Noname manuscript No.
(will be inserted by the editor)

Detecting edges from non-uniform Fourier data using Fourier frames

Anne Gelb · Guohui Song

the date of receipt and acceptance should be inserted later

Abstract Edge detection plays an important role in identifying regions of interest in an underlying signal or image. In some applications, such as magnetic resonance imaging (MRI) or synthetic aperture radar (SAR), data are sampled in the Fourier domain. Many algorithms have been developed to efficiently extract edges of images when uniform Fourier data are acquired. However, in cases where the data are sampled non-uniformly, such as in non-Cartesian MRI or SAR, standard inverse Fourier transformation techniques are no longer suitable. Methods exist for handling these types of sampling patterns, but are often ill-equipped for cases where data are highly non-uniform or when the data are corrupted or otherwise not usable in certain parts of the frequency domain. This investigation further develops an existing approach to discontinuity detection, and involves the use of concentration factors. Previous research shows that the concentration factor technique can successfully determine jump discontinuities in non-uniform data. However, as the distribution diverges further away from uniformity so does the efficacy of the identification. Thus we propose a method that employs the finite Fourier approximation to specifically tailor the design of concentration factors. We also adapt the algorithm to incorporate appropriate smoothness assumptions in the piecewise smooth regions of the function. Numerical results indicate that our new design method produces concentration factors which can more precisely identify jump locations than those previously developed in both one and two dimensions.

Keywords Edge Detection · Fourier Frames · Non-Uniform Fourier Data

Department of Mathematics, Dartmouth College, 27 N. Main St., Hanover, NH 03755. (annegelb@math.dartmouth.edu)

Department of Mathematics, Clarkson University, Potsdam, NY 13699. (gsong@clarkson.edu)

Address(es) of author(s) should be given

1 Introduction

Edge detection is critical for identifying regions of interests in images in a variety of applications. For instance, in magnetic resonance imaging (MRI), edge detection aids in the process of tissue classification and tissue boundary identification, [25,20]. Determining the the edges in synthetic aperture radar (SAR) images can improve target identification. In these applications, data are acquired as Fourier samples. Since Fourier data are inherently global in nature, giving information about the underlying image as a whole, determining local features such as edges exclusively from Fourier samples is challenging. The concentration factor edge detection method, designed in [16], determines edges of a piecewise smooth function from a finite sampling of uniform Fourier data. This approach relies on a "concentration factor", which is essentially a bandpass filter, derived to satisfy certain admissibility conditions, to concentrate at the singular support of the underlying piecewise smooth function.

There have been various investigations on the use of concentration factors for non-uniform Fourier data. For example, in [14,15], concentration factors were explicitly determined to satisfy the admissibility criteria established in [16,17] for uniform data. In [28,30] the approach was altered so that the concentration factors were found by solving an optimization problem. Specific constraints on the problem allowed for customization. We follow the latter approach in this investigation, and improve the method further in two ways. First, we adopt the view in [14,15] that the given data represent Fourier frame data. In so doing, we obtain an accurate reconstruction when the Fourier data are non-uniformly sampled. Second, we modify the design so that additional smoothness constraints are satisfied away from the jump discontinuities. Such realistic assumptions yield a more refined approach to constructing concentration factors for non-uniform Fourier data. The first improvement extends the use of Fourier frame function reconstruction to detecting edges. The second improvement is a novel development of a high order expansion of the jump function, leading to additional constraints in the optimization problem and faster convergence to the underlying jump function. Our approach can also be modified to include other constraints for different applications.

Our method requires knowledge of the inverse (Fourier) frame operator, which we will approximate using the admissible frame algorithm developed in [27]. Of course, frames have been used for the purpose of reconstruction before, [8,5]. However, the method in [27] is shown to be convergent and it is also numerically efficient as it can be implemented using an NFFT, [15]. We note that recent investigations on stable reconstructions from non-uniform Fourier data [1–4] might also be useful in deriving the inverse frame operator approximations used in our method.

The rest of this paper is organized as follows. Section 2 discusses the concentration factor edge detection method for non-uniform Fourier data and lays the foundation for our problem formulation. In Section 3 we give a brief overview of frame theory, focusing specifically on results that are applicable to finite Fourier frames approximation approach to reconstruction. In Section 4

we formulate the optimization problem and design our concentration factors. Numerical results for two-dimensional problems are provided and discussed in Section 5. Section 6 contains concluding remarks.

2 Preliminaries

Consider an unknown, piecewise analytic function $f : \mathbb{R} \to \mathbb{R}$ that is supported on $[-1, 1]$. Later we will consider the two-dimensional case. The corresponding jump function $[f] : \mathbb{R} \to \mathbb{R}$ is defined as

$$[f](x) = f(x^+) - f(x^-). \tag{1}$$

Note that $[f]$ is well-defined since f is piecewise continuous and that $[f] = 0$ everywhere except at jump discontinuities, where it takes on the value of the jump. Since ultimately we will want to reconstruct the jump function on a grid, we discretize $x \in [-1, 1]$ as $\{x_j\}_{j=-J}^{J}$. Although there is no restriction on the distribution of grid points, for simplicity we define $x_j = \frac{j}{J}$. We also make the assumption that there is at most one jump within a cell $I_j = [x_j, x_{j+1})$. Thus, if $[f](x_j)$ is the value of the jump that occurs within the cell I_j, we can write

$$[f](x) = \sum_{j=-J}^{J-1} [f](x_j)\delta_{x_j}(x). \tag{2}$$

For the purposes of algorithmic development, we assume for now that there is only one discontinuity at $x = \xi \in (-1, 1)$. In this case,

$$[f](x) = [f](\xi)\delta_\xi(x), \tag{3}$$

where $\delta_\xi(x) = 1$ if $x = \xi$ and 0 otherwise. In 4 we demonstrate that our method also applies when there are multiple jumps.

Suppose we are given a finite sequence of Fourier samples of f,

$$\hat{f}(\lambda_k) = \int_{-1}^{1} f(x)e^{-\pi i \lambda_k x} dx, \tag{4}$$

where $\lambda_k \in \mathbb{R}$, $-M \leq k \leq M$, from which we seek an approximation to (3).

The assumption that f is sampled non-uniformly in Fourier space reflects the data distribution patterns in certain signal processing applications. Moreover, even technologies designed to sample uniformly sometimes experience uneven data distribution caused by machine error. To study an analogous one-dimensional problem, we consider three cases:

– Jittered sampling, defined by

$$\lambda_k = k \pm \nu_k, \qquad k = -M, \cdots M, \tag{5}$$

where ν_k are uniformly distributed within the interval and $|\theta| \leq 1/4$. This sampling pattern is designed to mimic samples taken on a theoretically Cartesian grid, while introducing the slight deviations that often occur in a real world setting.

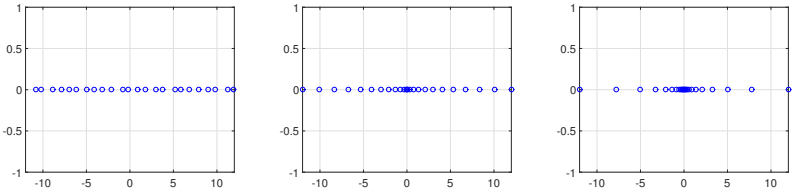
– Quadratic sampling, defined by

$$\lambda_k = sgn(k)\frac{1}{M}k^2, \qquad k = -M, \cdots, M. \tag{6}$$

This is a one-dimensional case designed to imitate a cross-section of a non-Cartesian sampling pattern that oversamples in the low frequencies, but is sparsely sampled in the high frequency part of the domain.

– Logarithmic sampling are acquired at logarithmic intervals, with more samples acquired in lower frequencies. Specifically, $\log|\lambda_k|$ is evenly distributed between $-v$ and $\log n$, with $v > 0$ and $2n+1$ being the total number of samples. This sampling scheme is even sparser than the quadratic sampling pattern in the high frequencies.

Figure 1 illustrates the sampling patterns that will be used to demonstrate our numerical results in one dimension.



(a) jittered sampling (b) quadratic sampling (c) logarithmic sampling

Fig. 1 Sampling patterns.

2.1 Edge detection from uniform data

Below is a brief review of the the concentration factor edge detection method, [16], which was developed to approximate (2) from the first $2M+1$ (uniform) Fourier coefficients of a piecewise smooth function f. It is given by

$$S_M^\sigma(f)(x) = i\sum_{|k|\le M} \hat{f}(k)\text{sgn}(k)\sigma\left(\frac{|k|}{M}\right)e^{\pi ikx}. \tag{7}$$

The convergence properties of (7) depend on the choice of factors $\sigma(u) = \sigma\left(\frac{|k|}{M}\right)$, termed *concentration factors*, which satisfy the admissibility conditions

1. $\displaystyle\sum_{k=1}^{M}\sigma\left(\frac{|k|}{M}\right)\sin(k\pi x)$ is odd
2. $\dfrac{\sigma(u)}{u} \in C^2(0,1)$

3. $\int_{\nu}^{1} \frac{\sigma(u)}{u} du \to -1$, $\nu = \nu(M) > 0$ being small

If these conditions are satisfied, $S_M^\sigma(f)$ "concentrates" at the singular support of f and the jump approximation obeys the concentration property, [11],

$$S_M^\sigma(f)(x) = [f](x) + \begin{cases} \mathcal{O}\left(\frac{\log M}{M}\right) & d(x) \leq \frac{\log M}{M} \\ \mathcal{O}\left(\frac{\log M}{(Md(x))^s}\right) & d(x) >> \frac{1}{M}. \end{cases} \tag{8}$$

Here, $d(x)$ denotes the distance between a point in the domain and the nearest discontinuity, while $s > 0$ depends on the concentration factor chosen. Note that (7) cannot be directly extended to the case when $\lambda_k \neq k$ because $\{e^{\pi i \lambda_k x}\}_{k=-M}^{M}$ does not form an orthogonal basis.

