Fast and direct inversion methods for the multivariate nonequispaced fast Fourier transform

Melanie Kircheis[*]   Daniel Potts[‡]

The well-known discrete Fourier transform (DFT) can easily be generalized to arbitrary nodes in the spatial domain. The fast procedure for this generalization is referred to as nonequispaced fast Fourier transform (NFFT). Various applications such as MRI, solution of PDEs, etc., are interested in the inverse problem, i. e., computing Fourier coefficients from given nonequispaced data. In this paper we survey different kinds of approaches to tackle this problem. In contrast to iterative procedures, where multiple iteration steps are needed for computing a solution, we focus especially on so-called direct inversion methods. We review density compensation techniques and introduce a new scheme that leads to an exact reconstruction for trigonometric polynomials. In addition, we consider a matrix optimization approach using Frobenius norm minimization to obtain an inverse NFFT.

Key words: inverse nonequispaced fast Fourier transform, nonuniform fast Fourier transform, direct inversion, density compensation, matrix optimization, iNFFT, NFFT, NUFFT

AMS *Subject Classifications*: 65Txx, 65T50, 65F05.

1 Introduction

The NFFT, short hand for nonequispaced fast Fourier transform or nonuniform fast Fourier transform (NUFFT), respectively, is a fast algorithm to evaluate a trigonometric polynomial

$$f(\boldsymbol{x}) = \sum_{\boldsymbol{k} \in \mathcal{I}_M} \hat{f}_{\boldsymbol{k}} \, \mathrm{e}^{2\pi \mathrm{i} \boldsymbol{k} \boldsymbol{x}} \tag{1.1}$$

---
[*]Corresponding author: melanie.kircheis@math.tu-chemnitz.de, Chemnitz University of Technology, Faculty of Mathematics, D–09107 Chemnitz, Germany

[‡]potts@math.tu-chemnitz.de, Chemnitz University of Technology, Faculty of Mathematics, D–09107 Chemnitz, Germany

with given Fourier coefficients $\hat{f}_{\boldsymbol{k}} \in \mathbb{C}$, $\boldsymbol{k} \in \mathcal{I}_{\boldsymbol{M}}$, at nonequispaced points $\boldsymbol{x}_j \in \left[-\frac{1}{2}, \frac{1}{2}\right)^d$, $j = 1, \dots, N$, $N \in \mathbb{N}$, where $\mathcal{I}_{\boldsymbol{M}} := \mathbb{Z}^d \cap \left[-\frac{M}{2}, \frac{M}{2}\right)^d$ with $|\mathcal{I}_{\boldsymbol{M}}| = M^d$. In case we are given equispaced points $\boldsymbol{x}_j$ and $|\mathcal{I}_{\boldsymbol{M}}| = N$, this evaluation can be realized by means of the well-known fast Fourier transform (FFT); an algorithm that is invertible. However, various applications such as magnetic resonance imaging (MRI), cf. [15, 19], solution of PDEs, cf. [21], etc., need to perform an inverse nonequispaced fast Fourier transform (iNFFT), i. e., compute the Fourier coefficients $\hat{f}_{\boldsymbol{k}}$ from given function evaluations $f(\boldsymbol{x}_j)$ of the trigonometric polynomial (1.1). Hence, we are interested in an inversion also for nonequispaced data.

In general, the number N of nodes $\boldsymbol{x}_j$ is independent from the number $|\mathcal{I}_{\boldsymbol{M}}|$ of Fourier coefficients $\hat{f}_{\boldsymbol{k}}$ and hence the nonequispaced Fourier matrix

$$\boldsymbol{A} := \left(\mathrm{e}^{2\pi\mathrm{i}\boldsymbol{k}\boldsymbol{x}_j}\right)_{j=1,\,\boldsymbol{k}\in\mathcal{I}_{\boldsymbol{M}}}^{N} \in \mathbb{C}^{N\times|\mathcal{I}_{\boldsymbol{M}}|},$$

which we would have to invert, is rectangular in most cases. Considering the corresponding linear system $\boldsymbol{A}\hat{\boldsymbol{f}} = \boldsymbol{f}$ with $\boldsymbol{f} := (f(\boldsymbol{x}_j))_{j=1}^{N}$ and $\hat{\boldsymbol{f}} := (\hat{f}_{\boldsymbol{k}})_{\boldsymbol{k}\in\mathcal{I}_{\boldsymbol{M}}}$, this can either be overdetermined, if $|\mathcal{I}_{\boldsymbol{M}}| \leq N$, or underdetermined, if $|\mathcal{I}_{\boldsymbol{M}}| > N$. Generally, this forces us to look for a pseudoinverse solution. Moreover, we also require that the nonequispaced Fourier matrix $\boldsymbol{A}$ has full rank. Eigenvalue estimates in [22, 6, 9, 45, 43] indeed confirm that this condition is satisfied for sufficiently nice sampling sets.

