Nonuniform Fast Fourier Transforms Using Min-Max Interpolation

Jeffrey A. Fessler, *Senior Member, IEEE,* and Bradley P. Sutton, *Member, IEEE*

Abstract—The fast Fourier transform (FFT) is used widely in signal processing for efficient computation of the 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. The min-max criterion is also useful for optimizing the parameters of interpolation kernels such as the Kaiser–Bessel function.

Index Terms—Discrete Fourier transform, gridding, imaging, min-max interpolation, magnetic resonance, tomography.

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 for an N-point signal, 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]–[20] and by iterative tomographic image reconstruction methods, where reprojection is based on the Fourier slice theorem [21]–[28]. These problems relate closely to the problem of reconstructing a bandlimited signal from nonuniform samples. Strohmer

argued compellingly for using trigonometric polynomials (complex exponentials) for finite-dimensional approximations in such problems [29] and proposed to use an iterative conjugate gradient reconstruction method with the nonuniform FFT (NUFFT) approach of [30] at its core. The min-max NUFFT approach presented here fits in that framework but provides higher accuracy. We explore these applications in more detail elsewhere [20], [28] 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], [31] and image compression [32], [33]. Warping methods apply only to special patterns of frequency locations and are insufficiently general for most applications.

In the scientific computing literature, several recent papers have described methods for approximating the one-dimensional (1-D) nonuniform FT by interpolating an oversampled FFT, beginning with [34] and including [8], [10], [30], and [35]–[41]. Related methods were known in astrophysics even earlier [42]. Such methods are often called the nonuniform FFT or NUFFT. Most of these algorithms have been presented only for 1-D signals, and many involve seemingly arbitrary choices for interpolation functions. This paper starts from first principles to derive a min-max approach to the interpolation problem. We find the fixed-width interpolator that minimizes the worst-case approximation error over all signals of unit norm. (Like all NUFFT methods, the user can trade off 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 [40]. We compare our approach to theirs in detail in Section IV-C.

This work is in the spirit of min-max approaches for other signal processing problems, such as bandlimited signal interpolation [43]–[49] and filter design [50], [51].

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 two-dimensional (2-D) NUFFT example.

II. THEORY: 1-D CASE

For simplicity, we first describe our min-max approach in the 1-D 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 $n = 0, \ldots, N-1$, with corresponding FT

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

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

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

The symbol "$\triangleq$" denotes "defined to be." The ω_ms can be arbitrary real numbers. This form has been called the nonuniform discrete Fourier transform (NDFT) [52, p. 194]. 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 [34], [40]. Sections III-F and G discuss alternative problems.

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

$$Y_k = \sum_{n=0}^{N-1} s_n x_n e^{-\imath \gamma k n}, \quad k = 0, \ldots, K-1, \tag{3}$$

where $\gamma \triangleq 2\pi/K$ is the fundamental frequency of the K-point DFT. The nonzero s_ns are algorithm design variables that have been called "scaling factors" [40]. We call $\boldsymbol{s} = (s_1, \ldots, s_N)$ the *scaling vector*. The purpose of $\boldsymbol{s}$ is to partially precompensate for imperfections in the subsequent frequency-domain interpolation. This first step requires $O(K \log N)$ operations if implemented efficiently as described in Section III-D.

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

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

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

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

$$X(\omega) = \sum_{n=0}^{N-1} x_n e^{-\imath \omega n} = \sum_{n=0}^{N-1} \left[\frac{1}{K} \sum_{k=0}^{K-1} Y_k e^{\imath \gamma k n} \right] e^{-\imath \omega n}$$

$$= \sum_{k=0}^{K-1} Y_k I\left(\frac{\omega}{\gamma} - k \right)$$

where the ideal interpolator kernel is

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

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

$$\delta_N(\kappa) \triangleq \frac{1}{N} \sum_{n=0}^{N-1} e^{\pm \imath \gamma \kappa(n - \eta_0)}$$

$$= \begin{cases} \frac{\sin\left(\frac{\pi \kappa N}{K}\right)}{N \sin\left(\frac{\pi \kappa}{K}\right)}, & \frac{\kappa}{K} \notin \mathbb{Z} \\ 1, & \frac{\kappa}{K} \in \mathbb{Z}. \end{cases} \tag{6}$$

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); therefore, it 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:

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

This offset satisfies the following shift property:

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

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

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

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 [34]. Tabei and Ueda also used such kernels in the specific context of direct Fourier tomographic reconstruction and included error analyses [53]. For even N and odd J only, Nguyen and Liu [40] considered interpolation of the form (9) with a choice for the u_js that arises from least-squares approximations of complex exponentials by linear combinations of other complex exponentials. We propose next an explicit min-max criterion for choosing the u_js, 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 x having unit norm. Hypothetically, this could yield shift-variant interpolation since each desired frequency location ω_m may have its own set of J interpolation coefficients.

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

$$\min_{s \in \mathbb{C}^N} \max_{\omega} \min_{u(\omega) \in \mathbb{C}^J} \max_{x \in \mathbb{C}^N : \|x\| \leq 1} \left| \hat{X}(\omega) - X(\omega) \right|. \quad (10)$$

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

Mathematically, our min-max criterion is the following:

$$\min_{u(\omega_m) \in \mathbb{C}^J} \max_{x \in \mathbb{C}^N : \|x\| \leq 1} \left| \hat{X}(\omega_m) - X(\omega_m) \right|. \quad (11)$$

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

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

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

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

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

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

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

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

$$g(\omega) = D(\omega) \left[S'C\Lambda(\omega)u(\omega) - b(\omega) \right] \quad (14)$$

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

$$D_{nn}(\omega) = e^{\imath\omega\eta_0} e^{\imath\gamma k_0(\omega)(n-\eta_0)} \quad (15)$$

$$C_{nj} = \frac{e^{\imath\gamma j(n-\eta_0)}}{\sqrt{N}} \quad (16)$$

$$\Lambda_{jj}(\omega) = e^{-\imath[\omega-\gamma(k_0(\omega)+j)]\eta_0} \quad (17)$$

$$b_n(\omega) = \frac{e^{\imath(\omega-\gamma k_0(\omega))(n-\eta_0)}}{\sqrt{N}}. \quad (18)$$

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

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

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

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

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

Inserting this case into the min-max criterion (19) and applying (14) and (15) reduces the min-max problem to the following (cf. [40, eqn. 10]):

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

The minimizer of this ordinary least-squares problem for $\omega = \omega_m$ is $u(\omega_m)$ where

$$u(\omega) = \Lambda'(\omega) \left[C'SS'C \right]^{-1} C'Sb(\omega) \quad (21)$$

(since Λ is unitary). This is a general expression for the min-max interpolator. Due to the shift property (8) and the definitions of $\Lambda(\omega)$ and $b(\omega)$, we see

$$u_j(\omega + \gamma l) = u_j(\omega), \quad \forall l \in \mathbb{Z}. \quad (22)$$

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

To apply the min-max interpolator (21), we must compute the interpolation coefficients $u(\omega)$ for each frequency location ω_m of interest. One method for computing (21) would be to use the following QR decomposition:

$$S'C = QR \quad (23)$$

where $Q \in \mathbb{C}^{N \times J}$ is a matrix with orthogonal columns, and R is an upper triangular invertible matrix. Since $S'C$ is independent of frequency location, we could precompute its QR decomposition and then precompute the matrix product $R^{-1}Q'$. We could then compute the interpolation coefficients by substituting (23) into (21), yielding

$$u(\omega) = \Lambda'(\omega) \left(R^{-1}Q' \right) b(\omega). \quad (24)$$

After precomputing $R^{-1}Q'$, this approach would require $2NJ$ operations per frequency location. These operations are independent of x, so this approach may be reasonable when one needs apply repeated NUFFT operations for the same set of frequency locations. (This mode is discussed further below.) However, the next subsection shows that if we use an L-term Fourier series for the s_ns, then we can reduce precomputation to $O((L+1+J)J)$ operations per frequency location. Usually, $L+1+J \ll N$; therefore, the savings can be significant for small L. However, very high accuracy computations may require large L, in which case, the above QR approach may be preferable.

C. Efficient Computation

An alternative expression for the interpolator (21) is

$$u(\omega) = \Lambda'(\omega)Tr(\omega) \quad (25)$$

where we define

$$T \triangleq \left[C'SS'C \right]^{-1} \quad (26)$$

$$r(\omega) \triangleq C'Sb(\omega). \quad (27)$$

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

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

$$s_n = \sum_{t=-L}^{L} \alpha_t e^{\iota \gamma \beta t (n - \eta_0)}, \quad n = 0, \ldots, N - 1. \quad (28)$$

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 αs are Hermitian symmetric, i.e., $\alpha_{-t} = \alpha_t^*$. We represent the coefficients by the vector $\boldsymbol{\alpha} \triangleq (\alpha_0, \alpha_1, \ldots, \alpha_L)$. As one special case of (28), the "cosine" scaling factors considered in [40] 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 (28). This expansion generalizes significantly the choices of scaling factors considered by Nguyen and Liu in [40] and can improve accuracy significantly, 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* [40].

Combining (28) and (26) with (16) yields

$$\begin{aligned}
\left[\boldsymbol{C}'\boldsymbol{S}\boldsymbol{S}'\boldsymbol{C}\right]_{l,j} &= \sum_{n=0}^{N-1} C_{nl}^* s_n s_n^* C_{nj} \\
&= \frac{1}{N} \sum_{n=0}^{N-1} \sum_{t=-L}^{L} \sum_{s=-L}^{L} \alpha_t \alpha_s^* e^{\iota \gamma (j - l + \beta(t-s))(n - \eta_0)} \\
&= \sum_{t=-L}^{L} \sum_{s=-L}^{L} \alpha_t \alpha_s^* \delta_N \left(j - l + \beta(t - s) \right) \quad (29)
\end{aligned}$$

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

The following properties of $\boldsymbol{T}$ are useful. From (26), $\boldsymbol{T}$ is Hermitian, and from (29), $\boldsymbol{T}$ is Toeplitz. In the usual case where the αs are real, $\boldsymbol{T}$ is a real matrix. If $K = N$ and $\boldsymbol{S} = \boldsymbol{I}$, then $\boldsymbol{T}^{-1} = \boldsymbol{C}'\boldsymbol{C} = \boldsymbol{I}$.

