Fourier reconstruction of univariate piecewise-smooth functions from non-uniform spectral data with exponential convergence rates[☆]

Rodrigo B. Platte[a,*], Alexander J. Gutierrez[b], Anne Gelb[a]

[a] *School of Mathematical and Statistical Sciences, Arizona State University, Tempe, AZ 85287-1804, USA*
[b] *Department of Mathematics, University of Minnesota, Minneapolis, MN 55455, USA*

Abstract

Reconstruction of piecewise smooth functions from non-uniform Fourier data arises in sensing applications such as magnetic resonance imaging (MRI). This paper presents a new method that uses edge information to recover the Fourier transform of a piecewise smooth function from data that is sparsely sampled at high frequencies. The approximation is based on a combination of polynomials multiplied by complex exponentials. We obtain super-algebraic convergence rates for a large class of functions with one jump discontinuity, and geometric convergence rates for functions that decay exponentially fast in the physical domain when the derivatives satisfy a certain bound. Exponential convergence is also proved for piecewise analytic functions of compact support. Our method can also be used improve initial jump location estimates, which are calculated from the available Fourier data, through an iterative process. Finally, if the Fourier transform is approximated at integer values, then the IFFT can be used to reconstruct the underlying function. Post-processing techniques, such as spectral reprojection, can then be used to reduce Gibbs oscillations.

Keywords: Resampling, non-uniform Fourier data, edge detection, Chebyshev polynomials, spectral convergence.

1. Introduction

Image reconstruction from Fourier data occurs naturally in several signal processing applications. In some of these applications it is advantageous to sample the data non-uniformly. For example, non-Cartesian two dimensional sampling patterns have been developed to reduce motion artifacts and simplify the collection process in magnetic resonance imaging (MRI) [1–5]. Of particular interest is the spiral sampling trajectory ([4] and references therein) in which Fourier data is densely sampled at the lower frequencies and gradually becomes more sparsely sampled at the higher frequencies. Reconstruction from non-uniform sampling patterns is not straightforward because the corresponding series does not typically constitute a basis for functions of practical interest [6, 7]. This problem is generally treated from a pragmatic perspective [4, 8], and typically involves mapping the non-harmonic Fourier data to its corresponding integer coefficients (e.g., via interpolation). Despite many algorithmic advances, such as the development of the non-uniform FFT [9–12], the mathematical consequences of non-uniform collection are often treated as secondary issues in the subsequent processing. Two commonly used algorithms in the MR imaging community, convolutional gridding [8] and uniform resampling [4], both generate non-physical oscillations throughout the domain as a result of resampling [6, 7, 10, 11]. In addition, since the underlying images in these applications are often only piecewise smooth, the reconstruction also suffers from the Gibbs phenomenon, manifested by spurious

[☆] This work is supported in part by NSF-DMS-FRG 0652833, NSF-DMS 1216559, and NSF-DMS-CSUMS 0703587.
[*] Corresponding author
Email addresses: rbp@asu.edu (Rodrigo B. Platte), alexg@math.umn.edu (Alexander J. Gutierrez), anne.gelb@asu.edu (Anne Gelb)

oscillations near the jump discontinuities and a reduction of the overall convergence rate to first order [13]. Filtering helps to reduce the Gibbs phenomenon [14], but does very little to mollify the oscillations caused by resampling [6]. High order post-processing techniques such as spectral reprojection can eliminate the Gibbs oscillations [13–15]. Moreover, when the underlying image is assumed to not have much variation, these methods have the added advantage of using only the (accurate) low frequency information, while the inaccurate high frequency modes are mostly suppressed. However, for functions with more variation, accurate (uniform) high frequency Fourier coefficients are also required, and so when starting with non-uniform Fourier data, the resampling error still dominates [6]. Finally, since accurate internal edge information is also critical to the success of high order post-processing methods that eliminate the Gibbs phenomenon, we are motivated to create both better resampling methods and better edge detection techniques given non-uniform Fourier data.

In this paper we develop a method to recover Fourier transforms of piecewise smooth functions from highly non-uniform spectral data in one dimension. We will refer to our method as the *polynomial resampling method* (PRM), since a polynomial expansion is used to resample non-uniform Fourier data, usually onto uniform nodes, so that the IFFT can be applied. Fourier transforms of discontinuous functions are oscillatory in nature with slow decay at high frequencies, and the PRM exploits an expansion based on jump locations that circumvents this difficulty. More specifically, because the frequency of oscillations is determined by the jump locations, we are able to produce a highly accurate decomposition that takes the form of complex exponentials multiplied by polynomial series expansions. The complex exponentials depend on the jump locations ("edges") and represent the oscillatory behavior, while the polynomial part captures the non-oscillatory contribution. A change of variables is used to transplant the problem from the ray to a bounded interval, which allows the representation of the transform to be accurate at the ends of the spectrum, even when there are very few samples in the high frequency part of the domain.

For functions with one jump, we show that sampling in the Fourier domain at the reciprocal values of Chebyshev nodes yields highly accurate transform approximations. For piecewise functions with multiple jumps we prove that if each piece in physical space is C^∞, then convergence is super-algebraic for a large class of functions. With additional bounds on the growth of derivatives, we are also able to derive geometric rates of convergence. Ideal sampling patterns are less clear for functions with multiple jump discontinuities, but our results indicate that the PRM is robust for sampling schemes that are denser at lower frequencies, such as logarithmic sampling distributions, which is the one dimensional analogue to spiral sampling patterns. For functions that are piecewise analytic with compact support, we prove that this expansion can also converge geometrically and present numerical experiments supporting these estimates.

When (non-uniform) Fourier data is available but no *a priori* information is known about the location of the edges, we combine our approximation scheme with a Fourier data based edge detector. The PRM can also be used to improve the accuracy of the approximation of the true edges yielded by these edge detection methods, which is useful in its own right in some contexts. Specifically, we use the scheme in [16] to obtain an initial estimate of the jump locations and then use the residual in the recovered Fourier transform as the objective function to improve edge locations. Finally, we evaluate the recovered transform at integer frequencies so that the inverse FFT can be used to recover the underlying function, and Gibbs suppression post-processing techniques, such as spectral reprojection, [13–15], can be directly applied.

This paper is organized in two main parts: Section 2 addresses the approximation problem in Fourier space, i.e., the recovery of the transform from spectral data, while Section 3 introduces the PRM algorithm and discusses its implementation. Numerical results that corroborate the theory are presented in both sections. We compare the PRM with the uniform resampling and convolutional gridding methods in Section 4. Final remarks are presented in Section 5.

2. Approximation of Fourier Transforms Using Edge Information

2.1. Functions with a Single Jump Discontinuity

Let $f \in L^1(\mathbb{R})$ be a bounded and piecewise smooth function. We define the jump function $[f] : \mathbb{R} \to \mathbb{R}$ of f as

$$[f](x) = \lim_{t \to x^+} f(t) - \lim_{t \to x^-} f(t), \tag{1}$$

that is, $[f](x) = 0$ in smooth regions, while at points of discontinuity, $[f]$ takes the value of the jump.

For a function with m piecewise smooth derivatives, we can relate the jump function (1) to the Fourier transform,

$$\hat{f}(\omega) := \int_{-\infty}^{\infty} f(x) e^{-i\omega x} dx, \tag{2}$$

using integration by parts. For ease of presentation, suppose that f has a single jump discontinuity at a known location ξ such that, for $\omega \neq 0$,

$$\begin{aligned}
\hat{f}(\omega) &= \int_{-\infty}^{\xi^-} f(x) e^{-i\omega x} dx + \int_{\xi^+}^{\infty} f(x) e^{-i\omega x} dx \\
&= \frac{f(\xi^-)}{-i\omega} e^{-i\omega\xi} - \frac{f(\xi^+)}{-i\omega} e^{-i\omega\xi} + \frac{1}{i\omega} \int_{\mathbb{R}\setminus\xi} f'(x) e^{-i\omega x} dx \\
&= \frac{[f](\xi)}{i\omega} e^{-i\omega\xi} + \frac{1}{i\omega} \int_{\mathbb{R}\setminus\xi} f'(x) e^{-i\omega x} dx,
\end{aligned} \tag{3}$$

where $\int_{\mathbb{R}\setminus\xi} f(x)dx = \lim_{\varepsilon\to 0} \int_{-\infty}^{\xi-\varepsilon} f(x)dx + \int_{\xi+\varepsilon}^{\infty} f(x)dx$. After m additional iterations of this process we obtain

$$\begin{aligned}
\hat{f}(\omega) &= \sum_{\ell=1}^{m} \frac{[f^{(\ell-1)}](\xi)}{(i\omega)^\ell} e^{-i\omega\xi} + \frac{1}{(i\omega)^m} \int_{\mathbb{R}\setminus\xi} f^{(m)}(x) e^{-i\omega x} dx \\
&= e^{-i\xi\omega} \frac{1}{\omega} \sum_{\ell=1}^{m} \frac{\lambda_\ell}{\omega^{\ell-1}} + \epsilon_m(\omega),
\end{aligned} \tag{4}$$

with the assumption that the indefinite integrals above remain bounded after differentiation. Here we have defined $\lambda_\ell = [f^{(\ell-1)}](\xi)(-i)^\ell$ and $\epsilon_m(\omega)$ as the contribution of the remaining integral. Finally, by allowing $s(\omega) = \frac{1}{\omega}$ we have

$$\hat{f}(\omega) = \hat{f}\left(\frac{1}{s}\right) = e^{-i\xi/s} s \sum_{\ell=1}^{m} \lambda_\ell s^{\ell-1} + \epsilon_m(1/s). \tag{5}$$

Using the expansion above as motivation, we introduce the approximation

$$\hat{f}\left(\frac{1}{s}\right) \approx \tilde{f}\left(\frac{1}{s}\right) := e^{-i\xi/s} s P_d(s), \tag{6}$$

where P_d is a polynomial of degree at most d. Assuming f is real-valued, we focus on $s \geq 0$, since in this case $\hat{f}(-w) = \overline{\hat{f}(w)}$. In particular, we will use (6) to approximate $\hat{f}$ in $[\omega_{min}, \infty)$, with $\omega_{min} > 0$ defined so as to be the minimum of the sampled non-zero frequencies.

The benefit of using this technique over current methodologies, such as convolutional gridding or uniform resampling, [6, 8, 10, 11], lies in the extraction of the oscillatory content, $e^{-i\xi\omega}$, from the Fourier transform. Because of this extraction, we are able to approximate the remaining (non-oscillatory) part of the Fourier transform with a polynomial. This can be illustrated using the following function:

$$f_1(x) = \begin{cases} e^{-(x-2)} & x \geq 2 \\ 0 & x < 2. \end{cases}$$

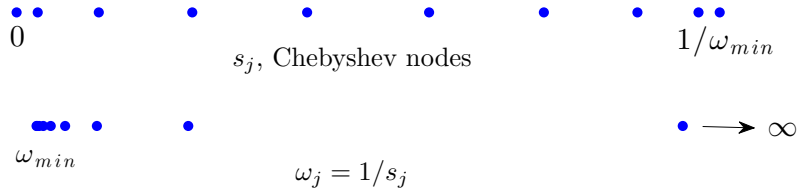

Figure 1: Chebyshev nodes and their reciprocals, as defined in (10) with $\omega_{max} = \infty$.

Its Fourier transform is given by $\hat{f}_1(w) = e^{-2iw}/(1+iw)$ so that $e^{2i/s}\hat{f}_1(1/s) = s/(s+i)$. The latter function is not only analytic for all s in $[0, 1/\omega_{min}]$ but also free of oscillations. In this case, it is possible to recover $\hat{f}_1$ very accurately from a polynomial approximation of $e^{i\xi/s}\hat{f}_1(1/s)$.