The partial sum approximation in (7) does not directly prescribe how the concentration factors should be chosen. Indeed, as observed in [18], the admissibility conditions allow for significant flexibility in the characterization of concentration factors, which can be exploited to refine the jump function approximation using non-linear post-processing algorithms. A different approach was suggested in [30,28] that uses the formulation in (7) to explicitly design concentration factors that yield certain desirable properties. This can be useful when certain assumptions can be made about the underlying function. Another advantage of this approach, which will be described later in Section 4.1, is that it provides a suitable framework for situations where the data are not acquired as standard uniform Fourier data. For example, in [30] the concentration factors were designed for the case when bands of Fourier data were not available, and in [28] the non-uniform case was considered.

3 Finite frame reconstruction

Our adaptation of the concentration factor edge detection method involves the finite Fourier frame approximation method described in [27]. This methodology makes use of results from frame theory, [19,10,21]. Below is a brief review of relevant results, starting with key definitions.

Definition 1 *A* frame *for a Hilbert space $\mathcal{H}$ is a sequence of vectors $\{\varphi_k : k \in \mathbb{Z}\} \subseteq \mathcal{H}$ for which there exists constants $0 < A \leq B < \infty$ such that, for every $f \in \mathcal{H}$, we have*

$$A||f||^2 \leq \sum_{k\in\mathbb{Z}} |\langle f, \varphi_k\rangle|^2 \leq B||f||^2. \tag{9}$$

In essence, a frame is a generalized notion of a basis which requires that the frame elements span $\mathcal{H}$ but are not necessarily linearly independent. For example, if $\mathcal{H}$ is finite dimensional, then any frame of cardinality greater than the dimension of $\mathcal{H}$ is linearly dependent. This property may be advantageous for function or image reconstructions since the associated "redundancy" can help recover losses during signal acquisition, [19].

Definition 2 *If $\{\varphi_k : k \in \mathbb{Z}\} \subseteq \mathcal{H}$ is a frame for $\mathcal{H}$ then the associated* frame operator $S : \mathcal{H} \to \mathcal{H}$ *is defined as*

$$Sf = \sum_{k\in\mathbb{Z}} \langle f, \varphi_k\rangle \varphi_k. \tag{10}$$

Observe that the conditions outlined in (9) indicate that its frame operator is bounded, invertible, positive, and self-adjoint, [19]. From this, we can conclude that both a right- and left-hand identity for S exists. In this paper, we will consider $\mathcal{H} = L^2(-1, 1)$ and $\varphi_k(x) = \mathrm{e}^{i\lambda_k \pi x}$. The left-hand identity is then used to reconstruct the underlying function as

$$f = S^{-1}Sf = \sum_{k\in\mathbb{Z}} \langle f, \varphi_k\rangle S^{-1}\varphi_k = \sum_{k\in\mathbb{Z}} \hat{f}(\lambda_k)\tilde{\varphi}_k, \tag{11}$$

where $\tilde{\varphi}_k = S^{-1}\varphi_k$, $k \in \mathbb{Z}$, is the canonical dual frame and $\hat{f}(\lambda_k)$ is given in (4).

3.1 Admissible frame approximation

Computing (11) is difficult since in general no closed form for constructing S^{-1} exists. In [27] a method was developed to construct a finite-dimensional approximation of $\tilde{\varphi}_k$ by projecting $\{\varphi_k : k \in \mathbb{Z}\}$ onto an *admissible frame* $\{\psi_l : l \in \mathbb{Z}\}$, which satisfies the following criteria:

Definition 3 *A frame $\{\psi_l : l \in \mathbb{Z}\}$ is admissible with respect to the frame $\{\varphi_k : k \in \mathbb{Z}\}$ if the following two conditions hold:*

1. *If it is intrinsically self-localized, that is, there exists $c_0 \in \mathbb{R}^+$ and $t > 1$ such that*
$$|\langle \psi_k, \psi_l\rangle| \le c_0(1 + |k - l|)^{-t}.$$
2. *There exists $c_1 \in \mathbb{R}^+$ and $s > 1/2$,*
$$|\langle \varphi_k, \psi_l\rangle| \le c_1(1 + |k - l|)^{-s}.$$

If $\{\psi_l : l \in \mathbb{Z}\}$ is an admissible frame with respect to $\{\varphi_k : k \in \mathbb{Z}\}$, then the dual frame $\{\tilde{\varphi}_k : k \in Z\}$ can be approximated [27] by

$$\tilde{\varphi}_k \approx \sum_{|l|\le N} b_{l,k}\ \psi_l =: \tilde{\varphi}_{N,k}, \tag{12}$$

where $b_{l,k}$ is the $(l, k)^{th}$ entry of the matrix $\mathbf{B} = \boldsymbol{\Psi}^\dagger$. Here $\dagger$ denotes the Moore-Penrose pseudo-inverse, and $\boldsymbol{\Psi}$ is given by

$$\Psi = [\langle \varphi_k, \psi_l\rangle],\ |k| \le M,\ |l| \le N.$$

In our case, since $\varphi_k(x) = e^{\pi i \lambda_k x}$, we choose $\psi_l(x) = e^{\pi i l x}$ to be the corresponding admissible frame yielding

$$\langle \varphi_k, \psi_l \rangle = \int_{-1}^{1} e^{\pi i (\lambda_k - \ell) x} = 2sinc(\lambda_k - l).$$

The approximation of $\tilde{\varphi}_k$ in (12) is then substituted into (11) to complete the reconstruction of f yielding the approximation

$$T_M f = \sum_{|l| \leq N} \sum_{|k| \leq M} \hat{f}(\lambda_k) b_{l,k} \psi_l = \sum_{|k| \leq M} \hat{f}(\lambda_k) \tilde{\varphi}_{N,k}(x). \tag{13}$$

In [27] it was shown that under certain smoothness assumptions on f and conditions on the relationship between M and N that $||T_M f - f||_{L^2(-1,1)} \to 0$ as $M \to \infty$.

4 Concentration factor design for non-uniform Fourier data

We now return to our previous discussion of determining edges of piecewise smooth functions from their non-uniform Fourier data. Since (7) does not apply in this case, we develop an approach that uses (13) to construct an approximation to (3).[1] First, let us define

$$T_M^\sigma f(x) = \sum_{|k| \leq M} \sigma_k \hat{f}(\lambda_k) \tilde{\varphi}_{N,k}(x) \tag{14}$$

as the approximation to $[f](x)$. We now seek $\sigma \in \mathbb{C}^{2M+1}$ such that, for a given set of reconstruction points, $x_j = \frac{j}{J}$, $-J \leq j \leq J$, for some $J \in \mathbb{Z}^+$, we have[2]

$$T_M^\sigma f(x_j) = \sum_{|k| \leq M} \sigma_k \hat{f}(\lambda_k) \tilde{\varphi}_{N,k}(x_j) \approx [f](\xi) \delta_\xi(x_j). \tag{15}$$

To formulate the optimization problem for determining σ in (15), we first make some observations regarding the underlying function, f. Specifically, if f is piecewise-analytic (with a finite number of jump discontinuities), then a superposition of scaled and shifted ramp functions, defined in (16), provides a first order approximation. While using such an approximation is ill-advised for *function* reconstruction, it is perfectly suitable for approximating $[f](x)$, as described below. Viewing edge detection in this way is advantageous since the Fourier coefficients of a ramp function are explicitly known.

[1] This idea was first investigated in [23].

[2] Recall that we assume that the discontinuities occur only on grid points x_j. For convenience we choose $x_j = \frac{j}{J}$, $-J \leq j \leq J$ so that the value $x = 0$ falls on the grid point x_0. The system can be designed for any chosen gridpoints, however.

We now proceed in constructing the optimization problem by defining the ramp function, $r(x) : [-1, 1] \to \mathbb{R}$ as

$$r(x) = \begin{cases} -\frac{x+1}{2} & x \leq 0 \\ -\frac{x-1}{2} & x > 0. \end{cases} \tag{16}$$

The associated jump function for $r_\xi(x) = r(x - \xi)$, where $\xi \in (-1, 1)$, is given by

$$[r_\xi](x) = \begin{cases} 1 & x = \xi \\ 0 & \text{otherwise.} \end{cases} \tag{17}$$

A first order approximation to $f(x)$ with a single jump discontinuity located at $x = \xi$ is then

$$f(x) \approx a r_\xi(x), \tag{18}$$

where $a \in R$. Note that the corresponding jump function of f is exactly $[f](x) = a[r_\xi](x)$, an idea that was exploited in [30] to design concentration factors from uniform Fourier data.