Conveniently, in this min-max framework the matrix-vector product defining $\boldsymbol{r}(\omega)$ in (27) also simplifies

$$\begin{aligned}
r_j(\omega) &= \sum_{n=0}^{N-1} C_{nj}^* s_n b_n(\omega) \\
&= \sum_{t=-L}^{L} \alpha_t \frac{1}{N} \sum_{n=0}^{N-1} e^{\iota \gamma [\omega/\gamma - k_0(\omega) - j + \beta t](n - \eta_0)} \\
&= \sum_{t=-L}^{L} \alpha_t \delta_N \left(\frac{\omega}{\gamma} - k_0(\omega) - j + \beta t \right) \quad (30)
\end{aligned}$$

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.

In the usual case where the αs are real, the vector $\boldsymbol{r}(\omega)$ is real. Therefore, the only complex component of the min-max interpolator $\boldsymbol{u}(\omega)$ in (25) 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 (25) for each ω_m using the analytical results (29) and (30). Since $\boldsymbol{C}'\boldsymbol{S}\boldsymbol{S}'\boldsymbol{C}$ is only $J \times J$, where J is usually less than 10, we always precompute $\boldsymbol{T}$ in (26) prior to all other calculations.

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

D. 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 fast memory is available, and then apply (9) directly to compute the NUFFT as needed.

Precomputing each $\boldsymbol{u}(\omega_m)$ using (25) requires only $O(J(L + 1 + J))$ operations. A key property of (29) and (30) is that they collapse the summations over n into the easily computed function δ_N, thereby significantly reducing the precomputation operations.

After precomputing each $\boldsymbol{u}(\omega_m)$, every subsequent NUFFT interpolation step (9) requires only $O(JM)$ operations. Excluding the precomputation, the overall operation count per NUFFT is $O(K \log N) + O(JM)$. An accuracy-computation time tradeoff is available through the choices for the oversampling factor K/N and the neighborhood size J. Typically, we use $K \approx 2N$, $L \leq 13$, $J \leq 10$, and $M \approx N$; therefore, 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!

E. Reduced Memory Mode

In unusual cases where storing all JM coefficients is infeasible, one can evaluate each $\boldsymbol{u}(\omega_m)$ as needed using (25), (30), and the precomputed $\boldsymbol{T}$ in (26). In this mode, the operation count for the NUFFT interpolation step increases to $O(J(L + 1 + J)M)$, but the storage requirements for the interpolator decrease to J^2.

Alternatively, one could decrease the interpolation operation count to roughly $2JM$ by finely tabulating $\boldsymbol{T}\boldsymbol{r}(\omega)$ over a uniform grid (cf. Fig. 1) and using table lookup with polynomial interpolation to determine the $u_j(\omega_m)$s "on the fly." This approach reduces both storage and interpolation operations but presumably decreases accuracy.

Table I summarizes these various modes.

F. "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

$$\lim_{N \to \infty} \delta_N(t) = \operatorname{sinc}\left(\frac{t}{\mu}\right)$$

TABLE I
COMPUTE OPERATIONS AND INTERPOLATOR STORAGE REQUIREMENTS FOR VARIOUS MODES, DISREGARDING SMALL FACTORS INDEPENDENT OF M

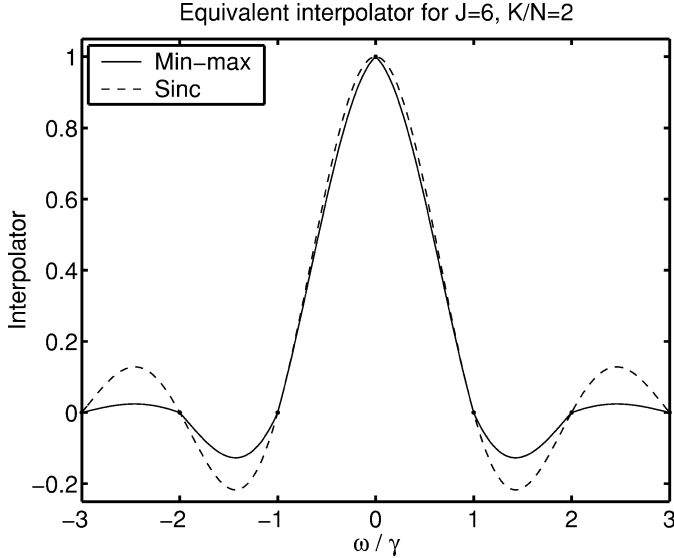| Method	Precomputation	Interpolation	Storage	Accuracy
QR precomputed	$J2NM$	JM	JM	very high
$\boldsymbol{Tr}$ precomputed	$J(L+1+J)M$	JM	JM	high
$\boldsymbol{Tr}$ partial	0	$J(L+1+J)M$	J^2	high
Table / linear interp.	(table size)	$2JM$	(table size)	medium high

Fig. 1. Illustration of the min-max interpolator corresponding to (33) for $J = 6$, $N = 128$, $K/N = 2$ and uniform scaling factors.

where $\text{sinc}(t) \triangleq \sin(\pi t)/(\pi t)$. Therefore, for large N, $\boldsymbol{T} \approx \tilde{\boldsymbol{T}}$, where

$$\left[\tilde{\boldsymbol{T}}^{-1}\right]_{l,j} \triangleq \sum_{t=-L}^{L} \sum_{s=-L}^{L} \alpha_t \alpha_s^* \text{sinc}\left(\frac{j-l+\beta(t-s)}{\mu}\right). \quad (31)$$

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

$$\tilde{r}_j(\omega) \triangleq \sum_{t=-L}^{L} \alpha_t \text{sinc}\left(\frac{\frac{\omega}{\gamma} - k_0(\omega) - j + \beta t}{\mu}\right). \quad (32)$$

Combining these with (25), the interpolator we consider hereafter is

$$\tilde{\boldsymbol{u}}(\omega) \triangleq \boldsymbol{\Lambda}'(\omega)\tilde{\boldsymbol{T}}\tilde{\boldsymbol{r}}(\omega). \quad (33)$$

From (29) and (30), 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 (33) is very reasonable.

G. Effective Interpolation Kernel

Most interpolation methods *start* with a specific functional form for the kernel, such as a Gaussian bell 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 that $L = 0$ and $\alpha_0 = 1$). To visualize the min-max interpolator (33), we can vary ω/γ over the interval $[-J/2, J/2]$ and evaluate $\tilde{\boldsymbol{T}}\tilde{\boldsymbol{r}}(\omega)$ using (33), yielding real functions such as those shown in Figs. 1 and 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.

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 ($\boldsymbol{s} = \boldsymbol{1}$), 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.

III. EXTENSIONS AND VARIATIONS

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 2-D, we oversample the 2-D 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 1-D interpolators. Specifically, for a 2-D 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 $\tilde{\boldsymbol{T}}$ in (31) becomes a Kronecker product (denoted "$\otimes$"):

$$\tilde{\boldsymbol{T}}_{2D} = \tilde{\boldsymbol{T}}_{1D}(J_2, \mu_2) \otimes \tilde{\boldsymbol{T}}_{1D}(J_1, \mu_1) \quad (34)$$
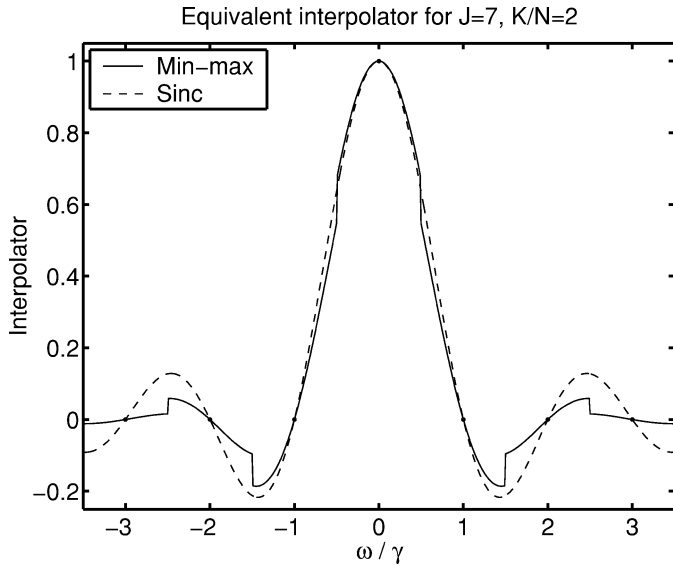
Fig. 2. Illustration of the min-max interpolator corresponding to (33) for $J = 7$, $N = 128$, $K/N = 2$, and uniform scaling factors.

as does the vector $\tilde{\boldsymbol{r}}$ in (32):

$$\tilde{\boldsymbol{r}}_{2\mathrm{D}} = \tilde{\boldsymbol{r}}_{1\mathrm{D}}\left(J_2, \mu_2\right) \otimes \tilde{\boldsymbol{r}}_{1\mathrm{D}}\left(J_1, \mu_1\right). \tag{35}$$

Subroutines for Matlab are freely available online.[1]

B. Shifted Signals

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

$$\sum_{n=0}^{N-1} x_n e^{-\imath(n-\tau)\omega} = e^{\imath\omega\tau} \sum_{n=0}^{N-1} x_n e^{-\imath n\omega}. \tag{36}$$

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_ms 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 nonsquare neighborhoods, e.g., approximate balls.

