support of the non-zero DFT coefficients. Lastly, for the noisy case, we assume that the non-zero DFT coefficients belong to an arbitrarily large but finite constellation such that the effective signal-to-noise ratio is finite for analysis purpose.

**Related work:** A number of previous works [5, 6, 7, 8, 9, 10, 11, 1] have addressed the problem of computing a 1D-DFT of an  $N$ -length signal that has a  $k$ -sparse Fourier transform, in sub-linear time and sample complexity. Unlike the 1D-DFT, there are few algorithms designed for a sparse 2D-DFT: The algorithm in Gilbert et al. [6] achieves  $O(k \log^c N)$  sample and time complexity for computing a  $k$ -sparse  $N = N_x \times N_y$  2D-DFT, for some constant  $c$ . In Ghazi et al. [13], the algorithm achieves  $O(k)$  sample complexity and  $O(k \log k)$  computational complexity only when  $N_x = N_y = \sqrt{N}$ . For the general sub-linear sparsity regime, the computational complexity is  $O(k \log k + k(\log \log N)^c)$  for some constant  $c$ . In addition, the algorithm succeeds only with a *constant probability* that does not approach 1, which generally translates to inferior empirical results. In [12], Indyk et al. describe an algorithm achieving  $O(k \log N)$  sample complexity and  $O(N \log^c N)$  computational complexity for general sub-linear sparsity  $k$ . In contrast, the proposed 2D-FFAST algorithm, in the absence of any noise, computes a  $k$ -sparse 2D-DFT using  $O(k)$  samples in  $O(k \log k)$  computational complexity, for any sub-linear sparsity  $k$ , with a probability of success that approaches 1 asymptotically. In the case when the signal is corrupted by noise, 2D-FFAST computes a 2D-DFT using  $O(k \log^3 N)$  measurements and in sub-linear time of  $O(k \log^4 N)$ .

## 2. PROBLEM FORMULATION

We consider the problem of computing the 2D-DFT  $\mathbf{X}$  from the spatial-domain samples  $\mathbf{x}$ , when the transform  $\mathbf{X}$  is known to be sparse. Specifically, we show that when  $\mathbf{X}$  has precisely  $k$  non-zero DFT coefficients with a uniformly random support, one can achieve significant gains in both the number of samples used and the computational complexity.

We assume that the sparsity  $k$  of the 2D-DFT of a signal is sub-linear with respect to the signal dimension, that is  $k \propto N^\delta$ , where  $N = N_x \times N_y$  and  $0 \leq \delta < 1$ . Our 2D-FFAST algorithm further requires that the 2D signal dimension  $N$  can be factorized into  $d$  approximately same order co-prime factors  $\{Q_i\}_{i=0}^{d-1}$ , where  $d$  is an appropriately chosen constant (usually chosen as 3). For example, suppose we want to target a 2D DFT computation with dimension  $N = N_x \times N_y$  around  $512 \times 256$  and sparsity  $k \approx 2500 \approx N^{2/3}$ . Then choosing  $d = 3$ , we have each factor  $Q_i \approx 50$ . Hence, we can choose  $Q_0 = 49$ ,  $Q_1 = 50$ ,  $Q_2 = 51$ , and split the factors between  $N_x$  and  $N_y$  to obtain  $N_x = 51 \times 10 = 510$  and  $N_y = 5 \times 49 = 245$ , which are close to the targeted 2D signal dimension. While the dimension assumption limits the application to arbitrary 2D dimensions, this constraint is flexible in practice as long as the user has reasonable control on the signal dimension as shown in the previous example. We also consider the case where the spatial-domain samples are corrupted by additive Gaussian noise and assume that the non-zero DFT coefficients belong to a finite constellation for analysis purpose.

