Nonuniform Fast Fourier Transforms Using Min-Max Interpolation

Jeffrey A. Fessler*
4240 EECS, The University of Michigan, Ann Arbor, MI 48109-2122
fessler@umich.edu

Bradley P. Sutton
BME Department, The University of Michigan
bpsutton@umich.edu

January 14, 2002

ABSTRACT

The FFT is used widely in signal processing for efficient computation of the Fourier transform (FT) of finite-length signals over a set of uniformly-spaced frequency locations. However, in many applications, one requires nonuniform sampling in the frequency domain, *i.e.*, a *nonuniform FT*. Several papers have described fast approximations for the nonuniform FT based on interpolating an oversampled FFT. This paper presents an interpolation method for the nonuniform FT that is optimal in the min-max sense of minimizing the worst-case approximation error over all signals of unit norm. The proposed method easily generalizes to multidimensional signals. Numerical results show that the min-max approach provides substantially lower approximation errors than conventional interpolation methods.

Keywords: *Nonuniform FFT, discrete Fourier transform, min-max interpolation, tomography, magnetic resonance imaging*

I. INTRODUCTION

The fast Fourier transform (FFT) is used ubiquitously in signal processing applications where uniformly-spaced samples in the frequency domain are needed. The FFT requires only $O(N \log N)$ operations, whereas direct evaluation of the discrete Fourier transform requires $O(N^2)$ operations. However, a variety of applications require nonuniform sampling in the frequency domain, as has been recognized for at least 30 years [1]. Examples include radar imaging [2–6], computing oriented wavelets via the Radon transform [7], computational electromagnetics [8–12], and FIR filter design, *e.g.*, [13–15]. Such problems require a *nonuniform Fourier transform* [16], yet one would like to retain the computational advantages of fast algorithms like the FFT, rather than resorting to brute-force evaluation of the nonuniform FT.

Our work on this problem was motivated by iterative magnetic resonance image (MRI) reconstruction [17–19], and by iterative tomographic image reconstruction methods where reprojection is based on the Fourier slice theorem [20–25]. We explore these applications in more detail elsewhere and focus here on the broadly-applicable general principles.

In the signal processing literature, many papers have discussed frequency warping approaches for filter design [1, 14, 15, 26] and compression [27, 28]. Warping methods apply only to special patterns of nonuniform frequency locations, so are insufficiently general for most applications.

Recently several papers in the scientific computing literature have described methods for approximating the 1D nonuniform FT by interpolating an oversampled FFT, beginning with [29] and including [8, 10, 30–37]. Such methods are often called the nonuniform FFT, or NUFFT. Most of these algorithms have been presented only for 1D signals, and many involve seemingly arbitrary choices for interpolation functions. In contrast, this paper presents a min-max approach to the interpolation problem that is derived from first principles. We derive the fixed-width interpolator that minimizes the worst-case approximation error over all signals of unit norm. (Like all NUFFT methods, the user can tradeoff computation time and accuracy.) This method generalizes naturally to multidimensional signals such as the imaging problems that motivated this work. This work was inspired by the paper of Nguyen and Liu [36]. We compare our approach to theirs in detail in Section IV-C.

*This work was supported in part by NIH grant CA-60711, NSF grant BES-9982349, the UM Center for Biomedical Engineering Research, and the Whitaker Foundation.

This work is in the spirit of min-max approaches for other signal processing problems, such as bandlimited signal interpolation [38–44] and filter design [45, 46].

Section II derives the min-max NUFFT method. Section III describes extensions including multidimensional signals. Section IV analyzes the approximation error of the min-max method. Section V compares the min-max method to conventional methods. Section VI gives a practical 2D NUFFT example.

II. Theory: 1D Case

For simplicity, we first describe our min-max approach in the 1D case. The basic idea is to first compute an oversampled FFT of the given signal, and then interpolate optimally onto the desired nonuniform frequency locations using small local neighborhoods in the frequency domain.

A. Problem statement

We are given equally-spaced signal samples x_n, for $n = 0, \dots, N-1$, with corresponding FT

$$X(\omega) = \sum_{n=0}^{N-1} x_n e^{-\imath\omega n}. \quad (1)$$

We wish to compute the FT at a collection of (nonuniformly spaced) frequency locations $\{\omega_m\}$:

$$X_m \triangleq X(\omega_m) = \sum_{n=0}^{N-1} x_n e^{-\imath\omega_m n}, \quad m = 1, \dots, M. \quad (2)$$

The symbol "$\triangleq$" denotes "defined to be." The ω_m's can be arbitrary real numbers. This form has been called the nonuniform discrete Fourier transform (NDFT) [47, p. 190]. Directly evaluating (2) would require $O(MN)$ operations, which would be undesirably slow. Fast computation of (2) is called the NUFFT. This is "Problem 2" in the nomenclature of [29, 36].

The first step of most NUFFT algorithms is to choose a convenient $K \geq N$ and compute a weighted K-point FFT of $\{x_n\}$:

$$Y_k = \sum_{n=0}^{N-1} s_n x_n e^{-\imath\gamma kn}, \quad k = 0, \dots, K-1, \quad (3)$$

where $\gamma \triangleq 2\pi/K$ is the fundamental frequency of the K-point DFT. The nonzero s_n's are algorithm design variables that have been called "scaling factors" [36]. We call $\boldsymbol{s} = (s_1, \dots, s_N)$ the *scaling vector*. The purpose of $\boldsymbol{s}$ is to partially pre-compensate for imperfections in the subsequent frequency-domain interpolation. This first step requires $O(K \log K)$ operations, or essentially $O(N \log N)$ since one chooses K proportional to N (*cf.* (40)).

The second step of most NUFFT methods is to approximate each X_m by interpolating the Y_k's using some of the neighbors of ω_m in the DFT frequency set $\Omega_K \triangleq \{\gamma k : k = 0, \dots, K-1\}$. Linear interpolators have the following general form:

$$\hat{X}(\omega_m) = \sum_{k=0}^{K-1} Y_k v_{mk}^\star = \langle \boldsymbol{Y}, \boldsymbol{v}_m \rangle, \quad m = 1, \dots, M, \quad (4)$$

where the v_{mk}'s denote interpolation coefficients, "$\star$" denotes complex conjugate, and $\boldsymbol{v}_m \triangleq (v_{m1}, \dots, v_{mK})$. The design problem is choosing the scaling vector $\boldsymbol{s}$ and the interpolators $\{\boldsymbol{v}_m\}$.

Given the Y_k's, an ideal linear "interpolator" could first recover $\boldsymbol{x} = (x_0, \dots, x_{N-1})$ by computing the inverse FFT from (3) and then compute explicitly the desired FT values $X(\omega_m)$ using (2). Specifically, for $\boldsymbol{s} = \mathbf{1}$:

$$\begin{aligned} X(\omega) &= \sum_{n=0}^{N-1} x_n e^{-\imath\omega n} = \sum_{n=0}^{N-1} \left[\frac{1}{K} \sum_{k=0}^{K-1} Y_k e^{\imath\gamma kn} \right] e^{-\imath\omega n} \\ &= \sum_{k=0}^{K-1} Y_k I(\omega/\gamma - k), \end{aligned}$$

where the ideal interpolator kernel is:

$$I(\kappa) \triangleq e^{-\imath\gamma\kappa\eta_0} \frac{N}{K} \delta_N(\kappa), \quad (5)$$

where $\eta_0 \triangleq (N-1)/2$ and where $\delta_N(\cdot)$ denotes the following Dirichlet-like "periodic sinc" function:

$$\begin{aligned} \delta_N(\kappa) &\triangleq \frac{1}{N} \sum_{n=0}^{N-1} e^{\pm\imath\gamma\kappa(n-\eta_0)} \\ &= \begin{cases} \dfrac{\sin(\pi\kappa N/K)}{N \sin(\pi\kappa/K)}, & \kappa/K \notin \mathbb{Z} \\ 1, & \kappa/K \in \mathbb{Z}. \end{cases} \end{aligned} \quad (6)$$

Oversampling is of no benefit to this ideal interpolator. Applying this ideal interpolator would require $O(MK)$ operations and would use all MK of the v_{mk}'s in (4), so is impractical.

To contain computational requirements, most NUFFT methods constrain each $\boldsymbol{v}_m$ to have at most J nonzero elements corresponding to the J nearest neighbors to ω_m in the set Ω_K. With this practical restriction, the interpolation step requires $O(MJ)$ operations, where $J \ll K$.

Define the integer offset $k_m = k_0(\omega_m)$ as follows:

$$k_0(\omega) \triangleq \begin{cases} (\arg\min_{k \in \mathbb{Z}} |\omega - \gamma k|) - \dfrac{J+1}{2}, & J \text{ odd} \\ (\max\{k \in \mathbb{Z} : \omega \geq \gamma k\}) - \dfrac{J}{2}, & J \text{ even}. \end{cases} \quad (7)$$

This offset satisfies the following shift property:

$$k_0(\omega + l\gamma) = l + k_0(\omega), \qquad \forall l \in \mathbb{Z}. \tag{8}$$

Let $u_j(\omega_m)$, $j = 1, \ldots, J$, denote the J possibly nonzero entries of $\boldsymbol{v}_m$. Then the interpolation formula (4) becomes

$$\hat{X}(\omega_m) = \sum_{j=1}^{J} Y_{\{k_m+j\}_K} \, u_j^\star(\omega_m), \tag{9}$$

where $\{\cdot\}_K$ denotes the modulo-K operation (ensuring that $\hat{X}(\omega)$ is 2π periodic). To apply this formula, one must choose the JM interpolation coefficients $\{u_j(\omega_m)\}$, and compute the M indices $\{k_m\}$. One would like to choose each interpolation coefficient vector $\boldsymbol{u}(\omega_m) = (u_1(\omega_m), \ldots, u_J(\omega_m))$ such that $\hat{X}(\omega_m)$ is an accurate approximation to X_m, *and* such that $\boldsymbol{u}(\cdot)$ is relatively easy to compute. Dutt and Rokhlin used gaussian bell kernels for their interpolation method [29]. For even N and odd J only, Nguyen and Liu [36] considered interpolation of the form (9) with a choice for the u_j's that arises from least-squares approximations of complex exponentials by linear combinations of other complex exponentials. We propose an explicit min-max criterion for choosing the u_j's next, with uniform treatment of both even and odd J and N using (7).