D. 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

$$Y_k = \begin{cases} \sum_{n=0}^{N-1} \left(s_n x_n\right) e^{-\imath(2\pi/N)n(k/2)}, & k \text{ even} \\ \sum_{n=0}^{N-1} \left(s_n x_n e^{-\imath\gamma n}\right) e^{-\imath(2\pi/N)n((k-1)/2)}, & k \text{ odd.} \end{cases} \tag{37}$$

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

E. Adjoint Operator

Since the NUFFT method described above is a linear operator, it corresponds implicitly to some $M \times N$ matrix, say $\boldsymbol{G}$. In other words, we can express (3) and (9) in matrix-vector form as $\hat{\boldsymbol{X}} = \boldsymbol{G}\boldsymbol{x}$, where $\boldsymbol{G} = \boldsymbol{V}\boldsymbol{W}\boldsymbol{S}$, where $\boldsymbol{S}$ was defined below (14), $\boldsymbol{W}$ is the $K \times N$ oversampled DFT matrix with elements $w_{kn} = e^{-\imath\gamma kn}$, and $\boldsymbol{V}$ is the (sparse) $M \times K$ interpolation matrix with elements v_{mk}. (This matrix representation is for analysis only and not for implementation.) For iterative image reconstruction algorithms, one also needs the *adjoint* of the NUFFT operator, i.e., one must perform matrix-vector multiplications of the form $\boldsymbol{G}'\hat{\boldsymbol{y}}$ for some vector $\tilde{\boldsymbol{y}} \in \mathbb{C}^M$. Since $\boldsymbol{G}$ 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 $\boldsymbol{G}'\tilde{\boldsymbol{y}} = \boldsymbol{S}'\boldsymbol{W}'\boldsymbol{V}'\tilde{\boldsymbol{y}}$ by "reversing" (not inverting!) the algorithm steps described in Section II.

The adjoint corresponding to (4), i.e., the $\boldsymbol{V}'$ term, is

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

(This step is akin to "gridding.") When the (sparse) interpolation matrix is precomputed and stored, this interpolation step requires $O(JM)$ operations. For (3), i.e., for $\boldsymbol{W}'$, the adjoint is

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

which is the K-point inverse DFT of $\tilde{X}_k$ scaled by K, discarding all but the first N signal values. This step requires $O(K \log K)$ operations. One can reduce this to $O(K \log N)$ by using the adjoint of the reduced FFT (37). The final step for $\boldsymbol{S}'$ is to scale each $\tilde{x}_n$ by $s_n^\star$.

F. 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, $k = 0, \ldots, K-1$ corresponding to frequencies $\{\gamma k\}$, we can compute

$$x\left(t_n\right) = \sum_{k=0}^{K-1} X_k e^{\imath\gamma k t_n}, \quad n = 1, \ldots, N \tag{38}$$

using the same type of approach with min-max interpolation. This is again "Problem 2" in the terminology of [34] and [40].

G. Inverse NUFFT

The formulation (2) is called "Problem 2" in the terminology of [34] and [40]. We view the imaging problems that motivated this work as being the *inverse* of (2). 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 other words, the $X(\omega_m)$s are given, and we must find the x_ns. One can formulate such applications as *inverse problems* in a maximum-likelihood or penalized-likelihood framework, e.g., [54]. For example, a least-squares formulation would be

$$\hat{x} = \arg\min_{x} \|X - Gx\|$$

where G was defined in Section III-E. Lee and Yagle analyze the conditioning of such problems [55]. Lacking an efficient method for solving this inverse problem directly (for large N), one applies iterative algorithms. These iterative algorithms require repeated calculation of the "forward problem" (from object space to frequency space and the adjoint thereof) [18]–[20], [28], [56]. Those forward problems are exactly of the "Problem 2" type addressed in this paper; therefore, the methods herein enable fast and accurate iterative solutions to "inverse NUFFT" problems.

In yet another family of problems, one would like to compute an expression of the form

$$\sum_{l} F_l e^{\iota \omega_l t} \tag{39}$$

where the ts of interest are uniformly spaced, but the given ω_ls are not. This is called "Problem 1" in [34] and [40]. It has been called "the Fourier transform of nonuniformly spaced data," although it differs from the usual Fourier transforms considered in signal processing. One can use NUFFT methods to compute accurate approximations to (39) [34], [40]. Such methods are known as "gridding" in the imaging literature, e.g., [25]. The interpolator proposed in this paper *may* be useful for (39), but here, we have been unable to formulate any claim of optimality. In the context of imaging problems known to us, we believe that iterative inverse NUFFT approaches will improve image quality relative to formulations of the form (39), albeit at the expense of greater computation. Nevertheless, there may be other applications where "Problem 1" is the natural formulation, and for these problems, we recommend the general guidelines provided in reviews like [41].

IV. ERROR ANALYSIS

Combining (20) and (21) and simplifying yields the following expression for the worst-case error at frequency ω:

$$\frac{\mathcal{E}_{\text{exact}}(\omega)}{\sqrt{N}} = \left\| S'C \left[C'SS'C \right]^{-1} C'Sb(\omega) - b(\omega) \right\|$$
$$= \left\| (I - QQ') b(\omega) \right\| \tag{40}$$

where Q was defined in (23). The error *bounds* given in NUFFT papers are often described as pessimistic. In contrast, the *exact* worst-case error given by (40) is achievable. Of course, the unit-norm signal that achieves this worst-case error may not be

representative of many problems of interest; therefore, the "typical" performance may appear better than (40).

Alternatively, combining (25) and (20) yields

$$\frac{\mathcal{E}_{\text{exact}}(\omega)}{\sqrt{N}} = \left\| S'C\Lambda(\omega) \left[\Lambda'(\omega) Tr(\omega) \right] - b(\omega) \right\|$$
$$= \sqrt{1 - r'(\omega) T r(\omega)}. \tag{41}$$

When $J < 10 - K/N$, the simpler form (41) is usually adequate. For larger J, the subtraction within the square root is numerically imprecise, so we revert to (40).

To simplify analysis for modest values of J, one can use the "large N" approximations (31) and (32) and normalize out the $\sqrt{N}$ dependence. Specifically, the following approximation is usually very accurate:

$$\frac{\mathcal{E}_{\text{exact}}(\omega)}{\sqrt{N}} \approx \mathcal{E}(\omega) \triangleq \sqrt{1 - \tilde{r}'(\omega) \tilde{T} \tilde{r}(\omega)}. \tag{42}$$

We focus on this normalized error $\mathcal{E}(\cdot)$ hereafter when $J < 10 - K/N$.

Due to the shift-invariance property (22), 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

$$\mathcal{E}_{\max} \triangleq \max_{\omega} \mathcal{E}(\omega) \tag{43}$$

usually occurs either at the midpoint between DFT samples or at the DFT samples themselves. (See Fig. 6 for examples.) Unfortunately this does not always hold, so we apply numerical methods to evaluate (43). We begin with the simplest case: uniform scaling factors ($s = 1$).

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 ($s = 1$). 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:

$$\mathcal{E}_{\max} \approx 0.75 \exp\left(-J\left[0.29 + 1.03\log\left(\frac{K}{N}\right)\right]\right). \tag{44}$$

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

To create Fig. 3, we used (40) because for large values of J and K/N, the matrix $C'C$ becomes very poorly conditioned, and (41) becomes numerically inaccurate. Using a truncated SVD to compute the pseudo-inverse of $C'C$ did not seem to help.

B. Multidimensional Case

Using (34) and (35), the 2-D error has the form

$$\mathcal{E}_{\text{2D}} = \sqrt{1 - \tilde{r}'_{\text{2D}} \tilde{T}_{\text{2D}} \tilde{r}_{\text{2D}}}$$
$$= \sqrt{1 - \tilde{r}'_2 \tilde{T}_2 \tilde{r}_2 \tilde{r}'_1 \tilde{T}_1 r_1}$$
$$= \sqrt{1 - (1 - \mathcal{E}_2^2)(1 - \mathcal{E}_1^2)} \leq \sqrt{\mathcal{E}_1^2 + \mathcal{E}_2^2}$$

where $\mathcal{E}_1$ and $\mathcal{E}_2$ denote the 1-D errors in (42). This gives an upper bound on the potential accuracy "penalty" in 2-D rela-
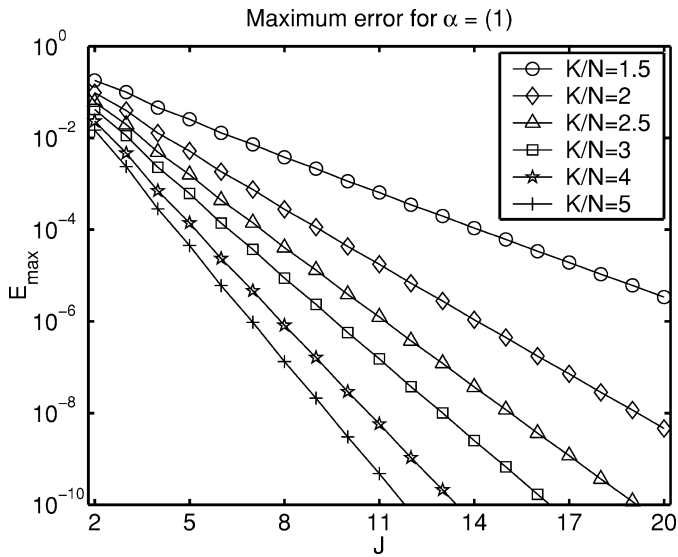
Fig. 3.   Maximum error $\mathcal{E}_{\max}$ of min-max interpolator with uniform scaling vector ($s = 1$) for various neighborhood sizes J and oversampling factors K/N.
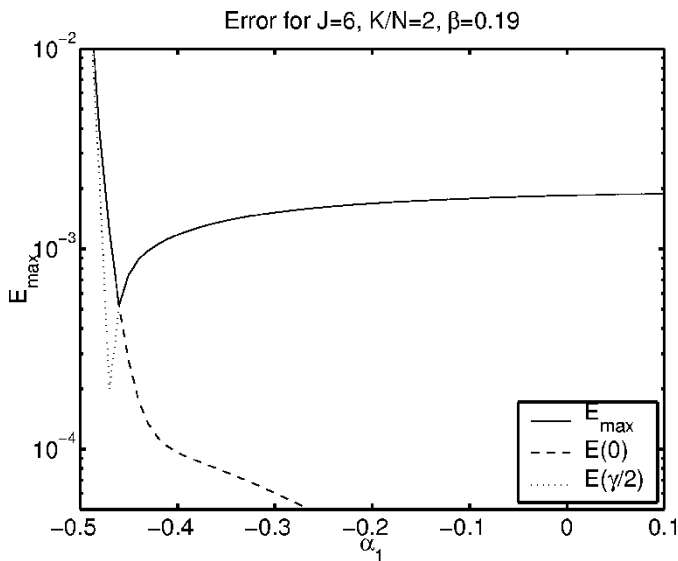

Fig. 4.   Maximum error $\mathcal{E}_{\max}$ as a function of α_1 for $L = 1$ and $\alpha_0 = 1$. Since the minimum is not at $\alpha_1 = 0$, uniform scaling factors are suboptimal.