## 3. MAIN RESULT

Consider the signal model as stated in Section 2, our following theorems state the main result:

**Theorem 3.1.** *For any  $0 \leq \delta < 1$ , and large enough  $N = N_x N_y$ , the 2D-FFAST algorithm computes the  $k$ -sparse 2D-DFT of an  $N_x \times N_y$ -size input  $\mathbf{x}$ , where  $k = O(N^\delta)$ , with the following properties:*

1. **Sample complexity:** *The algorithm needs  $m = O(k)$  measurements*
2. **Computational complexity:** *The computational complexity of the 2D-FFAST algorithm is  $O(m \log m)$*
3. **Probability of success:** *The probability that the algorithm correctly computes the 2D-DFT  $\mathbf{X}$  is at least  $1 - O(1/m)$*

*Proof.* For the case when  $N_x$  and  $N_y$  are co-prime, there exists a unique one-to-one mapping from 2D-DFT to a 1D-DFT [14, 15]. For the general case, we show that there is a direct correspondence between the sparse graph codes induced by the 2D-FFAST architecture and the 1D-FFAST architecture. Please see extended paper on arXiv [16] for detail.  $\square$

**Theorem 3.2.** *For a sufficiently high signal-to-noise-ratio, any  $0 \leq \delta < 1$ , and a large enough  $N = N_x N_y$ , the 2D-FFAST algorithm computes the  $k$ -sparse 2D-DFT of an  $(N_x \times N_y)$ -size input  $\mathbf{x}$ , where  $k = O(N^\delta)$ , with the following properties:*

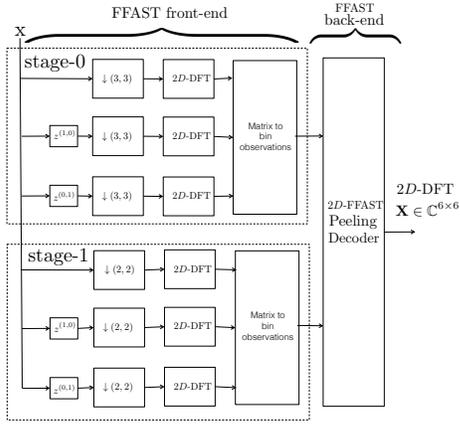
1. **Sample complexity:** *The algorithm needs  $m = O(k \log^3(N))$  measurements*
2. **Computational complexity:** *The computational complexity of the 2D-FFAST algorithm is  $O(k \log^4(N))$*
3. **Probability of success:** *The probability that the algorithm correctly computes the 2D-DFT  $\mathbf{X}$  is at least  $1 - O(1/m)$*

*Proof.* Please see extended paper on arXiv [16]  $\square$

## 4. 2D-FFAST ARCHITECTURE

Throughout this section, we will use a simple 2D example signal to describe the proposed 2D-FFAST architecture. Specifically, we consider a  $6 \times 6$  2D signal  $\mathbf{x}$ , such that its 2D-DFT  $\mathbf{X}$  is 4-sparse:  $X[1][3]$ ,  $X[2][0]$ ,  $X[2][3]$  and  $X[4][0]$ .

Our 2D-FFAST architecture is built on the design principles of 1D-FFAST in [1] and consists of a deterministic sub-sampling front-end and an associated back-end peeling-decoder algorithm. The 2D-FFAST front-end sub-samples the input signal and its circularly shifted version through multiple stages. In addition, each stage has multiple delay paths and is parametrized by a single pair of sampling factors: one for horizontal subsampling, the other one for vertical subsampling. For example, the 2D-FFAST architecture of Fig. 2 has 2 stages and 3 delay (circular shift) paths per stage.

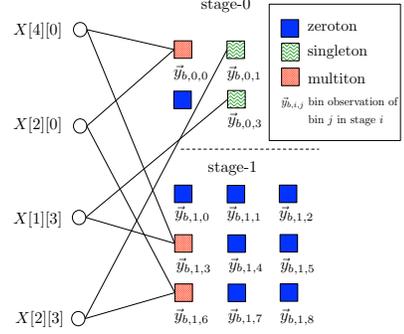


**Fig. 2.** A 2D-FFAST architecture of  $d = 2$  stages. Each stage further has 3 delay chains and a common sub-sampling factor. Bin-observations are formed by collecting one scalar output from each of the 3 delay chains, which are further processed by a ‘peeling-decoder’ to reconstruct  $\mathbf{X}$

### 4.1. Front-end: 2D-subsampling and delays

As shown in Figure 2, our 2D-FFAST front-end takes in a 2D signal  $\mathbf{x}$  and outputs bin observations. A bin observation is simply a vector formed by collecting the scalar outputs from each of the delay chains in a stage of the 2D-FFAST front-end. The bin observations, in the terms of the non-zero DFT coefficients  $\mathbf{x}$  of the signal, can be computed using the basic signal processing identities of aliasing and circular shifts. For our example signal, the bin observation vector of bin 1 in stage 0 is given by  $\vec{y}_{b,0,1} = (X[2][3], e^{2\pi i 2/6} X[2][3], e^{2\pi i 3/6} X[2][3])^\top$ .