2.2. Chebyshev Interpolation

We now assume that we are given a finite vector of non-uniformly distributed discrete values of the Fourier transform, $\hat{f}(\boldsymbol{\omega})$ for $\omega \in [\omega_{min}, \omega_{max}]$. We can approximate (6) for given $\{\hat{f}(\omega_j)\}_{j=1}^N$ using polynomial interpolation or discrete least-squares. To this end, we rewrite $\tilde{f}$ in (6) using the Chebyshev polynomial basis as

$$\tilde{f}\left(\frac{1}{s}\right) = e^{-i\xi/s} s \sum_{\ell=1}^{d} \tilde{\lambda}_\ell T_{\ell-1}(\tilde{s}), \tag{7}$$

where $\tilde{\lambda}_\ell$, $\ell = 1, ..., d$, are the coefficients of the Chebyshev polynomials in the expansion and $\tilde{s}$ is a linear transformation that maps $s \in [1/\omega_{max}, 1/\omega_{min}]$ to $[-1, 1]$. Specifically, $s = \alpha\tilde{s} + \beta$ where $\alpha = \frac{1}{2}\left(\frac{1}{\omega_{min}} - \frac{1}{\omega_{max}}\right)$ and $\beta = \frac{1}{2}\left(\frac{1}{\omega_{min}} + \frac{1}{\omega_{max}}\right)$. Note that (7) is written as a function of both s and $\tilde{s}$ to simplify notation. We will also assume that the samples are sorted so that $\omega_1 = \omega_{min}$ and $\omega_N = \omega_{max}$.

We can approximate the coefficients in (7) for N data points by solving the system

$$\mathbf{A}\tilde{\boldsymbol{\lambda}} = \hat{f}(\boldsymbol{\omega}) \tag{8}$$

for $\tilde{\boldsymbol{\lambda}}$, where $\mathbf{A}$ is the $N \times d$ matrix based on (7). Here $\tilde{\boldsymbol{\lambda}} = (\tilde{\lambda}_1, ..., \tilde{\lambda}_d)^T$, $\hat{f}(\boldsymbol{\omega}) = (\hat{f}(\omega_1), ..., \hat{f}(\omega_N))^T$, and each element of $\mathbf{A}$ is given by

$$a_{j,\ell} = \exp(-i\xi\omega_j)\cos((\ell-1)\cos^{-1}\tilde{s}_j)s_j. \tag{9}$$

This system is always full rank and for $N = d$ the approximation is interpolatory. Equivalently, one may use the barycentric formula [17] to efficiently evaluate polynomial interpolants.

A natural sampling choice for polynomial interpolation is a Chebyshev distribution in s, as Chebyshev interpolation is near-optimal [18]. The reciprocals of standard Chebyshev nodes in ω, therefore, are good nodes for approximating Fourier transforms using (7). Figure 1 shows nodes in the transformed variable s and the original variable w. To generate N points in $[\omega_{min}, \omega_{max}]$ we use Chebyshev nodes of the second kind:

$$\omega_j = \frac{1}{s_j}, \quad s_j = \alpha\cos\left(\frac{(j-1)\pi}{N-1}\right) + \beta, \quad 1 \le j \le N. \tag{10}$$

Notice that in this case, $\mathbf{A}$ can be seen as the product of a diagonal matrix, whose non-zero elements are $\exp(-i\xi/s_j)s_j$, and a Chebyshev interpolation matrix. Polynomial interpolation on Chebyshev nodes is known to be well conditioned [19].

Convergence of polynomial interpolation on these nodes is well understood [18–20] and we are able to provide precise rates of convergence. In the following theorem, we take $\omega_{max} = \infty$, but a similar error estimate also holds when ω_{max} is finite.

Theorem 1. *Suppose $f \in L^1 \cap C^{m-1}(\mathbb{R}\backslash\xi)$ is bounded and $f^{(m)}(x)(x-\xi)^\mu$, with $\mu = \lfloor (m+1)/2 \rfloor$, is of bounded variation in $\mathbb{R}\backslash\xi$. If $\tilde{f}$ interpolates $\hat{f}$ on N nodes given by (10), then*

$$\|\hat{f} - \tilde{f}\|_{[\omega_{min},\infty)} = O(N^{-\mu}), \quad \text{as } N \to \infty, \tag{11}$$

where $\omega_{min} > 0$ and $\|\cdot\|_{[\omega_{min},\infty)}$ is the sup-norm over the interval $[\omega_{min}, \infty)$.

PROOF. Let $g(s) := e^{i\xi/s}\hat{f}(1/s)$ and notice that $|\hat{f}(\omega) - \tilde{f}(\omega)| = |g(s) - sP_d(s)|$, where $sP_d(s)$ is the polynomial interpolant of $g(s)$ at Chebyshev nodes in $[0, 1/\omega_{min}]$. We point out that by the Riemann–Lebesgue lemma $g(0) = 0$. The s factor in front of P_d, therefore, is consistent with the interpolatory condition at $s = 0$. In what follows we show that $g \in C^\mu[0, 1/\omega_{min}]$, which is sufficient to establish the estimate (11); see e.g. [21].

Using equation (4), direct differentiation of g gives

$$\begin{aligned}
g(s) &= \sum_{\ell=1}^{m} \lambda_\ell s^\ell + (-i)^m s^m \int_{\mathbb{R}\backslash\xi} f^{(m)}(x) e^{-i(x-\xi)\omega} dx, \qquad (12)\\
g'(s) &= \sum_{\ell=2}^{m} \lambda_\ell \ell s^{\ell-1} + (-i)^m m s^{m-1} \int_{\mathbb{R}\backslash\xi} f^{(m)}(x) e^{-i(x-\xi)\omega} dx \\
&\quad -(-i)^{m+1} s^{m-2} \int_{\mathbb{R}\backslash\xi} f^{(m)}(x)(x-\xi) e^{-i(x-\xi)\omega} dx, \\
&\vdots \\
g^{(\mu)}(s) &= \sum_{\ell=\mu+1}^{m} \lambda_\ell \Pi_{j=0}^{\mu-1}(\ell - j) s^{\ell-\mu} \\
&\quad +(-i)^m \Pi_{j=0}^{\mu-1}(m-j) s^{m-\mu} \int_{\mathbb{R}\backslash\xi} f^{(m)}(x) e^{-i(x-\xi)\omega} dx \\
&\quad + \ \ldots \ + (-1)^\mu (-i)^{m+\mu} s^{m-2\mu} \int_{\mathbb{R}\backslash\xi} f^{(m)}(x)(x-\xi)^\mu e^{-i(x-\xi)\omega} dx. \qquad (13)
\end{aligned}$$

The decay of $f^{(m)}$ ensures that all integrands in these expressions belong to $L^1(\mathbb{R}\backslash\xi)$, implying that these integrals are bounded and uniformly continuous functions of ω. The choice $\mu = \lfloor (m+1)/2 \rfloor$ guarantees that $m - 2\mu \geq -1$ and the last term in (13) remains bounded as $s \to 0$ because the last integral is $O(1/\omega)$ as $\omega \to \infty$, or $O(s)$ as $s \to 0$. All other powers of s are non-negative.

We consider the following example to illustrate algebraic rates of convergence:

$$f_2^k(x) = \begin{cases} (2-x)^k & 1 \leq x \leq 2, \\ 0 & \text{otherwise.} \end{cases} \tag{14}$$

Note that the kth derivative of f_2^k is of bounded variation on $\mathbb{R}\backslash\xi$. Figure 2 presents the error in the approximation of the Fourier transform of these functions in the interval $[0.1, \infty)$ under the assumption that the jump location $\xi = 1$ is known *a priori*. As predicted by Theorem 1, the interpolants converge algebraically.

Super-algebraic convergence is possible for functions that are in $f \in L^1 \cap C^\infty(\mathbb{R}\backslash\xi)$ for which derivatives decay sufficiently fast. For example, consider the function

$$f_3(x) = \begin{cases} 0 & x < 1 \\ \frac{1}{x^2} & x \geq 1. \end{cases} \tag{15}$$

Figure 3 presents the error in the approximation of its Fourier transform in the interval $[0.1, \infty)$. The convergence plot shows that the error decays faster than any algebraic rate, as predicted by Theorem 1.

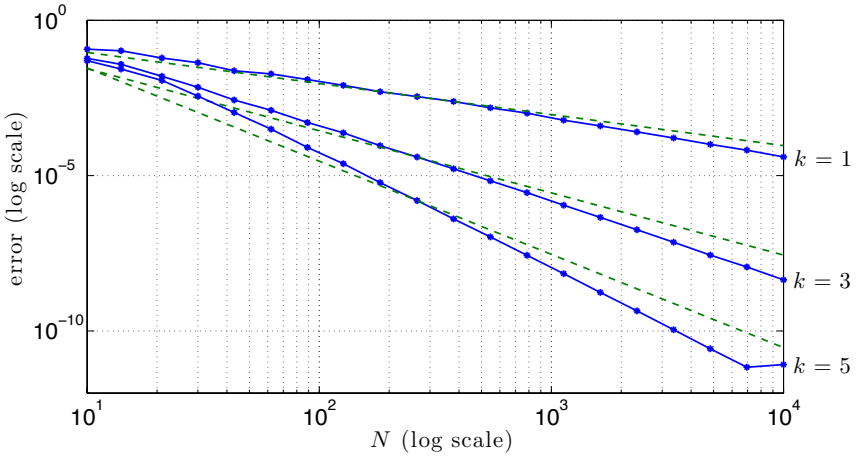

Figure 2: Sup-norm error in frequency space as a function of N, the number of Fourier samples, in the recovery of $\hat{f}_2^k$ with $k = 1, 3$, and 5 (14). The error was computed for $\omega \in [0.1, \infty)$. Dashed lines correspond to the error estimate in Theorem 1, $O(N^{(k+1)/2})$.

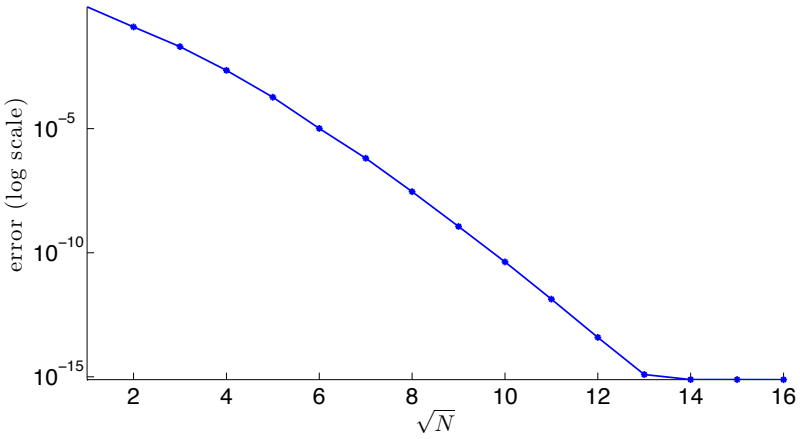

Figure 3: Error in frequency space as a function of $\sqrt{N}$ in the recovery of $\hat{f}_3$ (15), where $\omega \in [0.1, \infty)$.

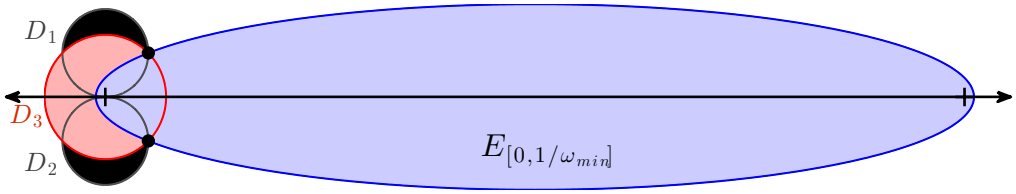

Figure 4: Ellipse used to estimate ρ in Theorem 2. The function g has an analytic extension into the complex plane. Possible singularities in the extension are restricted to the black regions exterior to D_3 and interior to D_1 and D_2. The critical ellipse passes through the intersection points of D_3 with D_1 and D_2. The radius of D_3 is $1/c_1$ and of D_1 and D_2 is $1/(2c_2)$. Tick marks on the real axis mark the ellipse's foci at $z = 0$ and $z = 1/\omega_{min}$.

