EMPIRICAL EVALUATION OF TWO DETERMINISTIC SPARSE FOURIER TRANSFORMS

M. A. Iwen

Institute for Mathematics and Its Applications (IMA)
University of Minnesota
http://www.ima.umn.edu/~iwen/index.html
iwen@ima.umn.edu

ABSTRACT

This paper empirically evaluates a recently proposed **D**eterministic **S**parse **F**ourier **T**ransform algorithm (hereafter called **DSFT**) for the first time. Our experiments indicate that DSFT is capable of guaranteed general frequency-sparse signal recovery using subNyquist sampling for realistic bandwidth values. Furthermore, we show that both variants of DSFT have fast reconstruction runtimes. In fact, the sublinear-time DSFT variant is shown to be faster than a traditional Fast Fourier Transform (FFT) for highly-sparse wideband signals.

Index Terms— Fourier transforms, Discrete Fourier transforms, Algorithms, Signal Processing

1. INTRODUCTION

Compressed Sensing (CS) is an exciting signal acquisition and recovery paradigm in which highly compressible signals can be recovered from a few linear measurements [4]. In this paper we focus on a particular type of compressible signal, namely signals consisting of a small number of significant Fourier modes. Thus, we sample a frequency-sparse signal f on a small deterministic sample set and then reconstruct the signal by returning a list of its predominant frequencies. This sensing paradigm has proven useful in many areas, including MR imaging [5, 6] and wideband analog-to-digital converter (ADC) design [7, 8].

Existing CS-based Fourier algorithms [4, 9, 10, 11] are all capable of producing incorrect results with some small probability, making them inappropriate for failure intolerant applications. Furthermore, the reconstruction algorithms of [4, 9] (based on linear programming and orthogonal matching pursuit, respectively) exhibit polynomial runtime dependence on the signal's bandwidth. Hence, their runtime requirements can be prohibitively expensive for wideband signals. Recently proposed Deterministic Sparse Fourier Transform (DSFT) [1, 2, 3] methods are both deterministic and (sub)linear-time in bandwidth. Furthermore, DSFT is consistent with recently proposed subNyquist ADC designs [7, 8]. These ADC designs, which are based on random sampling, currently require the implementation of random clocks, pseudo-random switches, etc.. Due to its deterministic nature, DSFT would allow one to build similar circuits with fixed sample sets in the hardware, thus simplifying circuit design.

DSFT is the first deterministic CS-based Fourier algorithm guaranteed to exactly reconstruct every N-bandwidth signal consisting of $k \ll N$ non-zero frequencies. In this paper, we empirically evaluate two variants of DSFT. Our experiments show that DSFT can reconstruct Fourier-sparse signals using subNyquist sampling at modest bandwidth values. Furthermore, we show that both DSFT variants are fast. The sublinear-time DSFT variant is faster than a traditional Fast Fourier Transform (FFT) for highly-sparse wideband signals.

The remainder of this paper is organized as follows. In Section 2 we introduce necessary background and terminology. Section 3 gives general information concerning our empirical evaluation of DSFT. Sections 4 and 5 then give specific sampling and runtime results, respectively. Finally, we conclude with a brief discussion in Section 6.

2. PRELIMINARIES

Throughout the remainder of this paper we will restrict our attention to complex-valued signals, $f : [0, 2\pi] \to \mathbb{C}$, which are band-limited and frequency-sparse. Hence, we assume there exists an $N \in \mathbb{N}$ such that for each signal f,

$$\Omega_f = \left\{ \omega \in \mathbb{Z} \;\middle|\; \widehat{f}(\omega) \not\approx 0 \right\} \subsetneq \left(-\left\lceil \frac{N}{2} \right\rceil, \left\lfloor \frac{N}{2} \right\rfloor \right].$$