B. Min-max interpolator

We adopt a *min-max criterion* for choosing the interpolation coefficients $\{u_j(\omega_m)\}$. For each desired frequency location ω_m, we determine the coefficient vector $\boldsymbol{u}(\omega_m) \in \mathbb{C}^J$ that minimizes the *worst case* approximation error between X_m and $\hat{X}(\omega_m)$ over all signals $\boldsymbol{x}$ having unit norm. In general this could be a shift-variant interpolation method since each desired frequency location ω_m can have its own set of J interpolation coefficients.

Both the scaling vector $\boldsymbol{s}$ and the interpolators $\{\boldsymbol{u}(\omega_m)\}$ are design variables, so ideally we would optimize simultaneously over both sets using the following criterion:

$$\min_{\boldsymbol{s} \in \mathbb{C}^N} \max_{\omega} \min_{\boldsymbol{u}(\omega) \in \mathbb{C}^J} \max_{\boldsymbol{x} \in \mathbb{C}^N \,:\, \|\boldsymbol{x}\| \le 1} |\hat{X}(\omega) - X(\omega)|. \tag{10}$$

As discussed in Section IV, the outer optimization requires numerical methods. Thus, we focus next on optimizing the interpolation coefficients $\boldsymbol{u}(\omega_m)$ for a fixed scaling vector $\boldsymbol{s}$, and address choice of $\boldsymbol{s}$ in Section IV-C.

Mathematically, our min-max criterion is the following:

$$\min_{\boldsymbol{u}(\omega_m) \in \mathbb{C}^J} \max_{\boldsymbol{x} \in \mathbb{C}^N \,:\, \|\boldsymbol{x}\| \le 1} |\hat{X}(\omega_m) - X(\omega_m)|. \tag{11}$$

Remarkably, this min-max problem has an analytical solution, as derived next.

From (9) and (2), we have the following expression for the error:

$$|\hat{X}(\omega_m) - X_m| = \left| \sum_{j=1}^{J} Y_{\{k_m+j\}_K} u_j^\star(\omega_m) - X(\omega_m) \right|. \tag{12}$$

Using (3) and (12), this error expression becomes

$$\begin{aligned} &\sum_{j=1}^{J} u_j^\star(\omega_m) \left[\sum_{n=0}^{N-1} s_n x_n e^{-\imath\gamma(k_m+j)n} \right] - \sum_{n=0}^{N-1} x_n e^{-\imath\omega_m n} \\ &= \sum_{n=0}^{N-1} x_n \left(s_n \left[\sum_{j=1}^{J} e^{-\imath\gamma(k_m+j)n} u_j^\star(\omega_m) \right] - e^{-\imath\omega_m n} \right) \\ &= \sqrt{N} \, \langle \boldsymbol{x}, \, \boldsymbol{g}(\omega_m) \rangle, \end{aligned} \tag{13}$$

where $\boldsymbol{g}(\cdot)$ is an N-vector with elements

$$g_n(\omega) \triangleq s_n^\star \left[\sum_{j=1}^{J} \frac{1}{\sqrt{N}} e^{\imath\gamma(k_0(\omega)+j)n} u_j(\omega) \right] - \frac{1}{\sqrt{N}} e^{\imath\omega n},$$

for $n = 0, \ldots, N-1$. In matrix-vector form:

$$\boldsymbol{g}(\omega) = \boldsymbol{D}(\omega) \boldsymbol{S}' \boldsymbol{C} \boldsymbol{\Lambda}(\omega) \boldsymbol{u}(\omega) - \boldsymbol{b}(\omega), \tag{14}$$

where $\boldsymbol{S} = \operatorname{diag}\{s_n\}$, "$'$" denotes Hermitian transpose, $\boldsymbol{D}(\omega)$ is a $N \times N$ diagonal matrix, $\boldsymbol{C}$ is a $N \times J$ matrix, $\boldsymbol{\Lambda}(\omega)$ is a $J \times J$ diagonal matrix, and $\boldsymbol{b}(\omega)$ is a N-vector with respective entries:

$$\begin{aligned} D_{nn}(\omega) &= e^{\imath\omega\eta_0} e^{\imath\gamma k_0(\omega)(n-\eta_0)} && (15) \\ c_{nj} &= e^{\imath\gamma j(n-\eta_0)}/\sqrt{N} && (16) \\ \Lambda_{jj}(\omega) &= e^{-\imath[\omega-\gamma(k_0(\omega)+j)]\eta_0} && (17) \\ b_n(\omega) &= e^{\imath\omega n}/\sqrt{N}. && (18) \end{aligned}$$

(We chose these definitions with considerable hindsight to simplify subsequent expressions.)

In this form, the min-max problem (11) becomes

$$\min_{\boldsymbol{u} \in \mathbb{C}^J} \max_{\boldsymbol{x} \in \mathbb{C}^N : \|\boldsymbol{x}\| \le 1} \sqrt{N} \, |\langle \boldsymbol{x}, \, \boldsymbol{g}(\omega) \rangle| \, . \tag{19}$$

By the Cauchy-Schwarz inequality, for a given frequency ω, the worst-case signal is $\boldsymbol{x} = \boldsymbol{g}^\star(\omega)/\|\boldsymbol{g}(\omega)\|$, *i.e.*,

$$\max_{\boldsymbol{x} \,:\, \|\boldsymbol{x}\| = 1} |\langle \boldsymbol{x}, \, \boldsymbol{g}(\omega) \rangle| \;\; = \;\; \|\boldsymbol{g}(\omega)\| \, .$$

Inserting this case into the min-max criterion (19) and applying (14) reduces the min-max problem to the following (cf [36, eqn. 10]):

$$\min_{\boldsymbol{u} \in \mathbb{C}^J} \sqrt{N} \left\| \boldsymbol{D}(\omega) \boldsymbol{S}' \boldsymbol{C} \boldsymbol{\Lambda}(\omega) \boldsymbol{u}(\omega) - \boldsymbol{b}(\omega) \right\| . \tag{20}$$

The minimizer of this ordinary least-squares problem for $\omega = \omega_m$ is $\boldsymbol{u}(\omega_m)$, where (since $\boldsymbol{D}$ and $\boldsymbol{\Lambda}$ are unitary):

$$\begin{aligned} \boldsymbol{u}(\omega) &= \boldsymbol{\Lambda}'(\omega)[\boldsymbol{C}'\boldsymbol{S}\boldsymbol{S}'\boldsymbol{C}]^{-1}\boldsymbol{C}'\boldsymbol{S}\boldsymbol{D}'(\omega)\boldsymbol{b}(\omega) \quad (21) \\ &= \boldsymbol{\Lambda}'(\omega)\boldsymbol{R}\boldsymbol{r}(\omega), \quad (22) \end{aligned}$$

where we define

$$\begin{aligned} \boldsymbol{R} &\triangleq [\boldsymbol{C}'\boldsymbol{S}\boldsymbol{S}'\boldsymbol{C}]^{-1} \quad (23) \\ \boldsymbol{r}(\omega) &\triangleq \boldsymbol{C}'\boldsymbol{S}\boldsymbol{D}'(\omega)\boldsymbol{b}(\omega). \quad (24) \end{aligned}$$

Fortuitously, the inverse of the $J \times J$ matrix $\boldsymbol{C}'\boldsymbol{S}\boldsymbol{S}'\boldsymbol{C}$ is independent of frequency sample location so it can be precomputed.

To facilitate computing $\boldsymbol{C}'\boldsymbol{S}\boldsymbol{S}'\boldsymbol{C}$, we expand the s_n's in terms of a (usually truncated) Fourier series:

$$s_n = \sum_{t=-L}^{L} \alpha_t e^{\imath\gamma\beta t(n-\eta_0)}, \qquad n = 0, \ldots, N-1. \quad (25)$$

The natural fundamental frequency corresponds to $\beta = K/N$, but we consider the general form above since orthogonality is not required here and β can be a design parameter. We assume that the α's are Hermitian symmetric, *i.e.*, $\alpha_{-t} = \alpha_t^\star$. We represent the coefficients by the vector $\boldsymbol{\alpha} \triangleq (\alpha_0, \alpha_1, \ldots, \alpha_L)$. As one special case of (25), the "cosine" scaling factors considered in [36] correspond to $\beta = 1/2$ and $\boldsymbol{\alpha} = (0, 1/2)$. For $\beta \leq K/N$, there is no loss of generality in using the expansion (25). This expansion significantly generalizes the choices of scaling factors considered by Nguyen and Liu in [36], and leads to significantly reduced errors as shown in Section IV-C. Nguyen and Liu referred to matrices of the form $\boldsymbol{C}'\boldsymbol{C}$ as $(K/N, N, J-1)$ *regular Fourier matrices* [36].

Combining (25) and (23) with (16) yields

$$\begin{aligned} \left[\boldsymbol{C}'\boldsymbol{S}\boldsymbol{S}'\boldsymbol{C}\right]_{l,j} &= \sum_{n=0}^{N-1} c_{nl}^\star s_n s_n^\star c_{nj} \\ &= \frac{1}{N}\sum_{n=0}^{N-1}\sum_{t=-L}^{L}\sum_{s=-L}^{L} \alpha_t \alpha_s^\star e^{\imath\gamma(j-l+\beta(t-s))(n-\eta_0)} \\ &= \sum_{t=-L}^{L}\sum_{s=-L}^{L} \alpha_t \alpha_s^\star \delta_N(j-l+\beta(t-s)), \quad (26) \end{aligned}$$

for $l, j = 1, \ldots, J$, where $\delta_N(\cdot)$ was defined in (6).

The following properties of $\boldsymbol{R}$ are useful. From (23), $\boldsymbol{R}$ is Hermitian, and from (26), $\boldsymbol{R}$ is Toeplitz. In the usual case where the α's are real, $\boldsymbol{R}$ is a real matrix. If $K = N$ and $\boldsymbol{S} = \boldsymbol{I}$, then $\boldsymbol{R}^{-1} = \boldsymbol{C}'\boldsymbol{C} = \boldsymbol{I}$.