Notice that this semi-logarithmic plot indicates that the error decays approximately as $O(\rho^{-\sqrt{N}})$, for some $\rho > 1$, which is consistent with supper-algebraic but sub-geometric convergence.

Geometric convergence rates are possible for functions that decay exponentially fast at infinity, provided that derivatives grow at most exponentially with the differentiation order.

Theorem 2. *Suppose $f \in C^\infty(\mathbb{R}\backslash\xi)$ is bounded and that there exist positive constants c_0, c_1, and c_2, such that, for all $m \in \mathbb{N}$,*

$$\left|f^{(m)}(x)\right| \leq c_0 c_1^m e^{-c_2|x-\xi|}, \quad x \in \mathbb{R}\backslash\xi. \tag{16}$$

If $\tilde{f}_d$ interpolates $\hat{f}$ on N nodes given by (10), then there exists a $\rho > 1$ such that

$$\|\hat{f} - \tilde{f}\|_{[\omega_{min},\infty)} = O(\rho^{-N}), \quad \text{as } N \to \infty, \tag{17}$$

where

$$\begin{aligned} \rho &= b + \sqrt{1+b^2} - \varepsilon, \quad \forall\ \varepsilon > 0, \\ b &= \tfrac{\sqrt{2\omega_{min}}}{c_1}\sqrt{\omega_{min} - \sqrt{c_1^2 - c_2^2} + \sqrt{\omega_{min}^2 - 2\omega_{min}\sqrt{c_1^2 - c_2^2} + c_1^2}}. \end{aligned} \tag{18}$$

Proof. As in the proof of Theorem 1, we have that $|\hat{f}(\omega) - \tilde{f}(\omega)| = |g(s) - sP_d(s)|$, where $sP_d(s)$ is the polynomial interpolant of $g(s)$ at Chebyshev nodes in $[0, 1/\omega_{min}]$. Polynomial interpolation at these nodes converges with the rate (17) if g has an analytic extension in a region of the complex plane that includes an ellipse with foci at $z = 0$ and $z = 1/\omega_{min}$ [21]. The exponential decay of $f(x)$ as $|x| \to \infty$ implies its Fourier transform $\hat{f}$ has an extension in the complex plane that is analytic inside the strip $|\text{Im }\omega| < c_2$ [22]. The map $1/z$ is smooth, except at $z = 0$, and maps this strip into the exterior of two circles with centers at $\pm i/(2c_2)$ and radius $r = 1/(2c_2)$. The function $g(s) = \exp(i\xi/s)\hat{f}(1/s)$ is, therefore, analytic in the exterior of these two disks. This mapping is illustrated in Figure 4, where the two disks are denoted by D_1 and D_2. To show that g is analytic in an open region that includes the interval $[0, 1/\omega_{min}]$, we next prove that g has a convergent power series centered at $s = 0$.

Notice that for $s \in \mathbb{R}$, (16) gives

$$\begin{aligned} \left|(-i)^m s^m \int_{\mathbb{R}\backslash\xi} f^{(m)}(x) e^{-i(x-\xi)\omega} dx\right| &\leq |s|^m \int_{\mathbb{R}\backslash\xi} |f^{(m)}(x)| dx \\ &\leq 2c_0(c_1|s|)^m/c_2. \end{aligned} \tag{19}$$

Referring again to (12), and recalling that $\lambda_\ell = [f^{(\ell-1)}](\xi)(-i)^\ell$, we have

$$|\lambda_\ell| < c_0 c_1^{\ell-1},$$

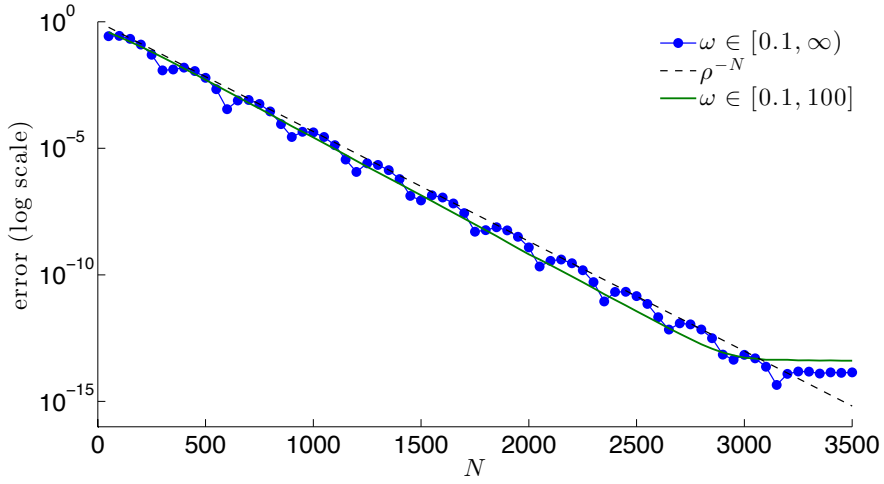

Figure 5: Error in frequency space as a function of N in the recovery of $\hat{f}_4$ (21). The error is shown for $\omega \in [0.1, \infty)$ and $\omega \in [0.1, 100]$, together with the estimate ρ^{-N}, where $\rho \approx 1.01$.

and hence, the power series $\sum_{\ell=1}^{\infty} \lambda_\ell s^\ell$ is convergent inside the disk $|s| < 1/c_1$. Finally combining this result with (19), we have that

$$g(s) = \sum_{\ell=1}^{\infty} \lambda_\ell s^\ell, \quad |s| < 1/c_1. \tag{20}$$

In order to estimate ρ, we refer to Figure 4, where the disk in which (20) holds is denoted by D_3. Any possible singularities in g, therefore, are restricted to the region exterior to D_3 and interior to D_1 and D_2, marked in black. Thus, the critical ellipse with foci at $z = 0$ and $z = 1/\omega_{min}$ used to estimate (17) contains the intersection points of D_3 with D_1 and D_2. We obtain ρ by rescaling the ellipse so that the foci are linearly mapped to $z = \pm 1$. Finally, ρ is the sum of its semi-major and semi-minor axes. More specifically, we have that

$$\begin{aligned} b &= \frac{\sqrt{2\omega_{min}}}{c_1} \sqrt{\omega_{min} - \sqrt{c_1^2 - c_2^2} + \sqrt{\omega_{min}^2 - 2\omega_{min}\sqrt{c_1^2 - c_2^2} + c_1^2}}, \\ \rho &= b + \sqrt{1 + b^2} - \varepsilon, \quad \forall\, \varepsilon > 0, \end{aligned}$$

were b is the semi-minor axis of the critical Bernstein-ellipse, and $\sqrt{1+b^2}$ is the semi-major.

To show this theorem gives a tight estimate of the error, we consider the following test function:

$$f_4(x) = \begin{cases} 0 & x < 0, \\ e^{-(x-2)} \cos\left(10(x-2)\right) & x \geq 2. \end{cases} \tag{21}$$

Its Fourier transform is given by $\hat{f}_4(\omega) = e^{-2i\omega}(1+i\omega)/\left(100 + (1+i\omega)^2\right)$. Moreover, f_4 satisfies the bounds in (16) with $c_1 = \sqrt{101}$ and $c_2 = 1$. Plugging them into (18), together with $\omega_{min} = 0.1$, we obtain $\rho \approx 1.01$ (using the limit $\varepsilon \to 0$). Figure 5 presents the error in Fourier domain, $\|\hat{f}_4 - \tilde{f}_4\|_{[\omega_{min}, \infty)}$, together with ρ^{-N}. Numerical results show very good agreement between the error and the estimate.

We point out that, although the theorem is stated for the case $\omega_{max} = \infty$, estimates are similar for the bounded domain case. In fact, our numerical results indicate that rates of convergence are not very sensitive to ω_{max}. Figure 5 also shows the error for approximations in $[0.1, 100]$. In this case, samples of the Fourier transform are also taken using (10), but with $\omega_{max} = 100$. It can be observed that errors for bounded and unbounded intervals are similar, with the rate being slightly better for the finite domain case.

Notice that we assume that the Fourier transform $\hat{f}$ vanishes at infinity. Hence, in the case $\omega_N = \omega_{max} = \infty$, the highest sampled frequency is ω_{N-1}. With $\omega_{max} = \infty$, the sampling points in (10) are given by $\omega_j = 1/s_j$, where

$$s_j = \frac{1}{2\omega_{min}} \left(\cos\left(\frac{(j-1)\pi}{N-1} \right) + 1 \right),$$

and

$$\omega_{N-1} = \frac{2\omega_{min}}{\cos\left(\frac{N-2}{N-1}\pi\right) + 1} = \frac{2\omega_{min}}{1 - \cos\left(\frac{\pi}{N-1}\right)} \sim \frac{4\omega_{min}}{\pi^2}(N-1)^2.$$

In other words, the length of the interval being sampled in Fourier space increases as $O(N^2)$. As illustrated in Fig. 1, however, very few samples are required at high-frequencies and most of the data used in the reconstruction is sampled at low frequencies.

On the other hand, if $\omega_{max} < \infty$ is fixed due to practical restrictions, $\hat{f}$ can still be accurately recovered on the interval $[\omega_{min}, \omega_{max}]$ as illustrated in Fig. 5. In this case, however, the Gibbs phenomenon will have a larger contribution to the error in the reconstruction of f. The approximation can be further improved by filtering and other post-processing techniques available to mitigate Gibbs effects.

It is important to mention that in practice, one often needs to determine the number of samples required to achieve certain accuracy. In order to apply the theorems above, one needs information about the regularity of the target function that is often unavailable. Notice, however, that

$$|\hat{f}(\omega) - \tilde{f}(\omega)| = s\,|g(s)/s - P_d(s)|\,.$$

That is, the approximation depends on how well P_d approximates $g(s)/s$. A well known strategy in polynomial approximation to obtain *a posteriori* error estimates is the following. Write P_d using an orthogonal polynomial expansion (say Chebyshev or Legendre polynomials, for instance). Because of orthogonality, the expansion coefficients can be used to estimate the error. The high-degree term coefficients are usually good estimates for the error. This approach is used, for instance, in the Chebfun system [23, 24], where the degree of the approximation is found adaptively using only samples of the function being approximated.

2.3. Functions with Multiple Jump Discontinuities

The problem becomes more complicated when the underlying function has multiple jump discontinuities, as the generalization of the asymptotic expansion in (4) yields multiple sums multiplied by different complex exponentials. For example, the Fourier transform of a function with two jump discontinuities at the points $x = \xi_1$ and ξ_2 is

$$\begin{aligned} \hat{f}(\omega) &= e^{-i\omega\xi_1} \sum_{\ell=1}^{m} \frac{[f^{(\ell-1)}](\xi_1)}{(i\omega)^\ell} + e^{-i\omega\xi_2} \sum_{\ell=1}^{m} \frac{[f^{(\ell-1)}](\xi_2)}{(i\omega)^\ell} \\ &\quad + \frac{1}{(i\omega)^m} \int_{\mathbb{R}\setminus\{\xi_1,\xi_2\}} f^{(m)}(x) e^{-i\omega x} dx, \end{aligned} \tag{22}$$

where $\int_{\mathbb{R}\setminus\{\xi_1,\xi_2\}} = \lim_{\varepsilon\to 0} \int_{-\infty}^{\xi_1-\varepsilon} + \int_{\xi_1+\varepsilon}^{\xi_2-\varepsilon} + \int_{\xi_2+\varepsilon}^{\infty}$.

The extension of the approximation (6) now becomes

$$\tilde{f}\left(\frac{1}{s}\right) := e^{-i\xi_1/s} s P_{1,d_1}(s) + e^{-i\xi_2/s} s P_{2,d_2}(s), \tag{23}$$

where P_{1,d_1} and P_{2,d_2} are polynomials of degree d_1 and d_2 respectively. It is now less clear what a good sampling scheme is. The existence of an accurate approximation of $\hat{f}$ using (23), however, is established in the following theorem, while the following section will outline a possible way to find such polynomials. As in Theorem 1 we let $\omega_{max} = \infty$ and assume that jump locations ξ_1 and ξ_2 are known a priori.