In literature a variety of approaches for an inverse NFFT (iNFFT) can be found. This is why we give a short overview.

Iterative inversion methods We start surveying the iterative methods. For the one-dimensional setting $d = 1$ with $|\mathcal{I}_{\boldsymbol{M}}| = N$ an algorithm was published in [60], which is specially designed for jittered equispaced points and is based on the CG method in connection with low rank approximation. An approach for the overdetermined case $|\mathcal{I}_{\boldsymbol{M}}| \leq N$ can be found in [22], where the Toeplitz structure of the matrix product $\boldsymbol{A}^*\boldsymbol{W}\boldsymbol{A}$ with a diagonal matrix $\boldsymbol{W} := \mathrm{diag}(w_j)_{j=1}^{N}$ of Voronoi weights is used to compute the solution iteratively by means of the CG algorithm.

For higher dimensional problems with $d \geq 1$ there are several approaches that compute a least squares approximation to the linear system $\boldsymbol{A}\hat{\boldsymbol{f}} = \boldsymbol{f}$. In the overdetermined case $|\mathcal{I}_{\boldsymbol{M}}| \leq N$, the given data can typically only be approximated up to a residual $\boldsymbol{r} := \boldsymbol{A}\hat{\boldsymbol{f}} - \boldsymbol{f}$. Therefore, the weighted least squares problem

$$\underset{\hat{\boldsymbol{f}}\in\mathbb{C}^{|\mathcal{I}_{\boldsymbol{M}}|}}{\text{Minimize}} \quad \sum_{j=1}^{N} w_j \left| \sum_{\boldsymbol{k}\in\mathcal{I}_{\boldsymbol{M}}} \hat{f}_{\boldsymbol{k}}\, \mathrm{e}^{2\pi\mathrm{i}\boldsymbol{k}\boldsymbol{x}_j} - f(\boldsymbol{x}_j) \right|^2$$

is considered, which is equivalent to solving the weighted normal equations of first kind $\boldsymbol{A}^*\boldsymbol{W}\boldsymbol{A}\hat{\boldsymbol{f}} = \boldsymbol{A}^*\boldsymbol{W}\boldsymbol{f}$ with the diagonal matrix $\boldsymbol{W} := \mathrm{diag}(w_j)_{j=1}^{N}$ of weights in time domain. In [69, 24, 41] these normal equations are solved iteratively by means of CG using the NFFT to realize fast matrix-vector multiplications involving $\boldsymbol{A}$, whereas in [56] a fast convolution is used.

2

In the consistent underdetermined case $|\mathcal{I_M}| > N$ the data can be interpolated exactly and therefore one can choose a specific solution, e. g. the one that solves the constrained minimization problem

$$\underset{\hat{\boldsymbol{f}} \in \mathbb{C}^{|\mathcal{I}_M|}}{\text{Minimize}} \quad \sum_{\boldsymbol{k} \in \mathcal{I}_M} \frac{|\hat{f}_{\boldsymbol{k}}|^2}{\hat{w}_{\boldsymbol{k}}} \quad \text{subject to} \quad \boldsymbol{A}\hat{\boldsymbol{f}} = \boldsymbol{f}.$$

This interpolation problem is equivalent to the weighted normal equations of second kind $\boldsymbol{A}\hat{\boldsymbol{W}}\boldsymbol{A}^*\boldsymbol{y} = \boldsymbol{f}$, $\hat{\boldsymbol{f}} = \hat{\boldsymbol{W}}\boldsymbol{A}^*\boldsymbol{y}$ with the diagonal matrix $\hat{\boldsymbol{W}} := \text{diag}(\hat{w}_{\boldsymbol{k}})_{\boldsymbol{k} \in \mathcal{I}_M}$ of weights in frequency domain. In [44] the CG method was used in connection with the NFFT to iteratively compute a solution to this problem, see also [53, Section 7.6.2].