Conveniently, in this min-max framework the matrix-vector product defining $\boldsymbol{r}(\omega)$ in (24) also simplifies:

$$\begin{aligned} r_j(\omega) &= \sum_{n=0}^{N-1} c_{nj}^\star s_n D_{nn}^\star(\omega) b_n(\omega) \\ &= \sum_{t=-L}^{L} \alpha_t \frac{1}{N}\sum_{n=0}^{N-1} e^{\imath\gamma[\omega/\gamma - k_0(\omega) - j + \beta t](n-\eta_0)} \\ &= \sum_{t=-L}^{L} \alpha_t \delta_N(\omega/\gamma - k_0(\omega) - j + \beta t), \quad (27) \end{aligned}$$

for $j = 1, \ldots, J$. This is a Dirichlet-like function of the distances between the desired frequency location and the nearest points in the set Ω_K. Due to the shift property (8),

$$r_j(\omega + \gamma l) = r_j(\omega), \qquad \forall l \in \mathbb{Z}, \quad (28)$$

so the min-max interpolator is γ-periodic and "shift invariant" in the sense appropriate for periodic interpolators.

In the usual case where the α's are real, the vector $\boldsymbol{r}(\omega)$ is real. So the only complex component of the min-max interpolator $\boldsymbol{u}(\omega)$ in (22) is the complex phases in $\boldsymbol{\Lambda}(\omega)$. By (17), these phases coincide with the linear phase of the ideal interpolator (5).

To summarize, we compute the min-max interpolation coefficients in (22) for each ω_m using the analytical results (26) and (27). Since $\boldsymbol{C}'\boldsymbol{S}\boldsymbol{S}'\boldsymbol{C}$ is only $J \times J$, where J is usually less than 10, we always precompute $\boldsymbol{R}$ in (23) prior to all other calculations.

As described next, there are two natural methods for using the above formulas, depending on one's tradeoff between memory and computation.

C. Precomputed mode

In problems like iterative image reconstruction, one must compute the NUFFT (2) several times for the same set of frequencies $\{\omega_m\}$, but for different signals $\boldsymbol{x}$. In such cases, it is preferable to *precompute* and store all JM of the interpolation coefficients $u_j(\omega_m)$, if sufficient memory is available, and then apply (9) directly to compute the NUFFT as needed.

Precomputing the $u_j(\omega_m)$'s requires only $O((2L+1)^2 J^2 M)$ operations. A key property of (26) and (27) is that they eliminate the summations over n, thereby making the precomputing practical. (In contrast, using (21) directly would require $O(NJ^2M)$ operations.)

After the $u_j(\omega_m)$'s are precomputed, each subsequent NUFFT interpolation step (9) requires only $O(JM)$ operations. Excluding the precomputation, the overall operation count per NUFFT is $O(K \log K) + O(JM)$. An

accuracy-computation time tradeoff is available through the choices for the oversampling factor K/N and the neighborhood size J. Typically we use $K \approx 2N$, $L \leq 5$, $J \leq 10$, and $M \approx N$, so the overall computational requirements are akin to an FFT but with a larger constant. The larger constant is an unavoidable consequence of needing accurate nonuniform frequency samples!

D. Reduced memory mode

In unusual cases where storing all JM coefficients is infeasible, one can evaluate each $\boldsymbol{u}(\omega_m)$ as needed using (22), (27) and the precomputed $\boldsymbol{R}$ in (23). In this mode the operation count for the NUFFT interpolation step increases to $O((2L+1)^2 J^2 M)$, but the storage requirements decrease to $O(M)$.

One could decrease the operation count to $O((2L+1)^2 JM)$ by tabulating $\boldsymbol{R}\boldsymbol{r}(\omega)$ (cf Fig. 1) and using table lookup/interpolation, presumably with a slight decrease in accuracy.

E. "Large" N interpolator

The dependence of the interpolator on the signal-length N can be inconvenient since it would seem to necessitate designing a new interpolator for each signal length of interest. To simplify the design, we consider hereafter cases where N is "large." These are of course the cases where fast algorithms are particularly desirable.

Defining $\mu \triangleq K/N$, from (6) one easily sees that

$$\lim_{N\to\infty} \delta_N(t) = \operatorname{sinc}(t/\mu)\,,$$

where $\operatorname{sinc}(t) \triangleq \sin(\pi t)/(\pi t)$. So for large N, $\boldsymbol{R} \approx \tilde{\boldsymbol{R}}$ where

$$[\tilde{\boldsymbol{R}}^{-1}]_{l,j} \triangleq \sum_{t=-L}^{L} \sum_{s=-L}^{L} \alpha_t \alpha_s^\star \operatorname{sinc}\left(\frac{j-l+\beta(t-s)}{\mu}\right). \tag{29}$$

Similarly, $\boldsymbol{r}(\omega) \approx \tilde{\boldsymbol{r}}(\omega)$ where

$$\tilde{r}_j(\omega) \triangleq \sum_{t=-L}^{L} \alpha_t \operatorname{sinc}\left(\frac{\omega/\gamma - k_0(\omega) - j + \beta t}{\mu}\right). \tag{30}$$

Combining these with (22), the interpolator we consider hereafter is

$$\tilde{\boldsymbol{u}}(\omega) \triangleq \boldsymbol{\Lambda}'(\omega)\tilde{\boldsymbol{R}}\tilde{\boldsymbol{r}}(\omega). \tag{31}$$

From (26) and (27), the maximum argument of δ_N is $2(J+\beta L)$ and typically is less than 30. From (6), as long as this argument is much smaller than K, the sinc approximation will be very accurate. For example, even for N as small as 32, the sinc and δ_N differ by less than 1% for arguments less than 30. Thus, focusing on the sinc-based interpolator (31) is very reasonable.

F. Effective interpolation kernel

Most interpolation methods *start* with a specific functional form for the kernel, such as a gaussian or B-spline. In contrast, we have started with only the min-max criterion and no other constraints except using the J nearest neighbors. Consider the case of uniform scaling factors ($s_n = 1$, so $L = 0$ and $\alpha_0 = 1$). To visualize the min-max interpolator (31), we can vary ω/γ over the interval $[-J/2, J/2]$ and evaluate $\tilde{\boldsymbol{R}}\tilde{\boldsymbol{r}}(\omega)$ using (31), yielding real functions such as those shown in Fig. 1 and Fig. 2 for the cases $J = 6$ and $J = 7$ respectively, using $\mu = 2$. The figures also show (part of) a sinc interpolator (*cf.* (5)) for comparison. For even J, the min-max interpolator is not differentiable at integer arguments. For odd J, the min-max interpolator has discontinuities at the midpoints between DFT samples since the neighborhood changes at that point (cf (7)). These properties depart significantly from classical interpolators but they need not be surprising since regularity was not part of the min-max formulation.

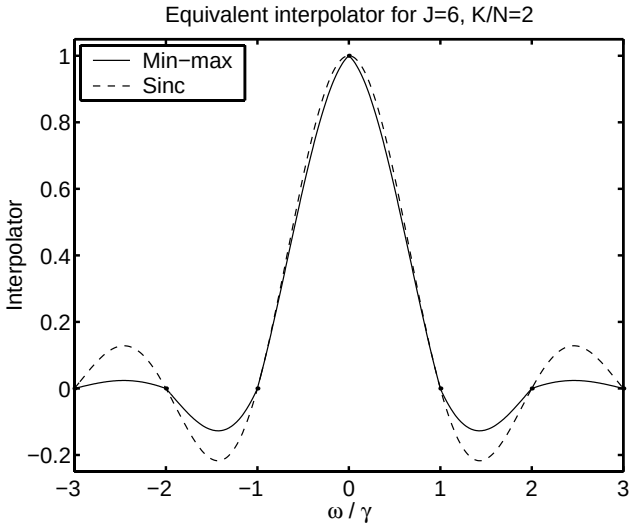

Figure 1: Illustration of the min-max interpolator corresponding to (31) for $J = 6$, $N = 128$, $K/N = 2$, and uniform scaling factors.

Although we have not attempted to prove this analytically, we have found empirically that the interpolation coefficient vector $\tilde{\boldsymbol{u}}(\omega)$ seems to satisfy the property that $\sum_{j=1}^{J} \tilde{u}_j$ is close to unity (particularly as J increases). This is an expected property of interpolators, but our formulation did not enforce this constraint *a priori*. Interestingly, it seems to have arisen naturally from the min-max framework. With uniform scaling factors ($\boldsymbol{s} = \mathbf{1}$), the kernel also satisfies the property that it is unity at $\omega = 0$ and zero at each other γk. This expected property follows directly from the min-max formulation.

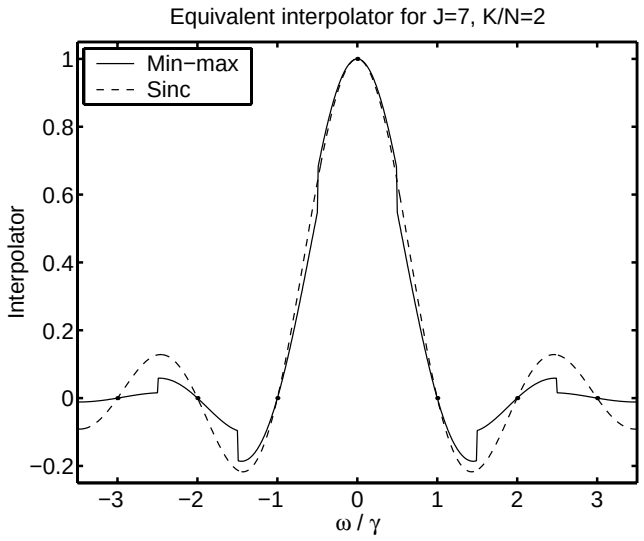

Figure 2: Illustration of the min-max interpolator corresponding to (31) for $J = 7$, $N = 128$, $K/N = 2$, and uniform scaling factors.

III. EXTENSIONS

This section describes some extensions to the min-max NUFFT developed above.