Theorem 3. *Suppose $f \in L^1 \cap C^{m-1}(\mathbb{R}\backslash\{\xi_1,\xi_2\})$ is bounded and $f^{(m)}(x)(x-\xi_2)^\mu$, with $\mu = \lfloor (m+1)/2 \rfloor$, is of bounded variation in $\mathbb{R}\backslash\{\xi_1,\xi_2\}$. There exist polynomials P_{1,d_1} and P_{2,d_2} in (23) of degree at most N such that*

$$\|\hat{f} - \tilde{f}\|_{[\omega_{min},\infty)} = O(N^{-\mu}), \quad \text{as } N \to \infty,$$

where $\omega_{min} > 0$.

PROOF. Let $sP_{1,d_1}(s) = \sum_{\ell=1}^{d_1} (-i)^\ell [f^{(\ell-1)}](\xi_1) s^\ell$ and let $sP_{2,d_2}(s)$ be the polynomial interpolant on Chebyshev nodes on $[0, 1/\omega_{min}]$ (as defined in (10)) of

$$g_2(s) := \sum_{\ell=1}^{m} \frac{[f^{(\ell-1)}](\xi_2)}{(i\omega)^\ell} + e^{i\xi_2/s}(-is)^m \int_{\mathbb{R}/\{\xi_1,\xi_2\}} f^{(m)}(x) e^{-ix/s} dx.$$

Similarly to Theorem 1, we have that $\left|\hat{f}(w) - \tilde{f}(w)\right| = |g_2(s) - sP_{2,d_2}(s)|$ whenever $N \geq m$. Repeating the procedure in (12)–(13) shows that $g_2 \in C^\mu[0, 1/\omega_{min}]$.

Geometric (exponential) convergence can also be proved using the same construction in Theorem 3 and bounds on $f^{(m)}$ by repeating arguments presented in Theorem 2.

Corollary 1. *Suppose $f \in C^\infty(\mathbb{R}\backslash\{\xi_1,\xi_2\})$ is bounded and that there exist positive constants c_0, c_1, and c_2, such that, for all $m \in \mathbb{N}$,*

$$\left|f^{(m)}(x)\right| \leq c_0 c_1^m e^{-c_2|x-\xi|}, \quad x \in \mathbb{R}\backslash\{\xi_1,\xi_2\}.$$

If the power series $\sum_{\ell=1}^{\infty} (-i)^\ell [f^{(\ell-1)}](\xi_1) s^\ell$ is convergent for $|s| \leq 1/\omega_{min}$, then there exist polynomials P_{1,d_1} and P_{2,d_2} in (23) of degree at most N such that

$$\|\hat{f} - \tilde{f}\|_{[\omega_{min},\infty)} = O(\rho^{-N}), \quad \text{as } N \to \infty, \tag{24}$$

for some $\rho > 1$. A possible construction is to take $sP_{1,d_1} = \sum_{\ell=1}^{d_1} (-i)^\ell [f^{(\ell-1)}](\xi_1) s^\ell$ and $sP_{2,d_2}(s)$ the Chebyshev interpolant of $g_2(s)$.

2.3.1. Compactly Supported Functions

We point out that the bound in the magnitude of derivatives in the corollary above is not satisfied by many piecewise analytic functions, since analytic functions often have m^{th} derivatives that grow as fast as $m!$. When f is compactly supported, it is possible to derive less restrictive conditions for geometric convergence. This is fortunate, as in the MR applications images are compactly supported, and often assumed to be piecewise smooth. To this end, notice that our proposed expansion can be exact for piecewise polynomials. To make this statement more precise, we generalize (23) for a function with J smooth pieces supported in $[\xi_1, \xi_{J+1}]$, where the jump locations are denoted by $x = \xi_j$, $j = 1, \ldots, J+1$:

$$\tilde{f}(1/s) = \sum_{j=1}^{J+1} e^{-i\xi_j/s} sP_{j,d_j}(s), \tag{25}$$

and each P_{j,d_j} is a polynomial of degree d_j.

Lemma 1. *Suppose f is piecewise polynomial with support in $[\xi_1, \xi_{J+1}]$ and $f(x) = f_j(x)$, $\xi_j < x < \xi_{j+1}$, j=1,...,J. Then $\hat{f}(1/s) = \tilde{f}(1/s)$, d_1 =deg(f_1), d_{J+1} =deg(f_{J+1}), and d_j =max(deg(f_{j-1}),deg(f_j)), $j = 2, \ldots, J$.*

PROOF. We have that $\hat{f}(w) = \sum_{j=1}^{J} \int_{\xi_j}^{\xi_{j+1}} f_j(x) e^{-i\omega x} dx$. Integration by parts of each integral in this sum completes the proof.

Piecewise smooth functions are well approximated by piecewise polynomials and we use this result to obtain the following estimates.

Theorem 4. *Suppose $f \in L^2(\mathbb{R})$ is a piecewise function with support in $[\xi_1, \xi_{J+1}]$. If f_p is a piecewise polynomial approximation of f, then*

$$\|\hat{f} - \tilde{f}_p\|_{L^2([\omega_{min},\infty))} \leq \sqrt{2\pi}\|f - f_p\|_{L^2(\mathbb{R})},$$

where $\tilde{f}_p$ is the Fourier transform of f_p in (25).

Moreover, if $f(x) = f_j(x)$, $\xi_j < x < \xi_{j+1}$, j=1,...,J, and each piece f_j is analytic in a neighborhood of $[\xi_j, \xi_{j+1}]$, then there exist P_{j,d_j} in (25) such that

$$\|\hat{f} - \tilde{f}\|_{L^2([\omega_{min},\infty))} = O(\rho^{-N}), \quad \text{as } N \to \infty,$$

for some $\rho > 1$.

Proof. The first part follows directly from Lemma 1 and Plancherel's theorem [25], i.e.,

$$\|\hat{f} - \tilde{f}_p\|_{L^2([\omega_{min},\infty))} = \|\hat{f} - \hat{f}_p\|_{L^2([\omega_{min},\infty))} \leq \|\hat{f} - \hat{f}_p\|_{L^2(\mathbb{R})} = \sqrt{2\pi}\|f - f_p\|_{L^2(\mathbb{R})}.$$

The second part follows from well known error bounds for polynomial approximation of analytic functions [20].

2.3.2. Discrete Least Squares Approximations

The discussion above addresses the existence of accurate expansions of the form (25). One way to construct polynomials P_{j,d_j} is to use jump value of $f^{(k)}$ at each ξ_j. Jump values in derivatives, however, are not available in most practical applications and are difficult, if not impossible, to estimate. Instead, we propose constructing P_{j,d_j} by a least-squares fit of (25) to the given Fourier data. Specifically, as in (7), we expand P_{j,d_j} using Chebyshev polynomials:

$$\tilde{f}\left(\frac{1}{\tilde{s}}\right) = \sum_{j=1}^{J+1} e^{-i\xi_j/s} s \sum_{\ell=1}^{d_j} \tilde{\lambda}_{\ell,j} T_{\ell-1}(\tilde{s}), \tag{26}$$

and use discrete least squares to find $\tilde{\lambda}_{\ell,j}$. Here again $\tilde{s}$ is a linear transformation of s such that $\tilde{s} \in [-1, 1]$, and we write $\tilde{f}$ as a function of both s and $\tilde{s}$ for convenience of notation.

The corresponding linear system to (26) gives

$$\begin{bmatrix} A_1 & | & \cdots & | & A_{J+1} \end{bmatrix} \begin{bmatrix} \tilde{\lambda}_{1,1} \\ \vdots \\ \tilde{\lambda}_{d_{J+1},J+1} \end{bmatrix} = \begin{bmatrix} \hat{f}(\omega_1) \\ \vdots \\ \hat{f}(\omega_N) \end{bmatrix}, \tag{27}$$

where the ℓ^{th} column of A_j is defined by $A_{j_\ell}(\tilde{\mathbf{s}}) = \exp(-i\boldsymbol{\omega}\xi_j)\mathbf{s}T_{\ell-1}(\tilde{\mathbf{s}}), \ell = 1, ..., d_j$, and the vectors are multiplied pointwise. In contrast to the one jump case, which is well conditioned on the (reciprocal of) Chebyshev nodes, the condition number of the matrix above may be large and, in many cases, the matrix is numerically rank deficient. The particular solution that we have found to work for our purposes is the *basic solution* [26, 27]:

Definition 1. *A* basic solution *to $\mathbf{A}x = b$ where $\mathbf{A}$ is an $m \times n$ matrix and $rank(\mathbf{A}) = r < n$ is a solution x that minimizes the residual $||\mathbf{A}x - b||_2$ such that x has at most r non-zero components.*

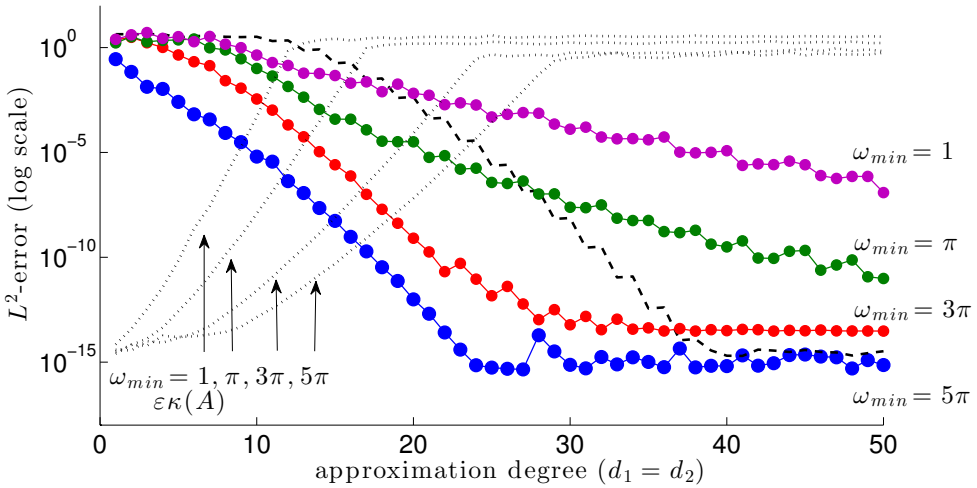

Figure 6: Error in frequency space in the recovery of $\hat{f}_5$ (28) for $\omega_{min} = 1, \pi, 3\pi, 5\pi$. The dashed line shows the error in the Chebyshev polynomial interpolant of f in $[-1, 1]$ (physical space); cf. Theorem 4. The dotted lines show the condition numbers of A, $\kappa(A)$, times the machine precision, $\varepsilon = 2^{-52}$.

The basic solution, which is not necessarily unique, was calculated using QR factorization with column pivoting [26], and our computations were carried out using "backslash" in MATLAB. Least squares has the added advantage of helping to stabilize the method when data is given on more general nodes distributions such as logarithmic sampling.

We use the following function to demonstrate the accuracy of this approach and how the error relates to the estimate in Theorem 4:

$$f_5(x) = \begin{cases} \cos(4\pi x) & -1 \le x \le 1, \\ 0 & \text{otherwise.} \end{cases} \tag{28}$$

As before, we assume that the edge locations $\xi_1 = -1$ and $\xi_2 = 1$ are known *a priori*. Figure 6 presents the error in Fourier space, $\|\hat{f}_5 - \tilde{f}_5\|_{L^2([\omega_{min},\infty))}$, for several values of ω_{min}. The coefficients in (26) were computed using discrete least squares. The number of Fourier samples N at reciprocal of Chebyshev nodes (10) is three times the approximating degree ($d = d_1 = d_2$). To compare the computed error with the estimate in Theorem 4, the error in the polynomial approximation of f_5 in $[-1, 1]$ is also presented. This approximation was computed using Chebyshev interpolation. The error agrees with the prediction in the theorem whenever the system is well conditioned. When the condition number of the system is large, i.e. $\varepsilon\kappa(A)$ is bigger than the truncation error, the rates of convergence deteriorate. The figure also illustrates the relationship of $\kappa(A)$ and ω_{min}. Fortunately, even when the system is numerically rank deficient, it is possible to obtain an accurate approximation, as we will detail in the following section.