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

TABLE II
COEFFICIENTS IN (28) OF CONVENTIONAL AND NUMERICALLY OPTIMIZED SCALING FACTORS FOR $K/N = 2$

L	J	β	$\boldsymbol{\alpha}$				$\mathcal{E}_{\mathbf{max}}$
0	6	0	(1) (uniform)				$2 \cdot 10^{-3}$
0	6	1/2	(0 1/2) (cosine)				$6 \cdot 10^{-3}$
1	6	0.19	(1 -0.46)				$5 \cdot 10^{-4}$
2	2	0.34	(1 -0.2	-0.04)			$5 \cdot 10^{-2}$
2	4	0.56	(1 -0.47	0.085)			$1 \cdot 10^{-3}$
2	6	0.43	(1 -0.57	0.14)			$1 \cdot 10^{-4}$
2	8	0.47	(1 -0.54	0.16)			$2 \cdot 10^{-5}$
2	10	0.43	(1 -0.57	0.185)			$6 \cdot 10^{-7}$
3	4	0.6339	(1 -0.5319	0.1522	-0.0199)		$3 \cdot 10^{-4}$
3	6	0.2254	(1 -0.6903	0.2138	-0.0191)		$1 \cdot 10^{-4}$

irrelevant.) Intuition suggests that for good interpolators, the s_ns should be fairly smooth, so a low-order expansion in (28) should be adequate. (This is consistent with the smooth choices that have been used in the literature, e.g., [34], [37], and [40].) Using the series expansion (28) 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 such extra effort is warranted only when one needs many NUFFTs for the same J and K/N. We also investigated

C. Choice of Scaling Factors

Both $\tilde{\boldsymbol{r}}$ and $\tilde{\boldsymbol{T}}$ in the error expression (42) depend on the choice of scaling vector $\boldsymbol{s}$, as seen in (31) and (32). 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_ns are
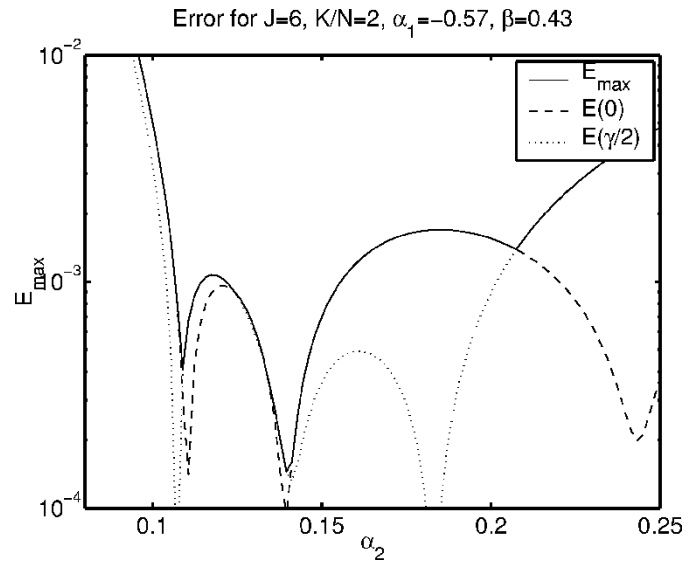
tive to 1-D. It also suggests that tensor products of good 1-D min-max interpolators should work well in higher dimensions; therefore, we can focus the efforts in optimizing $\boldsymbol{\alpha}$ and β on the 1-D case.

complex values for α_1 and found that the minimizer was always a real-valued α_1.

For $L = 2$, $J = 6$, and $K/N = 2$, we numerically minimized $\mathcal{E}_{\max}((1, \alpha_1, \alpha_2), \beta)$ over α_1, α_2, and β. The minimizer was $\boldsymbol{\alpha} = (1, -0.57, 0.14)$ and $\beta = 0.43$. Fig. 5 shows $\mathcal{E}_{\max}((1, -0.57, \alpha_2), 0.43)$ versus α_2. Again, in the neighborhood of the minimum, $\mathcal{E}_{\max}$ can be fairly sensitive to $\boldsymbol{\alpha}$.

Table II summarizes the optimized $\boldsymbol{\alpha}$s and βs 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 [40]. As acknowledged by Nguyen and Liu, "the cosine scaling factors … are by no mean[s] the "best" ones," which is a point that Fig. 6 confirms. We found in many such experiments (for a variety of Js and K/Ns) that uniform scaling factors yielded consistently lower errors than cosine scaling factors.[2]

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; therefore, there is little reason to "favor" the DFT sample locations.

The interpolators shown in Figs. 1 and 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.

Our emphasis here has been on worst-case error, and the error values given in Fig. 3 differ from those reported in [40]. This "discrepancy" has two explanations. First, we consider "Problem 2" in (2), whereas the figures in [40] are for "Problem 1." These problems may have different error properties. Second, the errors reported in [40] and related papers are for particular experiments involving pseudo-random data and sample locations; the characteristics of such data may 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 min-max interpolation. To enable comparisons, this section analyzes the worst-case error of conventional shift-invariant interpolation.

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

$$\hat{X}(\omega) = \sum_{k=0}^{K-1} Y_k \tilde{\psi}\left(\frac{\omega}{\gamma} - k\right) \qquad (45)$$

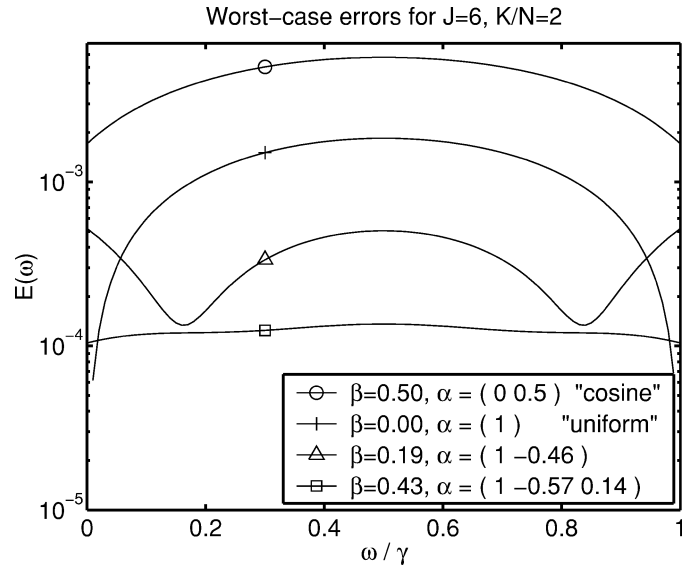[2]There is an error in the second to last equation on [40, p. 292] regarding uniform scaling factors.

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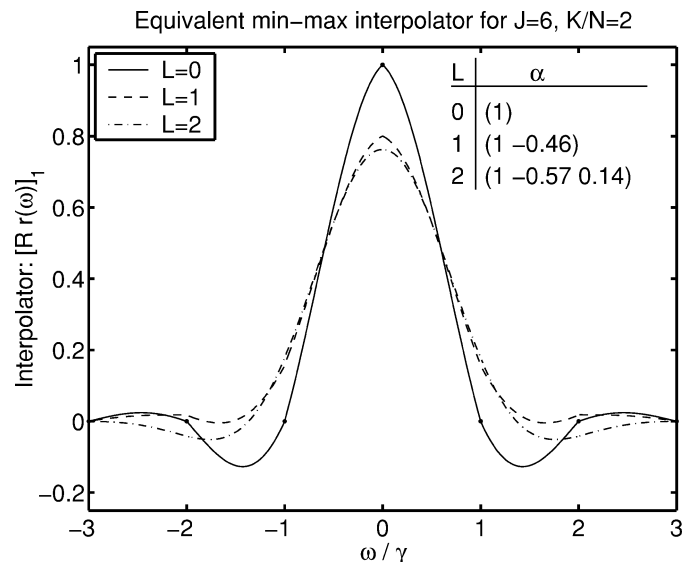
Fig. 7. Effective min-max interpolator for $J = 6$ and $K/N = 2$ for optimized $\boldsymbol{\alpha}$ and β.

where Y_k was defined in (3), and $\tilde{\psi}$ denotes the K-periodic and phase-modulated (cf. (5)) version of ψ:

$$\tilde{\psi}(\kappa) \triangleq \sum_{l=-\infty}^{\infty} e^{-\imath\gamma(\kappa-lK)\eta_0}\psi(\kappa - lK).$$

A. Min-Max Error Analysis

Mimicking (13), the error for interpolator (45) is

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

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

$$q_n(\omega) \triangleq \frac{1}{\sqrt{N}} \sum_{k=0}^{K-1} e^{\imath\gamma kn}\tilde{\psi}^{\star}\left(\frac{\omega}{\gamma} - k\right).$$