Regularization methods Moreover, there also exist several regularization techniques for the multidimensional setting $d \geq 1$. For example, [65, 71, 1] all solve the ℓ_1-regularized problem

$$\underset{\hat{\boldsymbol{f}} \in \mathbb{C}^{|\mathcal{I}_M|}}{\text{Minimize}} \quad \tfrac{1}{2}\|\boldsymbol{A}\hat{\boldsymbol{f}} - \boldsymbol{f}\|_2^2 + \lambda\|\mathcal{L}^m\hat{\boldsymbol{f}}\|_1$$

with regularization parameter $\lambda > 0$ and the m-th order polynomial annihilation operator $\mathcal{L}^m \in \mathbb{R}^{N \times |\mathcal{I}_M|}$ as sparsifying transform, see [2]. Based on this, weighted ℓ_p-schemes

$$\underset{\hat{\boldsymbol{f}} \in \mathbb{C}^{|\mathcal{I}_M|}}{\text{Minimize}} \quad \tfrac{1}{2}\|\boldsymbol{A}\hat{\boldsymbol{f}} - \boldsymbol{f}\|_2^2 + \tfrac{1}{p}\|\boldsymbol{W}\mathcal{L}^m\hat{\boldsymbol{f}}\|_p^p$$

were introduced in [10, 11, 14, 48], which are designed to reduce the penalty at locations where $\mathcal{L}^m\hat{\boldsymbol{f}}$ is nonzero. For instance, [13, 61] each state a two step method, that firstly uses edge detection to create a mask, i. e., a weighting matrix which indicates where non-zero entries are expected in the TV domain, and then targets weighted ℓ_2-norm TV regularization appropriately to smooth regions of the function in a second minimization step.

Direct inversion methods In contrast to these iterated procedures, there are also so-called direct methods that do not require multiple iteration steps. Already in [18] a direct method was explained for the setting $d = 1$ and $|\mathcal{I}_M| = N$ which uses Lagrange interpolation in combination with fast multiple methods. Based on this, further methods were deduced for the same setting, which also use Lagrange interpolation, but additionally incorporate an imaginary shift in [63], or utilize fast summation in [37] for the fast evaluation of occurring sums, see also [53, Section 7.6.1].

In the overdetermined setting $|\mathcal{I}_M| \leq N$ another approach for computing an inverse NFFT can be obtained by using the fact that $\boldsymbol{A}^*\boldsymbol{A}$ is of Toeplitz structure. To this end, the Gohberg-Semencul formula, see [32], can be used to solve the normal equations $\boldsymbol{A}^*\boldsymbol{A}\hat{\boldsymbol{f}} = \boldsymbol{A}^*\boldsymbol{f}$ exactly by analogy with [5]. Here the computation of the components of the Gohberg-Semencul formula can be viewed as a precomputational step. In addition, also a frame-theoretical approach is known from [26], which provides a link between the adjoint NFFT and frame approximation, and could therefore be seen as a way to invert

the NFFT. Note that the method in [26] is based only on optimizing a diagonal matrix (the matrix $\boldsymbol{D}$ defined in (2.13)), whereas in [37] similar ideas were used to modify a sparse matrix (the matrix $\boldsymbol{B}$ defined in (2.15)).

For the multidimensional setting $d > 1$ several methods have been developed that are tailored to the special structure of the linogram or pseudo-polar grid, respectively, see Figure 5.2b, such that the inversion involves only one-dimensional FFTs and interpolations. On the one hand, in [4] a least squares solution is computed iteratively by using a fast multiplication technique with the matrix $\boldsymbol{A}$, which can be derived in case of the linogram grid. On the other hand, [3] is based on a fast resampling strategy, where a first step resamples the linogram grid onto a Cartesian grid, and the second phase recovers the image from these Cartesian samples. However, these techniques are exclusively applicable for the special case of the linogram grid, see Figure 5.2b, or the polar grid by another resampling, cf. Figure 5.2a. Since we are interested in more generally applicable methods, a brief introduction to direct inversion for general sampling patterns can be found in [38].

Current approach   In this paper we focus on direct inversion methods that are applicable for general sampling patterns and present new methods for the computation of an iNFFT. Note that direct method in the context of the linear system $\boldsymbol{A}\hat{\boldsymbol{f}} = \boldsymbol{f}$ means, that for a fixed set of points $\boldsymbol{x}_j$, $j = 1, \ldots, N$, the reconstruction of $\hat{\boldsymbol{f}}$ from given $\boldsymbol{f}$ can be realized with the same number of arithmetic operations as a single application of an adjoint NFFT (see Algorithm 2.4). To achieve this, a certain precomputational step is compulsory, since the adjoint NFFT does not yield an inversion of the NFFT per se, cf. (3.3). Although this precomputations might be rather costly, they need to be done only once for a given set of points $\boldsymbol{x}_j$, $j = 1, \ldots, N$. Therefore, direct methods are especially beneficial in case of fixed nodes for several measurement vectors $\boldsymbol{f}$.

For this reason, the current paper is concerned with two different approaches of this kind. Firstly, we consider the very well known approach of so-called sampling density compensation, which can be written as $\hat{\boldsymbol{f}} \approx \boldsymbol{A}^*\boldsymbol{W}\boldsymbol{f}$ with a diagonal matrix $\boldsymbol{W} \coloneqq \mathrm{diag}(w_j)_{j=1}^{N}$ of weights. The already mentioned precomputations then consist of computing suitable weights w_j, while the actual reconstruction step includes only one adjoint NFFT applied to the scaled coefficient vector $\boldsymbol{W}\boldsymbol{f}$. In this paper we examine several existing approaches for computing the weights w_j and introduce a new method, such that the iNFFT is exact for all trigonometric polynomials (1.1) of degree $\boldsymbol{M}$.