3. The Polynomial Resampling Method

3.1. Basic Algorithm

In applications such as MRI, the objective is the recovery of f from Fourier data. In this section we present a resampling algorithm based on the approximation (26). The main idea is to evaluate $\tilde{f}$, the approximate Fourier transform, on integers and then use a discrete inverse Fourier transform to recover f in physical space. To reduce Gibbs oscillations, we can then apply spectral reprojection [13–15]. In its simplest form, the method consists of two steps: the computation of coefficients $\lambda_{\ell,j}$ in (26), which we refer to as the analysis portion; and the evaluation of $\tilde{f}$ on integers, $\mathbf{k} = (k_1, ..., \lfloor \omega_{\max} \rfloor)^T$, which we call the synthesis part of the method. Together these two steps define the *polynomial resampling method (PRM).*

Algorithm 1 describes the PRM when there are multiple jump discontinuities. The analysis portion of the PRM uses discrete least squares. We point out, however, that for functions with only one jump discontinuities interpolation will work well on reciprocal of Chebyshev nodes or similar distributions.

Algorithm 1 The PRM for (known) multiple jump discontinuities

Input: the non-uniform Fourier data $(\boldsymbol{\omega}, \hat{f}(\boldsymbol{\omega}))$ where $\boldsymbol{\omega} = (\omega_1, ..., \omega_N)^T$ and $\hat{f}(\boldsymbol{\omega}) = (\hat{f}(\omega_1), ..., \hat{f}(\omega_N))^T$. Also a vector of jump discontinuities $\boldsymbol{\xi} = (\xi_1, ..., \xi_{J+1})$ of a piecewise analytic and compactly supported function on the interval $[\xi_1, \xi_{J+1}]$.
Output: The vector $\tilde{f}(\mathbf{k})$, for $\mathbf{k} = (k_1, k_2, ..., \lfloor \omega_N \rfloor)^T$ where $k_1 \geq 1$ is an integer and $k_{j+1} = k_j + 1$.

1. Choose the number of columns m in the matrix (27) as $m = \sum_{j=1}^{J+1} d_j \leq N$. We let each jump discontinuity in f have equal weight by setting $d_i = d_j$ for all i, j. In practice we usually choose $m \approx \frac{N}{3}$.

2. Define the elements of the vector $\mathbf{s}$ by $s_j := \frac{1}{\omega_j}, j = 1, ..., N$. The linear transformation vector $\tilde{\mathbf{s}}$ is defined by its elements as $s_j = \alpha \tilde{s}_j + \beta$, where α and β are defined just below (7).

3. **Analysis step 1:** Generate the matrix $\mathbf{A}_{N \times m}$ to be such that $\mathbf{A}\tilde{\boldsymbol{\lambda}} = \hat{f}(\boldsymbol{\omega})$ by constructing the elements of the $N \times \frac{m}{p}$ matrix $\mathbf{A}_t, t = 1, \cdots, J+1$, corresponding to each of the summands in (26) using (9):

$$a_{t_{j,l}} = \exp(-i\xi_t \omega_j) s_j \cos((l-1)\cos^{-1}(\tilde{s}_j))$$

   The $N \times m$ matrix $\mathbf{A}$ is formed by aggregating all of these matrices as

$$\mathbf{A} = [\mathbf{A}_1 \ \mathbf{A}_2 \ ... \ \mathbf{A}_{J+1}]$$

4. **Analysis step 2:** Solve the linear system $\mathbf{A}\tilde{\boldsymbol{\lambda}} = \hat{f}(\boldsymbol{\omega})$ for $\tilde{\boldsymbol{\lambda}}$, using a basic solution if the matrix $\mathbf{A}$ is rank deficient.

5. **Synthesis step 1:** Generate a matrix $\mathbf{B}_j$ with

$$b_{t_{j,l}} = \exp(-i\xi_t k_j) s_j \cos((l-1)\cos^{-1}(\tilde{r}_j))$$

   where $\tilde{r}_j$ correspond to the integers k_j mapped to the interval $[-1, 1]$.

   The matrix $\mathbf{B}$ is formed by aggregating all of these matrices as

$$\mathbf{B} = [\mathbf{B}_1 \ \mathbf{B}_2 \ ... \ \mathbf{B}_{J+1}]$$

6. **Synthesis step 2:** Multiply $\mathbf{B}\tilde{\boldsymbol{\lambda}}$ to get $\tilde{f}(\mathbf{k})$, an approximation of $\hat{f}(\mathbf{k})$.

3.2. Numerical experiments

As proof of concept we will consider jittered and log sampling schemes besides reciprocal of Chebyshev nodes. More specifically, we shall consider the following sampling patterns:

(a) Log Sampling: Samples in Fourier space are acquired at logarithmic intervals. This sampling pattern is motivated by the two-dimensional spiral sampling found in MRI, which oversamples in the lower frequencies and undersamples in the higher frequencies:

$$\omega_{min} = \omega_1 = 10^{-v}, ..., \omega_N = \omega_{max}, \tag{29}$$

where $\log(\omega_{j+1}) - \log(\omega_j) = \text{constant}$, $1 \leq j \leq N-1$ for some $v \geq 0$.

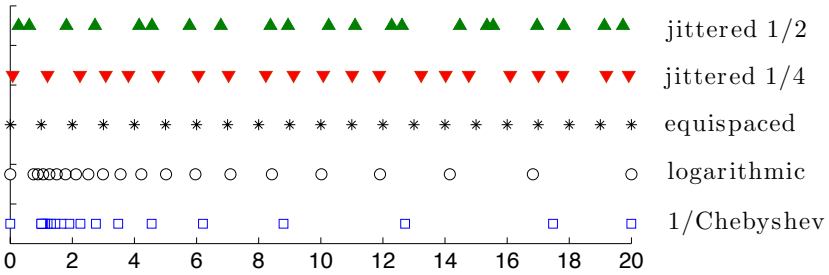

Figure 7: Examples of sampling patterns in the frequency domain.

(b) Reciprocal of Chebyshev Sampling: This is the ideal distribution for the PRM in the case when there is a single jump discontinuity. It assumes that $\hat{f}(0)$ is known and then samples $N-1$ points in the domain $[1, \omega_{max}]$ for some maximum frequency ω_{max} in the reciprocal of the Chebyshev distribution (10). The qualitative proximity of the inverse Chebyshev distribution to the logarithmic distribution suggests that the PRM will perform well for (one-dimensional) non-Cartesian spiral sampling patterns considered useful in MRI.

(c) Jittered Sampling:

$$\omega_j = j \pm \tau_j, \quad \tau_j \sim U[0, \theta],\ j = 0, \cdots, N. \tag{30}$$

Here $U[a, b]$ denotes a uniform distribution on the interval $[a, b]$. The τ_j's are independent, identically distributed (iid) random variables, and represent a uniform jitter about the equispaced nodes with a maximal jitter of θ. Further, both positive and negative jitters are equiprobable, with the sign of jitter at each node being independent of the sign of jitter at any other node. Reconstructions using this scheme can be considered as representative of the quality of real-world Cartesian (equispaced) sampling trajectories, since inaccuracies in generating magnetic field gradients can cause stochastic jitters in the scanning trajectory. Note that we assume these jitter locations are known.

In particular, throughout this paper we will consider logarithmic sampling with $v = 0$ and jittered distributions with $\theta = \frac{1}{4}$ and $\frac{1}{2}$. Note that the case when $\theta = \frac{1}{4}$ is the largest possible jitter that satisfies Kadec's one-quarter theorem [28] whereas the $\theta = \frac{1}{2}$ case represents a more non-uniform distribution. Figure 7 displays the various sampling schemes.

We now consider the following example:

$$f_6(x) := \begin{cases} 3/2 & \frac{-3\pi}{4} \le x < \frac{-\pi}{2} \\ 7/4 - x/2 + \sin(x - 1/4) & \frac{-\pi}{4} \le x < \frac{\pi}{8} \\ 11x/4 - 5 & \frac{3\pi}{8} \le x < \frac{3\pi}{4} \\ 0 & \text{otherwise} \end{cases} \tag{31}$$

To demonstrate the effectiveness of the PRM in Algorithm 1, we calculate the pointwise error at the integer coefficients in the Fourier domain. Figure 8 displays the test function f_6 alongside the pointwise error of the method in frequency domain. Here 128 logarithmically sampled Fourier coefficients (29) were used to find $\tilde{f}_6(k)$ at $k = 1..., 200$. Note that the edges are assumed to be known exactly. We also calculate the pointwise error of the Fourier partial sum as the difference between the sum for exact and for PRM coefficients, $\left| S_N \tilde{f}_6(x) - S_N \hat{f}_6(x) \right|$, where

$$S_N \hat{g} = \frac{1}{2\pi} \sum_{k=-N}^{N} \hat{g}(k) e^{ikx} \tag{32}$$

is the standard Fourier partial sum given Fourier coefficients $\hat{g}(k)$ at integer modes. Figure 9 illustrates these results for f_6 as the number of sampled modes varies, demonstrating exponential convergence and that, with

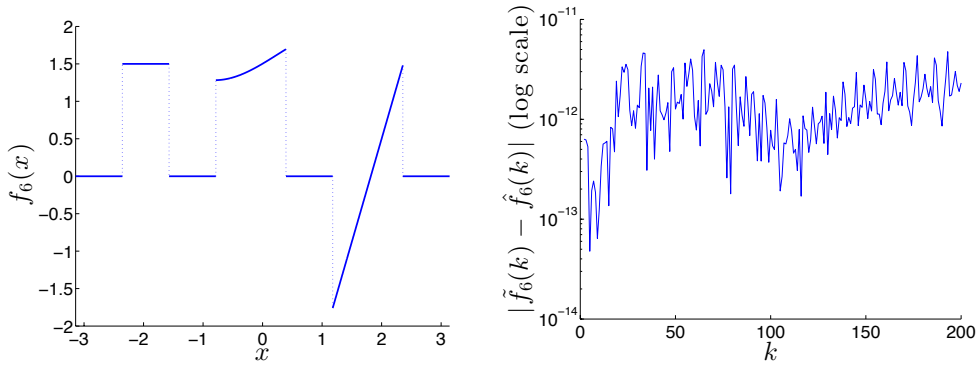

Figure 8: Left: plot of f_6 in $[-\pi, \pi]$. Right: pointwise error, $\left|\hat{f}_6(\mathbf{k}) - \tilde{f}_6(\mathbf{k})\right|$, where $\tilde{f}_6(\mathbf{k})$ is determined by Algorithm 1 for $N = 128$ logarithmically sampled points (29) with $v = 0$ and $\omega_{max} = 200$.

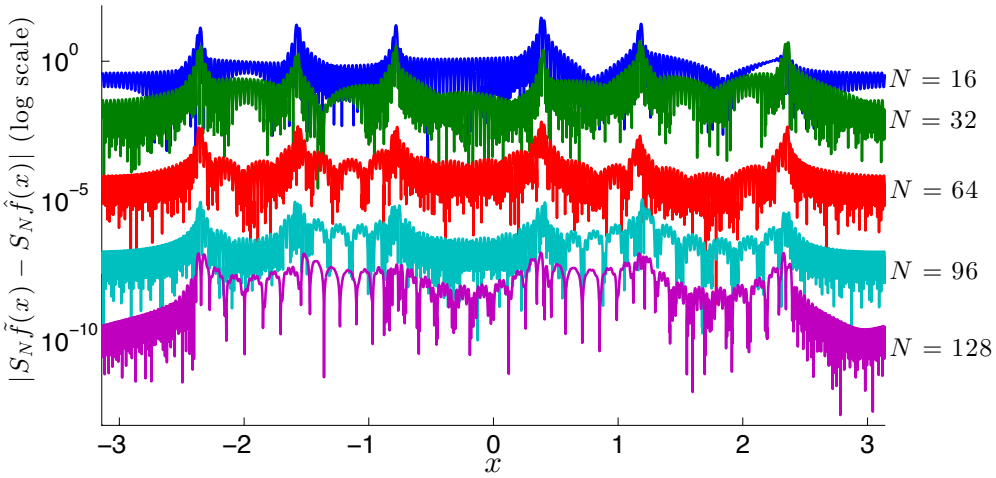

Figure 9: Pointwise error for $N = 16, 32, 64, 128$, $x \in [-\pi, \pi]$, for logarithmically distributed samples (29) where $v = 0$ and $\omega_{max} = 200$.

sufficient resolution, high order post-processing techniques that remove the Gibbs oscillations from $S_N\hat{f}$ can be effectively applied to $S_N\tilde{f}$ as well.