A. Multidimensional NUFFT

The extension of the min-max method to two dimensions and higher is conceptually very straightforward. In 2D, we oversample the 2D FFT in both directions, and precompute and store the min-max interpolator for each desired frequency location using the nearest $J \times J$ sample locations. The storage requirements are $O(J^2 M)$ if the interpolation coefficients are precomputed. Precomputing the interpolator involves simple Kronecker products of the the 1D interpolators. Specifically, for a 2D image, if we use a $J_1 \times J_2$ neighborhood, with oversampling factors $\mu_1 = K_1/N_1$ and $\mu_2 = K_1/N_2$ in the two dimensions respectively, then the matrix $\tilde{\boldsymbol{R}}$ in (29) becomes a Kronecker product (denoted "$\otimes$"):

$$\tilde{\boldsymbol{R}}_{\text{2D}} = \tilde{\boldsymbol{R}}_{\text{1D}}(J_2, \mu_2) \otimes \tilde{\boldsymbol{R}}_{\text{1D}}(J_1, \mu_1), \qquad (32)$$

as does the vector $\tilde{\boldsymbol{r}}$ in (30):

$$\tilde{\boldsymbol{r}}_{\text{2D}} = \tilde{\boldsymbol{r}}_{\text{1D}}(J_2, \mu_2) \otimes \tilde{\boldsymbol{r}}_{\text{1D}}(J_1, \mu_1). \qquad (33)$$

Subroutines for Matlab are freely available online[1].

[1] `http://www.eecs.umich.edu/~fessler`

B. Shifted signals

Applications often need a "shifted" version of (1):

$$\sum_{n=0}^{N-1} x_n e^{-\imath(n-\tau)\omega} = e^{\imath\omega\tau} \sum_{n=0}^{N-1} x_n e^{-\imath n\omega}. \qquad (34)$$

Incorporating the $e^{\imath\omega\tau}$ phase term into the precomputed interpolation coefficients $\tilde{u}_j(\omega)$ induces this shift efficiently.

C. Adaptive neighborhoods

In the approach described above, the same number J of neighboring DFT samples is used for each frequency location ω_m of interest. This simplifies implementation, but is suboptimal in terms of both memory and computation. Some of the ω_m's are likely to fall very close to the DFT samples in the set Ω_K, and for those locations a smaller value of J may suffice (depending on $\boldsymbol{\alpha}$, see Fig. 6). An interesting extension would be to specify a maximum error tolerance, and then for each ω_m use the smallest J_m that guarantees that error tolerance, assuming that one has made a reasonable choice for K/N.

In higher dimensions, one could consider using non-square neighborhoods, *e.g.*, approximate balls.

D. Adjoint operator

Since the NUFFT method described above is a linear operator, it corresponds implicitly to some $M \times N$ matrix, say $\boldsymbol{F}$. For iterative image reconstruction algorithms, one must also apply the adjoint of this operator, *i.e.*, effectively perform matrix-vector multiplication of the form $\boldsymbol{F}'\tilde{\boldsymbol{y}}$ for some vector $\tilde{\boldsymbol{y}} \in \mathbb{C}^M$. Since $\boldsymbol{F}$ itself is too large to store in the imaging problems of interest, and since direct matrix-vector multiplication would be computationally inefficient, we must evaluate $\boldsymbol{F}'\tilde{\boldsymbol{y}}$ by "reversing" (not inverting!) the algorithm steps described in Section II.

The adjoint corresponding to (4) is

$$\tilde{X}_k = \sum_{m=1}^{M} \tilde{y}_m v_{mk}$$

and likewise for (3) the adjoint is

$$\tilde{x}_n = \sum_{k=0}^{K-1} \tilde{X}_k e^{\imath 2\pi k n/K}, \qquad n = 0, \ldots, N-1,$$

which is the K-point inverse DFT of $\tilde{X}_k$ scaled by K, followed by extracting the appropriate N signal values.

Thus, to evaluate $\boldsymbol{F}'\tilde{\boldsymbol{y}}$ we first apply the Hermitian transpose of the interpolation matrix $\{v_{mk}\}$. When the (sparse) interpolation matrix is precomputed and stored,

the interpolation step requires $O(JM)$ operations. We then apply the K-point inverse FFT, scale by K, and keep the N relevant values. This step is $O(K \log K)$.

E. Nonuniform inverse FFT

By duality, *i.e.*, by changing the sign in the exponent of (1), one could apply the min-max approach to cases where one has uniformly-spaced frequency samples and wants to evaluate the inverse FT on a nonuniform set of spatial locations. Given $X_k,\ k = 0, \dots, K-1$ corresponding to frequencies $\{\gamma k\}$, we can compute

$$x(t_n) = \sum_{k=0}^{K-1} X_k e^{\imath \gamma k t_n}, \qquad n = 1, \dots, N \tag{35}$$

using the same type of approach with min-max interpolation. This is again "Problem 2" in the terminology of [29, 36].

F. Reduced FFT

Since (3) corresponds to an oversampled FFT, when K/N is an integer, one can evaluate (3) by combining K/N invocations of an N-point FFT routine, reducing the operation count for (3) from $O(K \log K)$ to $O(K \log N)$. As a concrete example, if $K/N = 2$ then

$$Y_k = \begin{cases} \sum_{n=0}^{N-1} (s_n x_n)\, e^{-\imath \frac{2\pi}{N} n \left(\frac{k}{2}\right)}, & k \text{ even} \\ \sum_{n=0}^{N-1} (s_n x_n e^{-\imath \gamma n})\, e^{-\imath \frac{2\pi}{N} n \left(\frac{k-1}{2}\right)}, & k \text{ odd.} \end{cases} \tag{36}$$

One can evaluate each of these two expressions using an N-point FFT. In general, one needs K/N FFT's, where the modulation needed for the m'th FFT is $e^{-\imath \gamma m n}$, $m = 0, \dots, K/N - 1$.

IV. Error Analysis

Combining (22) and (20) and simplifying yields the following expression for the worst-case error at frequency ω:

$$\begin{aligned} \frac{\mathcal{E}_{\text{exact}}(\omega)}{\sqrt{N}} &= \left\| \boldsymbol{D}(\omega) \boldsymbol{S}' \boldsymbol{C} \boldsymbol{\Lambda}(\omega) [\boldsymbol{\Lambda}'(\omega) \boldsymbol{R} \boldsymbol{r}(\omega)] - \boldsymbol{b}(\omega) \right\| \\ &= \sqrt{1 - \boldsymbol{r}'(\omega) \boldsymbol{R} \boldsymbol{r}(\omega)}. \end{aligned} \tag{37}$$

The error *bounds* given in NUFFT papers are often described as pessimistic. In contrast, the *exact* worst-case error given by (37) is achievable. Of course, the unit-norm signal that achieves this worst-case error may not be representative of many problems of interest, so the "typical" performance may appear better than (37).

To simplify analysis, we use the "large N" approximations (29) and (30) and normalize out the $\sqrt{N}$ dependence. Specifically, the following approximation is *very* accurate:

$$\frac{\mathcal{E}_{\text{exact}}(\omega)}{\sqrt{N}} \approx \mathcal{E}(\omega) \triangleq \sqrt{1 - \tilde{\boldsymbol{r}}'(\omega) \tilde{\boldsymbol{R}} \tilde{\boldsymbol{r}}(\omega)}. \tag{38}$$

We focus on this normalized error $\mathcal{E}(\cdot)$ hereafter.

Due to the shift-invariance property (28), the error $\mathcal{E}(\omega)$ is periodic with period γ. One can also show that $\mathcal{E}(\omega)$ has a local extremum when ω is midway between the nearest two DFT samples $\{\gamma k\}$. The *maximum* error

$$\mathcal{E}_{\max} \triangleq \max_{\omega} \mathcal{E}(\omega) \tag{39}$$

usually occurs either at the midpoint between DFT samples, or at the DFT samples themselves. (See Fig. 6 below for examples.) Unfortunately this does not always hold, so we apply numerical methods to evaluate (39). We begin with the simplest case: uniform scaling factors ($\boldsymbol{s} = \boldsymbol{1}$).

A. Uniform scaling factors

Fig. 3 plots $\mathcal{E}_{\max}$ for a variety of choices of neighborhood size J and oversampling factor K/N for uniform scaling factors ($\boldsymbol{s} = \boldsymbol{1}$). As expected, increasing J or K/N reduces the error, with diminishing returns as K/N increases. By examining many such curves, we fit the following empirical formula for the error:

$$\mathcal{E}_{\max} \approx 0.75 \exp(-J[0.29 + 1.03 \log(K/N)])\,. \tag{40}$$

This might serve as a guide for choosing J and K/N.

We would have liked to show empirical results with lower worst-case errors as J increases further. Unfortunately, as J and K/N increase, the matrix $\boldsymbol{C}'\boldsymbol{C}$ becomes very poorly conditioned. Using a truncated SVD to compute the pseudo-inverse of $\boldsymbol{C}'\boldsymbol{C}$ did not seem to help reduce $\mathcal{E}_{\max}$. Although we expect that small values for J and K/N will likely be adequate in practice (*cf.* Section VI or [25]), it would still be more satisfying theoretically to have an approach that can provide the user the full tradeoff between computation and accuracy. Exploiting the Toeplitz structure of $\boldsymbol{C}'\boldsymbol{C}$ or using a QR decomposition of $\boldsymbol{C}$ are two possible avenues for further exploration, *e.g.*, [48].