A key observation of the 2D-FFAST is that the relation between the unknown non-zero DFT coefficients of the signal and the output bin observations can be represented as a bi-partite graph as shown in Fig. 3. The left nodes in the graph represent the non-zero 2D-DFT coefficients and the right nodes represent the bins with vector observations. A bin is called a zero-ton if it has no left neighbor in the graph. A singleton bin has exactly one left neighbor and a multiton bin has more than one neighbor on left in the graph.



**Fig. 3.** A bi-partite graph representing the relation between the bin observations and the non-zero 2D-DFT coefficients  $\mathbf{X}$  for our  $6 \times 6$  example.

### 4.2. Back-end: 2D-FFAST peeling-decoder

Given the bin observations, we would like to recover the non-zero 2D-DFT coefficients. Now, suppose a ‘genie’ informs the decoder which bins in the graph are zero-tons, singletons, and multitons along with the locations and values of the connected DFT coefficients for singletons. Then, effectively we have access to the bi-partite graph and our 2D-FFAST peeling-decoder repeats the following steps:

1. Select all the edges in the graph with right degree 1.
2. Remove these edges from the graph as well as the associated left and right nodes.
3. Remove all edges connected to the left nodes removed in step-2. When an edge is removed, its contribution is subtracted from the connected bin.

Decoding is successful if all the edges from the graph have been removed in the end. In [1, 16], we have shown that if the 2D sub-sampling factors are carefully chosen following the Chinese-Remainder-Theorem (CRT), the induced bipartite graph is such that the 2D-FFAST peeling-decoder succeeds with probability approaching 1, asymptotically in  $k$ .

To replace the ‘genie’, we make the following important observation: if a bin is a singleton, then the phases of the bin observations allow the decoder to identify the support. This is because each delay chain produces a linear phase proportional to the DFT support. By shifting horizontally and vertically separately in each stage, the decoder can extract the 2D support of the singletons even without the help of the genie. In addition, in [1], we show that the bin observation allows us to almost surely identify zero-ton, singleton and multiton bins, thus eliminating the need for a ‘genie’.

### 4.3. Noise-Robust 2D-FFAST

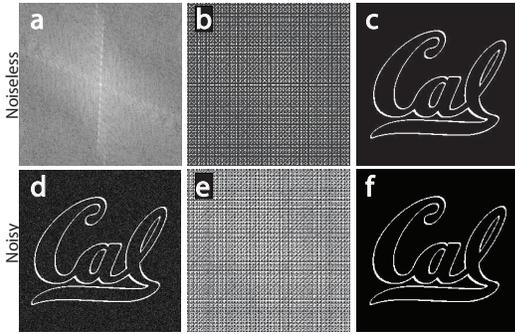
In this section, we provide a brief overview of robust extensions of 2D-FFAST. In [2], we have shown that a noiseless 1D-FFAST framework can be made *noise-robust* by using  $O(\log^3 N)$  number of delay-chains per sub-sampling stage, achieving sub-linear computational complexity. Since for 2D-DFT, we follow a *separable* approach and perform 2 indepen-

dent ratio-tests to determine the 2D support of a single-ton bin, all the results from [2] follow. With  $O(\log^3 N)$  specially designed delays, the 2D-FFAST algorithm computes a  $k$ -sparse 2D-DFT in sub-linear time of  $O(k \log^4 N)$  using  $O(k \log^3 N)$  noise-corrupted measurements.

## 5. SIMULATIONS

In this section, we empirically evaluate performance of the 2D-FFAST algorithm for, both synthetic as well as real-world magnetic resonance images that go well beyond the signal model assumed in Section 2. Thus, providing an empirical evidence that the 2D-FFAST architecture is applicable to a wider class of input signals beyond the theoretical analysis provided in the paper. Additional simulation results can be found in our extended arXiv paper [16].