Akin to (19), the worst-case unit-norm signal is $\boldsymbol{x} = (\boldsymbol{S}'\boldsymbol{q} - \boldsymbol{b})/\|\boldsymbol{S}'\boldsymbol{q} - \boldsymbol{b}\|$, so the worst-case error for frequency ω, normalized by $1/\sqrt{N}$, is

$$\mathcal{E}(\boldsymbol{s}, \omega) = \left\| \boldsymbol{S}'\boldsymbol{q}(\omega) - \boldsymbol{b}(\omega) \right\|. \tag{46}$$

Expanding, an alternate expression is

$$\mathcal{E}^2(\boldsymbol{s}, \omega) = \frac{1}{N} \sum_{n=0}^{N-1} \left| s_n^\star \sqrt{N} q_n(\omega) - e^{\imath\omega n} \right|^2$$
$$= \frac{1}{N} \sum_{n=0}^{N-1} \left| s_n z_n \left(\frac{\omega}{\gamma} \right) - 1 \right|^2 \tag{47}$$

where

$$z_n(\rho) \triangleq e^{\imath\rho\gamma n} \sqrt{N} q_n^\star(\rho\gamma) = \sum_{k=0}^{K-1} e^{\imath\gamma(\rho-k)n} \tilde{\psi}(\rho - k). \tag{48}$$

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

$$z_n(\rho) = \sum_{k=0}^{K-1} e^{\imath\gamma(\rho-k)n} \tilde{\psi}(\rho - k)$$
$$= \sum_{j=-J/2+1}^{J/2} e^{\imath\gamma(\rho-j)(n-\eta_0)} \psi(\rho - j).$$

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

For a given interpolation kernel ψ, 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. One practical "do no harm" approach would be to minimize the worst-case error *at the DFT frequency locations*:

$$\min_{\boldsymbol{s} \in \mathbb{C}^N} \max_{\omega \in \Omega_K} \mathcal{E}(\boldsymbol{s}, \omega). \tag{49}$$

Considering (47), the solution to (49) is simply

$$s_n = \frac{1}{z_n(0)} = \frac{1}{\sum\limits_{k=-J/2}^{J/2-1} e^{\imath\gamma k(n-\eta_0)} \psi(k)}. \tag{50}$$

If the kernel $\psi(\cdot)$ satisfied the classical interpolation properties $\psi(0) = 1$ and $\psi(k) = 0$ for $k \neq 0$, then (50) would reduce to uniform scaling factors ($\boldsymbol{s} = \boldsymbol{1}$).

One calculates the worst-case error of conventional interpolators of the form (45) by substituting (50) into (47). Since (47) approaches a finite limit as $N \to \infty$, we again focus on this "large N" approximation.

With the choice (50), $\mathcal{E}(\omega) = 0$ 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 $\psi(\cdot)$ is Hermitian symmetric about zero, then $\mathcal{E}(\omega)$ has a stationary point at $\omega = \gamma/2$ for the choice (50). Lacking a proof, we compute numerically the maximum error $\mathcal{E}_{\max} = \max_\omega \mathcal{E}(\boldsymbol{s}, \omega)$.

B. Aliasing Error Analysis

The error formula (47) is convenient for computation but seems to provide little insight. Here, we summarize an alternate form for the error that is somewhat more intuitive, following related analyses of "gridding" methods, e.g., [25] and [41].

Since $\tilde{\psi}$ is K-periodic, it has a Fourier series expansion of the form

$$\tilde{\psi}(\kappa) = \sum_{n=-\infty}^{\infty} \frac{c_n}{K} e^{-\imath\gamma n\kappa}$$

(assuming sufficient regularity), where the c_ns are samples of the inverse Fourier transform of ψ:

$$c_n \triangleq \int_{-J/2}^{J/2} \tilde{\psi}(\kappa) e^{\imath\gamma n\kappa} d\kappa = \int_{-J/2}^{J/2} \psi(\kappa) e^{\imath\gamma(n-\eta_0)\kappa} d\kappa.$$

Substituting into (45)

$$\hat{X}(\omega) = \sum_{k=0}^{K-1} Y_k \left[\sum_{n=-\infty}^{\infty} \frac{c_n}{K} e^{-\imath\gamma n(\omega/\gamma - k)} \right]$$
$$= \sum_{n=-\infty}^{\infty} c_n e^{-\imath\omega n} \left[\frac{1}{K} \sum_{k=0}^{K-1} Y_k e^{\imath\gamma nk} \right]$$
$$= \sum_{n=-\infty}^{\infty} c_n e^{-\imath\omega n} x_{\{n\}_K} s_{\{n\}_K}$$
$$= \sum_{n=0}^{N-1} x_n s_n c_n e^{-\imath\omega n}$$
$$+ \sum_{n=0}^{N-1} x_n \left[s_n \sum_{l \neq 0} c_{n+lK} e^{-\imath\omega(n+lK)} \right]. \tag{51}$$

Viewed in this form, the natural choice for the scaling factors s_n is the following (assuming these c_ns are nonzero):

$$s_n = \frac{1}{c_n} = \frac{1}{\int_{-J/2}^{J/2} \psi(\kappa) e^{\imath\gamma(n-\eta_0)\kappa} d\kappa} \tag{52}$$

for $n = 0, \ldots, N-1$. For this choice, the error is

$$\left| \hat{X}(\omega) - X(\omega) \right| = \sum_{n=0}^{N-1} x_n \left[\frac{\sum\limits_{l \neq 0} c_{n+lK} e^{-\imath\omega(n+lK)}}{c_n} \right]$$
$$\leq \|\boldsymbol{x}\|_1 \max_{n \in \{0, \ldots, N-1\}} \frac{\sum\limits_{l \neq 0} |c_{n+lK}|}{|c_n|}.$$

For this error to be small, we want to choose ψ such that the Fourier series coefficients c_n are small for $n \notin \{0, \ldots, N-1\}$. Since ψ has finite support $[-J/2, J/2]$, the c_ns cannot all be zero, so one must choose ψ considering the usual time-frequency tradeoffs.

C. Comparisons of Min-Max to Conventional

The purpose of the preceding analysis was to enable a fair comparison of the min-max interpolator (33) with conventional interpolators (45) while using good scaling factors for the latter. The following subsections report comparisons with Dirichlet, Gaussian bell, and Kaiser-Bessel interpolators.

1) Apodized Dirichlet: 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. Using (43) and (47), Fig. 8 compares the maximum error for the min-max interpolator and for the truncated Dirichlet interpolator $I(\omega)\text{rect}(\omega/(\gamma J))$, for $K/N = 2$, where $\text{rect}(\cdot)$ is unity on $(-1/2, 1/2)$ and zero otherwise. Fig. 8 also shows the $\cos^3$-tapered Dirichlet interpolator proposed in [57] and [58]. Both uniform scaling factors and numerically optimized αs were used for the min-max case. Min-max interpolation can yield much less error than truncated or tapered Dirichlet interpolation. The seemingly minor differences in Fig. 1 can strongly affect maximum error!

2) Truncated Gaussian Bell: Many NUFFT papers have focused on truncated Gaussian bell interpolation using

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

For fair comparisons, for each J, we optimized the Gaussian bell width parameter σ using (47) by exhaustive search. We investigated both (50) and (52) as the scaling factors and found the latter to provide 10–45% lower maximum error; therefore, we focused on (52). Empirically the min-max width agreed closely with the approximation $\sigma \approx 0.31 * J^{0.52}$.

Fig. 9 compares the worst-case error of min-max interpolation and optimized Gaussian bell interpolation. 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. After optimizing the width σ for the Gaussian bell interpolator, we compute its scaling factors using (52). Then, we use ordinary least-squares linear regression with $L \approx 6$ in (28) to find a $\boldsymbol{\alpha}$ for (28) that closely matches the optimized Gaussian bell scaling factors. Then, we use that $\boldsymbol{\alpha}$ in (43) 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 1-D search over the Gaussian bell 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.)

Fig. 9 illustrates several important points. First, the min-max interpolator with simple uniform scaling factors has comparable error to the exhaustively-optimized Gaussian bell interpolator. Second, optimizing the scaling factors very significantly reduces the min-max interpolation error, outperforming both the Gaussian bell interpolator and the min-max interpolator with uniform scaling factors. Third, for $J \leq 6$, exhaustive optimization of $\boldsymbol{\alpha}$ with $L = 2$ yields comparable maximum error to the simpler least-squares fit (using $L = 6$) to the optimized Gaussian bell scaling factors (50), 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 $L > 2$.
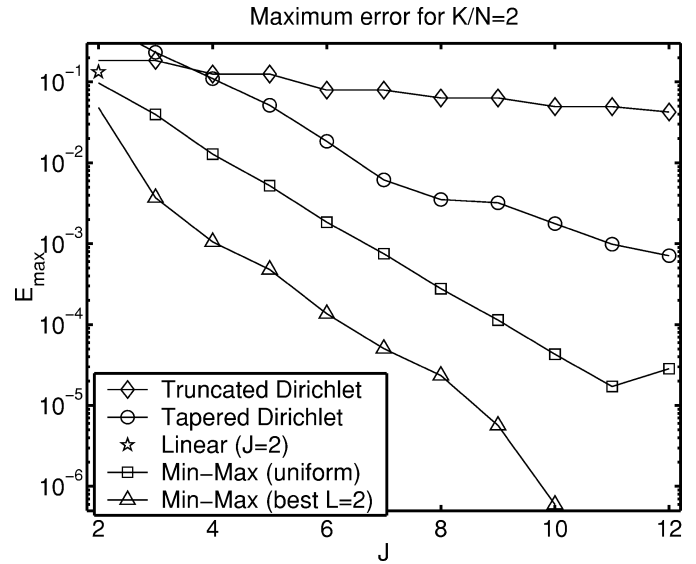