Furthermore, we assume that $k = |\Omega_f| \ll N$. We will refer to any such f as a *k-frequency superposition*. For any k-frequency superposition f, we will refer to the k non-zero elements of Ω_f as f's *energetic frequencies*. Furthermore, we will refer to the process of either calculating or measuring f at any $t \in [0, 2\pi]$ as *sampling* from f. Finally, let $p_0 = 1$ and p_l be the l^{th} prime natural number. We define $q \in \mathbb{N}$ to be such that

$$p_{q-1} < k \leq p_q.$$

Recently, a Deterministic Sparse Fourier Transform algorithm (DSFT) [1, 2, 3] was developed by building upon the

Algorithm 1 LINEAR-TIME DSFT

Input: Signal pointer f, integers $k \leq N$
Output: $\hat{\mathbf{R}}^s$, a sparse representation for $\hat{f}$
Initialize $\hat{\mathbf{R}}^s \leftarrow \emptyset$
Set $K = 2 \cdot k \cdot \lfloor \log_k N \rfloor$, q so that $p_{q-1} < k \leq p_q$
for j from 0 to K **do**
  $\mathbf{A}_{p_{q+j}} \leftarrow f(0), f\left(\frac{2\pi}{p_{q+j}}\right), \ldots, f\left(\frac{2\pi(p_{q+j}-1)}{p_{q+j}}\right)$
  $\widehat{\mathbf{A}_{p_{q+j}}} \leftarrow \mathbf{FFT}[\mathbf{A}_{p_{q+j}}]$
end for
for ω from $1 - \lceil \frac{N}{2} \rceil$ to $\lfloor \frac{N}{2} \rfloor$ **do**
  $\mathbb{R}\mathrm{e}\{C_\omega\} \leftarrow \text{median}\left\{\mathbb{R}\mathrm{e}\left\{\widehat{\mathbf{A}_{\mathrm{p_{q+j}}}}(\omega \bmod \mathrm{p_{q+j}})\right\} \,\middle|\, 0 \leq \mathrm{j} \leq \mathrm{K}\right\}$
  $\mathbb{I}\mathrm{m}\{C_\omega\} \leftarrow \text{median}\left\{\mathbb{I}\mathrm{m}\left\{\widehat{\mathbf{A}_{\mathrm{p_{q+j}}}}(\omega \bmod \mathrm{p_{q+j}})\right\} \,\middle|\, 0 \leq \mathrm{j} \leq \mathrm{K}\right\}$
end for
$\hat{\mathbf{R}}^s \leftarrow (\omega, C_\omega)$ entries for k largest magnitude C_ω's

number theoretic hashing techniques first proposed in [12, 13]. For a given input signal f, DSFT outputs f's energetic frequencies along with their Fourier coefficients. See Algorithm 1 for a variant of DSFT with runtime linear in the input signal f's bandwidth, N. We will hereafter refer to Algorithm 1 as *Linear-Time DSFT*.

The following theorems are proven in [2]. Theorem 1 concerns Linear-Time DSFT. Theorem 2 concerns a more complicated and faster variant of DSFT, hereafter called *Sublinear-Time DSFT*.

Theorem 1. *Suppose f is a k-frequency superposition. Then, Linear-Time DSFT (i.e., 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 samples taken from f will be $O\left(k^2 \cdot \log_k^2 N \cdot \log(k \log N)\right)$.*

Theorem 2. *Suppose f is a k-frequency superposition. Then, Sublinear-Time DSFT (see [2]) can exactly recover f in $O\left(k^2 \cdot \frac{\log^2 N \cdot \log^2(k \log N) \cdot \log^2 \frac{N}{k}}{\log^2 k \cdot \log \log \frac{N}{k}}\right)$ time. The number of samples taken from f will be $O\left(k^2 \cdot \frac{\log^2 N \cdot \log(k \log N) \cdot \log^2 \frac{N}{k}}{\log^2 k \cdot \log \log \frac{N}{k}}\right)$.*

Suppose that $k = \Theta(N^\alpha)$. In this case Theorem 1 tells us that Linear-Time DSFT exactly recovers k-frequency superpositions in $O(N \cdot k \cdot \log(N))$ time using $O\left(k^2 \log(N)\right)$ samples. Similarly, Theorem 2 guarantees that Sublinear-Time DSFT exactly recovers k-frequency superpositions in $O\left(k^2 \log^4 N\right)$ time using $O\left(k^2 \log^3 N\right)$ samples. In signal processing applications the subNyquist sampling required to compute either DSFT variant's FFTs can be accomplished via $O(k \cdot \text{polylog}(N))$ parallel $O(k \cdot \text{polylog}(N))$-rate analog-to-digital converters. Finally, we note that both Linear-Time and Sublinear-Time DSFT are tolerant to arbitrary bounded noise. See [2] for details.