Figure 10 demonstrates the results of applying Algorithm 1 when the assumption that the exact edge locations are known is relaxed. The convergence of the PRM depends on both the number of sampled frequencies as well as on the accuracy of our approximation of the locations of each jump discontinuity. In this figure we approximate the true edge location vector $\boldsymbol{\xi}$ by its elements as $\hat{\xi}_j = \xi_j \pm h_j$, $j = 1, \cdots, J+1$, for $J+1$ jump discontinuities and $h_j \sim U[0, 10^{-i}]$ for $i = 1, ..., 5$, and plot the error $\|S_N\tilde{f}_6 - S_N\hat{f}_6\|_2$ as N varies.

3.3. Edge Location Optimization

In applications the exact edge locations of the underlying piecewise smooth function are typically not known. In this section we demonstrate how the PRM can also be used to improve on a set of initial edge location estimates. The initial edge location estimates for non-uniform Fourier data are provided by the algorithm described in [16], based on the concentration factor paradigm developed in [29]. As mentioned in the introduction, the accuracy of any high order post-processing method for reconstructing piecewise

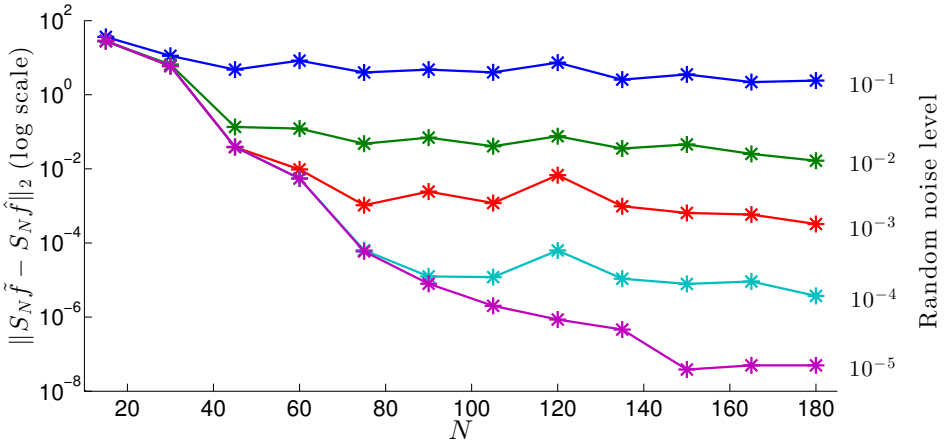

Figure 10: The L^2-error in $[-\pi, \pi]$ when each edge location is perturbed as $\hat{\xi}_j = \xi_j \pm h_j$, $j = 1, \cdots, J+1$, and h_j is uniformly distributed random noise in $[0, 10^{-i}], j = 1, ..., 6,\ i = 1, ..., 5$.

smooth functions from Fourier data critically depends on the edge locations. Obtaining these edge locations accurately can be especially difficult when the data is corrupted by noise. Figure 10 shows that the PRM, as presented in Algorithm 1, also needs accurate edge location information to obtain high order convergence, more accurate than can be expected of current edge detection methods from non-uniform Fourier data. Fortunately, the implementation in Algorithm 1 suggests a means of obtaining more accurate edge location estimates given an initial approximation. That is, given an initial estimate of jump locations, we can use Algorithm 1 to refine their location by finding the edge locations that minimize the least-squares residual $\left\| \mathbf{A}_{\hat{\boldsymbol{\xi}}} \tilde{\boldsymbol{\lambda}} - \hat{f}(\boldsymbol{\omega}) \right\|_2$, where we denote the matrix $\mathbf{A}$ in Algorithm 1 by $\mathbf{A}_{\hat{\boldsymbol{\xi}}}$ to emphasize its dependence on the estimated jump locations $\hat{\boldsymbol{\xi}}$.

Figure 11 shows the (one-parameter) objective function for

$$f_7(x) = \begin{cases} e^{-(x+\pi/2)^2} & x \geq -\pi/2, \\ 0 & x < -\pi/2. \end{cases}$$

The plot, on a semi-logarithmic scale, shows the residual of the least squares approximation of $\hat{f}(\boldsymbol{\omega})$ as the estimated location of the single jump discontinuity ($\xi = -\pi/2$) varies. The idea behind our next algorithm is to find the minimum of this objective function. For functions with multiple jump discontinuities, where the optimization space is multi-dimensional, it is necessary to have a good initial approximation of the edges in order to ensure that the global minimum (as opposed to a local minimum) is recovered from the algorithm. In this paper we used Nelder-Mead simplex direct search [27, 30] as our minimization procedure. It took this method 439 iterations and 744 function evaluations to find the minimum used in Figure 12(left).

Finally, Algorithm 2 describes the full PRM method including edge location optimization, and Figure 12 demonstrates its effectiveness for both logarithmic and jittered sampling distributions. By comparing Figure 12(left) to Figure 8 (which is equivalent except it assumes that the edges are known exactly) we see that the PRM yields high accuracy whether the edges are known exactly or whether we use Algorithm 2 to optimize an initial estimate. Indeed, the figures are nearly indistinguishable.

For jittered sampling distributions (30), Figure 12(right) demonstrates that the PRM accurately approximates the Fourier transform $\hat{f}(\omega)$ when ω is large enough, e.g., $\omega \geq 8$ in this example. The jittered sampling pattern yields a poorly conditioned matrix. Because the error in the PRM approximation in this case is concentrated in the low frequency Fourier coefficients the PRM is still useful even when the conditioning is poor, as other methods such as URS [4] do a very good job of approximating just the lower frequency coefficients. Notice especially the ability to approximate the transform from both jittered 1/2 and 1/4 distributions.

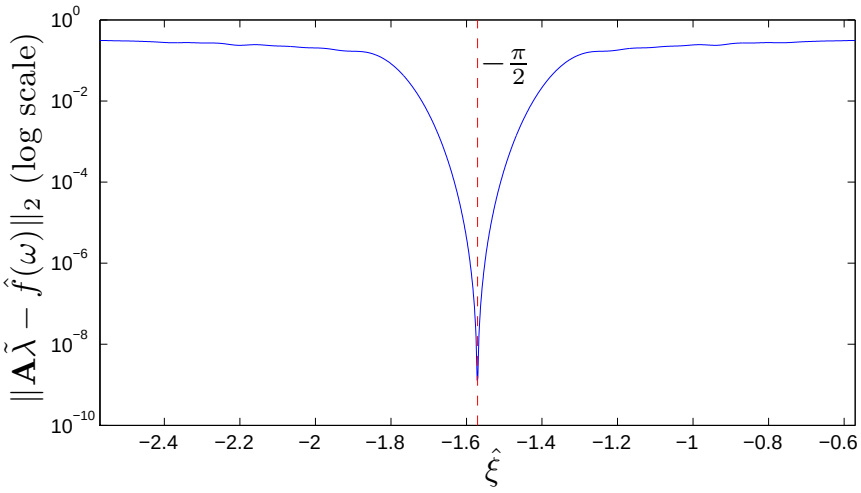

Figure 11: Residual $\left\|\mathbf{A}_{\hat{\boldsymbol{\xi}}}\tilde{\boldsymbol{\lambda}} - \hat{f}(\boldsymbol{\omega})\right\|_2$ as (estimated) jump location $\hat{\xi}$ of f_7 varies.

Algorithm 2 PRM with multiple edge location and optimization

Input: N non-uniform Fourier data points $(\boldsymbol{\omega}, \hat{f}(\boldsymbol{\omega}))$ where $\boldsymbol{\omega} = (\omega_1, ..., \omega_N)^T, \hat{f}(\boldsymbol{\omega}) = (\hat{f}(\omega_1), ..., \hat{f}(\omega_N))^T$
Output The vector $\tilde{f}(\mathbf{k})$, for $\mathbf{k} = (k_1, ..., \lfloor\omega_{\max}\rfloor)^T$ where $k_1 \geq 1$ and a vector $\hat{\boldsymbol{\xi}}$ of optimized edges for a piecewise smooth function f.

1. Let $\hat{\boldsymbol{\xi}}_0$ be an estimate of the jump locations. In our program we use the algorithm described in [16] to obtain $\hat{\boldsymbol{\xi}}_0$ from non-uniform Fourier data.

2. Optimize $\hat{\boldsymbol{\xi}}$ by minimizing the residual $\left\|\mathbf{A}_{\hat{\boldsymbol{\xi}}}\tilde{\boldsymbol{\lambda}} - \hat{f}(\boldsymbol{\omega})\right\|_2$ in Algorithm 1. In our code we use the Nelder-Mead simplex direct search [27, 30] as our minimization procedure, with $\hat{\boldsymbol{\xi}}_0$ as the initial guess for edge location.

3. Approximate the Fourier transform at the integers using these optimized edge locations. (Algorithm 1)

Remark: The optimization procedure may fail to converge to the absolute minimum when a very poor initial estimate of the edges is used in Step 2. In practice the estimates given by the method described in [16] worked well for some choice of parameters. It is also important that the number of edges given by the initial estimate is correct.

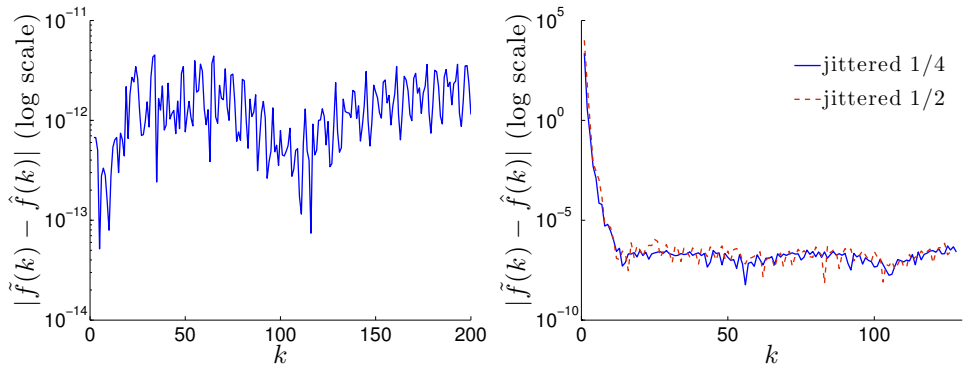

Figure 12: Pointwise error $\left|\hat{f}_6(\mathbf{k}) - \tilde{f}_6(\mathbf{k})\right|$ where $\tilde{f}_6(\mathbf{k})$ is determined by Algorithm 2, $N = 128$ logarithmic sampling pattern (29) for $\omega_k \in [1, 200]$ (left) and jittered sampling pattern for $N = 128$ (30) (right).