Remark 1 For functions with multiple discontinuities, we can replace (18) with

$$f(x) \approx \sum_{j=1}^{J} a_j r_{\xi_j}(x). \tag{19}$$

Due to the linearity of (15) on the given Fourier data, the results that follow for a single jump similarly hold for multiple jumps. Of course, the global nature of the Fourier data will cause additional interfering oscillations in each jump discontinuity neighborhood. Hence we assume that the jump discontinuities are located sufficiently far apart. More discussion on the validity of (15) for multiple jumps (in the uniform data case) can be found in [30]. We will also demonstrate its validity in our numerical experiments.

For the non-uniform case, we discretize $ar(x - \xi)$ and substitute into (15) to obtain

$$aT_M^\sigma r_\xi(x_j) = a \sum_{|k| \leq M} \sigma_k \widehat{r_\xi}(\lambda_k) \tilde{\varphi}_{N,k}(x_j) \approx a\delta_\xi(x_j), \tag{20}$$

where $\widehat{r_\xi}$ are the Fourier coefficients of $r_\xi(x)$. Translating the above system yields

$$\sum_{|k| \leq M} \sigma_k \hat{r}(\lambda_k) \tilde{\varphi}_{N,k}(x_j) \approx \delta_0(x_j), \tag{21}$$

where $\hat{r}(\lambda_k)$ are the Fourier coefficients of $r(x)$ explicitly given by

$$\hat{r}(\lambda_k) = \begin{cases} 0 & \lambda_k = 0 \\ \frac{(\sin(\pi\lambda_k) - \pi\lambda_k)i}{(\pi\lambda_k)^2} & \text{otherwise.} \end{cases} \tag{22}$$

We note that $\delta_\xi(x)$ only takes non-trivial values on a set of measure 0 and does not have a non-trivial Fourier expansion. Hence we will instead consider a smooth approximation of $\delta_\xi(x)$, given by

$$h_\xi(x) \approx \delta_\xi(x), \qquad \epsilon > 0, \tag{23}$$

where $h_\xi = h(\frac{x-\xi}{\epsilon})$ for some function h and positive constant ϵ. Typically h is an (essentially) compactly supported bump function such that h_ξ is supported in $[\xi-\epsilon, \xi+\epsilon]$ with $h_\xi(\xi) = 1$. As $\epsilon \to 0$, $h_\xi(x) \to \delta_\xi(x)$. The choice of ϵ is critical to how the method performs. As ϵ increases, the approximation is more regularized, but the edges are not as well localized. This trade off can be decided on a case by case basis, depending on other external influences, such as the amount of and corruption in the data, see e.g. [30,14]. Replacing $\delta_\xi(x)$ with $h_\xi(x)$ in (3) yields

$$[f](x) \approx [f](\xi)h_\xi(x). \tag{24}$$

Thus, (21) becomes

$$\sum_{|k|\le M} \sigma_k \hat{r}(\lambda_k)\tilde{\varphi}_{N,k}(x_j) \approx h_0(x) \approx \sum_{|k|\le M} \widehat{h_0}(\lambda_k)\tilde{\varphi}_{N,k}(x_j) \tag{25}$$

or more simply,

$$\sigma_k = \frac{\widehat{h_0}(\lambda_k)}{\hat{r}(\lambda_k)}, \quad k = -M, \cdots, M. \tag{26}$$

Remark 2 Several approximations are needed to obtain the explicit expression for σ_k in (26). First, since $\delta_\xi(x)$ does not have a non-trivial frame expansion, we use (23), which leads directly to the second approximation (24). The error is controlled by the width of ϵ. The final approximation in (25) is a finite truncation of the infinite frame expansion and a numerical dual frame approximation. Its error analysis is presented in [27].

This explicit form for σ in (26) is what was originally derived in [16] for different choices of h_ξ given uniform Fourier data, and in [14] in the non-uniform case. Some examples include

1. Step function

$$h_\xi(x) = \begin{cases} 1 & \text{if } \xi-\epsilon \le x \le \xi \le \xi+\epsilon \\ 0 & \text{else.} \end{cases} \tag{27}$$

2. Hat function

$$h_\xi(x) = \begin{cases} \frac{1}{\epsilon}(x-\xi+\epsilon) & \text{if } \xi-\epsilon \le x \le \xi \\ -\frac{1}{\epsilon}(x-\xi-\epsilon) & \text{if } \xi \le x \le \xi+\epsilon \\ 0 & \text{else.} \end{cases} \tag{28}$$

3. Gaussian

$$h_\xi(x) = exp\big(-5(\frac{x-\xi}{\epsilon})^2\big). \tag{29}$$

In this investigation, we chose $\epsilon = .07$ for all three cases of h_ξ and all sampling patterns. Observe that $\widehat{h_0}(\lambda_k)$ can be explicitly determined for all three cases and (14) directly applied. This straightforward approach has been demonstrated to be successful in pinpointing edges after some additional post-processing, [18]. We remark that h can also be chosen to be oscillatory, and in some applications this may be desirable since in this case h is more easily generated by a Fourier partial sum. Details of the properties of h can be found in [17]. No attempt has been made here to determine optimal smooth approximation functions and corresponding parameters.

We first consider

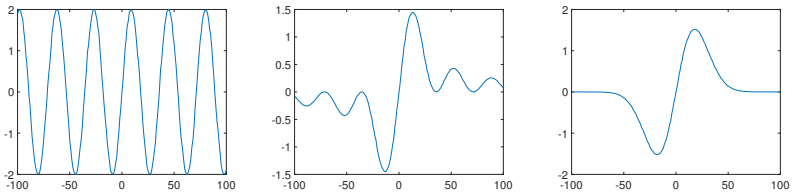
Example 1

$$f_1(x) = \begin{cases} \cos(\pi x - \frac{\pi x}{2}\,\mathrm{sign}(-x - 1/2)) & -1 \le x \le 0; \\ \cos(\frac{5\pi x}{2} + \pi x\,\mathrm{sign}(x - 1/2)) & 0 < x \le 1. \end{cases}$$

with

$$[f_1](x) = \begin{cases} -\sqrt{2} & x = -.5; \\ \sqrt{2} & x = .5; \\ 0 & \text{otherwise.} \end{cases}$$

Figure 2 displays the concentration factors (CF) determined from (26) using the three examples of $h_\xi(x)$ given in (27), (28), and (29), for the jittered sampling case with $M = N = 100$. The corresponding jump function approximations for Example 1, given by (14), are shown in Figure 3. In Section 4.1 we discuss how this slow convergence may be improved.



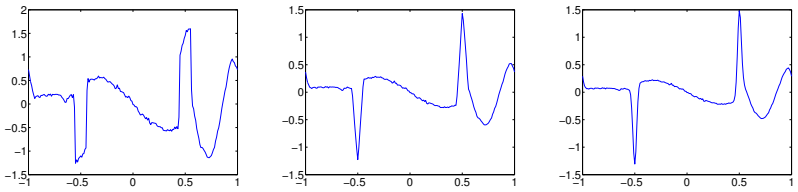
(a) CF (27) (b) CF (28) (c) CF (29)

Fig. 2 Concentration factors determined by (26) with three examples of $h_\xi(x)$.

We now consider

Example 2

$$f_2(x) = \begin{cases} -\frac{1}{2}(1 - x^2)^2, & -1 \le x \le -1/2; \\ \cos(4\pi x), & -1/2 < x < 1/2; \\ (1 - x^2)^4, & 1/2 \le x \le 1; \\ 0, & \text{otherwise.} \end{cases}$$



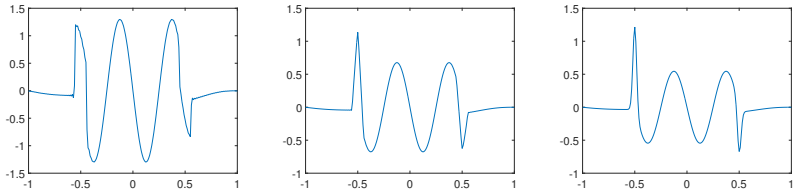
(a) $T_M^\sigma f_1](x)$ with (27) (b) $T_M^\sigma f_1(x)$ with (28) (c) $T_M^\sigma f_1(x)$ with (29)

Fig. 3 Approximation of $[f_1](x)$ using (15) with concentration factors determined by (26).

with

$$[f_2](x) = \begin{cases} \frac{42}{31} & x = -1/2; \\ -\frac{175}{256} & x = 1/2; \\ 0 & \text{otherwise.} \end{cases}$$

As before, for $M = N = 100$ and the jittered sampling case, we use (26) with (27), (28), and (29), to compute the concentration factors for the jump function approximation, (15). The results are displayed in Figure 4.



(a) $T_M^\sigma f_2(x)$ with (27) (b) $T_M^\sigma f_2(x)$ with (28) (c) $T_M^\sigma f_2(x)$ with (29)

Fig. 4 Approximation of $[f_2](x)$ using (15) with concentration factors determined by (26).