3. EMPIRICAL EVALUATION

Both DSFT variants were coded in C. All experiments were run on a QuadCore2 2.4 Ghz Ubuntu Linux machine with 3 GB of RAM. We used FFTW 3.2 [14] with an FFTW_MEASURE plan as our FFT for runtime comparisons in Section 5. All bandwidth values (i.e., array lengths) used for generating our graphs were powers of two.

All N-bandwidth k-frequency superpositions used for tests below where constructed as follows. First, k frequencies were selected uniformly at random from $\left(-\left\lceil \frac{N}{2} \right\rceil, \left\lfloor \frac{N}{2} \right\rfloor\right]$. Next, each randomly selected frequency was given a uniformly random phase. Their coefficient magnitudes were left as 1.

Despite the fact that both DSFT variants are guaranteed to deterministically recover any such superposition, every data point in the next two sections' graphs is the result of 1000 runs on randomly generated superpositions. During all runs the errors of both DSFT variants were monitored. Their precisions were within an order of magnitude of FFTW's for all energetic superposition frequency coefficients in all tests reported on below. Hence, in the noise-free case DSFT behaves as expected in terms of 'exactly reconstructing' sparse superpositions.

4. DSFT SAMPLING REQUIREMENTS

Figure 1 contains graphs of our DSFT implementations' sampling requirements. The left graph contains the number of samples (i.e., function evaluations) used by Sublinear-Time DSFT to recover k-frequency superpositions at four different bandwidth values. The left graph's vertical axis is in terms of bandwidth-fraction sampled $\left(\text{i.e., } \frac{\text{DSFT samples}}{\text{bandwidth } N} \text{ for each curve}\right)$. Thus, for example, we can see that Sublinear-Time DSFT can recover any 8-frequency superposition with bandwidth 2^{19} by sampling the superposition less than 2^{18} times (i.e., by using less than half the samples a full FFT requires).

The right graph in Figure 1 compares the sampling requirements of Linear-Time and Sublinear-Time DSFT. This graph plots, for various bandwidth values, the maximum number of frequencies a superposition can contain and still be recovered by (Sub)Linear-Time DSFT using subNyquist sampling. Looking at Figure 1's right graph we can see that Linear-Time DSFT can use a sublinear number of samples to recover superpositions containing (roughly) an order of magnitude more frequencies than can be recovered by Sublinear-Time DSFT using sublinear sampling. For example, Sublinear-Time DSFT can recover superpositions containing at most 14 frequencies at bandwidth 2^{19} using fewer than 2^{19} samples, whereas Linear-Time DSFT can recover superpositions containing over 100 frequencies.

Figure 2 explores the sampling required for both DSFT variants' coprime FFTs (i.e., the signals' sizes whose FTs we must calculate). In the left graph of Figure 2 we fixed the bandwidth at 1024 and plotted, as the number of super-