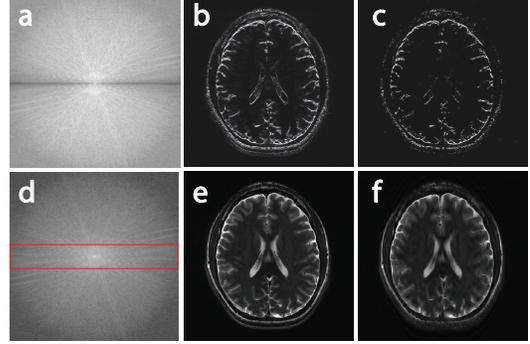
### 5.1. Exact $k$ -sparse synthetic image



**Fig. 4.** Simulation results on ‘Cal’ image of size  $N_x \times N_y = 280 \times 280$  and sparsity  $k = 3509$ :

- (a) Log-intensity plot of the noiseless 2D-DFT
- (b) Front-end subsampled noiseless 2D-DFT with sampling factors 5, 7, and 8 and 3 delays per stage. Total samples:  $m = 4.75k = 16668$
- (c) Perfectly reconstructed noiseless ‘Cal’ image using 2D FFAST
- (d) Noisy image with SNR = 13 dB (Contrast enhanced by 20x for display)
- (e) Front-end subsampled noisy 2D-DFT with sampling factors 5, 7, and 8 and 7 delays per stage. Total samples:  $m = 10.06k = 35308$
- (f) Perfect location recovered noisy ‘Cal’ image using 2D FFAST with NMSE= 0.0136

Consider a  $280 \times 280$  dimensional synthetically generated ‘Cal’ image, shown in Fig. 4, that has  $k = 3509$  non-zero pixels. We input the 2D-inverse Fourier transform of the cal image to the 2D-FFAST algorithm. Note, that the ‘Cal’ image is perfectly  $k$ -sparse but the support of the non-zero pixels is not distributed uniformly at random. For noiseless, the 2D-FFAST perfectly reconstructs the image with  $4.75k$  measurements. For noisy, 2D-FFAST recovers the exact support with  $10.06k$  measurements, and results in a normalized MSE of 0.0136, when the input has SNR of 13 dB.



**Fig. 5.** Simulation results on ‘Brain’ image of size  $N_x \times N_y = 504 \times 504$ : (a) Log intensity plot of 2D-DFT of the original ‘Brain’ image, after application of the vertical difference operation. (b) Differential ‘Brain’ image obtained using the vertical difference operation on the original ‘Brain’ image. (c) Differential ‘Brain’ image reconstructed using a 3-stage 2D FFAST algorithm from 56.71% of Fourier samples. (d) Log intensity plot of the 2D-DFT of the original ‘Brain’ image. The red enclosed region is fully sampled and used for the stable inversion. (e) Original ‘Brain’ image in spatial domain. (f) Reconstructed ‘Brain’ image using the 2D-FFAST architecture. The total number of Fourier samples used is 60.18%.

### 5.2. Application of the 2D-FFAST for MR imaging

In this section, we apply the 2D-FFAST algorithm to reconstruct a brain image acquired on an MR scanner (Fig. 5). In MR imaging the samples are acquired in the Fourier domain and the task is to reconstruct the spatial image from less Fourier samples. To reconstruct the full brain image using 2D-FFAST, we perform the following two-step procedure:

- *Differential space signal acquisition:* We perform a vertical finite difference operation on the image by multiplying the 2D-DFT signal with  $1 - e^{2\pi i \omega_0}$ , which creates an approximately sparse differential image, as shown in Fig. 5b, and can be reconstructed using 2D-FFAST.
- *Inversion using fully sampled center frequencies:* After reconstructing the differential brain image, as shown in Fig. 5, we invert the finite difference operation by dividing the 2D-DFT samples with  $1 - e^{2\pi i \omega_0}$ . Since the inversion is not stable near the center of the Fourier domain, only the non-center frequencies are inverted, with the center region of the 2D-DFT additionally sampled.

Overall we use a total of 60.18% of Fourier measurements to reconstruct the brain image using the 2D-FFAST algorithm along with the fully sampled center frequencies. The resulting signal-to-noise ratio of the reconstructed image is 4.5173 dB. While the reconstruction error is not as good as state-of-the-art compressed sensing MRI results, we emphasize that the 2D-FFAST has *both low computational complexity and low sample complexity*, which none of the state-of-the-art compressed sensing MR reconstruction can achieve. Thus, providing a new direction of research for designing faster, efficient and high fidelity MR acquisition systems.