As a second part, we reconsider and enhance our approach introduced in [38]. Here the idea is using the matrix representation $\boldsymbol{A} \approx \boldsymbol{B}\boldsymbol{F}\boldsymbol{D}$ of the NFFT to modify one of the matrix factors, such that an inversion is given by $\hat{\boldsymbol{f}} \approx \boldsymbol{D}^*\boldsymbol{F}^*\boldsymbol{B}_{\mathrm{opt}}^*\boldsymbol{f}$. Then the precomputational step consists of optimizing the matrix $\boldsymbol{B}$, while the actual reconstruction step includes only one modified adjoint NFFT applied to the coefficient vector $\boldsymbol{f}$.

Outline of this paper   This paper is organized as follows. In Section 2 we introduce the already mentioned algorithm, the NFFT, as well as its adjoint version, the adjoint NFFT. Secondly, in Section 3 we introduce the inversion problem and deal with direct

4

methods using so-called sampling density compensation. We start our investigations with trigonometric polynomials in Section 3.1. Here the main formula (3.14) yields exact reconstruction for all trigonometric polynomials of degree $\boldsymbol{M}$. We also discuss practical computation schemes for the overdetermined as well as the underdetermined setting. Subsequently, in Section 3.2 we go on to bandlimited functions and show that the same numerical procedures as in Section 3.1 can be used in this setting as well. Section 3.3 then summarizes the previous findings by presenting a general error bound on density compensation factors computed by means of (3.14) in Theorem 3.14. In addition, this also yields an estimate on the condition number of a specific matrix product, as shown in Theorem 3.15. In Section 3.4 we have a look at certain approaches from literature and their connection among each other as well as to the method presented in Section 3.1. Afterwards, we examine another direct inversion method in Section 4, where we aim to modify the adjoint NFFT by means of matrix optimization such that this yields an iNFFT. Finally, in Section 5 we show some numerical examples to investigate the accuracy of our approaches.

2 Nonequispaced fast Fourier transform

Let

$$\mathbb{T}^d := \mathbb{R}^d \setminus \mathbb{Z}^d \cong \left[-\tfrac{1}{2}, \tfrac{1}{2}\right)^d = \left\{ x \in \mathbb{R}^d \colon\ -\tfrac{1}{2} \le x_t < \tfrac{1}{2},\, t = 1, \ldots, d \right\}$$

denote the d-dimensional torus with $d \in \mathbb{N}$. For $\boldsymbol{M} := (M, \ldots, M)^T$, $M \in 2\mathbb{N}$, we define the multi-index set

$$\mathcal{I}_{\boldsymbol{M}} := \mathbb{Z}^d \cap \left[-\tfrac{M}{2}, \tfrac{M}{2}\right)^d = \left\{ \boldsymbol{k} \in \mathbb{Z}^d \colon\ -\tfrac{M}{2} \le k_t < \tfrac{M}{2},\, t = 1, \ldots, d \right\}$$

with cardinality $|\mathcal{I}_{\boldsymbol{M}}| = M^d$. The inner product of two vectors shall be defined as usual as $\boldsymbol{k}\boldsymbol{x} := k_1 x_1 + \cdots + k_d x_d$. Additionally, we define the componentwise product as $\boldsymbol{x} \odot \boldsymbol{y} := (x_1 y_1, \ldots, x_d y_d)^T$, the all ones vector $\boldsymbol{1}_d := (1, \ldots, 1)^T$ and the reciprocal of a vector $\boldsymbol{x}$ with nonzero components shall be given by $\boldsymbol{x}^{-1} := \left(x_1^{-1}, \ldots, x_d^{-1}\right)^T$.

We consider the Hilbert space $L_2(\mathbb{T}^d)$ of all 1-periodic, complex-valued functions, which possesses the orthonormal basis $\{e^{2\pi i \boldsymbol{k}\boldsymbol{x}} \colon \boldsymbol{k} \in \mathbb{Z}^d\}$. It it known that every function $f \in L_2(\mathbb{T}^d)$ is uniquely representable in the form

$$f(\boldsymbol{x}) = \sum_{\boldsymbol{k} \in \mathbb{Z}^d} c_{\boldsymbol{k}}(f)\, e^{2\pi i \boldsymbol{k}\boldsymbol{x}} \tag{2.1}$$