Fig. 8.   Maximum error $\mathcal{E}_{\max}$ of truncated Dirichlet interpolator, of $\cos^3$-tapered Dirichlet interpolator, of linear interpolator ($J = 2$), and of min-max interpolator for various neighborhood sizes J and for oversampling factor $K/N = 2$. Despite similarities in Fig. 1, the min-max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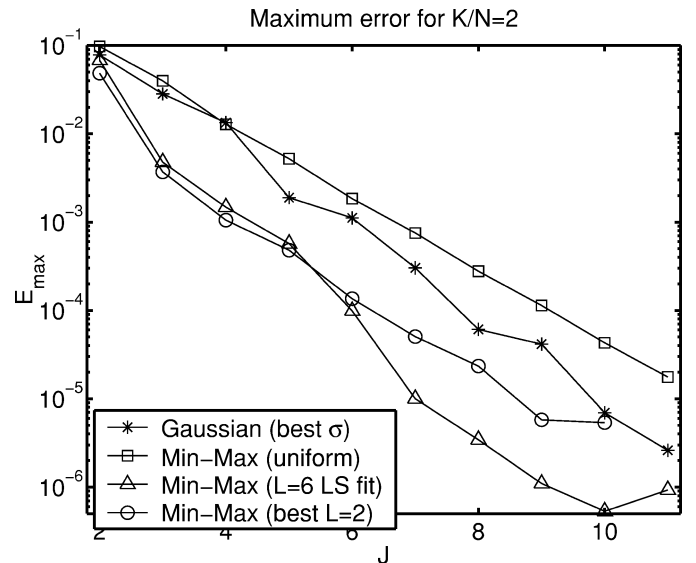

Fig. 9.   Maximum error $\mathcal{E}_{\max}$ of min-max interpolators and truncated Gaussian bell interpolator versus neighborhood size J for oversampling factor $K/N = 2$. For each J, the Gaussian bell width σ was optimized numerically by exhaustive search to minimize worst-case error. Three choices of scaling factors (s_ns) for the min-max method are shown: uniform, numerically optimized, and LS fit of (28) to optimized Gaussian bell (s_ns) given by (50).

3) Kaiser–Bessel: An alternative to the Gaussian bell interpolator is the generalized Kaiser-Bessel function [59], [60]

$$\psi(\kappa) = f_J^m(\kappa)\frac{I_m\left(\alpha f_J(\kappa)\right)}{I_m(\alpha)}$$

where I_m denotes the modified Bessel function of order m, and

$$f_J(\kappa) \triangleq \begin{cases} \sqrt{1 - \left(\frac{\kappa}{J/2}\right)^2}, & |\kappa| < \frac{J}{2} \\ 0, & \text{otherwise.} \end{cases}$$
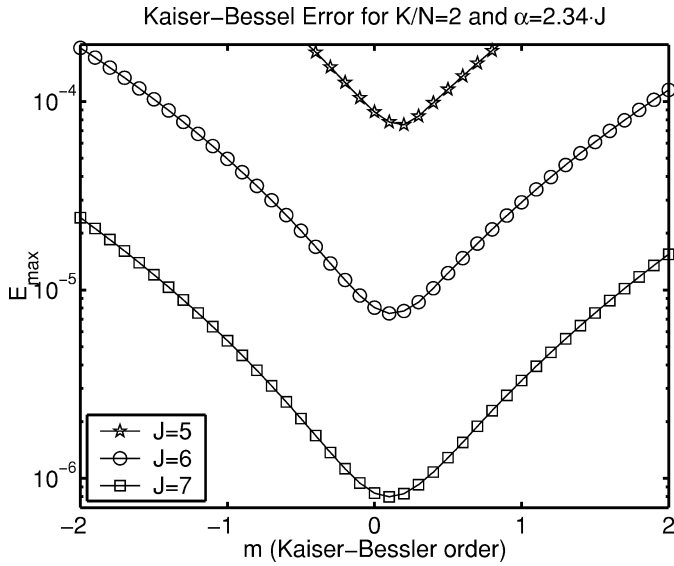
Fig. 10. Maximum error $\mathcal{E}_{\max}$ of Kaiser–Bessel interpolator versus order m for $\alpha = 2.34J$. Surprisingly, the minimum is near $m = 0$.
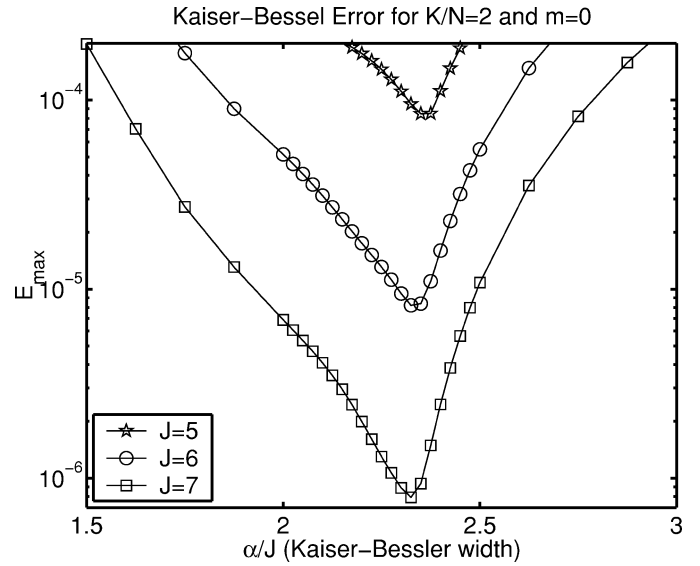

Fig. 11. Maximum error $\mathcal{E}_{\max}$ of Kaiser–Bessel interpolator versus width parameter α for $m = 0$.

The width of this function is related to the "shape parameter" α. This function is popular in "gridding" methods for imaging problems, e.g., [61], but has been largely ignored in the general NUFFT literature to our knowledge.

Again, for fair comparisons, we used (43) and (46) to optimize both the order m and α numerically to minimize the worst-case error. Initially, we had planned to use $m = 2$ since this provides continuity of the kernel and its first derivative at the endpoints $\kappa = \pm J/2$. However, we found numerically that the min-max optimal order is near $m = 0$. This property is illustrated in Fig. 10. Choosing $m = 0$ reduces the maximum error by a factor of more than 10 relative to the "conventional" $m = 2$ choice. For $m = 0$, we found that the optimal α was about $2.34J$ for $K/N = 2$. Fig. 11 shows examples.

For the scaling factors, we compared the "do no harm" choice (50) to the Fourier choice (52), i.e., $s_n = \Psi((n - \eta_0)/K)$, where [59]

$$\Psi(u) = \left(\frac{1}{2}\right)^m \pi^{d/2} \left(\frac{J}{2}\right)^d \alpha^m \frac{\Lambda(z(u))}{I_m(\alpha)}$$

where $d = 1$ (for the 1-D case), $\nu = d/2 + m$, $z(u) = \sqrt{(\pi J u)^2 - \alpha^2}$, and $\Lambda(z) = (z/2)^{-\nu} J_\nu(z)$, where J_ν denotes the Bessel function of the first kind of order ν. The Fourier choice (52), which is conventional in gridding methods, yielded about 25–65% lower errors than (50) for $m = 0$.

Fig. 12 compares the maximum errors of the (optimized) Kaiser-Bessel interpolator, the (optimized) Gaussian bell interpolator, and a few min-max interpolators. We investigated three choices of scaling factors: uniform, the numerically optimized choices for $L = 2$ shown in Table II, and a third case in which we used the scaling factors computed by least-squares fit of (28) with $L = 13$ and $\beta = 1$ to the Kaiser–Bessel scaling factors from (52).

As expected, the min-max interpolator yields lower errors than both the optimized Gaussian bell and the optimized Kaiser–Bessel interpolators. For the choices of scaling factors investigated here (particularly the least-squares fitting
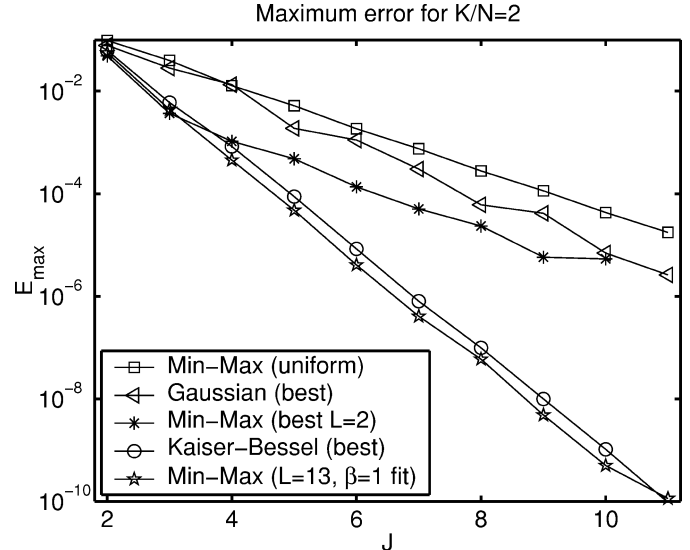

Fig. 12. Maximum error $\mathcal{E}_{\max}$ of min-max interpolators, truncated Gaussian bell interpolator (with numerically optimized width), and Kaiser–Bessel interpolator (with numerically optimized shape) versus neighborhood size J for oversampling factor $K/N = 2$. Three choices of scaling factors (s_ns) for the min-max method are shown: uniform, numerically optimized for $L = 2$, and LS fit of (28) to optimized Kaiser-Bessel s_ns given by (50).

approach), the reduction in error relative to the Kaiser–Bessel interpolator is 30–50% for $J \leq 10$. It is plausible that larger error reductions would be possible if a practical method for optimizing the scaling parameters (e.g., α for larger L) were found. Lacking such a method, it seems that the Kaiser-Bessel interpolator, with suitably optimized parameters, represents a very reasonable compromise between accuracy and simplicity.