Clearly, using (26) to determine the concentration factor when given jittered Fourier data, (5), is even less effective for Example 2. All of the jump function approximations exhibit large oscillatory artifacts away from the two true jump locations, $\xi = \pm\frac{1}{2}$. Such artifacts, caused by regularizing $\delta_\xi(x)$, may lead to false detections away from the true edges, especially when the function has considerable variation in smooth regions. This issue has been addressed before, and has been mitigated by using non-linear post-processing on the results, [17,18]. The accuracy decreases as the Fourier data distribution becomes more irregular, however. Moreover, although there is an explicit formula for the concentration factors given in (25), there may be more suitable ways to construct the concentration factors given prior information about the function. For example, in [30] it was recognized that the sparsity of the jump function, $[f]$ in (2), could be exploited in developing concentration factors. The resulting algorithm, described in Section 4.1, has the additional advantage of taking into

consideration user concerns, such as the undesirability of oscillatory artifacts or corrupted and/or unusable data.

4.1 Exploiting sparsity of the jump function for concentration design

Determining concentration factors using (25) provides one type of regularization for (21). However, as was shown in [30] given uniform Fourier data, exploiting the sparsity of $[f]$ provides another method of regularization. In this case we introduce an optimization problem as a way to determine σ. We start by defining the operator

Definition 4 *Let $D_M^{\mathbf{x}} : \mathbb{C}^{2M+1} \to \mathbb{C}^{2J+1}$ be a linear operator such that*

$$(D_M^{\mathbf{x}}\sigma)_j = \sum_{|k|\le M} \sigma_k \hat{r}(\lambda_k)\tilde{\varphi}_{N,k}(x_j), \ \forall \sigma \in \mathbb{C}^{2M+1}. \tag{30}$$

In order to set up the optimization problem, we rewrite (30) as a linear system and define the matrix $\mathbf{D}$ as

$$\mathbf{D}_{k,j} = \hat{r}(\lambda_k)\tilde{\varphi}_{N,k}(x_j). \tag{31}$$

To satisfy (21), we therefore require

$$(\mathbf{D}\boldsymbol{\sigma})_j = \delta_{j,0} = \begin{cases} 1 & j = 0 \\ 0 & \text{elsewhere,} \end{cases} \tag{32}$$

where, as before, $x_j = \frac{j}{J}$, $j = -J, \cdots, J$, are the associated grid points yielding $x_0 = 0$. Hence (32) establishes a constraint for determining σ. To determine the cost function, observe that the underlying function f has only a single jump at 0, and that the jump function $[f](x)$ is zero almost everywhere except at $x = 0$. Thus, as was established in [30], we seek to minimize $\mathbf{D}\sigma$ over the domain $[-1, 1]$, leading to the constrained optimization problem:

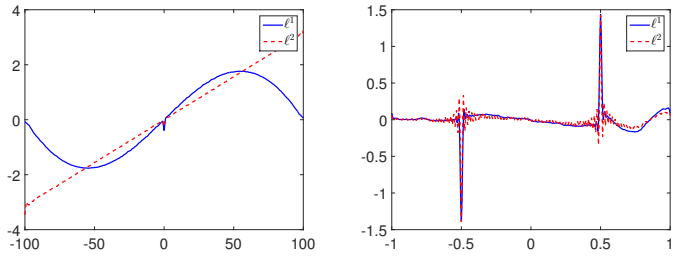
$$\underset{\sigma}{\text{minimize}} \quad ||\mathbf{D}\sigma|| \quad \text{subject to} \quad (\mathbf{D}\sigma)_0 = 1. \tag{33}$$

Following [30], we employed ℓ^1 and ℓ^2 regularization to solve (33), yielding the algorithm:

Algorithm 1 *[Concentration factor design for non-uniform Fourier data.][3] Define $\mathbf{D}$ as in (31). Determine σ as the solution to*

$$\underset{\sigma}{\textit{minimize}} \quad ||\mathbf{D}\sigma||_{\ell^1 or \ell^2} \quad \textit{subject to} \quad (\mathbf{D}\sigma)_0 = 1. \tag{34}$$

Once the vector of concentration factors σ is determined, it can be directly incorporated into (14) to approximate $[f](x)$. Figure 5(a) shows the concentration factors for jittered sampling with $M = N = 100$ using both the ℓ_1 and



(a) σ determined from Algorithm 1 (b) $T_M^\sigma f_1(x)$

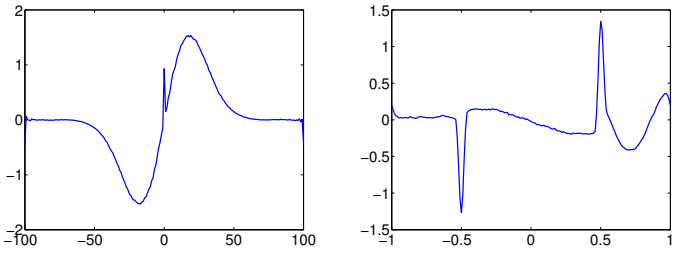
Fig. 5 $T_M^\sigma f_1(x)$ where σ is determined from Algorithm 1 given jittered Fourier samples. Here $M = N = 100$.

ℓ_2 minimization. Figure 5(b) displays the jump function approximation, (14), for Example 1 using the concentration factors calculated by Algorithm 1.

There are several ways to modify Algorithm 1 that may be advantageous under different circumstances. For example, in the case of uniform Fourier data, an extra constraint was added that forced $\sigma_k = 0$ whenever $k_1 \leq |k| \leq k_2$, (see [30]). This would be relevant in situations where the Fourier data in certain intervals are missing or corrupted. Another modification could be made by replacing $\delta_0(x)$ with $h_0(x)$. Specifically,

$$\underset{\sigma}{\text{minimize}} \quad \|\mathbf{D}\sigma\|_{\ell^1} \quad \text{subject to} \quad \|\mathbf{D}\sigma - \mathbf{h}\|_{\ell^2} < \delta, \tag{35}$$

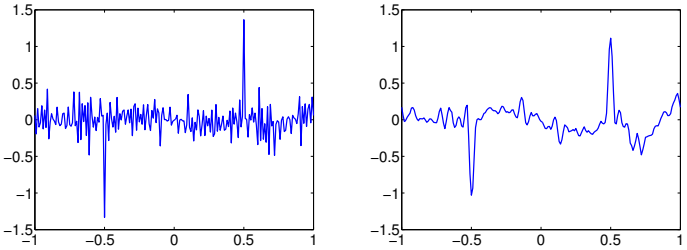
where $\mathbf{h} = \{h_0(x_j), -J, \leq j \leq J\}$, and $\delta > 0$ is some pre-determined error tolerance. Using (35) might be beneficial when the data are noisy or under-sampled. Figures 6 demonstrate the results using (35) for Example 1 with the jittered sampling pattern and $M = N = 100$, where h_0 is given by (29). In Figure 7, Gaussian noise with mean 0 and standard deviation of 0.02 was added to the Fourier data.



(a) σ determined using (35) (b) $T_M^\sigma f_1(x)$

Fig. 6 $T_M^\sigma f_1(x)$ where σ is determined from (35) given jittered Fourier samples. Here $M = N = 100$.

[3] The numerical results using Algorithm 1 were first reported in [23].



(a) $T_M^\sigma f_1(x)$ using Algorithm 1 (b) $T_M^\sigma f_1(x)$ using (35)

Fig. 7 $T_M^\sigma f_1(x)$ when given jittered noisy Fourier data. Here $M = N = 100$.

4.2 Refinement to concentration factor design

While Algorithm 1 and its modifications provide a general technique to determine the concentration factors for (15), it does not take into account any additional information about f. However, in many applications, f is in a smaller class of functions, for example, piecewise polynomials or piecewise trigonometric polynomials. In this case, assuming once again that f has a single jump discontinuity at $x = \xi$, we can write

$$f(x) = s(x) + [f](\xi) r_\xi(x), \tag{36}$$

where $r_\xi(x)$ is defined in (16), and $s(x)$ denotes the continuous part of f. In other words, (36) is a higher order extension of (18).

As before, we still seek the convergence of (20). However, we now add the constraint that T_M^σ in (15) satisfies

$$T_M^\sigma s(x_j) \approx 0$$

for all $x_j \in (-1, 1)$ since there is no jump in $s(x)$. By so doing, we require that the smooth regions of f be annihilated in the approximation of $[f](x)$.[4] Once again translating the jump to $x = 0$, we can construct the following system to determine σ from a set of grid points $x_j = \frac{j}{J}$, $j = -J, \cdots, J$, as

$$\begin{aligned} \sum_{|k|\le M} \sigma_k \hat{s}(\lambda_k) \tilde{\varphi}_{N,k}(x_j) &\approx 0 \\ \sum_{|k|\le M} \sigma_k \hat{r}(\lambda_k) \tilde{\varphi}_{N,k}(x_j) &\approx \delta_0(x_j). \end{aligned} \tag{37}$$

The system in (37) allows for some flexibility in setting up the optimization problem. As discussed previously, there are a variety of ways to approximate

[4] Indeed, a related idea was examined in [30] for suppressing higher order terms in (36) given uniform samples, but in this case we design $s(x)$ to more closely resemble the smooth part of the underlying function.

$\delta_\xi(x)$ that satisfy the admissibility conditions for concentration factors, for example those given in (27), (28) and (29).