with the coefficients

$$c_{\boldsymbol{k}}(f) := \int_{\mathbb{T}^d} f(\boldsymbol{x})\, e^{-2\pi i \boldsymbol{k}\boldsymbol{x}} \, \mathrm{d}\boldsymbol{x}, \quad \boldsymbol{k} \in \mathbb{Z}^d, \tag{2.2}$$

where the sum in (2.1) converges to f in the $L_2(\mathbb{T}^d)$-norm, cf. [53, Thm. 4.5]. A series of the form (2.1) is called *Fourier series* with the *Fourier coefficients* (2.2). Numerically,

the Fourier coefficients (2.2) are approximated on the uniform grid $\{\boldsymbol{M}^{-1} \odot \boldsymbol{\ell}, \boldsymbol{\ell} \in \mathcal{I}_{\boldsymbol{M}}\}$ by the trapezoidal rule for numerical integration as

$$c_{\boldsymbol{k}}(f) \approx \frac{1}{|\mathcal{I}_{\boldsymbol{M}}|} \sum_{\boldsymbol{\ell} \in \mathcal{I}_{\boldsymbol{M}}} f(\boldsymbol{M}^{-1} \odot \boldsymbol{\ell}) \, \mathrm{e}^{-2\pi \mathrm{i} \boldsymbol{k}(\boldsymbol{M}^{-1} \odot \boldsymbol{\ell})}, \quad \boldsymbol{k} \in \mathbb{Z}^d, \tag{2.3}$$

which is an acceptable approximation for $\boldsymbol{k} \in \mathcal{I}_{\boldsymbol{M}}$, see e.g. [53, p. 214]. The fast evaluation of (2.3) can then be realized by means of the *fast Fourier transform (FFT)*. Moreover, it is know that this transformation is invertible and that the inverse problem of computing

$$f(\boldsymbol{M}^{-1} \odot \boldsymbol{\ell}) = \sum_{\boldsymbol{k} \in \mathcal{I}_{\boldsymbol{M}}} \hat{f}_{\boldsymbol{k}} \, \mathrm{e}^{2\pi \mathrm{i} \boldsymbol{k}(\boldsymbol{M}^{-1} \odot \boldsymbol{\ell})}, \quad \boldsymbol{\ell} \in \mathcal{I}_{\boldsymbol{M}}, \tag{2.4}$$

with $\hat{f}_{\boldsymbol{k}} \approx c_{\boldsymbol{k}}(f)$, $\boldsymbol{k} \in \mathcal{I}_{\boldsymbol{M}}$, can be realized by means of an *inverse fast Fourier transform (iFFT)*, which is basically the same algorithm except for some reordering and scaling, cf. [53, Lem. 3.17].

Now suppose we are given nonequispaced nodes $\boldsymbol{x}_j \in \mathbb{T}^d$, $j = 1, \ldots, N$, instead. Then, we consider the computation of the sums

$$f_j := f(\boldsymbol{x}_j) = \sum_{\boldsymbol{k} \in \mathcal{I}_{\boldsymbol{M}}} \hat{f}_{\boldsymbol{k}} \, \mathrm{e}^{2\pi \mathrm{i} \boldsymbol{k} \boldsymbol{x}_j}, \quad j = 1, \ldots, N, \tag{2.5}$$

for given $\hat{f}_{\boldsymbol{k}} \in \mathbb{C}$, $\boldsymbol{k} \in \mathcal{I}_{\boldsymbol{M}}$, as well as the adjoint problem of computing

$$h_{\boldsymbol{k}} = \sum_{j=1}^{N} f_j \, \mathrm{e}^{-2\pi \mathrm{i} \boldsymbol{k} \boldsymbol{x}_j}, \quad \boldsymbol{k} \in \mathcal{I}_{\boldsymbol{M}}, \tag{2.6}$$

for given $f_j \in \mathbb{C}$, $j = 1, \ldots, N$. By defining the nonequispaced Fourier matrix

$$\boldsymbol{A} = \boldsymbol{A}_{|\mathcal{I}_{\boldsymbol{M}}|} := \left(\mathrm{e}^{2\pi \mathrm{i} \boldsymbol{k} \boldsymbol{x}_j} \right)_{j=1, \, \boldsymbol{k} \in \mathcal{I}_{\boldsymbol{M}}}^{N} \in \mathbb{C}^{N \times |\mathcal{I}_{\boldsymbol{M}}|}, \tag{2.7}$$

as well as the vectors $\boldsymbol{f} := (f_j)_{j=1}^{N}$, $\hat{\boldsymbol{f}} := (\hat{f}_{\boldsymbol{k}})_{\boldsymbol{k} \in \mathcal{I}_{\boldsymbol{M}}}$ and $\boldsymbol{h} := (h_{\boldsymbol{k}})_{\boldsymbol{k} \in \mathcal{I}_{\boldsymbol{M}}}$, the computation of sums of the form (2.5) and (2.6) can be written as $\boldsymbol{f} = \boldsymbol{A}\hat{\boldsymbol{f}}$ and $\boldsymbol{h} = \boldsymbol{A}^*\boldsymbol{f}$, where $\boldsymbol{A}^* := \overline{\boldsymbol{A}}^T$ denotes the adjoint matrix of $\boldsymbol{A}$.