## 6. REFERENCES

- [1] Sameer Pawar and Kannan Ramchandran, "Computing a  $k$ -sparse  $n$ -length Discrete Fourier Transform using at most  $4k$  samples and  $O(k \log k)$  complexity," *arXiv preprint arXiv:1305.0870*, May 2013, Available online: <http://arxiv.org/abs/1305.0870v2>.
- [2] Sameer Pawar and Kannan Ramchandran, "A robust sub-linear time R-FFAST algorithm for computing a sparse DFT," *arXiv preprint arXiv:1501.00320*, Jan. 2015, Available online: <http://arxiv.org/abs/1501.00320v1>.
- [3] D L Donoho, "Compressed sensing," *Information Theory, IEEE Transactions on*, vol. 52, no. 4, pp. 1289–1306, Apr. 2006.
- [4] E J Candes, J Romberg, and T Tao, "Robust uncertainty principles: exact signal reconstruction from highly incomplete frequency information," *Information Theory, IEEE Transactions on*, vol. 52, no. 2, pp. 489–509, 2006.
- [5] A C Gilbert, S Guha, P Indyk, S Muthukrishnan, and M Strauss, "Near-optimal sparse fourier representations via sampling," in *the thirty-fourth annual ACM symposium*, New York, New York, USA, May 2002, pp. 152–161, ACM.
- [6] A C Gilbert, S Muthukrishnan, and M Strauss, "Improved Time Bounds for Near-Optimal Sparse Fourier Representations," *Optics & Photonics 2005*, vol. 5914, pp. 1–15, July 2005.
- [7] Anna C Gilbert, Martin J Strauss, and Joel A Tropp, "A Tutorial on Fast Fourier Sampling," *IEEE Signal Processing Magazine*, vol. 25, no. 2, pp. 57–66, Mar. 2008.
- [8] Anna C Gilbert, Piotr Indyk, Mark Iwen, and L Schmidt, "Recent Developments in the Sparse Fourier Transform: A compressed Fourier transform for big data," *IEEE Signal Processing Magazine*, vol. 31, no. 5, pp. 91–100, Sept. 2014.
- [9] Haitham Hassanieh, Piotr Indyk, Dina Katabi, and Eric Price, "Nearly optimal sparse fourier transform," in *the 44th symposium*, New York, New York, USA, May 2012, pp. 563–578, ACM.
- [10] H Hassanieh, P Indyk, D Katabi, and E Price, "Simple and practical algorithm for sparse Fourier transform," *Proceedings of the twenty-third annual ACM-SIAM symposium on Discrete Algorithms*, 2012.
- [11] P Indyk, M Kapralov, and E Price, "(Nearly) sample-optimal sparse Fourier transform," in *Proceedings of the Twenty-Fifth Annual ACM-SIAM Symposium on Discrete Algorithms*. 2014, pp. 480–499, SIAM.
- [12] Piotr Indyk and Michael Kapralov, "Sample-Optimal Fourier Sampling in Any Constant Dimension," *2014 IEEE 55th Annual Symposium on Foundations of Computer Science (FOCS)*, pp. 514–523, 2014.
- [13] Badih Ghazi, Haitham Hassanieh, Piotr Indyk, Dina Katabi, Erik Price, and Lixin Shi, "Sample-optimal average-case sparse Fourier Transform in two dimensions," in *2013 51st Annual Allerton Conference on Communication, Control, and Computing (Allerton)*. 2013, pp. 1258–1265, IEEE.
- [14] I J Good, "The interaction algorithm and practical Fourier analysis," *Journal of the Royal Statistical Society. Series B (Methodological)*, 1958.
- [15] L H Thomas, *Using a computer to solve problems in physics*, Applications of Digital Computers, 1963.
- [16] Frank Ong, Sameer Pawar, and Kannan Ramchandran, "Fast and Efficient Sparse 2D Discrete Fourier Transform using Sparse-Graph Codes," *arXiv preprint arXiv:1509.05849*, Sept. 2015, Available online: <http://arxiv.org/abs/1509.05849v1>.