From Fig. 12, one sees that $J = 9$ is sufficient for single-precision (10^{-8}) accuracy in the min-max sense. (Practical problems are usually not worst case, so $J = 9$ is probably overkill.) For $J = 9$ and $K/N = 2$, using Matlab's cputime command, we found that the interpolation step (with precomputed coefficients) required roughly twice the CPU time required by the oversampled FFT step.
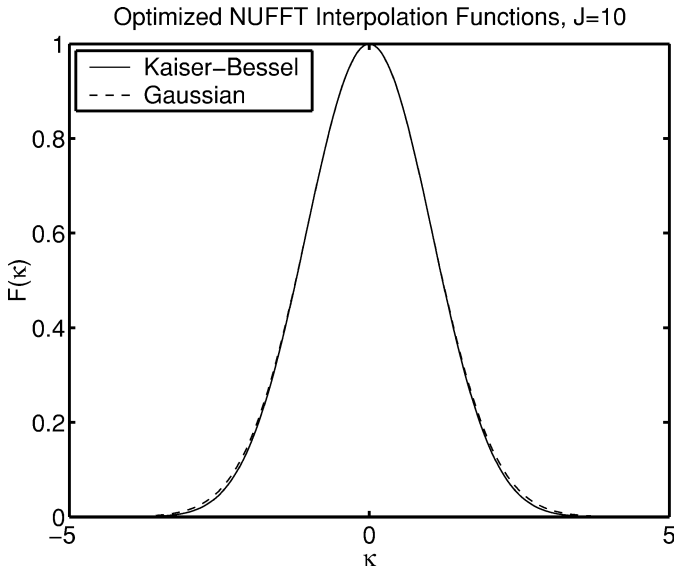
Fig. 13. Optimized Kaiser–Bessel ($m = 0$, $\alpha = 2.34 J$) and Gaussian bell ($\sigma = 1.04$) interpolation kernels for $J = 10$.

Fig. 13 compares the shape of the optimized Kaiser–Bessel and Gaussian bell interpolation kernels. Superficially, the kernels appear to be very similar, but $J = 10$ can provide errors on the order of 10^{-9} with the Kaiser–Bessel kernel; therefore, even subtle departures in the kernel shape may drastically affect the interpolation error.

VI. 2-D EXAMPLE

To illustrate the accuracy of the NUFFT method in a practical context, we considered the classical 128×128 Shepp–Logan image [62], [63]. We generated $10\,000$ random frequency locations (ω_ms) in $(-\pi, \pi) \times (-\pi, \pi)$ and computed the 2-D FT *exactly* (to within double precision in Matlab) and with the min-max 2-D NUFFT method with $J = 6$ and $K/N = 2$. The relative percent error

$$\frac{\max_m \left| \hat{X}\left(\omega_m\right) - X\left(\omega_m\right) \right|}{\max_m \left| X\left(\omega_m\right) \right|} \times 100\%$$

was less than 0.14% when uniform scaling factors were used and less than 0.011% when the optimized scaling factors for $L = 2$ in Table II were used and less than $2.1 \cdot 10^{-4}\%$ when the scaling factors were based on least-squares fits to Kaiser–Bessel scaling factors, as described in Section V-C3. These orders-of-magnitude error reductions are consistent with the reductions shown in Figs. 3 and 12 and confirm 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 four orders of magnitude.

VII. DISCUSSION

This paper has presented a min-max framework for the interpolation step of NUFFT methods. This criterion leads to a novel high-accuracy interpolator and aids in the optimization of the shape parameters of conventional interpolators as well. These optimized interpolators for the NUFFT 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 when expressed using an appropriate Fourier series. This optimization led to considerably reduced errors compared with the previously considered uniform and cosine scaling factors [40]. Optimizing the scaling factors further remains an challenging open problem; perhaps iterations like those used in gridding [61], [64] are required.

Based on the results in Fig. 12, we recommend the following strategies. In applications where precomputing and storing the interpolation coefficients is practical and where multiple NUFFTs of the same size are needed, such as for iterative reconstruction in the imaging problems that motivated our work, using the proposed min-max approach with scaling factors fit to the Kaiser–Bessel s_ns provides the highest accuracy of the methods investigated and, therefore, allows the reduction of the neighborhood size J and, hence, minimization of computation per iteration. On the other hand, if memory constraints preclude storing the interpolation coefficients, then based on Figs. 9 and 12, we see that a Gaussian bell or Kaiser–Bessel interpolator, suitably optimized, provides accuracy comparable to the min-max interpolator *if* one is willing to use a modestly larger neighborhood J.

Alternatively, one could finely tabulate any of these interpolators and use table lookup (with polynomial interpolation) to compromise between computation and storage. The accuracy of such approaches requires investigation.

One remaining open problem is that the $J \times J$ matrix $\boldsymbol{C}'\boldsymbol{C}$ becomes ill-conditioned as J increases beyond about 10 (likewise for $\boldsymbol{C}'\boldsymbol{SS}'\boldsymbol{C}$, 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 (29) and (30) appear may depend on the norm.

Another possible generalization would be to use *different* scaling factors for the two FFTs in (37). It is unclear how much, if any, error reduction this generalization could provide, but the additional computational cost would be very minimal.

Although detailed analyses of the errors associated with NUFFT methods for "Problem 1" are available, e.g., [41], to our knowledge, no provably optimal interpolator has been found for Problem 1; therefore, this remains an interesting open problem.

Finally, one could extend the min-max approach to related transforms such as Hankel and cosine [12], [65].

REFERENCES

[1] A. V. Oppenheim, D. Johnson, and K. Steiglitz, "Computation of spectra with unequal resolution using the fast Fourier transform," *Proc. IEEE*, vol. 59, 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, pp. 917–925, Aug. 1983.

[3] D. C. Munson and J. L. Sanz, "Image reconstruction from frequency-offset Fourier data," *Proc. IEEE*, vol. 72, pp. 661–669, June 1984.

[4] H. Choi and D. C. Munson, "Direct-Fourier reconstruction in tomography and synthetic aperture radar," *Int. J. Imaging Syst. Tech.*, vol. 9, no. 1, pp. 1–13, 1998.

[5] E. Larsson, P. Stoica, and J. Li, "SAR image construction from gapped phase-history data," in *Proc. IEEE Int. Conf. Image Process.*, vol. 3, 2001, pp. 608–611.

[6] Y. Wu and D. Munson, "Multistatic passive radar imaging using the smoothed pseudo Wigner-Ville distribution," in *Proc. IEEE Int. Conf. Image Process.*, vol. 3, 2001, pp. 604–607.

[7] E. J. Candes and D. L. Donoho, "Ridgelets: A key to higher-dimensional intermittency?," *Philos. Trans. R. Soc. Lond. A, Math. Phys. Eng. Sci.*, vol. 357, no. 1760, pp. 2495–2509, Sept. 1999.

[8] Q. H. Liu and N. Nguyen, "An accurate algorithm for nonuniform fast Fourier transforms (NUFFTs)," *IEEE Microw. Guided Wave Lett.*, vol. 8, 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 *Proc. IEEE Geosci. Remote Sensing Symp.*, vol. 1, 1998, pp. 288–290.

[10] Q. H. Liu and X. Y. Tang, "Iterative algorithm for nonuniform inverse fast Fourier transform," *Electron. Lett.*, vol. 34, no. 20, pp. 1913–1914, Oct. 1998.

[11] X. M. Xu and Q. H. Liu, "The conjugate-gradient nonuniform fast Fourier transform (CG-NUFFT) method for one- and two-dimensional media," *Microw. Opt. Technol. Lett.*, vol. 24, no. 6, pp. 385–389, 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 Trans. Geosci. Remote Sensing*, vol. 38, pp. 1551–1560, July 2000.

[13] E. Angelidis and J. E. Diamessis, "A novel method for designing FIR digital filters with nonuniform frequency samples," *IEEE Trans. Signal Processing*, vol. 42, pp. 259–267, Feb. 1994.

[14] S. Bagchi and S. K. Mitra, "The nonuniform discrete Fourier transform and its applications in filter design. I—1-D," *IEEE Trans. Circuits Syst. II*, vol. 43, pp. 422–433, June 1996.

[15] A. Makur and S. K. Mitra, "Warped discrete-Fourier transform: Theory and applications," *IEEE Trans. Circuits Syst. I*, vol. 48, pp. 1086–1093, Sept. 2001.

[16] S. Bagchi and S. Mitra, *The Nonuniform Discrete Fourier Transform and its Applications in Signal Processing*. Boston, MA: Kluwer, 1999.

[17] K. P. Pruessmann, M. Weiger, M. B. Scheidegger, and P. Boesiger, "SENSE: Sensitivity encoding for fast MRI," *Magn. Resonance Med.*, vol. 42, pp. 952–962, 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. Int. Soc. Mag. Res. Med.*, 2001, p. 763.

[19] ——, "Iterative MR image reconstruction using sensitivity and inhomogeneity field maps," in *Proc. Int. Soc. Mag. Res. Med.*, 2001, p. 771.

[20] B. P. Sutton, D. Noll, and J. A. Fessler, "Fast, iterative, field-corrected image reconstruction for MRI," *IEEE Trans. Med. Imaging*, 2002, submitted for publication.

[21] C. R. Crawford and Elscint, "System for reprojecting images using transform techniques," US Patent 4 616 318, 1986.

[22] 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. Imag. II: Image Formation, Detection, Process., Interpretation*, 1988, pp. 311–318.

[23] C. W. Stearns, D. A. Chesler, and G. L. Brownell, "Three-dimensional image reconstruction in the Fourier domain," *IEEE Trans. Nuclear Sci.*, vol. 34, pp. 374–378, Feb. 1987.

[24] ——, "Accelerated image reconstruction for a cylindrical positron tomograph using Fourier domain methods," *IEEE Trans. Nuclear Sci.*, vol. 37, pp. 773–777, Apr. 1990.

[25] H. Schomberg and J. Timmer, "The gridding method for image reconstruction by Fourier transformation," *IEEE Trans. Med. Imaging*, vol. 14, pp. 596–607, Sept. 1995.