We now propose an algorithm for designing concentration factors that exploits prior assumptions about the class of functions f may belong to. The idea is to determine *an optimal* $\hat{\boldsymbol{h}} = \{\hat{h}_k\}_{k=-M}^{M}$, where $\hat{h}_k = \widehat{h_0}(\lambda_k)$, by formulating a discrete minimization problem rather than using a pre-determined regularization that makes no assumption on $s(x)$. To do this, we will determine $\hat{\boldsymbol{h}}$ to satisfy the two constraints in (37). In this way we are able to further adapt the algorithm for the particular application in mind. For example, if it is expected that the underlying signal f can be decomposed as in (36), the concentration factor design in Algorithm 2 should yield optimal results. To this end, we rewrite the constraints (37) in terms of $\hat{\boldsymbol{h}}$ as

$$\sum_{|k|\le M} \frac{\hat{h}_k}{\hat{r}(\lambda_k)} \hat{s}(\lambda_k)\tilde{\varphi}_{N,k}(x_j) \approx 0 \quad \text{and} \quad \sum_{|k|\le M} \hat{h}_k \tilde{\varphi}_{N,k}(x_j) \approx \delta_0(x_j).$$

Algorithm 2 *[Refined Concentration factor design for non-uniform Fourier data.] Given* $2M+1$ *Fourier coefficients of a piecewise smooth function,* $\hat{f}(\lambda_k)$ *in (4):*

1. *Choose the smooth part* $s(x)$ *to be consistent with (36) and define*

$$\mathsf{F} = \left[\tilde{\varphi}_{N,k}(x_j)\right]_{j=-J,k=-M}^{J,M} \quad \textit{and} \quad \mathsf{S} = \left[\frac{\hat{s}(\lambda_k)}{\hat{r}(\lambda_k)}\tilde{\varphi}_{N,k}(x_j)\right]_{j=-J,k=-M}^{J,M}.$$

2. *Determine* $\hat{\boldsymbol{h}}$ *as the solution to*

$$\min_{\hat{\boldsymbol{h}}\in\mathbb{R}^{2M+1}} \|\mathsf{F}\hat{\boldsymbol{h}}\|_{\ell^1} + \mu\|\mathsf{S}\hat{\boldsymbol{h}}\|_{\ell^1},$$

   for $x_j = \frac{j}{J}, j = -J, \cdots J$ *and some positive constant* μ.
3. *Define* $\sigma_k = \frac{\hat{h}_k}{\hat{r}(\lambda_k)}$, $k = -M, \cdots, M$.

We chose $\mu = 1000$ for each performance of Algorithm 2. As before, once the vector of concentration factors σ is determined, it can be directly incorporated into (14) to approximate $[f](x)$. It is also straightforward to modify Algorithm 2 to account for missing bands of data, or to impose other constraints on the solution. What remains is how to choose $s(x)$, which depends on what prior information is known about the underlying function f. For example, choosing $s(x) = 0$ returns Algorithm 1. Here we consider two other options.

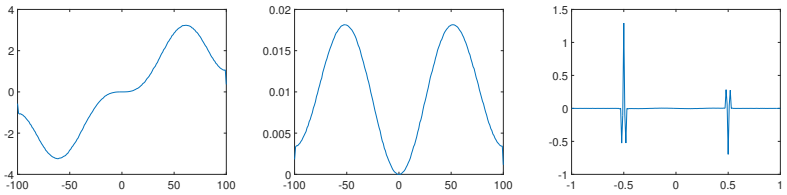
1. $s_1(x)$ is a hat function

$$s_1(x) = \begin{cases} -\frac{x+1}{2}, & -1 \le x \le 0; \\ \frac{x-1}{2}, & 0 < x \le 1. \end{cases} \tag{38}$$

2. $s_2(x)$ is some varying smooth function

$$s_2(x) = \begin{cases} -\frac{(x+1)^3}{12}, & -1 \le x \le 0; \\ -\frac{(x-1)^3}{12} - \frac{1}{6}, & 0 < x \le 1. \end{cases} \tag{39}$$

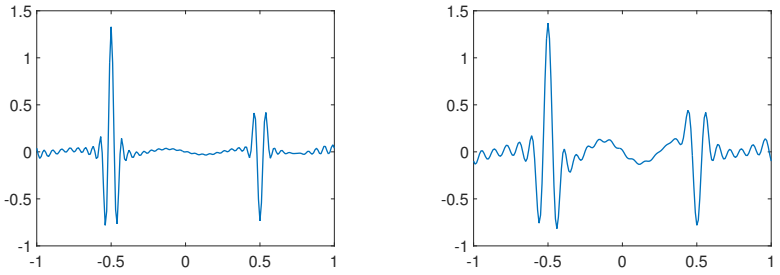
We consider these two cases to represent very different scenarios. On the one hand, $s_1(x)$ suggests that the underlying function has discontinuities in the derivatives as well. On the other hand, $s_2(x)$ may represent the situation where the underlying function is basically piecewise constant or linear with smooth variation between the jumps. For example, variation may be apparent in different tissue types, but should not be misconstrued as being discontinuous.



(a) CF from Algorithm 2 (b) $\hat{\boldsymbol{h}}$ from Algorithm 2 (c) $T_M^\sigma f_2(x)$

Fig. 8 $T_M^\sigma f_2(x)$ by Algorithm 2 for the jittered sampling pattern and $M = N = 100$.

We first consider $f_2(x)$. Observe that $f_2'(x)$ also has discontinuities at the locations. Thus using $s_1(x)$ in (36) increases the expansion order of f. Figure 8 demonstrates the results using Algorithm 2 using jittered sampling and $M = N = 100$, while Figure 9 displays the corresponding results for the quadratic ($M = 100$, $N = 40$) and logarithmic ($M = 100$, $N = 25$) sampling. Observe that as the sampling becomes increasingly non-uniform, the ratio of N to M decreases. This is consistent with the results in [27] for admissible frames.



(a) quadratic sampling ($N = 40$) (b) logarithmic sampling ($N = 25$)

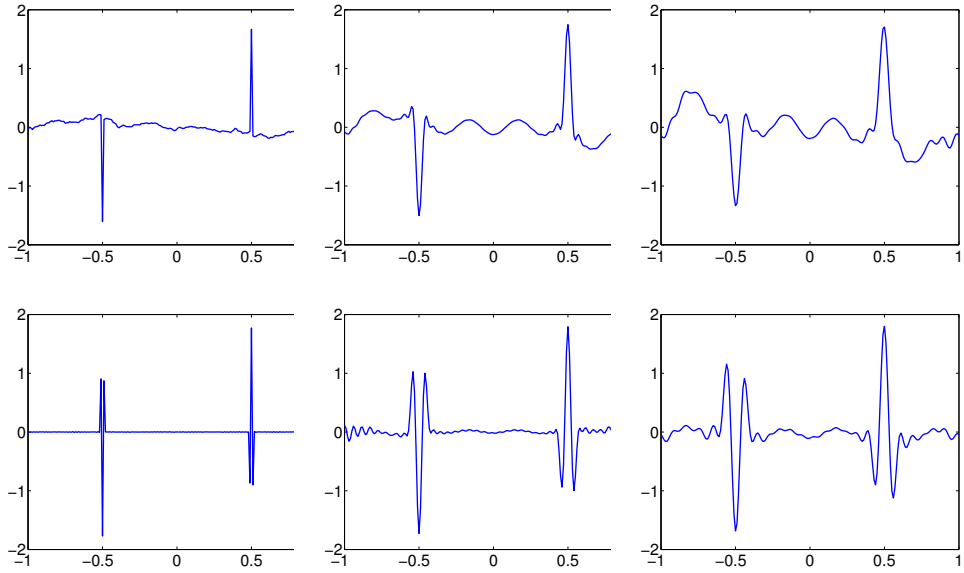
Fig. 9 $T_M^\sigma f_2(x)$ by Algorithm 2 for the quadratic and logarithmic sampling patterns and $M = 100$.

To demonstrate the efficacy of Algorithm 2 using $s_2(x)$, we consider

Example 3

$$f_3(x) = \begin{cases} \pi(1-x^2)^2, & -1 \le x \le -1/2 \text{ or } 1/2 \le x \le 1; \\ -\frac{1}{6}\sin(6\pi x), & -1/2 < x < 1/2; \\ 0, & \text{otherwise.} \end{cases}$$

$$[f_3](x) = \begin{cases} -\frac{9}{16}\pi, & x = -1/2; \\ \frac{9}{16}\pi, & x = 1/2; \\ 0, & \text{otherwise.} \end{cases}$$



(a) jittered sampling ($N = 100$)(b) quadratic sampling ($N = 40$) (c) log sampling ($N = 25$)

Fig. 10 $T_M^\sigma f_3(x)$ using Algorithms 1 (top row) and 2 (bottom row) for Example 3 with jittered sampling (a); quadratic sampling (b); and logarithmic sampling (c). Here $M = 100$.

Observe that $f_3(x)$ does not have discontinuities in the derivative, which suggests that $s_2(x)$ in (39) will give a better approximation in (36). Figure 10 compares $T_M^\sigma f_3(x)$ using Algorithms 1 and 2 for the three sampling cases when $M = 100$.