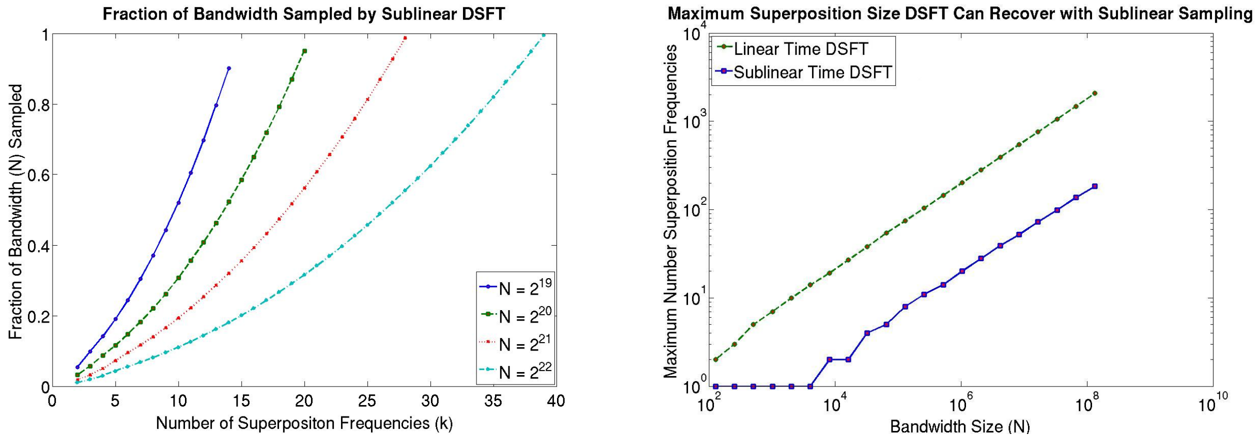

Fig. 1. Sublinear-Time DSFT Sampling Requirements, and Both Variants' Maximum Sparsity with subNyquist Sampling

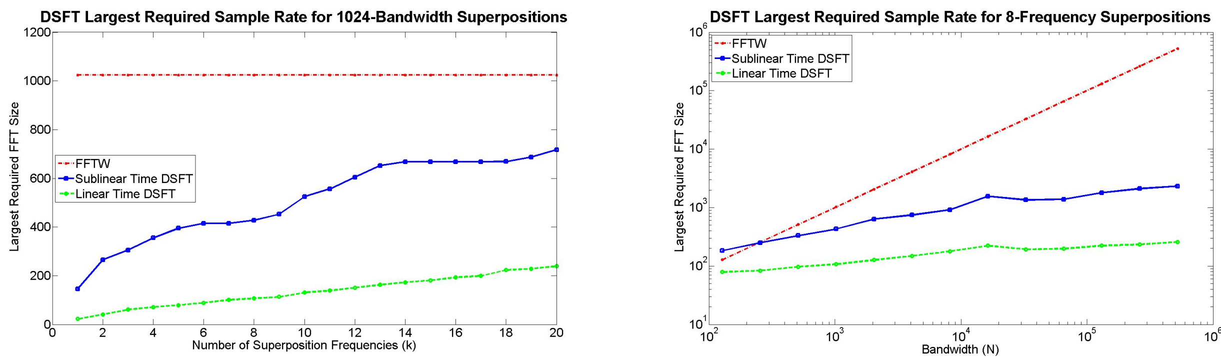

Fig. 2. Largest FT Required for 1024-bandwidth Superpositions, and Largest FT Required for 8-Frequency Superpositions

position frequencies varied for both variants of DSFT, the maximum signal's size whose FT we had to calculate. FFTW is also included for reference. In the right graph we fixed the number of superposition frequencies at 8 and plotted, as bandwidth size varied, the maximum signal's size whose FT we had to calculate. Looking at Figure 2 we note that for highly-sparse signals with bandwidths larger than 128, both DSFT variants' parallel FFTs are all of sublinear-size signals (i.e., their parallel ADCs would all sample at subNyquist rates). Furthermore, as before, we note that Linear-Time DSFT has milder sampling (rate) requirements than Sublinear-Time DSFT.

5. DSFT RUNTIME

Figure 3 contains graphs of both DSFT variants' runtimes (averaged over 1000 runs per data point as per Section 3). The left graph compares Linear-Time and Sublinear-Time DSFTs' runtimes (in milliseconds) for 1024-bandwidth superpositions containing various numbers of energetic frequencies. FFTW's runtime is included for reference. As we can see, Sublinear-Time DSFT is indeed faster than Linear-Time DSFT for all sparsity levels. However, both variants required less than 5 ms for all recorded runs.