B. Multidimensional case

Using (32) and (33), the 2D error has the form

$$\begin{aligned} \mathcal{E}_{\text{2D}} &= \sqrt{1 - \tilde{\boldsymbol{r}}'_{\text{2D}} \tilde{\boldsymbol{R}}_{\text{2D}} \tilde{\boldsymbol{r}}_{\text{2D}}} \\ &= \sqrt{1 - \tilde{\boldsymbol{r}}'_2 \tilde{\boldsymbol{R}}_2 \tilde{\boldsymbol{r}}_2 \tilde{\boldsymbol{r}}'_1 \tilde{\boldsymbol{R}}_1 \boldsymbol{r}_1} \\ &= \sqrt{1 - (1 - \mathcal{E}_2^2)(1 - \mathcal{E}_1^2)} \le \sqrt{\mathcal{E}_1^2 + \mathcal{E}_2^2}, \end{aligned}$$

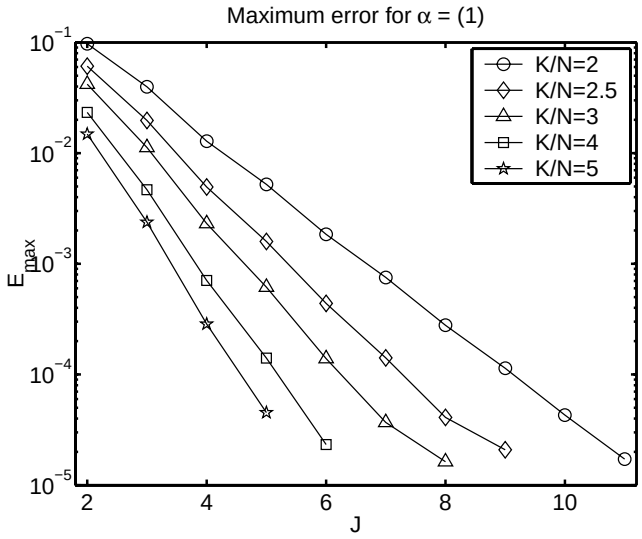

Figure 3: Maximum error $\mathcal{E}_{\max}$ of min-max interpolator with $\boldsymbol{s} = \mathbf{1}$ for various neighborhood sizes J and oversampling factors K/N.

where $\mathcal{E}_1$ and $\mathcal{E}_2$ denote the 1D errors in (38). This gives an upper bound on the potential accuracy "penalty" in 2D relative to 1D. It also suggests that tensor products of good 1D min-max interpolators should work well in higher dimensions, so we can focus the efforts in optimizing $\boldsymbol{\alpha}$ and β on the 1D case.

C. Choice of scaling factors

Both $\tilde{\boldsymbol{r}}$ and $\tilde{\boldsymbol{R}}$ in the error expression (38) depend on the choice of scaling vector $\boldsymbol{s}$ as seen in (29) and (30). Returning to (10), ideally we would like to choose the scaling factors using the following criterion:

$$\min_{\boldsymbol{s} \in \mathbb{C}^N} \max_{\omega} \mathcal{E}(\omega).$$

Unfortunately, an analytical solution to this optimization problem has proven elusive. For the ideal interpolator (5), uniform scaling factors are optimal. (In fact the s_n's are irrelevant.) Intuition suggests that for good interpolators, the s_n's should be fairly uniform, so a low-order expansion in (25) should be adequate. (This is consistent with the smooth choices that have been used in the literature, *e.g.*, [29, 33, 36].) Using the series expansion (25) and denoting the dependence of $\mathcal{E}_{\max}$ on the Fourier series coefficients $\boldsymbol{\alpha}$ and on β, for a given L, we would like to solve

$$\min_{\boldsymbol{\alpha}, \beta} \mathcal{E}_{\max}(\boldsymbol{\alpha}, \beta).$$

Lacking an analytical solution, we have explored this minimization numerically using brute-force global search for small values of L, by searching jointly over β and $\boldsymbol{\alpha} = (1, \alpha_1, \ldots, \alpha_L)$. For example, for the case $L = 1$, $J = 6$, and $K/N = 2$, we searched jointly over β and α_1 in $\boldsymbol{\alpha} = (1, \alpha_1)$. The best β was 0.19, and Fig. 4 plots $\mathcal{E}_{\max}$ versus α_1 for that β. The minimizer is $\alpha_1 = -0.46$, rather than 0, so clearly uniform scaling factors are suboptimal. Because the minimum in Fig. 4 is sharp, this minimization required a fine search, so that extra effort is warranted only when one needs many NUFFTs for the same J and K/N. We also investigated complex values for α_1 and found that the minimizer was always a real-valued α_1.

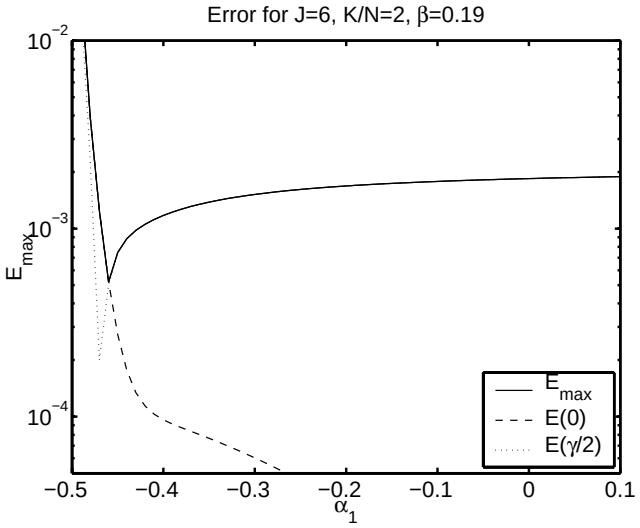

Figure 4: Maximum error $\mathcal{E}_{\max}$ as a function of α_1 for $L = 1$ and $\alpha_0 = 1$. Since the minimum is not at $\alpha_1 = 0$, uniform scaling factors are suboptimal.

For $L = 2$, $J = 6$, and $K/N = 2$, we numerically minimized $\mathcal{E}_{\max}((1, \alpha_1, \alpha_2), \beta)$ over $\alpha_1, \alpha_2, \beta$. The minimizer was $\boldsymbol{\alpha} = (1, -0.57, 0.14)$ and $\beta = 0.43$. Fig. 5 shows $\mathcal{E}_{\max}((1, -0.57, \alpha_2), 0.14)$ versus α_2. Again, in the neighborhood of the minimum, $\mathcal{E}_{\max}$ can be fairly sensitive to $\boldsymbol{\alpha}$.

Table 1 summarizes the optimized $\boldsymbol{\alpha}$'s and β's for these and other cases.

Fig. 6 compares the accuracy of these optimized min-max interpolators to uniform scaling factors and to the cosine scaling factors emphasized in [36]. As acknowledged by Nguyen and Liu, 'the cosine scaling factors ... are by no mean[s] the "best" ones,' a point that Fig. 6 confirms. We found in many such experiments (for a variety of J's and K/N's) that uniform scaling factors yielded consistently lower errors than cosine scaling factors[2].

[2]There is an error in the second to last equation on p. 292 of [36] regarding uniform scaling factors.

L	J	β	$\boldsymbol{\alpha}$		$\mathcal{E}_{\max}$
1	6	0.19	(1 -0.46)		$5.2{\cdot}10^{-4}$
2	2	0.34	(1 -0.2	-0.04)	$4.8{\cdot}10^{-2}$
2	3	0.48	(1 -0.485	0.09)	$3.7{\cdot}10^{-3}$
2	4	0.56	(1 -0.47	0.085)	$1.1{\cdot}10^{-3}$
2	5	0.495	(1 -0.4825	0.12)	$4.8{\cdot}10^{-4}$
2	6	0.43	(1 -0.57	0.14)	$1.4{\cdot}10^{-4}$
2	7	0.65	(1 -0.465	0.07)	$5.1{\cdot}10^{-5}$
2	8	0.47	(1 -0.54	0.16)	$2.4{\cdot}10^{-5}$
2	9	0.325	(1 -0.625	0.14)	$5.7{\cdot}10^{-6}$
2	10	0.43	(1 -0.57	0.185)	$6.0{\cdot}10^{-7}$
3	4	0.6339	(1 -0.5319	0.1522 -0.0199)	$3.0{\cdot}10^{-4}$
3	6	0.2254	(1 -0.6903	0.2138 -0.0191)	$1.1{\cdot}10^{-4}$

Table 1: Coefficients in (25) of numerically optimized scaling factors for $K/N = 2$.

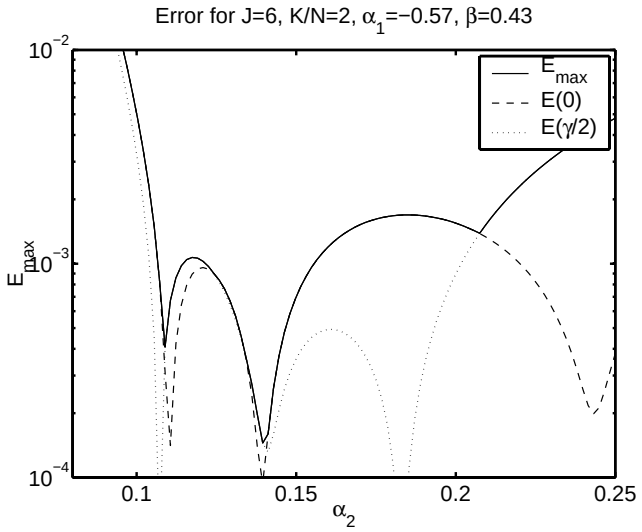

Figure 5: Maximum error $\mathcal{E}_{\max}$ as a function of α_2 for $L = 2$ and $\alpha_0 = 1$.

The shapes of the curves in Fig. 6 are noteworthy. Uniform scaling factors yield zero error at the DFT samples, and peak error at the midpoints. In contrast, optimized scaling factors tend to balance the error at the DFT samples and at the midpoints. We expect that the latter property will be preferable in practice, since the desired frequency locations often have essentially random locations so there is little reason to "favor" the DFT sample locations.

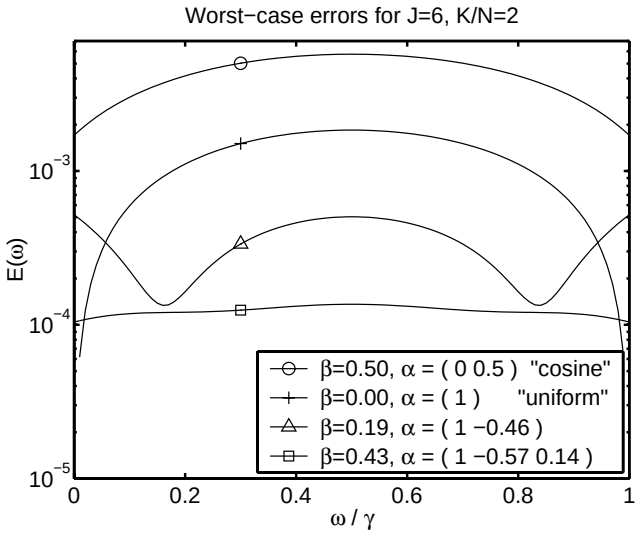

Figure 6: Worst-case error $\mathcal{E}(\omega)$ for various scaling vectors $\boldsymbol{\alpha}$. The "cosine" scaling factors are inferior to uniform scaling factors. Optimizing $\boldsymbol{\alpha}$ significantly reduces error.