4.3 Using l^1 regularization to recover $[f](x)$

Algorithms 1 and 2 provide a mechanism to recover concentration factors σ_k, $k = -M, \cdots, M$, which are in turn used in (14) to recover $[f](x)$. In

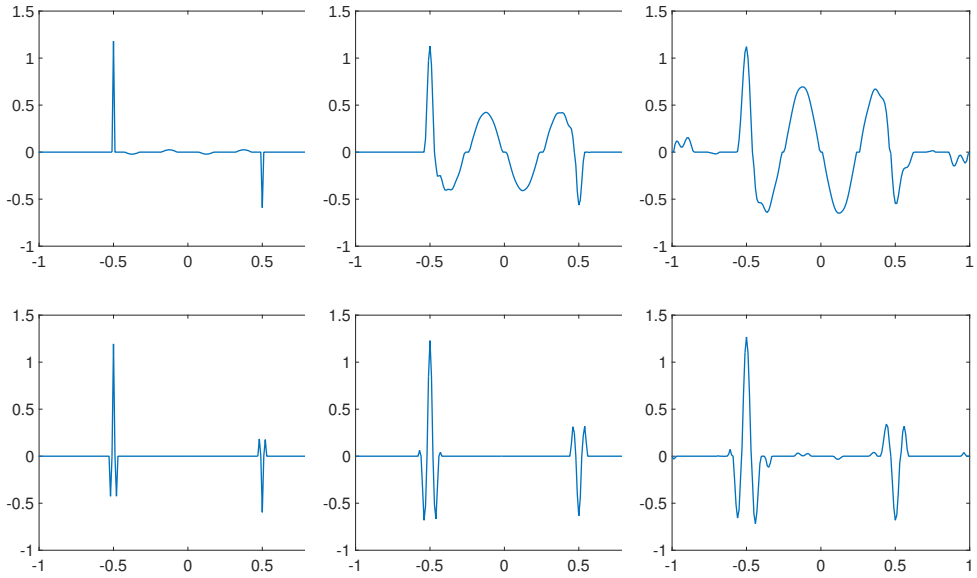
fact, the approximation can be further refined by exploiting the sparsity of $[f](x)$ which is evident in (2). This was first suggested in [28] for explicitly defined concentration factors. Using the results from the current investigation, the algorithm for recovering $[f](x)$ from non-uniform Fourier data is given as follows:

Algorithm 3 *[Reconstruction of $[f](x)$ from non-uniform Fourier data.] Given $2M+1$ Fourier coefficients of a piecewise smooth function, $\hat{f}(\lambda_k)$ in (4):*

1. *Determine σ from either Algorithm 1 or 2.*
2. *Define $\mathbf{T}$ such that $T(j,k) = \sigma_k \hat{f}(\lambda_k)\tilde{\varphi}_{N,k}(x_j)$, $k = -M, \cdots, M$ and $j = -J, \cdots, J$. Note that $T(j,0) = 0$. Also let $\hat{\mathbf{f}}$ be the vector of Fourier coefficients $\hat{f}(\lambda_k)$, $k = -M, \cdots, M$.*
3. *Define $\mathbf{p}$ as the vector with elements $[f](x_j)$, $j = 0, \cdots, J$. We seek $\mathbf{p}$ as the solution to*

$$\underset{\mathbf{p}}{minimize} \|\mathbf{p}\|_{\ell^1} + \mu\|\mathbf{p} - \mathbf{T}\hat{\mathbf{f}}\|_{\ell^2} \tag{40}$$

Figure 11 displays the results using Algorithm 3 for approximating $[f_2](x)$. We chose $\mu = 10$ for each performance of Algorithm 3.



(a) jittered sampling ($N = 100$) (b) quadratic sampling ($N = 40$) (c) log sampling ($N = 25$)

Fig. 11 Approximation of $[f_2](x)$ using Algorithms 1 & 3 (top row) and Algorithms 2 & 3 (bottom row) with three sampling patterns and $M = 100$.

As observed in Figure 11, although using Algorithm 3 improves the edge approximation, there are still large side lobes that occur in the neighborhoods

of the jumps. This is easily explained by the fidelity term in Algorithm 3. Namely, we are seeking a solution $[f]$ that most closely matches the projection of $\sum_{l=1}^{L}[f](\xi_l)h_{\xi_l}(x)$, where L is the number of edges, onto the space spanned by the Fourier frame elements given by (15). As such, the solution to (33) will be oscillatory. Since h_ξ is typically (essentially) compactly supported, in [28] it was noted that a better match to the jump function used in the fidelity (ℓ^2) term could be obtained using the approximation

$$W_M^\sigma * [f] \approx S_M^\sigma(f), \tag{41}$$

where $S_M^\sigma(f)$ is defined in (7) for some admissible σ. Here the waveform kernel

$$W_M^{\boldsymbol{\sigma}}(x) = \frac{1}{\gamma_M^\sigma}\sum_{k=-M}^{M}\sigma_k\frac{\cos kx}{k}, \tag{42}$$

where γ_M^σ is a normalization constant, is the Fourier partial sum approximation of $h_0(x)$ given uniform Fourier data, [28,30]. Observe that $W_M^{\boldsymbol{\sigma}}(x)$ in (42) is equivalent to the left hand side of (21) when $\lambda_k = k$. Thus to determine the corresponding waveform kernel for non-uniform Fourier data, we rewrite the left hand side of (21) using (22) as

$$\begin{aligned} W_{N(M)}^\sigma(x) &= \frac{1}{\gamma_{N(M)}^\sigma}\sum_{|k|\le M}\sigma_k\hat{r}(\lambda_k)\tilde{\varphi}_{N,k}(x) \\ &= \frac{1}{\gamma_{N(M)}^\sigma}\sum_{|k|\le M}\sum_{|l|\le N}\sigma_k\hat{r}(\lambda_k)b_{l,k}\mathrm{e}^{\pi i l x}, \end{aligned} \tag{43}$$

where $\gamma_{N(M)}^\sigma = \sum_{|k|\le M}\sum_{|l|\le N} b_{l,k}\sigma_k\hat{r}(\lambda_k)$ is the normalization constant.

Analogously to (41), we now write

$$W_{N(M)}^\sigma * [f] \approx T_M^\sigma(f). \tag{44}$$

To satisfy (41), we require their Fourier coefficients to be "close", that is

$$\frac{1}{\gamma_{N(M)}^\sigma}\sum_{|k|\le M} b_{l,k}\sigma_k\hat{r}(\lambda_k)\widehat{[f]}(l) \approx \sum_{|k|\le M} b_{l,k}\sigma_k\hat{f}(\lambda_k), \quad |l|\le N. \tag{45}$$

Let $\hat{\boldsymbol{f}} = (\hat{f}(\lambda_k) : |k| \le M)$ denote the set of non-harmonic measurements of the Fourier data and suppose $\boldsymbol{\sigma}$ is determined from either Algorithm 1 or Algorithm 2. We define

$$\boldsymbol{\Sigma} := \mathrm{diag}\Big(\frac{1}{\gamma_{N(M)}^\sigma}\sum_{|k|\le M} b_{l,k}\sigma_k\hat{r}(\lambda_k) : |l| \le N\Big), \tag{46}$$

and

$$\boldsymbol{y} := (\sum_{|k|\le M} b_{l,k}\sigma_k\hat{f}(\lambda_k) : |l| \le N). \tag{47}$$

We also denote $F \in \mathbb{C}^{(2N+1)\times(2J+1)}$ as the discrete non-harmonic Fourier matrix with elements

$$\mathsf{F}_{kj} = \exp\left[ikx_j\right], \quad k = -N, \ldots, N; \; j = -J, \ldots, J. \tag{48}$$

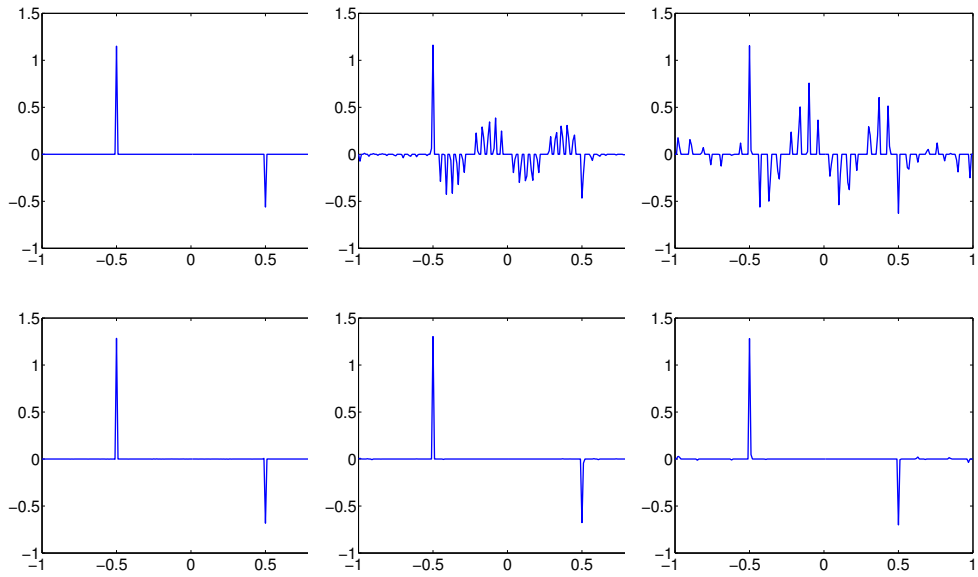
Using (46), (47) and (48) for the requirement (45), we arrive at the following alternative algorithm:[5]