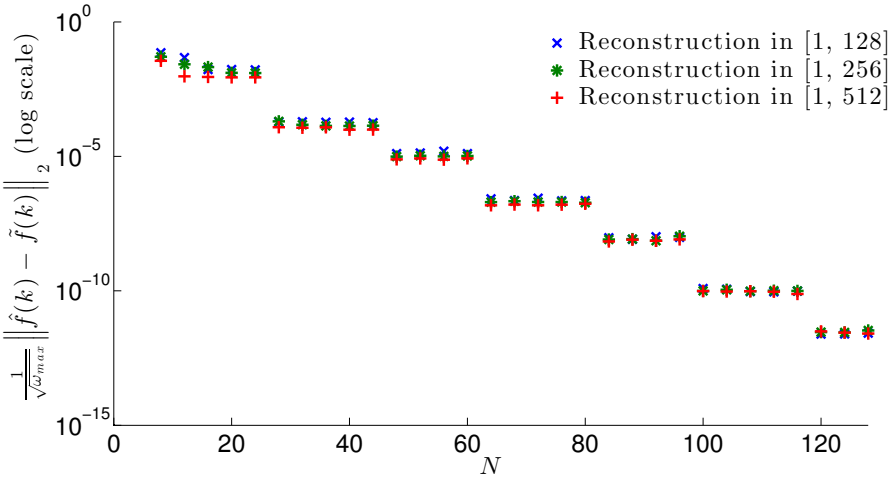

Figure 13: ℓ^2 error $\|\hat{f}_6(\mathbf{k}) - \tilde{f}_6(\mathbf{k})\|$ where $\mathbf{k} = (1, .., \omega_{max})^T$ and $\omega_{max} = 128, 256,$ and 512 respectively. Here the input data is $\hat{f}(\omega_j), j = 1, ..., N$, for ω_j logarithmically distributed (29) in $[1, \omega_{max}]$.

Figure 13 illustrates the PRM's ability to reconstruct ω in intervals of increasing size using the same amount of data. It plots the (scaled) ℓ^2 norm of the PRM approximation of the integer Fourier coefficients using logarithmically spaced data (29) with $v = 0$ and $\omega_{max} = 128, 256,$ and 512, respectively, as N, the number of sampled points in the interval $[1, \omega_{max}]$ increases. This figure demonstrates that given the same amount of input data the error per Fourier coefficient remains constant as the size of the domain increases, and thus the PRM does not necessarily need more data to approximate $\hat{f}$ in a larger domain. In this figure the edges are assumed to be known exactly as the edge detection method in [16] can only recover edges when $\omega_{\max}$ is less than or approximately equal to N. We point out that in [6] this same edge detection technique was shown to be able to extract additional Fourier data from higher frequencies, which may ultimately prove valuable when used in conjunction with the PRM. We leave this to future investigations.

Table 1 shows the results of Algorithm 2 for both clean and noisy Fourier data of f_6. No assumptions are made about the function and the only input is the non-uniform Fourier data sampled logarithmically as in (29) with $v = 0$ and $\omega_{max} = N$, the number of sampled Fourier coefficients. The first second and third columns in the table demonstrate the efficacy of the optimization procedure, showing the error in the initial approximation of the edges given by [16] in the second column and showing the error in the edge locations after the optimization procedure in the third column. The last column shows the error in the integer Fourier coefficients after optimization of the edge locations. The first sequence (set of numbers from 32-128) represents an unperturbed set of Fourier data, the next four sequences show Fourier data with noise levels starting at $O(10^{-6})$ and going through $O(10^{-2})$. That is, instead of putting in the true Fourier coefficients, we perturb the Fourier coefficients by uniformly distributed complex noise of the indicated order as described. All errors here are calculated with respect to the ℓ^2 norm. As the table demonstrates for the test function f_6, given a zero-mean random perturbation of $\hat{f}(\omega)$ of order 10^{-q}, the PRM can still achieve approximately 10^{-q} error in the approximation $\tilde{f}(\mathbf{k})$ of $\hat{f}(\mathbf{k})$.

The first four rows of this table use clean Fourier data whereas the following five columns use noisy Fourier data perturbed as follows. The input data is $(\boldsymbol{\omega}, \hat{f}(\boldsymbol{\omega}) \pm \epsilon^l) = ((\omega_1, ..., \omega_N)^T, (\hat{f}(\omega_1) \pm \epsilon_1^l, ..., \hat{f}(\omega_N) \pm \epsilon_N^l)$, for $l = 6, 5, 4, 3, 2$, and where $\epsilon_j^l = 10^{-l}(h_j \frac{\sqrt{2}}{2} \pm g_j \frac{i\sqrt{2}}{2})$ is a complex uniform random variable (i.e., $h_j \sim U[0, 1],\ g_j \sim U[0, 1]$).

Although the accuracy of the initial estimates provided by the edge detection method in [16] seems to be sufficient, it is important to analyze the results of the PRM when these estimates are poor. In particular, since the method inevitably depends on some thresholding parameters, it is possible that it will detect more edges than are actually present (false alarms), or alternatively it may undercount their true number (missed edges). Fortunately, these two outcomes have very noticeable ramifications for the PRM method, allowing *a posteriori* testing to ensure the correct number and locations of detected edges. Specifically, when the PRM method is used with fewer edges than actually exist, or when the detected edges are far from their true locations, the residual of the least squares system in Algorithm 2 will be very large (i.e. $O(1)$). Thus, if the PRM yields a large residual, it is reasonable to try changing the thresholding parameters in the initial edge detection method. On the other hand, if the PRM detects more edges than are actually present in the function then one of two things can happen. It may be that the optimization procedure does a very poor job of minimizing the residual in the least squares system, in which case the same tact can be taken; changing the thresholding parameters in the edge detection code to get a new initial estimate may yield a much better approximation. It is also possible that even with more jumps than occur in the actual function that Algorithm 3 will still do a good job of minimizing the residual. In this case there is no need to do anything; in practice we have found that when the residual of the least squares system is small the error in the resampled integer nodes is also small. In other words, the residual of the least squares system seems to be a good measure for approximation error over the entire domain, even when the jump information is inaccurate.

4. Comparison to Other Methods

Two techniques used in the MRI community for non-Cartesian sampling trajectories include uniform resampling and convolutional gridding [4, 8]. In this section we compare the results of these methods, which

Table 1: Errors as a function of number of sampled modes and data noise level.

noise level	N	initial edge estimate error	optimized edge error by Alg. 2	$\frac{1}{\sqrt{N}}\left\|\left\|\tilde{f}(\mathbf{k})-\hat{f}(\mathbf{k})\right\|\right\|_2$
0	32	0.195052	0.00191624	0.000212243
	64	0.0900712	0.000144707	8.9075×10^{-6}
	96	0.0480141	2.49348×10^{-7}	4.67659×10^{-9}
	128	0.0358682	2.3906×10^{-10}	2.2088×10^{-12}
10^{-6}	32	0.195052	0.00191718	0.000212311
	64	0.0900712	0.000144469	8.96852×10^{-6}
	96	0.0480141	3.35315×10^{-6}	4.85715×10^{-7}
	128	0.0358682	1.62186×10^{-5}	5.10052×10^{-7}
10^{-5}	32	0.195052	0.0019022	0.000213289
	64	0.0900712	0.000162127	1.02454×10^{-5}
	96	0.0480141	0.000134914	5.87128×10^{-6}
	128	0.0358682	0.000262969	4.84864×10^{-6}
10^{-4}	32	0.195052	0.00202702	0.000233146
	64	0.0900712	0.00026634	4.27438×10^{-5}
	96	0.0480141	0.0015836	7.68323×10^{-5}
	128	0.0358682	0.00207123	5.51233×10^{-5}
10^{-3}	32	0.195052	0.00618942	0.00106399
	64	0.0900712	0.00232162	0.00045983
	96	0.0480141	0.00567023	0.000565949
	128	0.0358682	0.0113682	0.000558678
10^{-2}	32	0.195052	0.0525086	0.00623687
	64	0.0725973	0.0569577	0.00679365
	96	0.0480141	0.0216089	0.00574286
	128	0.0445016	0.0850294	0.00575455

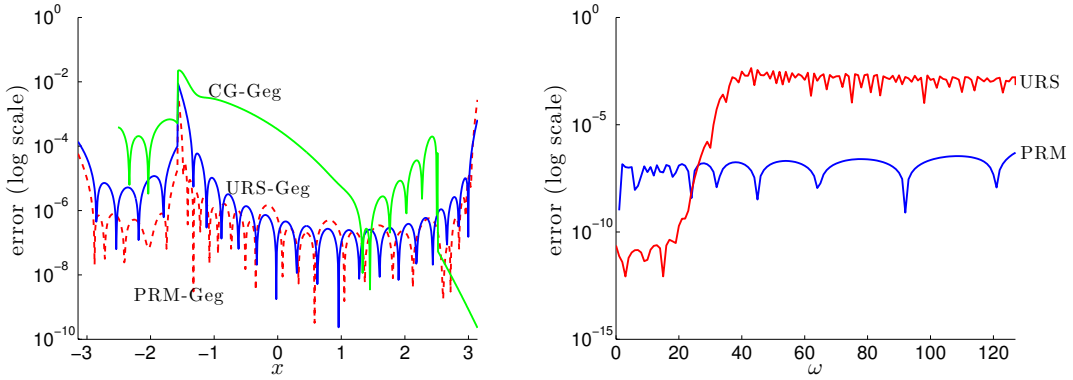

Figure 14: Left: Error in physical space in the recovery of f_7 for each of PRM, URS, and CG after post-processing with spectral reprojection. Right: Error in the Fourier domain, $|\tilde{f}_7(\omega) - \hat{f}_7(\omega)|$, for the URS and PRM.

are described below, to the PRM. Except where otherwise noted, the only input in the following experiments is the non-uniform Fourier data to allow for a fair comparison. In particular, no prior knowledge of the edge locations is assumed.

Uniform resampling (URS): The URS algorithm, like the PRM, first recovers the Fourier coefficients at equispaced nodes, and then uses the standard (filtered) FFT algorithm to reconstruct the underlying function. Under certain conditions (the Nyquist criterion), the non-uniform modes are related to the equispaced modes as

$$\hat{f}(\omega) = \sum_{l=-\infty}^{\infty} \hat{f}(l)\mathrm{sinc}(\omega - l). \tag{33}$$

The URS algorithm essentially truncates (33) and then solves the resulting finite linear system to obtain the needed integer values $\hat{f}(k)$, typically by applying the Moore-Penrose pseudo-inverse obtained by means of an SVD decomposition. Improvements to the computational efficiency of this method can be achieved by means of the "BURS" algorithm, also introduced in [4].

Convolutional gridding (CG): The CG algorithm is based on a discrete approximation of the convolution of the Fourier transform of the function with the Fourier transform $\hat{\phi}$ of a smooth "window" function ϕ. The convolved function can then be evaluated on a uniform grid, enabling standard FFT reconstruction. Finally, the window function ϕ is divided out from the Fourier partial sum to get an approximation to the standard reconstruction. It is important to note that convolutional gridding is heavily dependent on something called a density compensation factor (DCF). In our experiments we used iteratively computed weights inspired by those introduced in [3].

Since both the URS method and the PRM first approximate the Fourier coefficients at the integers, it is straightforward to compare their corresponding Fourier partial sums (32) or errors in Fourier space. By contrast, CG never directly approximates the Fourier coefficients of the underlying function, and hence it is more difficult to compare it to the PRM. To compare all three of these methods we apply spectral reprojection using Gegenbauer polynomials [15] to post-process the partial sum approximations generated by each method.

Figure 14 compares the PRM to the convolutional gridding and uniform resampling algorithms for the log sampling case, (29), for f_7. Figure 14(left) shows the error when spectral reprojection using Gegenbauer polynomials is applied to post-process the Fourier partial sum approximation [15]. The exact edge locations are assumed to be known. It is somewhat difficult to compare the results using spectral reprojection, since the parameters chosen in spectral reprojection can have some impact on the overall approximation, and in this case we see that for "optimal" parameters, the results using URS and PRM for f_7 are indeed quite similar.