Since the naive computation of (2.5) and (2.6) is of complexity $\mathcal{O}(N \cdot |\mathcal{I}_{\boldsymbol{M}}|)$, a fast approximate algorithm, the so-called *nonequispaced fast Fourier transform (NFFT)*, is briefly explained below. For more information see [17, 7, 66, 29, 36] or [53, pp. 377-381].

2.1 The NFFT

We firstly restrict our attention to problem (2.5), which is equivalent to the evaluation of a trigonometric polynomial

$$f(\boldsymbol{x}) = \sum_{\boldsymbol{k} \in \mathcal{I}_{\boldsymbol{M}}} \hat{f}_{\boldsymbol{k}} \, \mathrm{e}^{2\pi \mathrm{i} \boldsymbol{k} \boldsymbol{x}} \tag{2.8}$$

with given $\hat{f}_{\boldsymbol{k}} \in \mathbb{C}$, $\boldsymbol{k} \in \mathcal{I}_{\boldsymbol{M}}$, at given nonequispaced nodes $\boldsymbol{x}_j \in \mathbb{T}^d$, $j = 1, \dots, N$. Let $w \in L_2(\mathbb{R}^d) \cap L_1(\mathbb{R}^d)$ be a so-called *window function*, which is well localized in space and frequency domain. Now we define the 1-periodic function $\tilde{w}(\boldsymbol{x}) := \sum_{\boldsymbol{r} \in \mathbb{Z}^d} w(\boldsymbol{x} + \boldsymbol{r})$ with absolute convergent Fourier series. As a consequence, the Fourier coefficients of the periodization $\tilde{w}$ have the form

$$c_{\boldsymbol{k}}(\tilde{w}) = \int_{\mathbb{T}^d} \tilde{w}(\boldsymbol{x}) \, \mathrm{e}^{-2\pi \mathrm{i} \boldsymbol{k} \boldsymbol{x}} \, \mathrm{d}\boldsymbol{x} = \int_{\mathbb{R}^d} w(\boldsymbol{x}) \, \mathrm{e}^{-2\pi \mathrm{i} \boldsymbol{k} \boldsymbol{x}} \, \mathrm{d}\boldsymbol{x} =: \hat{w}(\boldsymbol{k}).$$

For a given oversampling factor $\sigma \geq 1$, we define $2\mathbb{N} \ni M_\sigma := 2\lceil \lceil \sigma M \rceil / 2 \rceil$ as well as $\boldsymbol{M}_{\boldsymbol{\sigma}} := M_\sigma \cdot \mathbf{1}_d$, and approximate f by a linear combination of translates of the periodized window function, i.e.,

$$f(\boldsymbol{x}) \approx s_1(\boldsymbol{x}) := \sum_{\boldsymbol{\ell} \in \mathcal{I}_{\boldsymbol{M}_{\boldsymbol{\sigma}}}} g_{\boldsymbol{\ell}} \, \tilde{w}\big(\boldsymbol{x} - \boldsymbol{M}_{\boldsymbol{\sigma}}^{-1} \odot \boldsymbol{\ell}\big), \tag{2.9}$$

where $g_{\boldsymbol{\ell}} \in \mathbb{C}$, $\boldsymbol{\ell} \in \mathcal{I}_{\boldsymbol{M}_{\boldsymbol{\sigma}}}$, are coefficients to be determined such that (2.9) yields a good approximation. By means of the convolution theorem (see [53, Lem. 4.1]), the approximant $s_1 \in L_2(\mathbb{T}^d)$ in (2.9) can be represented as

$$\begin{aligned} s_1(\boldsymbol{x}) &= \sum_{\boldsymbol{k} \in \mathbb{Z}^d} c_{\boldsymbol{k}}(s_1) \, \mathrm{e}^{2\pi \mathrm{i} \boldsymbol{k} \boldsymbol{x}} \\ &= \sum_{\boldsymbol{k} \in \mathcal{I}_{\boldsymbol{M}}} \hat{g}_{\boldsymbol{k}} \, c_{\boldsymbol{k}}(\tilde{w}) \, \mathrm{e}^{2\pi \mathrm{i} \boldsymbol{k} \boldsymbol{x}} + \sum_{\boldsymbol{r} \in \mathbb{Z}^d \setminus \{\boldsymbol{0}\}} \sum_{\boldsymbol{k} \in \mathcal{I}_{\boldsymbol{M}}} \hat{g}_{\boldsymbol{k}} \, c_{\boldsymbol{k} + \boldsymbol{M}_{\boldsymbol{\sigma}} \odot \boldsymbol{r}}(\tilde{w}) \, \mathrm{e}^{2\pi \mathrm{i} (\boldsymbol{k} + \boldsymbol{M}_{\boldsymbol{\sigma}} \odot \boldsymbol{r}) \boldsymbol{x}}, \end{aligned} \tag{2.10}$$