Algorithm 4 *[Reconstruction of $[f](x)$ from non-uniform Fourier data.] Given $2M+1$ Fourier coefficients of a piecewise smooth function, $\hat{f}(\lambda_k)$ in (4):*

1. *Determine $\boldsymbol{\sigma}$ from either Algorithm 1 or 2.*
2. *Construct Σ, $\boldsymbol{y}$, and $\mathbf{F}$ from (46), (47) and (48).*
3. *Define $\boldsymbol{p}$ as the vector with elements $[f](x_j)$, $j = -J, \cdots, J$. We seek $\boldsymbol{p}$ as the solution to*

$$\min_{\boldsymbol{p}} \|\boldsymbol{p}\|_{\ell^1} + \mu \|\Sigma \mathsf{F} \boldsymbol{p} - \boldsymbol{y}\|_{\ell^2} \tag{49}$$

Figure 12 displays the results using Algorithm 4 for approximating $[f_2](x)$. We chose $\mu = 100$ for each performance of Algorithm 4.



(a) jittered sampling ($N = 100$)(b) quadratic sampling ($N = 40$) (c) log sampling ($N = 25$)

Fig. 12 Approximation of $[f_2](x)$ using Algorithms 1 & 4 (top row) and Algorithms 2 & 4 (bottom row) with three sampling patterns and $M = 100$.

[5] Algorithm 4 closely follows the one provided in [28] for non-uniform coefficients, although the values obtained in (46) and (47) are substantially refined by Algorithms 1 and 2.

We note that Algorithm 4 yields better results than the algorithm developed in [28] since the non-uniformity of the samples is considered both in the design of the concentration factor and in the jump function reconstruction. By contrast, the algorithm in [28] uses convolutional gridding with density compensation factors given by the trapezoidal rule for the jump function reconstruction, which admits additional interpolation errors, [31]. It also employs a concentration factor designed to be admissible for the corresponding *uniform case*, i.e. (7). Thus Algorithm 4 is particularly effective when the samples become increasingly non-uniform. This will be critical in the two-dimensional applications discussed below.

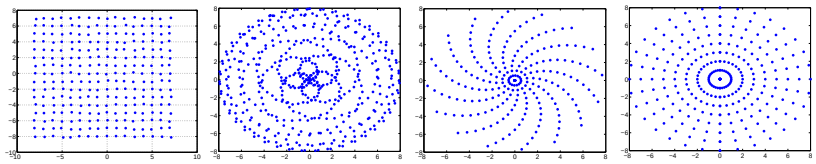
5 Edge detection for two-dimensional non-uniform Fourier data

We now discuss how to extend Algorithm 2 to two-dimensional non-uniform Fourier data. Specifically, we will consider a piecewise smooth function $f : [-1,1]^2 \to \mathbb{R}$ and try to recover its edge map from given non-uniform Fourier data:

$$\hat{f}(\boldsymbol{\lambda_k}) = \int_{-1}^{1}\int_{-1}^{1} f(x,y)\mathrm{e}^{-\pi i \lambda_{\boldsymbol{k},1} x}\mathrm{e}^{-\pi i \lambda_{\boldsymbol{k},2} y}dxdy,$$

where $\{\boldsymbol{\lambda_k} = (\lambda_{\boldsymbol{k},1}, \lambda_{\boldsymbol{k},2}) : |\boldsymbol{k}| \leq M\} \in \mathbb{R}^2$. We adopt the convention $|\boldsymbol{k}| \leq M$ to denote $|k_1| \leq M$ and $|k_2| \leq M$. We will consider the following non-uniform sampling patterns.

1. Jittered sampling:
$$\lambda_{\boldsymbol{j}} = \boldsymbol{j} + \boldsymbol{\epsilon_j},$$
where $\boldsymbol{\epsilon_j}$ is a small two-dimensional perturbation. Typically, we assume $\boldsymbol{\epsilon_j} \in [-1/4, 1/4]^2$. Note that the jittered sampling case constitutes a frame in $L^2[-1,1]$ when the perturbations lie in such a range, [29].
2. Rosette sampling:
$$\lambda_j = k_{max}(\cos(w_1 t_j)\cos(w_2 t_j), \cos(w_1 t_j)\sin(w_2 t_j)),$$
where k_{max}, w_1, w_2 are positive constants and $t_j \in [0,T]$ for some $T > 0$.
3. Spiral sampling:
$$\lambda_{\boldsymbol{j}} = (c\theta_{\boldsymbol{j}}\cos(2\pi\theta_{\boldsymbol{j}}), c\theta_{\boldsymbol{j}}\sin(2\pi\theta_{\boldsymbol{j}})),$$
where $c > 0$ and $\theta_{\boldsymbol{j}} > 0$.
4. Polar sampling:
$$\lambda_{\boldsymbol{j}} = (cr_{j_1}\cos(\theta_{j_2}), cr_{j_1}\cos(\theta_{j_2})),$$
where $c > 0$, $r_{j_1} = \frac{j_1}{R} \in [-1/2, 1/2)$, and $\theta_{j_2} = \frac{\pi j_2}{T} \in [-\pi/2, \pi/2)$.



(a) Jittered (b) Rosette (c) Spiral (d) Polar

Fig. 13 Examples of various sampling patterns

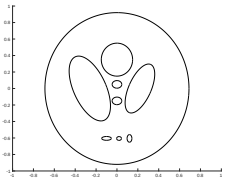


Fig. 14 The edge map of a synthetic phantom.

We display an example for each of the above sampling patterns in Figure 13.

We will test our algorithm on the classical Shepp Logan phantom, seen in Figure 14.

A two-dimensional inverse frame operator approximation technique was developed in [26]. As in the one-dimensional case, we choose the Fourier basis $\psi_{\mathbf{l}}(\mathbf{x}) = e^{\pi i \mathbf{l}\cdot\mathbf{x}}$ to be the corresponding admissible frame, so that f can be approximated from its non-uniform Fourier data $\hat{f}(\boldsymbol{\lambda}_{\boldsymbol{k}})$ as

$$T_M f(x,y) = \sum_{|\boldsymbol{l}|\le N} \sum_{|\boldsymbol{k}|\le M} \hat{f}(\boldsymbol{\lambda}_{\boldsymbol{k}}) b_{\boldsymbol{l},\boldsymbol{k}} \mathrm{e}^{\pi i l_1 x} \mathrm{e}^{\pi i l_2 y}, \tag{50}$$

where $\mathsf{B} = [b_{\boldsymbol{l},\boldsymbol{k}}]_{|\boldsymbol{l}|\le N,|\boldsymbol{k}|\le M}$ is the Moore-Penrose pseudo-inverse of $\Psi = [4\,\mathrm{sinc}(\boldsymbol{\lambda}_{\boldsymbol{k},1} - l_1)\,\mathrm{sinc}(\boldsymbol{\lambda}_{\boldsymbol{k},1} - l_1)]_{|\boldsymbol{k}|\le M,|\boldsymbol{l}|\le N}$. It was shown in [26] that for f smooth enough and M and N satisfying a certain relationship, $\|T_M f - f\| \to 0$ uniformly.

As in [6,7,13], we use a line-by-line approach to determine the edges in each direction while holding the other direction fixed. In particular, we construct

$$T_M^x f(x,y_s) = \sum_{|\boldsymbol{l}|\le N} \sum_{|\boldsymbol{k}|\le M} \hat{f}(\boldsymbol{\lambda}_{\boldsymbol{k}}) b_{\boldsymbol{l},\boldsymbol{k}} \mathrm{e}^{\pi i l_1 x} \mathrm{e}^{\pi i l_2 y_s}$$

and

$$T_M^y f(x_t,y) = \sum_{|\boldsymbol{l}|\le N} \sum_{|\boldsymbol{k}|\le M} \hat{f}(\boldsymbol{\lambda}_{\boldsymbol{k}}) b_{\boldsymbol{l},\boldsymbol{k}} \mathrm{e}^{\pi i l_1 x_t} \mathrm{e}^{\pi i l_2 y},$$

for each $y_s \in [-1,1]$ and $x_t \in [-1,1]$, and then apply Algorithm 3 with Algorithm 2 as the initial step on each directional approximation. The directional edge maps are then combined as

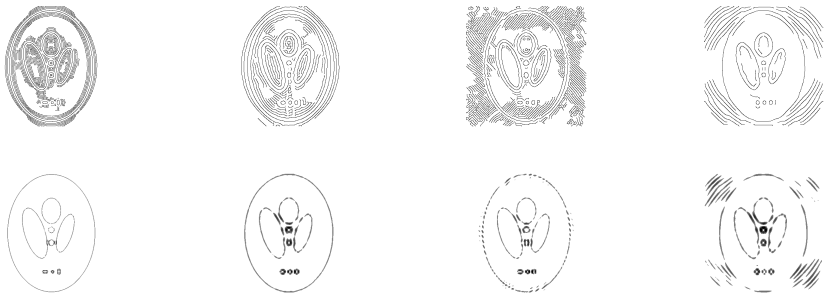
$$T_M^\sigma[f](x_t,y_s) = \max_{x_t,y_s}(|T_M^{x,\sigma} f(x_t,y_s)|, |T_M^{y,\sigma} f(x_t,y_s)|) \tag{51}$$

or

$$T_M^\sigma[f](x_t, y_s) = \sqrt{(T_M^{x,\sigma} f(x_t, y_s))^2 + (T_M^{y,\sigma} f(x_t, y_s))^2} \tag{52}$$

to form the final edge map of f. Although there were no noticeable differences in our experiments using (51) or (52), it is possible that further post-processing or other data considerations may favor one to the other. Further, $T_M^\sigma[f](x_t, y_s)$ can be constructed using other combinations as well.