The interpolators shown in Fig. 1 and Fig. 2 were for uniform scaling factors. Fig. 7 shows the effective interpolators for the optimized $\boldsymbol{\alpha}$'s described above for $L = 1, 2$.

The optimized interpolators ($L = 1, 2$) have lower sidelobes than the uniform case ($L = 0$) and are not unity at zero nor zero at other DFT samples.

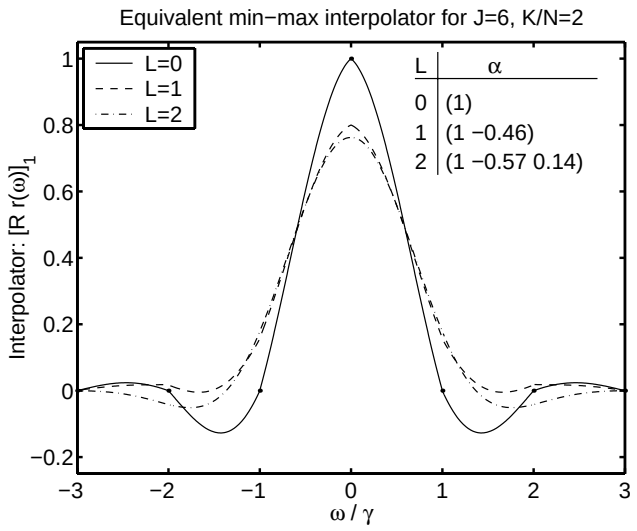

Figure 7: Effective min-max interpolator for $J = 6$ and $K/N = 2$ for optimized $\boldsymbol{\alpha}$ and β.

Our emphasis here has been on worst-case error, and the error values given in Fig. 3 differ from those reported in [36]. This "discrepancy" has two explanations. Firstly, we consider "Problem 2" in (2), whereas the figures in [36] are for "Problem 1" which is a transform from unequally-spaced data into uniformly spaced values. These problems may be fundamentally different in their error properties. Secondly, the errors reported in [36] and related papers are for a particular experiment involving pseudo-random data and sample locations, and it is likely that the characteristics of this data differ considerably from the "worst-case" signal $\boldsymbol{x}$ considered in the analysis here. Apparently one must be cautious about generalizing accuracies reported in particular experiments.

V. Conventional interpolators

The preceding error analysis was for the min-max interpolation method. One naturally wonders how the min-max method compares to conventional shift-invariant interpolators. This section analyzes the worst-case error of conventional interpolators. (See [49] for analysis of "gridding" methods using a Kaiser-Bessel kernel.)

A. Error analysis

Let $F(\cdot)$ denote a finite-support interpolation kernel satisfying $F(\kappa) = 0$ for $|\kappa| > J/2$. Conventional interpolation has the following form

$$\hat{X}(\omega) \;=\; \sum_{k=0}^{K-1} Y_k \tilde{F}(\omega/\gamma - k), \qquad (41)$$

where Y_k was defined in (3), and $\tilde{F}$ denotes the periodic version of F:

$$\tilde{F}(\kappa) \triangleq \sum_{l=-\infty}^{\infty} F(\kappa - lK),$$

ensuring that $\hat{X}(\omega)$ is 2π-periodic. Mimicking (13), the error is

$$|\hat{X}(\omega) - X(\omega)| = \sqrt{N}\, \left|\langle \boldsymbol{x},\, \boldsymbol{S}'\boldsymbol{q}(\omega) - \boldsymbol{b}(\omega)\rangle\right|,$$

where $\boldsymbol{S} = \mathrm{diag}\{s_n\}$, $\boldsymbol{b}(\omega)$ is defined as in (18), and

$$q_n(\omega) \triangleq \frac{1}{\sqrt{N}} \sum_{k=0}^{K-1} e^{\imath\gamma k n} \tilde{F}^{\star}(\omega/\gamma - k)\,.$$

As before, by Cauchy-Schwarz, the worst-case unit-norm signal is $\boldsymbol{x} = (\boldsymbol{S}'\boldsymbol{q} - \boldsymbol{b})/\left\|\boldsymbol{S}'\boldsymbol{q} - \boldsymbol{b}\right\|$, so the worst-case error for frequency ω, normalized by $1/\sqrt{N}$, is

$$\mathcal{E}(\boldsymbol{s}, \omega) = \left\|\boldsymbol{S}'\boldsymbol{q}(\omega) - \boldsymbol{b}(\omega)\right\|. \qquad (42)$$

Expanding, an alternate expression is

$$\begin{aligned}\mathcal{E}^2(\boldsymbol{s}, \omega) &= \frac{1}{N} \sum_{n=0}^{N-1} \left| s_n^{\star}\sqrt{N} q_n(\omega) - e^{\imath\omega n}\right|^2 \\ &= \frac{1}{N} \sum_{n=0}^{N-1} \left| s_n z_n(\omega/\gamma) - 1\right|^2 \qquad (43)\end{aligned}$$

where

$$z_n(\rho) \triangleq e^{\imath\rho\gamma n}\sqrt{N} q_n^{\star}(\rho\gamma) = \sum_{k=0}^{K-1} e^{\imath\gamma(\rho-k)n} \tilde{F}(\rho - k)\,. \qquad (44)$$

Since $z_n(\rho)$ has period unity, $\mathcal{E}(\cdot)$ is γ-periodic. Thus we focus on $\omega = \rho\gamma$ for $\rho \in [0, 1)$, for which

$$\begin{aligned} z_n(\rho) &= \sum_{k=0}^{K-1} e^{\imath\gamma(\rho-k)n} \tilde{F}(\rho - k) \\ &= \sum_{j=-J/2+1}^{J/2} e^{\imath\gamma(\rho-j)n} F(\rho - j)\,. \end{aligned}$$

(For odd J the summation limits are $-(J-1)/2$ to $(J-1)/2$.)

For a given interpolation kernel F, ideally we would like to choose the scaling factors $\boldsymbol{s}$ to minimize the *maximum* error via the following min-max criterion:

$$\min_{\boldsymbol{s} \in \mathbb{C}^N} \max_{\omega} \mathcal{E}(\boldsymbol{s}, \omega).$$

This maximization over ω seems intractable, so we adopt a simple "do no harm" strategy by minimizing the worst-case error *at the DFT frequency locations*:

$$\min_{\boldsymbol{s} \in \mathbb{C}^N} \max_{\omega \in \Omega_K} \mathcal{E}(\boldsymbol{s}, \omega). \tag{45}$$

Considering (43), the solution to (45) is simply

$$s_n = \frac{1}{z_n(0)} = \frac{1}{\sum_{j=-J/2}^{J/2-1} e^{\imath \gamma j n} F(j)}. \tag{46}$$

If the kernel $F(\cdot)$ satisfied the classical interpolation properties $F(0) = 1$ and $F(k) = 0$ for $k \neq 0$, then (46) would reduce to uniform scaling factors ($\boldsymbol{s} = \mathbf{1}$).

One calculates the worst-case error of conventional interpolators of the form (41) by substituting (46) into (43). Since (43) approaches a finite limit as $N \to \infty$, we again focus on this "large N" approximation (which again we have found to be very accurate even for modest values of N).

With the choice (46), $\mathcal{E}(\omega) = 0$ for all $\omega \in \Omega_K$, and we have observed empirically that the maximum error occurs at the midpoints between the DFT frequencies Ω_K as expected. We conjecture that if $F(\cdot)$ is Hermitian symmetric about zero, then $\mathcal{E}(\omega)$ has a stationary point at $\omega = \gamma/2$ for the choice (46). Lacking a proof, we compute numerically the maximum error $\mathcal{E}_{\max} = \max_\omega \mathcal{E}(\boldsymbol{s}, \omega)$.

B. Comparisons of min-max to conventional

The purpose of the preceding analysis was to enable a fair comparison of the min-max interpolator (31) with conventional interpolators (41) while using good scaling factors for the latter. In particular, the apparent similarity in Fig. 1 between the min-max interpolator and the (truncated) ideal Dirichlet interpolator (5) raises the question of how well a simple truncated Dirichlet interpolator would perform.

Fig. 8 compares the maximum error for the min-max interpolator and for the truncated Dirichlet interpolator $I(\omega) \operatorname{rect}\left(\frac{\omega}{\gamma J}\right)$, for $K/N = 2$, using (39) and (43). Also shown is the cos^3 tapered Dirichlet interpolator proposed in [50, 51]. Both uniform scaling factors and numerically optimized $\boldsymbol{\alpha}$'s were used for the min-max case. Our min-max interpolator can have much less error than the truncated or tapered Dirichlet interpolator. The seemingly subtle differences in Fig. 8 are quite important in terms of maximum error!

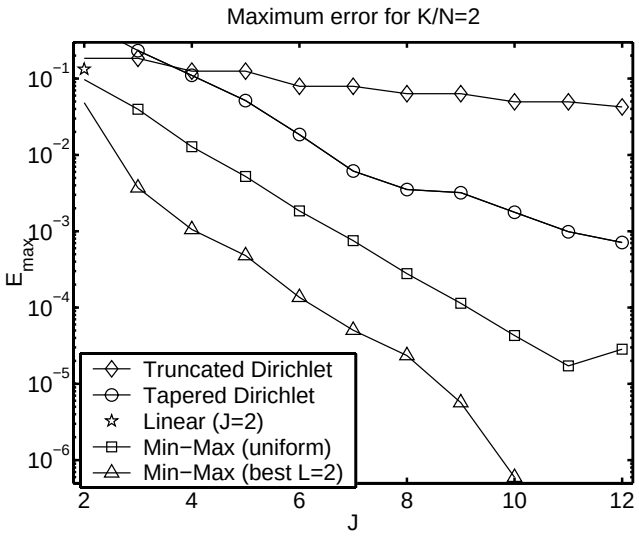

Figure 8: Maximum error $\mathcal{E}_{\max}$ of truncated Dirichlet interpolator, of cos^3-tapered Dirichlet interpolator, of linear interpolator ($J = 2$), and of min-max interpolator for various neighborhood sizes J, and oversampling factor $K/N = 2$. Despite similarities in Fig. 1, the min-max approach significantly reduces error relative to a truncated or tapered Dirichlet.