where the discrete Fourier transform of the coefficients $g_{\boldsymbol{\ell}}$ is defined by

$$\hat{g}_{\boldsymbol{k}} := \sum_{\boldsymbol{\ell} \in \mathcal{I}_{\boldsymbol{M}_{\boldsymbol{\sigma}}}} g_{\boldsymbol{\ell}} \, \mathrm{e}^{-2\pi \mathrm{i} \boldsymbol{k} (\boldsymbol{M}_{\boldsymbol{\sigma}}^{-1} \odot \boldsymbol{\ell})}, \quad \boldsymbol{k} \in \mathcal{I}_{\boldsymbol{M}}. \tag{2.11}$$

Comparing (2.5) and (2.10) then yields

$$\hat{g}_{\boldsymbol{k}} = \begin{cases} \dfrac{\hat{f}_{\boldsymbol{k}}}{\hat{w}(\boldsymbol{k})} & : \boldsymbol{k} \in \mathcal{I}_{\boldsymbol{M}}, \\ 0 & : \boldsymbol{k} \in \mathcal{I}_{\boldsymbol{M}_{\boldsymbol{\sigma}}} \setminus \mathcal{I}_{\boldsymbol{M}}. \end{cases}$$

Consequently, the coefficients $g_{\boldsymbol{\ell}}$ in (2.9) can be obtained by inverting (2.11), i.e., by the application of an iFFT.

Furthermore, we assume that w is well localized such that it is small outside the square $[-m/M_\sigma, m/M_\sigma]^d$ with truncation parameter $m \ll M_\sigma$. In this case, w can be approximated by the compactly supported function

$$w_m(\boldsymbol{x}) := \begin{cases} w(\boldsymbol{x}) & : \boldsymbol{x} \in \left[-\frac{m}{M_\sigma}, \frac{m}{M_\sigma} \right]^d, \\ 0 & : \text{otherwise.} \end{cases}$$

Thereby, we approximate s_1 by the short sums

$$f(\boldsymbol{x}_j) \approx s_1(\boldsymbol{x}_j) \approx s(\boldsymbol{x}_j) := \sum_{\boldsymbol{\ell} \in \mathcal{I}_{\boldsymbol{M}_{\boldsymbol{\sigma}}}} g_{\boldsymbol{\ell}} \, \tilde{w}_m\big(\boldsymbol{x}_j - \boldsymbol{M}_{\boldsymbol{\sigma}}^{-1} \odot \boldsymbol{\ell}\big)$$

$$= \sum_{\boldsymbol{\ell}\in\mathcal{I}_{\boldsymbol{M_\sigma},m}(\boldsymbol{x}_j)} g_{\boldsymbol{\ell}}\,\tilde{w}_m\big(\boldsymbol{x}_j - \boldsymbol{M}_{\boldsymbol{\sigma}}^{-1}\odot\boldsymbol{\ell}\big)\,,$$

where the index set

$$\mathcal{I}_{\boldsymbol{M_\sigma},m}(\boldsymbol{x}_j) := \Big\{\boldsymbol{\ell}\in\mathcal{I}_{\boldsymbol{M_\sigma}} : \exists \boldsymbol{z}\in\mathbb{Z}^d \text{ with } -m\cdot\boldsymbol{1}_d \leq \boldsymbol{M_\sigma}\odot\boldsymbol{x}_j - \boldsymbol{\ell} + \boldsymbol{z} \leq m\cdot\boldsymbol{1}_d\Big\} \tag{2.12}$$

contains at most $(2m+1)^d$ entries for each fixed $\boldsymbol{x}_j$. Thus, the obtained algorithm can be summarized as follows.

Algorithm 2.1 (NFFT).

For $d, N \in \mathbb{N}$ let $\boldsymbol{x}_j \in \mathbb{T}^d$, $j = 1,\ldots,N$, be given nodes as well as $\hat{f}_{\boldsymbol{k}} \in \mathbb{C}$, $\boldsymbol{k} \in \mathcal{I}_{\boldsymbol{M}}$, given Fourier coefficients. Furthermore, we are given the oversampling factor $\sigma \geq 1$, $2\mathbb{N} \ni M_\sigma := 2\lceil\lceil\sigma M\rceil/2\rceil$, $\boldsymbol{M_\sigma} := M_\sigma\cdot\boldsymbol{1}_d$, as well as the window function w, the truncated function w_m with truncation parameter $m \ll M_\sigma$, and their 1-periodic versions $\tilde{w}$ and $\tilde{w}_m$.