It is important to note that our method does not require prior reconstruction of the image for the purposes of edge detection. As was pointed out in [22,24], the oscillations produced in the Fourier reconstruction of the image makes it difficult to employ a pixel based edge detector, while filtering the reconstruction causes some edges to go unidentified. We demonstrate this again in Figure 15, where we compare our edge detection algorithm to the Canny edge detector, [9], for the Shepp Logan phantom on the four different sampling schemes. In the latter case, the physical image is first obtained from the non-uniform Fourier data using (50). We note that the method developed in [22] is essentially an example of using (26) in two dimensions with a separable function h. The convolutional gridding machinery from [12] is then employed. Thus it does not benefit from the more accurate frame theoretic NFFT developed in [26], nor from the refinement of the concentration factor design presented here.



(a) Jittered ($N = 100$) (b) Rosette ($N = 50$) (c) Spiral ($N = 60$) (d) Polar ($N = 45$)

Fig. 15 Edge maps using the Canny edge detection method (top row) on the reconstructed image and a line-by-line approach for Algorithms 2 & 4 (bottom row) for various two-dimensional non-uniform sampling patterns. Here $M = 100$.

6 Conclusion

In this investigation we presented a new methodology for constructing an edge map from non-uniform Fourier data. Our algorithm leads to marked improvement over those developed in [14,15,22,28]: Higher order concentration

factors are designed based on the sampling trajectory of the given Fourier data, and l^1 regularization is used to pinpoint the edges. Our algorithm is effective in one and two dimensions, especially for sampling patterns that are prototypical of those seen in magnetic resonance imaging. Further thresholding, designed for any particular application, can be employed as needed. Our results also demonstrate that our method is robust with respect to noise.

It is interesting to note that the rosette, spiral, and polar sampling patterns considered in this paper do not form Fourier frames, since in each case we assume that the finite set of non-uniform Fourier data comes from an infinite Fourier frame with common frame bounds that are independent of the number of samples. Nevertheless, using the frame theoretic approach is effective for these data sets. Future work will include testing our algorithm on MRI and SAR data. In each case, the algorithm will be adapted to the particular constraints on the data acquisition, specifically in designing the concentration factors.

Acknowledgments

This work is supported in part by grants NSF-DMS 1216559 (AG), NSF-DMS 1521600 (AG), NSF-DMS 1521661 (GS), NSF 1502640 (AG), and AFOSR FA9550-15-1-0152 (AG).

References

1. Adcock, B., Gataric, M., and Hansen, A. On stable reconstructions from nonuniform Fourier measurements. *SIAM Journal on Imaging Sciences 7*, 3 (2014), 1690–1723.
2. Adcock, B., Gataric, M., and Hansen, A. C. Stable nonuniform sampling with weighted Fourier frames and recovery in arbitrary spaces. In *Sampling Theory and Applications (SampTA), 2015 International Conference on* (2015), IEEE, pp. 105–109.
3. Adcock, B., Gataric, M., and Hansen, A. C. Weighted frames of exponentials and stable recovery of multidimensional functions from nonuniform Fourier samples. *Applied and Computational Harmonic Analysis* (2015).
4. Adcock, B., Gataric, M., and Romero, J. L. Computing reconstructions from nonuniform Fourier samples: Universality of stability barriers and stable sampling rates. *arXiv preprint arXiv:1606.07698* (2016).
5. Aldroubi, A., and Grochenig, K. Nonuniform sampling and reconstruction in shift-invariant spaces. *SIAM Rev. 43*, 4 (2001), 585–620.
6. Archibald, R., Chen, K., Gelb, A., and Renaut, R. Improving tissue segmentation of human brain mri through preprocessing by the gegenbauer reconstruction method. *NeuroImage 20*, 1 (2003), 489–502.
7. Archibald, R., and Gelb, A. A method to reduce the gibbs ringing artifact in mri scans while keeping tissue boundary integrity. *Medical Imaging, IEEE Transactions on 21*, 4 (2002), 305–319.
8. Benedetto, J. Irregular sampling and frames. In *Wavelets: A Tutorial in Theory and Applications*, C. Chui, Ed. Academic Press, 1992, pp. 445–507.
9. Canny, J. A computational approach to edge detection. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 6 (1986), 679–698.
10. Chebira, A., and Kovacevic, J. Life beyond bases: The advent of frames (part i). *IEEE Signal Process. Mag. 24*, 4 (2007), 86–104.

11. Engelberg, S., and Tadmor, E. Recovery of edges from spectral data with noise: a new perspective. *Siam J. Numer. Anal. 46*, 5 (2008), 2620–2635.
12. Fessler, J. A., and Sutton, B. P. Nonuniform fast Fourier transforms using min-max interpolation. *IEEE Trans. Signal Process. 51*, 2 (2003), 560–574.
13. Gelb, A., and Cates, D. Segmentation of images from Fourier spectral data. *Commun. Comput. Phys. 5*, 2-4 (2009), 326–349.
14. Gelb, A., and Hines, T. Detection of edges from nonuniform Fourier data. *J. Fourier Anal. Appl. 17*, 11 (2011), 1152 – 1179.
15. Gelb, A., and Song, G. A frame theoretric approach to the non-uniform fast Fourier transform. *Siam J. Numer. Anal. 52*, 3 (2014), 1222–1242.
16. Gelb, A., and Tadmor, E. Detection of edges in spectral data. *Appl. Comput. Harmon. Anal. 7*, 1 (1999), 101–135.
17. Gelb, A., and Tadmor, E. Detection of edges in spectral data. II. Nonlinear enhancement. *SIAM J. Numer. Anal. 38*, 4 (2000), 1389–1408 (electronic).
18. Gelb, A., and Tadmor, E. Adaptive edge detectors for piecewise smooth data based on the minmod limiter. *J. Sci. Comput. 28*, 2-3 (2006), 279–306.
19. Han, D., Kornelson, K., Larson, D., and Weber, E. *Frames for Undergraduates*. Student Mathematical Library 40. American Mathematical Society, 2007.
20. Jimenez, J., Medina, V., and Yanez, O. Data-driven brain mri segmentation supported on edge confidence and a priori tissue information. *IEEE Trans. Med. Imag. 25*, 1 (2006), 74–83.
21. Kovacevic, J., and Chebira, A. Life beyond bases: The advent of frames (part ii). *IEEE Signal Process. Mag. 24* (Sept. 2007), 115–125.
22. Martinez, A., Gelb, A., and Gutierrez, A. Edge detection from non-uniform Fourier data using the convolutional gridding algorithm. *J. Sci. Comput. 61*, 3 (2014), 490–512.
23. Moore, R. Designing concentration factors to detect edges from non-uniform Fourier data. *Arizona State University Undergraduate Honors Thesis* (2015).
24. Petersen, A., Gelb, A., and Eubank, R. Hypothesis testing for Fourier based edge detection methods. *J. Sci. Comput. 51* (2012), 608–630.
25. Shattuck, D. W., Sandor-Leahy, S. R., Schaper, K. A., Rottenberg, D. A., and Leahya, R. M. Magnetic resonance image tissue classification using a partial volume model. *Neuroimage 13*, 5 (2001), 856–876.
26. Song, G., Davis, J., and Gelb, A. A two-dimensional inverse frame operator approximation technique. *SIAM J. Numer. Anal.* (2016, to appear).
27. Song, G., and Gelb, A. Approximating the inverse frame operator from localized frames. *Appl. Comput. Harmon. Anal. 35*, 1 (2013), 94–110.
28. Stefan, W., Viswanathan, A., Gelb, A., and Renaut, R. Sparsity enforcing edge detection method for blurred and noisy Fourier data. *J. Sci. Comput. 50*, 3 (2012), 536–556.
29. Sun, W., and Zhou, X. On the stability of multivariate trigonometric systems. *Journal of Mathematical Analysis and Applications 235*, 1 (1999), 159 – 167.
30. Viswanathan, A., Gelb, A., and Cochran, D. Iterative design of concentration factors for jump dection. *J. Sci. Comput. 51* (2012), 631–649.
31. Viswanathan, A., Gelb, A., Cochran, D., and Renaut, R. On reconstruction from non-uniform spectral data. *J. Sci. Comput. 45* (2010), 487–513.