The focus in some previous NUFFT papers has been on gaussian bell interpolators. Fig. 9 compares the worst-case error between the min-max interpolator and a truncated gaussian bell interpolator

$$F(\kappa) = e^{-(\kappa/\sigma)^2/2} \operatorname{rect}\left(\frac{\kappa}{\gamma J}\right).$$

Using (43), we searched exhaustively over σ to optimize the gaussian interpolator and used (46) for its scaling factors. Errors for the min-max method are shown for both uniform scaling factors and least-squares fit scaling factors as described next.

Choosing the scaling vector by exhaustive minimization of $\mathcal{E}_{\max}$ becomes more tedious as L increases, and the presence of sharp local minima (*cf.* Fig. 5) is a challenge for local descent methods. We found the following approach to be a useful alternative. We numerically evaluate the worst-case error of a gaussian interpolator using (43) for a range of widths of the gaussian. For the width giving the lowest error, we compute the scaling factors using (46). Then we use ordinary least-squares linear regression with $L \approx 5$ in (25) to find a $\boldsymbol{\alpha}$ for (25) that closely matches the optimized gaussian scaling factors. Then we use that

$\boldsymbol{\alpha}$ in (39) to compute the error of this "optimized" min-max interpolator. An example is shown in Fig. 9. This approach reduces the nonlinear part of the search from an L-dimension search over $\boldsymbol{\alpha}$ to a 1D search over the gaussian width. Again this process is practical only when one plans to perform many NUFFTs for the same J and K/N. (Clearly analytical optimization of $\boldsymbol{s}$ for the min-max approach would be preferable.) In practice, we often simply use the scaling factors in Table 1 for expediency.

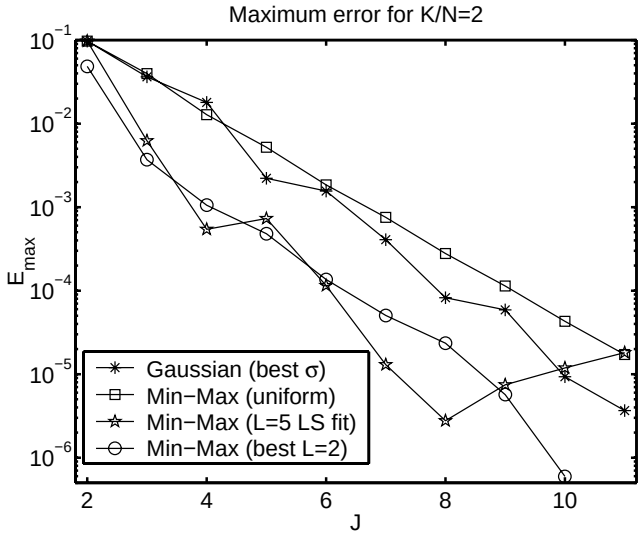

Figure 9: Maximum error $\mathcal{E}_{\max}$ of min-max interpolators and truncated gaussian interpolator vs neighborhood size J for oversampling factor $K/N = 2$. For each J, the gaussian width σ was optimized numerically by exhaustive search to minimize worst-case error. Three choices of scaling factors (s_n's) for the min-max method are shown: uniform, numerically optimized, and LS fit of (25) to optimized gaussian s_n's given by (46).

Fig. 9 illustrates several important points. Firstly, the min-max interpolator with simple uniform scaling factors has comparable error to the exhaustively-optimized gaussian interpolator. Secondly, optimizing the scaling factors very significantly reduces the min-max interpolation error, outperforming both the gaussian interpolator and the min-max interpolator with uniform scaling factors. Thirdly, for $J \leq 8$, exhaustive optimization of $\boldsymbol{\alpha}$ with $L = 2$ yields comparable maximum error to the simpler least-squares fit (using $L = 5$) to the optimized gaussian scaling factors (46), so the latter approach may be preferable in the practical use of the min-max method. However, even better results would be obtained if there were a practical method for optimizing $\boldsymbol{\alpha}$ for $L > 2$.

VI. 2D Example

To illustrate the accuracy of the NUFFT method in a practical context, we considered the classical 128×128 Shepp-Logan image [52, 53]. We generated 10000 random frequency locations (ω_m's) in $(-\pi, \pi) \times (-\pi, \pi)$ and computed the 2D FT *exactly* (to within double precision in Matlab) and with the 2D NUFFT method with $J = 6$ and $K/N = 2$. The relative percent error

$$\frac{\max_{\omega_m} |\hat{X}(\omega_m) - X(\omega_m)|}{\max_{\omega_m} |X(\omega_m)|} \times 100\%$$

was less than 0.14% when uniform scaling factors were used, and less than 0.012% when the optimized scaling factors for $L = 2$ in Table 1 were used. This order-of-magnitude error reduction is consistent with the reductions shown in Fig. 3, and confirms that minimizing the worst-case error can lead to significant error reductions even with practical signals of interest. The exact FT method required more than 100 times the CPU time of the NUFFT method as measured by Matlab's `tic/toc` functions. For comparison, classical bilinear interpolation yields a relative error of 6.7% for this problem. This large error is why linear interpolation is insufficiently accurate for tomographic reprojection by Fourier methods. The NUFFT approach with optimized min-max interpolation reduces this error by nearly two orders of magnitude.

VII. Discussion

The proposed min-max interpolation method for the NUFFT should have applications in a variety of signal processing and imaging problems where nonuniform frequency samples are required.

The min-max formulation provides a natural framework for optimizing the scaling factors. This optimization led to considerably reduced errors compared to the previously considered uniform and cosine scaling factors [36].

The min-max interpolator is readily useful in its own right, and is also useful as a benchmark for evaluating alternative interpolators. For example, Fig. 9 shows that the gaussian bell interpolator with optimized width provides accuracy comparable to the min-max interpolator *if* one is willing to use a modestly larger neighborhood for the gaussian. In applications where precomputing and storing the interpolation coefficients is impractical, using the somewhat simpler gaussian bell with a wider support may be a reasonable tradeoff. However, in the imaging problems that motivated our work, precomputing the $u_j(\omega_m)$'s is feasible, and in the interest of minimizing computation per iteration, we want to use the smallest support possible; thus the min-max approach is preferable.

One remaining open problem is that the $J \times J$ matrix $\boldsymbol{C}'\boldsymbol{C}$ becomes ill-conditioned as J increases. Likewise for $\boldsymbol{C}'\boldsymbol{S}\boldsymbol{S}'\boldsymbol{C}$, at least for the optimized scaling factors. Since J is small, we currently use a truncated SVD type of pseudo-inverse when such ill-conditioning appears. Perhaps a more sophisticated form of regularization of its inverse could further improve accuracy.

Several generalizations of the method are apparent. We have used the usual Euclidian norm $\|\boldsymbol{x}\|$ in our min-max formulation (10). In some applications alternative norms may be useful. The general theory accommodates any quadratic norm; however, whether simplifications of the form (26) and (27) appear may depend on the norm.

Another possible generalization would be to use *different* scaling factors for the two FFT's in (36). It is unclear how much, if any, error reduction this generalization could provide, but the additional computational cost would be very minimal.

Our focus has been "Problem 2" (*cf.* (2)) in the terminology of [29, 36]. The imaging problems that motivated this work might be considered by some readers to be somewhat akin to "Problem 1." For example, in magnetic resonance imaging with non-Cartesian k-space trajectories, we are given nonuniform samples in the spatial-frequency domain, and want to reconstruct uniformly-spaced object samples. In our view, this type of problem is best formulated as an *inverse problem* in a maximum-likelihood or penalized-likelihood framework, *e.g.*, [54]. The iterative algorithms associated with such approaches require repeated calculation of the "forward problem" (from object space to frequency space, and the adjoint thereof) [18, 19]. Those forward problems are exactly of the "Problem 2" type addressed in this paper. Nevertheless, there may be other applications where "Problem 1" is the natural formulation, so it may also be of interest to search for min-max interpolation strategies for "Problem 1."

Finally, one could extend the min-max approach to related transforms such as Hankel and cosine [12, 55].

VIII. Acknowledgement

The authors gratefully acknowledge Doug Noll for discussions about gridding and MR image reconstruction, and Robert Lewitt for references and comments on a draft of this paper.

References

[1] A V Oppenheim, D Johnson, and Ken Steiglitz, "Computation of spectra with unequal resolution using the fast Fourier transform," *Proc. IEEE*, vol. 59, no. 2, pp. 299–301, Feb. 1971.

[2] D C Munson, J D O'Brien, and W K Jenkins, "A tomographic formulation of spotlight mode synthetic aperture radar," *Proc. IEEE*, vol. 71, no. 8, pp. 917–25, Aug. 1983.

[3] D C Munson and J L Sanz, "Image reconstruction from frequency-offset Fourier data," *Proc. IEEE*, vol. 72, no. 6, pp. 661–9, June 1984.

[4] Hyeokho Choi and David C Munson, "Direct-Fourier reconstruction in tomography and synthetic aperture radar," *Intl. J. Imaging Sys. and Tech.*, vol. 9, no. 1, pp. 1–13, 1998.

[5] Eric Larsson, Peter Stoica, and J Li, "SAR image construction from gapped phase-history data," in *Proc. IEEE Intl. Conf. on Image Processing*, 2001, vol. 3, pp. 608–11.

[6] Y Wu and D Munson, "Multistatic passive radar imaging using the smoothed pseudo Wigner-Ville distribution," in *Proc. IEEE Intl. Conf. on Image Processing*, 2001, vol. 3, pp. 604–7.

[7] Emmanuel J Candes and D L Donoho, "Ridgelets: a key to higher-dimensional intermittency?," *R. Soc. Lond. Philos. Trans. Ser. A Math. Phys. Eng. Sci.*, vol. 357, no. 1760, pp. 2495–509, Sept. 1999.

[8] Q H Liu and N Nguyen, "An accurate algorithm for nonuniform fast Fourier transforms (NUFFT's)," *IEEE Microwave and Guided Wave Letters*, vol. 8, no. 1, pp. 18–20, Jan. 1998.