[26] S. Matej and R. M. Lewitt, "3-FRP: Direct Fourier reconstruction with Fourier reprojection for fully 3-D PET," *IEEE Trans. Nuclear Sci.*, vol. 48, pp. 1378–1385, Aug. 2001.

[27] J. A. Fessler and B. P. Sutton, "A min-max approach to the multidimensional nonuniform FFT: Application to tomographic image reconstruction," in *Proc. IEEE Int. Conf. Image Process.*, vol. 1, 2001, pp. 706–709.

[28] J. A. Fessler. (2001) Iterative tomographic image reconstruction using nonuniform fast Fourier transforms. Commun. and Signal Process. Lab, Dept. Elect. Eng. Comput. Sci., Univ. of Michigan, Ann Arbor, MI. [Online]. Available: http://www.eecs.umich.edu/~fessler

[29] T. Strohmer, "Numerical analysis of the nonuniform sampling problem," *J. Comput. Appl. Math.*, vol. 122, no. 1–2, pp. 297–316, Oct. 2000.

[30] G. Beylkin, "On the fast Fourier transform of functions with singularities," *Appl. Comput. Harmonic Ana.*, vol. 2, no. 4, pp. 363–381, Oct. 1995.

[31] S. Bagchi and S. K. Mitra, "The nonuniform discrete Fourier transform and its applications in filter design. II—2-D," *IEEE Trans. Circuits Syst. II*, vol. 43, pp. 434–444, June 1996.

[32] G. Evangelista and S. Cavaliere, "Discrete frequency warped wavelets: Theory and applications," *IEEE Trans. Signal Processing*, vol. 46, pp. 874–885, Apr. 1998.

[33] N. I. Cho and S. K. Mitra, "Warped discrete cosine transform and its application in image compression," *IEEE Trans. Circuits Syst. Vid. Technol.*, vol. 10, pp. 1364–1373, Dec. 2000.

[34] A. Dutt and V. Rokhlin, "Fast Fourier transforms for nonequispaced data," *SIAM J. Sci. Comput.*, vol. 14, no. 6, pp. 1368–1393, Nov. 1993.

[35] C. Anderson and M. D. Dahleh, "Rapid computation of the discrete Fourier transform," *SIAM J. Sci. Comput.*, vol. 17, no. 4, pp. 913–919, July 1996.

[36] A. Dutt and V. Rokhlin, "Fast Fourier transforms for nonequispaced data, II," *Appl. Comput. Harmonic Anal.*, vol. 2, pp. 85–100, 1995.

[37] G. Steidl, "A note on the fast Fourier transforms for nonequispaced grids," *Adv. Comput. Math.*, vol. 9, no. 3, pp. 337–352, 1998.

[38] A. F. Ware, "Fast approximate Fourier transforms for irregularly spaced data," *SIAM Rev.*, vol. 40, no. 4, pp. 838–856, Dec. 1998.

[39] A. J. W. Duijndam and M. A. Schonewille, "Nonuniform fast Fourier transform," *Geophys.*, vol. 64, no. 2, pp. 539–551, Mar. 1999.

[40] N. Nguyen and Q. H. Liu, "The regular Fourier matrices and nonuniform fast Fourier transforms," *SIAM J. Sci. Comput.*, vol. 21, no. 1, pp. 283–293, 1999.

[41] D. Potts, G. Steidl, and M. Tasche, "Fast Fourier transforms for nonequispaced data: A tutorial," in *Modern Sampling Theory: Mathematics and Application*, J. J. B. P. Ferreira, Ed. Boston, MA: Birkhauser, 2000, pp. 253–274.

[42] W. H. Press and G. B. Rybicki, "Fast algorithm for spectral analysis of unevenly sampled data," *Astrophys. J.*, vol. 338, pp. 227–280, Mar. 1989.

[43] T. W. Parks and D. P. Kolba, "Interpolation minimizing maximum normalized error for band-limited signals," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. ASSP–26, pp. 381–384, Aug. 1978.

[44] D. S. Chen and J. P. Allebach, "Analysis of error in reconstruction of two-dimensional signals from irregularly spaced samples," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. ASSP–35, pp. 173–180, Feb. 1987.

[45] R. G. Shenoy and T. W. Parks, "An optimal recovery approach to interpolation," *IEEE Trans. Signal Processing*, vol. 40, pp. 1987–1992, Aug. 1992.

[46] N. P. Willis and Y. Bresler, "Norm invariance of minimax interpolation," *IEEE Trans. Inform. Theory*, vol. 38, pp. 1177–1181, May 1992.

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

[48] H. Choi and D. C. Munson, "Analysis and design of minimax-optimal interpolators," *IEEE Trans. Signal Processing*, vol. 46, pp. 1571–1579, June 1998.

[49] D. D. Muresan and T. W. Parks, "Optimal recovery approach to image interpolation," in *Proc. IEEE Int. Conf. Image Process.*, vol. 3, 2001, pp. 848–851.

[50] T. W. Parks and J. H. McClellan, "Chebyshev approximation for nonrecursive digital filters with linear phase," *IEEE Trans. Circuit Theory*, vol. 19, no. CT–2, pp. 189–199, Mar. 1972.

[51] R. E. Crochiere and L. R. Rabiner, *Multirate Digital Signal Processing*. Englewood Cliffs, NJ: Prentice-Hall, 1983.

[52] S. K. Mitra, *Digital Signal Procesing: A Computer-Based Approach*, 2nd ed. New York: McGraw-Hill, 2001.

[53] M. Tabei and M. Ueda, "Backprojection by upsampled Fourier series expansion and interpolated FFT," *IEEE Trans. Image Processing*, vol. 1, pp. 77–87, Jan. 1992.

[54] A. J. W. Duijndam, M. A. Schonewille, and C. O. H. Hindriks, "Reconstruction of band-limited signals, irregularly sampled along one spatial direction," *Geophys.*, vol. 64, no. 2, pp. 524–538, Mar. 1999.

[55] B. Lee and A. E. Yagle, "A sensitivity measure for image reconstruction from irregular 2-D DTFT samples," in *Proc. IEEE Conf. Acoust. Speech Signal Process.*, vol. 4, 2002, pp. 3245–3248.

[56] K. P. Pruessmann, M. Weiger, P. Börnert, and P. Boesiger, "Advances in sensitivity encoding with arbitrary k-space trajectories," *Magn. Resonance Med.*, vol. 46, no. 4, pp. 638–651, Oct. 2001.

[57] M. Magnusson, P.-E. Danielsson, and P. Edholm, "Artefacts and remedies in direct Fourier tomographic reconstruction," in *Proc. IEEE Nuc. Sci. Symp. Med. Imag. Conf.*, vol. 2, 1992, pp. 1138–1340.

[58] S. Lanzavecchia and P. L. B., "A bevy of novel interpolating kernels for the Shannon reconstruction of high-bandpass images," *J. Visual Commun. Imag. Represent.*, vol. 6, no. 2, pp. 122–131, June 1995.

[59] R. M. Lewitt, "Multidimensional digital image representations using generalized Kaiser-Bessel window functions," *J. Opt. Soc. Amer. A*, vol. 7, no. 10, pp. 1834–1846, Oct. 1990.

[60] ——, "Alternatives to voxels for image representation in iterative reconstruction algorithms," *Phys. Med. Biol.*, vol. 37, no. 3, pp. 705–716, 1992.

[61] J. I. Jackson, C. H. Meyer, D. G. Nishimura, and A. Macovski, "Selection of a convolution function for Fourier inversion using gridding," *IEEE Trans. Med. Imaging*, vol. 10, pp. 473–478, Sept. 1991.

[62] L. A. Shepp and B. F. Logan, "The Fourier reconstruction of a head section," *IEEE Trans. Nuclear Sci.*, vol. NS-21, pp. 21–43, June 1974.

[63] A. C. Kak and M. Slaney, *Principles of Computerized Tomographic Imaging*. New York: IEEE Press, 1988.

[64] H. Sedarat and D. G. Nishimura, "On the optimality of the gridding reconstruction algorithm," *IEEE Trans. Med. Imaging*, vol. 19, pp. 306–317, Apr. 2000.

[65] Q. H. Liu and Z. Q. Zhang, "Nonuniform fast hankel transform (NUFHT) algorithm," *Appl. Opt.*, vol. 38, no. 32, pp. 6705–6708, Nov. 1999.

Jeffrey A. Fessler (SM'00) received the B.S.E.E. degree from Purdue University, West Lafayette, IN, in 1985, the M.S.E.E. degree from Stanford University, Stanford, CA, in 1986, and the M.S. degree in statistics in 1989 and the Ph.D. degree in electrical engineering in 1990, also from Stanford University.

From 1985 to 1988, he was a National Science Foundation Graduate Fellow at Stanford. From 1991 to 1992, he was a Department of Energy Alexander Hollaender Post-Doctoral Fellow with the Division of Nuclear Medicine, University of Michigan, Ann Arbor. From 1993 to 1995, he was an Assistant Professor in nuclear medicine and the Bioengineering Program. Since 1995, he has been with the Department of Electrical Engineering and Computer Science, where he is an Associate Professor. His research interests are in statistical aspects of imaging problems.

Dr. Fessler serves as an associate editor for IEEE TRANSACTIONS ON MEDICAL IMAGING and has served as an associate editor for the IEEE TRANSACTIONS ON IMAGE PROCESSING and the IEEE SIGNAL PROCESSING LETTERS. He was technical program co-chair of the 2002 IEEE International Symposium on Biomedical Imaging.

Bradley P. Sutton (M'02) received the B.S. degree in general engineering from the University of Illinois at Urbana-Champaign in 1998 and the M.S. degrees in electrical engineering and biomedical engineering from the University of Michigan, Ann Arbor, in 2001.

His current research interests are in iterative MR image reconstruction methods including the effects of field inhomogeneities and coil sensitivities.

Mr. Sutton is a member of ISMRM.