The right graph of Figure 3 plots the runtimes of both DSFT variants for 8-frequency superpositions with various bandwidths. Looking at the left graph we can see that Sublinear-Time DSFT is faster than Linear-Time DSFT for all bandwidth values greater than 128. Likewise, Sublinear-Time DSFT is faster than FFTW for all bandwidth values greater than 2^{18}. More generally, Sublinear-Time DSFT will be faster than FFTW for all highly-sparse wideband superpositions.

6. CONCLUSION

In this paper we empirically evaluated two variants of DSFT, a recently proposed Fourier Transform method for frequency-sparse signals. During the course of our evaluation we demonstrated that DSFT is capable of guaranteed sparse superposition recovery using subNyquist sampling for realistic bandwidth sizes.

We conclude by noting that Monte-Carlo versions of both

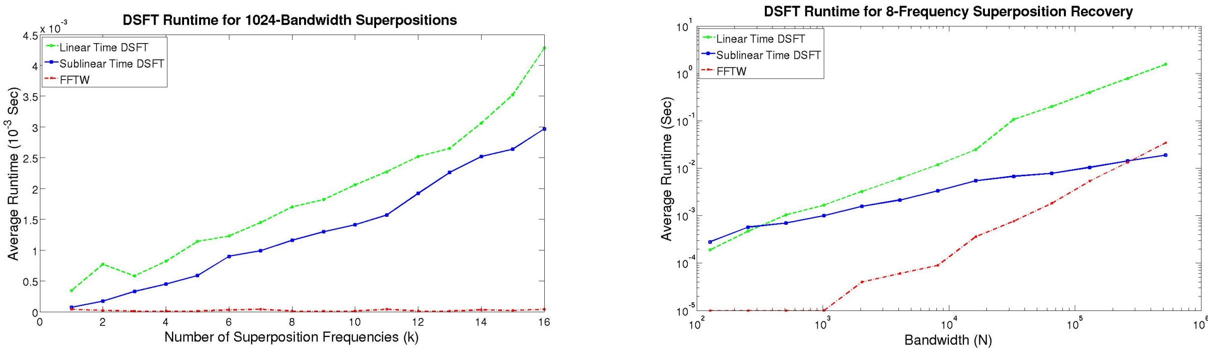

Fig. 3. Runtime of DSFT at Various Sparsity Levels, and Runtime of DSFT at Various Bandwidths

DSFT variants exist. If the user is willing to incorrectly calculate an N-bandwidth k-frequency superposition's FT with probability $\frac{1}{N^{\Theta(1)}}$, the number of samples required by Linear-Time DSFT can be reduced to $O(k \cdot \log_k N \cdot \log^2 N)$. It's runtime will be $O(N \cdot \log_k N \cdot \log^3 N)$. Similarly, Sublinear-Time DSFT's runtime/number of required samples can be reduced to $O(k \cdot \text{polylog}\, N)$. See [2] for details.

7. REFERENCES

[1] 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.

[2] M. A. Iwen, "Combinatorial sublinear-time fourier algorithms," *Submitted. Available at http://www.ima.umn.edu/~iwen/index.html*, 2008.

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

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

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

[6] 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.

[7] Jason Laska, Sami Kirolos, Yehia Massoud, Richard Baraniuk, Anna Gilbert, Mark Iwen, and Martin Strauss, "Random sampling for analog-to-information conversion of wideband signals," *Proc. IEEE Dallas Circuits and Systems Conference*, 2006.

[8] Sami Kirolos, Jason Laska, Michael Wakin, Marco Duarte, Dror Baron, Tamer Ragheb, Yehia Massoud, and Richard Baraniuk, "Analog-to-information conversion via random demodulation," *Proc. IEEE Dallas Circuits and Systems Conference*, 2006.

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

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

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

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

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

[14] M. Frigo and S. Johnson, "The design and implementation of fftw3," *Proceedings of IEEE 93 (2)*, pp. 216–231, 2005.