[9] Q H Liu, N Nguyen, and X Y Tang, "Accurate algorithms for nonuniform fast forward and inverse Fourier transforms and their applications," in *IEEE Geoscience and Remote Sensing Symposium Proceedings*, 1998, vol. 1, pp. 288–90.

[10] Qing Huo Liu and Xue Yuan Tang, "Iterative algorithm for nonuniform inverse fast Fourier transform," *Electronics Letters*, vol. 34, no. 20, pp. 1913–4, Oct. 1998.

[11] Xue Min Xu and Qing Huo Liu, "The conjugate-gradient nonuniform fast Fourier transform (CG-NUFFT) method for one- and two-dimensional media," *Microwave and optical technology letters*, vol. 24, no. 6, pp. 385–9, Mar. 2000.

[12] Q H Liu, X M Xu, B Tian, and Z Q Zhang, "Applications of nonuniform fast transform algorithms in numerical solutions of differential and integral equations," *IEEE Tr. Geosci. Remote Sensing*, vol. 38, no. 4, pp. 1551–60, July 2000.

[13] E Angelidis and J E Diamessis, "A novel method for designing FIR digital filters with nonuniform frequency samples," *IEEE Tr. Sig. Proc.*, vol. 42, no. 2, pp. 259–67, Feb. 1994.

[14] S Bagchi and S K Mitra, "The nonuniform discrete Fourier transform and its applications in filter design. I—1-D," *IEEE Tr. Circ. Sys. II, Analog and digital signal processing*, vol. 43, no. 6, pp. 422–33, June 1996.

[15] A Makur and S K Mitra, "Warped discrete-Fourier transform: Theory and applications," *IEEE Tr. Circ. Sys. I, Fundamental theory and applications*, vol. 48, no. 9, pp. 1086–93, Sept. 2001.

[16] Sonali Bagchi and Sanjit Mitra, *The nonuniform discrete Fourier transform and its applications in signal processing*, Kluwer, Boston, 1999.

[17] K P Pruessmann, M Weiger, M B Scheidegger, and P Boesiger, "SENSE: sensitivity encoding for fast MRI," *Magnetic Resonance in Medicine*, vol. 42, pp. 952–62, 1999.

[18] B P Sutton, J A Fessler, and D Noll, "A min-max approach to the nonuniform N-D FFT for rapid iterative reconstruction of MR images," in *Proc. Intl. Soc. Mag. Res. Med.*, 2001, p. 763.

[19] B P Sutton, J A Fessler, and D Noll, "Iterative MR image reconstruction using sensitivity and inhomogeneity field maps," in *Proc. Intl. Soc. Mag. Res. Med.*, 2001, p. 771.

[20] C R Crawford, "System for reprojecting images using transform techniques," 1986, US Patent 4,616,318. Filed 1983-6-7. Elscint.

[21] C R Crawford, J G Colsher, N J Pelc, and A H R Lonn, "High speed reprojection and its applications," in *Proc. SPIE 914, Med. Im. II: Im. Formation, Detection, Processing, and Interpretation*, 1988, pp. 311–8.

[22] C W Stearns, D A Chesler, and G L Brownell, "Three-dimensional image reconstruction in the Fourier domain," *IEEE Tr. Nuc. Sci.*, vol. 34, no. 1, pp. 374–8, Feb. 1987.

[23] C W Stearns, D A Chesler, and G L Brownell, "Accelerated image reconstruction for a cylindrical positron tomograph using Fourier domain methods," *IEEE Tr. Nuc. Sci.*, vol. 37, no. 2, pp. 773–7, Apr. 1990.

[24] S Matej and R M Lewitt, "3-FRP: direct Fourier reconstruction with Fourier reprojection for fully 3-D PET," *IEEE Tr. Nuc. Sci.*, vol. 48, no. 4-2, pp. 1378–1385, Aug. 2001.

[25] J A Fessler and B P Sutton, "A min-max approach to the multidimensional nonuniform FFT: Application to tomographic image reconstruction," in *Proc. IEEE Intl. Conf. on Image Processing*, 2001, vol. 1, pp. 706–9.

[26] S Bagchi and S K Mitra, "The nonuniform discrete Fourier transform and its applications in filter design. II—2-D," *IEEE Tr. Circ. Sys. II, Analog and digital signal processing*, vol. 43, no. 6, pp. 434–44, June 1996.

[27] G Evangelista and S Cavaliere, "Discrete frequency warped wavelets: Theory and applications," *IEEE Tr. Sig. Proc.*, vol. 46, pp. 874–85, Apr. 1998.

[28] N I Cho and S K Mitra, "Warped discrete cosine transform and its application in image compression," *IEEE Tr. Circ. Sys. Vid. Tech.*, vol. 10, no. 8, pp. 1364–73, Dec. 2000.

[29] A Dutt and V Rokhlin, "Fast Fourier transforms for nonequispaced data," *SIAM J. Sci. Comp.*, vol. 14, no. 6, pp. 1368–93, Nov. 1993.

[30] C Anderson and M D Dahleh, "Rapid computation of the discrete Fourier transform," *SIAM J. Sci. Comp.*, vol. 17, no. 4, pp. 913–9, July 1996.

[31] G Beylkin, "On the fast Fourier transform of functions with singularities," *Applied and Computational Harmonic Analysis*, vol. 2, no. 4, pp. 363–81, Oct. 1995.

[32] A Dutt and V Rokhlin, "Fast Fourier transforms for nonequispaced data, II," *Applied and Computational Harmonic Analysis*, vol. 2, pp. 85–100, 1995.

[33] G Steidl, "A note on the fast Fourier transforms for nonequispaced grids," *Advances in computational mathematics*, vol. 9, no. 3, pp. 337–52, 1998.

[34] A F Ware, "Fast approximate Fourier transforms for irregularly spaced data," *SIAM Review*, vol. 40, no. 4, pp. 838–56, Dec. 1998.

[35] A J W Duijndam and M A Schonewille, "Nonuniform fast Fourier transform," *Geophysics*, vol. 64, no. 2, pp. 539–51, Mar. 1999.

[36] Nhu Nguyen and Qing Huo Liu, "The regular Fourier matrices and nonuniform fast Fourier transforms," *SIAM J. Sci. Comp.*, vol. 21, no. 1, pp. 283–93, 1999.

[37] Daniel Potts, Gabriele Steidl, and Manfred Tasche, "Fast Fourier transforms for nonequispaced data: A tutorial," in *Modern Sampling Theory: Mathematics and Application*, J J Benedetto P Ferreira, Ed., pp. 253–74. Birkhauser, ?, 2000.

[38] T W Parks and D P Kolba, "Interpolation minimizing maximum normalized error for band-limited signals," *IEEE Tr. Acoust. Sp. Sig. Proc.*, vol. 26, no. 4, pp. 381–4, Aug. 1978.

[39] D S Chen and J P Allebach, "Analysis of error in reconstruction of two-dimensional signals from irregularly spaced samples," *IEEE Tr. Acoust. Sp. Sig. Proc.*, vol. 35, no. 2, pp. 173–80, Feb. 1987.

[40] Ram G Shenoy and Thomas W Parks, "An optimal recovery approach to interpolation," *IEEE Tr. Sig. Proc.*, vol. 40, no. 8, pp. 1987–92, Aug. 1992.

[41] N P Willis and Y Bresler, "Norm invariance of minimax interpolation," *IEEE Tr. Info. Theory*, vol. 38, no. 3, pp. 1177–81, May 1992.

[42] G Calvagno and D C Munson, "A frequency-domain approach to interpolation from a nonuniform grid," *Signal Processing*, vol. 52, no. 1, pp. 1–21, July 1996.

[43] H Choi and D C Munson, "Analysis and design of minimax-optimal interpolators," *IEEE Tr. Sig. Proc.*, vol. 46, no. 6, pp. 1571–9, June 1998.

[44] D Darian Muresan and Thomas W Parks, "Optimal recovery approach to image interpolation," in *Proc. IEEE Intl. Conf. on Image Processing*, 2001, vol. 3, pp. 848–51.

[45] T W Parks and J H McClellan, "Chebyshev approximation for nonrecursive digital filters with linear phase," *IEEE Tr. Circ. Theory*, vol. 19, no. 2, pp. 189–99, Mar. 1972.

[46] R E Crochiere and L R Rabiner, *Multirate digital signal processing*, Prentice-Hall, NJ, 1983.

[47] Sanjit K Mitra, *Digital signal procesing: A computer-based approach*, McGraw-Hill, New York, 1998.

[48] L Rodman and T Shalom, "On inversion of symmetric Toeplitz matrices," *SIAM J. Matrix. Anal. Appl.*, vol. 13, no. 2, pp. 530–49, 1992.

[49] F T A W Wajer, R Lethmate, J A C van Osch, D Graveron-Demilly, M Fuderer, and D van Ormondt, "Simple formula for the accuracy of gridding," in *Proc. Intl. Soc. Mag. Res. Med.*, 2001, p. 776.

[50] M Magnusson, P-E Danielsson, and P Edholm, "Artefacts and remedies in direct Fourier tomographic reconstruction," in *Proc. IEEE Nuc. Sci. Symp. Med. Im. Conf.*, 1992, vol. 2, pp. 1138–40.

[51] Salvatore Lanzavecchia and Pier Luigi Bellon, "A bevy of novel interpolating kernels for the Shannon reconstruction of high-bandpass images," *J. Visual Comm. Im. Rep.*, vol. 6, no. 2, pp. 122–31, June 1995.

[52] L A Shepp and B F Logan, "The Fourier reconstruction of a head section," *IEEE Tr. Nuc. Sci.*, vol. 21, no. 3, pp. 21–43, June 1974.

[53] A C Kak and M Slaney, *Principles of computerized tomographic imaging*, IEEE Press, New York, 1988.

[54] A J W Duijndam, M A Schonewille, and C O H Hindriks, "Reconstruction of band-limited signals, irregularly sampled along one spatial direction," *Geophysics*, vol. 64, no. 2, pp. 524–38, Mar. 1999.

[55] Qing Huo Liu and Zhong Qing Zhang, "Nonuniform fast Hankel transform (NUFHT) algorithm," *appopt*, vol. 38, no. 32, pp. 6705–8, Nov. 1999.