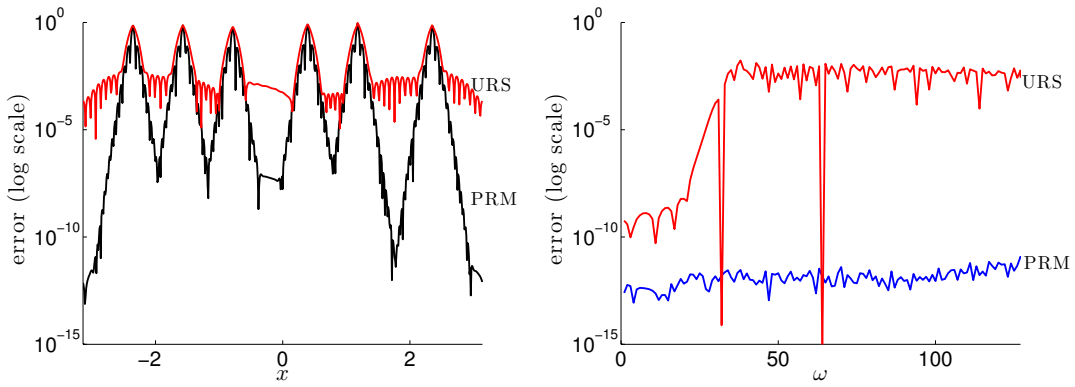

Figure 15: Left: Pointwise error of the filtered approximation in physical space using PRM and URS on f_6. Right: Corresponding pointwise error in Fourier domain.

However Figure 14(right) show siginificant difference in the URS and PRM approximations in the Fourier domain, in that URS exhibits particularly high accuracy in the lower frequencies while the PRM performs well throughout the domain. While the URS should never be solely relied upon to reconstruct a function with important high frequency information, such as in $f_7(x)$, these figures suggest that a combination of these approaches may be ideal in some cases.

This difference is further illustrated in Figure 15(left), which compares the filtered Fourier partial sum of f_6 for both schemes using an exponential filter [14]. No attempt was made to optimize the parameters. The corresponding error in the Fourier domain is shown in Figure 15(right). This figure was generated using $N = 128$ logarithmically sampled Fourier coefficients (29) with $v = 0$ and $\omega_{max} = 128$. The physical error is measured by the absolute difference between the physical function f_6 and the exponentially filtered Fourier partial sum created by PRM and URS respectively. The Fourier error is again measured by the absolute difference pointwise at the integers between the true Fourier coefficients and the integer Fourier coefficients resulting from each method.

Finally, we define the function $f_8 : [-\pi, \pi] \to \mathbb{R}$ by

$$f_8(x) = \begin{cases} \cos(10x) & |x| \leq \frac{\pi}{2} \\ 0 & \text{otherwise} \end{cases}$$

and use it to demonstrate that for functions with more variation, post-processing can be very effective for coefficients recovered by the PRM. Figure 16(left) shows the (pointwise) error for the filtered reconstruction in physical space by CG, URS, and PRM of f_8 using the exponential filter. The same filter was used in all cases, and the Fourier data (logarithmically sampled for $v = 0, N = 256$, and $\omega_{max} = 256$) was the only input into each method. Figure 16(right) shows the exponential decay in the error of the spectral reprojection approximation using Gegenbauer polynomials of f_8 for data sampled logarithmically (29) in $[1, 512]$. Figure 16(right) also assumes that the edge locations are known exactly.

5. Conclusions and Future Work

In this paper we demonstrate the Fourier transform of a piecewise smooth function can be approximated via a decomposition based on polynomials multiplied by complex exponentials. This decomposition exploits the relationship between jump locations in physical domain and frequency of oscillations in Fourier space. The approximation in Fourier space is guaranteed to yield algebraic convergence for piecewise smooth functions, with convergence rates that depends on the smoothness of each piece and decay of derivatives at infinity. For piecewise analytic functions with compact support, we show that the decomposition converges geometrically.

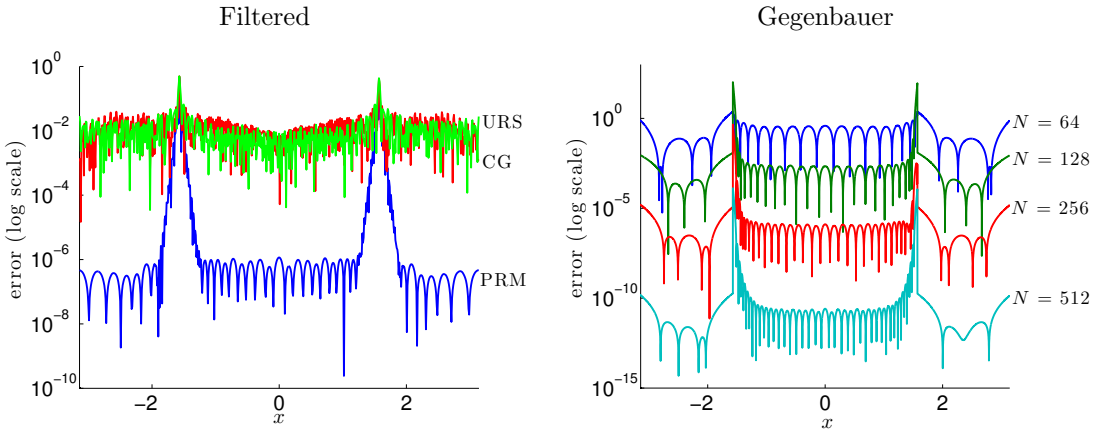

Figure 16: Pointwise error for post-processed non-uniform spectral data of f_8. The left plot shows the comparison between the exponentially filtered reconstruction for CG, URS, and PRM. The right graph shows the decay in the pointwise error of the Gegenbauer reprojection of the PRM Fourier coefficients as N varies.

The approximation is computed by a discrete least squares fit of data that is sampled more densely at low frequencies.

The method we develop here, the polynomial resampling method (PRM), exploits this decomposition. Once an approximation of the Fourier transform is obtained, it is evaluated at integer frequencies. Current edge detection algorithms, such as in [16], provide adequate initial estimates to ensure accuracy and robustness. Moreover, the PRM is able to improve the accuracy of the edge locations via a minimization technique. The PRM compares favorably in many one dimensional problems to several other currently used resampling methods, including uniform resampling and convolutional gridding. This is readily apparent when each method is post-processed to remove the Gibbs phenomenon.

The PRM is robust with respect to the distribution of the Fourier modes, and is amenable to logarithmic distributions. This feature is particularly attractive in applications such as MRI, since non-Cartesian sampling patterns are designed to collect more data in the low frequency range. Our numerical examples demonstrate robustness to noise in the Fourier data as well as to perturbations in the edge locations. Moreover, the PRM algorithm can be used for *a posteriori* testing, since false edges can often be observed in the minimization process. Future work includes investigating these ideas further, as well as extending the PRM to multiple dimensions. We point out, however, that the two-dimensional extension of the method is not trivial, as the two point evaluation in the integration by parts formula (3) becomes a contour integral along the jump discontinuity.

Acknowledgments

The authors thank Dr. Adityavikram Viswanathan for his very useful advice, edge detection code, and implementations of CG and Gegenbauer post-processing. Special thanks also to Ben Adcock for fruitful discussions on the solution of the least squares problem. Many of the numerical results presented were obtained with the aid of the Chebfun System [21, 23, 24]. We also thank the two anonymous reviewers for their valuable suggestions and comments.

References

[1] J. Jackson, C. Meyer, D. Nishimura, A. Macovski, Selection of a convolution function for Fourier inversion using gridding, IEEE T. Med. Imaging 10 (1991) 473–478.

[2] Y. Kadah, New solution to the gridding problem, in: Proc. SPIE, volume 4684, p. 2.

[3] J. Pipe, P. Menon, Sampling density compensation in MRI: Rationale and an iterative numerical solution, Magn. Reson. Med. 41 (1999) 179–186.

[4] D. Rosenfeld, An optimal and efficient new gridding algorithm using singular value decomposition, Magn. Reson. Med. 40 (1998) 14–23.

[5] J. Song, Q. Liu, Improving non-cartesian MRI reconstruction through discontinuity subtraction, Int. J. Bio. Imag. 2006 (2006) 1–9.

[6] A. Viswanathan, Imaging from Fourier Spectral Data: Problems in Discontinuity Detection, Non-harmonic Fourier Reconstruction and Point-spread Function Estimation, Ph.D. thesis, Arizona State University, 2010.

[7] A. Viswanathan, A. Gelb, D. Cochran, R. Renaut, On reconstruction from non-uniform spectral data, J. Sci. Comput. 45 (2010) 487–513.

[8] J. O'Sullivan, A fast sinc function gridding algorithm for Fourier inversion in computer tomography, IEEE T. Med. Imaging 4 (1985) 200–207.

[9] A. Dutt, V. Rokhlin, Fast Fourier transforms for nonequispaced data, SIAM J. Sci. Comput. 14 (1993) 1368–1393.

[10] A. Dutt, V. Rokhlin, Fast Fourier transforms for nonequispaced data, II, Appl. Comput. Harmon. A. 2 (1995) 85 – 100.

[11] K. Fourmont, Non-equispaced fast Fourier transforms with applications to tomography, J. Fourier Anal. Appl. 9 (2003) 431–450.

[12] L. Greengard, J.-Y. Lee, Accelerating the nonuniform fast Fourier transform, SIAM Rev. 46 (2004) 443–454.

[13] D. Gottlieb, S. Orszag, Numerical Analysis of Spectral Methods: Theory and Applications, volume 26, SIAM, Philadelphia, PA, 1993.

[14] J. Hesthaven, S. Gottlieb, D. Gottlieb, Spectral Methods for Time-Dependent Problems, volume 21, Cambridge University Press, Cambridge, UK, 2007.

[15] D. Gottlieb, C. Shu, On the Gibbs phenomenon and its resolution, SIAM Rev. 39 (1997) 644–668.

[16] W. Stefan, A. Viswanathan, A. Gelb, R. Renaut, Sparsity enforcing edge detection method for blurred and noisy Fourier data, J. Sci. Comput. 50 (2011) 1–21.

[17] J.-P. Berrut, L. N. Trefethen, Barycentric Lagrange interpolation, SIAM Rev. 46 (2004) 501–517.

[18] T. J. Rivlin, The Chebyshev Polynomials, Wiley-Interscience [John Wiley & Sons], New York, 1974. Pure and Applied Mathematics.

[19] J. C. Mason, D. Handscomb, Chebyshev Polynomials, Chapman and Hall/CRC, Boca Raton, 2003.

[20] P. J. Davis, Interpolation and Approximation, Dover Publications Inc., New York, 1975.

[21] Z. Battles, L. N. Trefethen, An extension of MATLAB to continuous functions and operators, SIAM J. Sci. Comput. 25 (2004) 1743–1770.

[22] R. E. A. C. Paley, N. Wiener, Fourier Transforms in the Complex Domain, volume 19 of *American Mathematical Society Colloquium Publications*, American Mathematical Society, Providence, RI, 1987.

[23] R. Pachón, R. B. Platte, L. N. Trefethen, Piecewise-smooth chebfuns, IMA J. Numer. Anal. 30 (2010) 898–916.

[24] L. N. Trefethen, et al., Chebfun Version 4.2, The Chebfun Development Team, 2011. http://www.maths.ox.ac.uk/chebfun/.

[25] Y. Katznelson, An Introduction to Harmonic Analysis, Cambridge Mathematical Library, Cambridge University Press, Cambridge, third edition, 2004.

[26] G. Golub, C. Van Loan, Matrix Computations, volume 3, Johns Hopkins Univ Pr, 1996.

[27] C. Moler, Numerical Computing with MATLAB, SIAM, Philadelphia, PA, 2004.

[28] M. Kadec, The exact value of the Paley-Wiener constant, in: Dokl. Akad. Nauk SSSR, volume 155, pp. 1253–1254.

[29] A. Gelb, E. Tadmor, Detection of edges in spectral data, Appl. Comput. Harmon. A. 7 (1999) 101.

[30] J. Nelder, R. Mead, A simplex method for function minimization, The Comp. J. 7 (1965) 308–313.