1. *Set*
$$\hat{g}_{\boldsymbol{k}} := \begin{cases} \frac{\hat{f}_{\boldsymbol{k}}}{\hat{w}(\boldsymbol{k})} & : \boldsymbol{k}\in\mathcal{I}_{\boldsymbol{M}}, \\ 0 & : \boldsymbol{k}\in\mathcal{I}_{\boldsymbol{M_\sigma}}\setminus\mathcal{I}_{\boldsymbol{M}}. \end{cases}$$
$\mathcal{O}(|\mathcal{I}_{\boldsymbol{M}}|)$

2. *Compute*
$$g_{\boldsymbol{\ell}} := \frac{1}{|\mathcal{I}_{\boldsymbol{M_\sigma}}|}\sum_{\boldsymbol{k}\in\mathcal{I}_{\boldsymbol{M}}} \hat{g}_{\boldsymbol{k}}\,\mathrm{e}^{2\pi\mathrm{i}\boldsymbol{k}(\boldsymbol{M}_{\boldsymbol{\sigma}}^{-1}\odot\boldsymbol{\ell})}, \quad \boldsymbol{\ell}\in\mathcal{I}_{\boldsymbol{M_\sigma}},$$
   by means of a d-variate iFFT.
$\mathcal{O}(|\mathcal{I}_{\boldsymbol{M}}|\log(|\mathcal{I}_{\boldsymbol{M}}|))$

3. *Compute the short sums*
$$\tilde{f}_j := \sum_{\boldsymbol{\ell}\in\mathcal{I}_{\boldsymbol{M_\sigma},m}(\boldsymbol{x}_j)} g_{\boldsymbol{\ell}}\,\tilde{w}_m\big(\boldsymbol{x}_j - \boldsymbol{M}_{\boldsymbol{\sigma}}^{-1}\odot\boldsymbol{\ell}\big)\,, \quad j = 1,\ldots,N.$$
$\mathcal{O}(N)$

Output: $\tilde{f}_j \approx f_j$, $j = 1,\ldots,N$, cf. (2.5). **Complexity:** $\mathcal{O}(|\mathcal{I}_{\boldsymbol{M}}|\log(|\mathcal{I}_{\boldsymbol{M}}|) + N)$

Remark 2.2. *Suitable window functions can be found e. g. in [17, 7, 16, 25, 29, 36, 54].*
□

Next we give the matrix-vector representation of the NFFT. To this end, we define

- the diagonal matrix

$$\boldsymbol{D} := \mathrm{diag}\left(\frac{1}{|\mathcal{I}_{\boldsymbol{M_\sigma}}|\cdot\hat{w}(\boldsymbol{k})}\right)_{\boldsymbol{k}\in\mathcal{I}_{\boldsymbol{M}}} \in \mathbb{C}^{|\mathcal{I}_{\boldsymbol{M}}|\times|\mathcal{I}_{\boldsymbol{M}}|}, \tag{2.13}$$

- the truncated Fourier matrix

$$\boldsymbol{F} := \left(\mathrm{e}^{2\pi \mathrm{i}\boldsymbol{k}(\boldsymbol{M}_{\boldsymbol{\sigma}}^{-1}\odot\boldsymbol{\ell})} \right)_{\boldsymbol{\ell}\in\mathcal{I}_{\boldsymbol{M}_{\boldsymbol{\sigma}}},\,\boldsymbol{k}\in\mathcal{I}_{\boldsymbol{M}}} \in \mathbb{C}^{|\mathcal{I}_{\boldsymbol{M}_{\boldsymbol{\sigma}}}|\times|\mathcal{I}_{\boldsymbol{M}}|}, \tag{2.14}$$

- and the sparse matrix

$$\boldsymbol{B} := \left(\tilde{w}_m\!\left(\boldsymbol{x}_j - \boldsymbol{M}_{\boldsymbol{\sigma}}^{-1}\odot\boldsymbol{\ell}\right) \right)_{j=1,\,l\in\mathcal{I}_{\boldsymbol{M}_{\boldsymbol{\sigma}}}}^{N} \in \mathbb{R}^{N\times|\mathcal{I}_{\boldsymbol{M}_{\boldsymbol{\sigma}}}|}, \tag{2.15}$$

where by definition (2.12) each row of $\boldsymbol{B}$ contains at most $(2m+1)^d$ nonzeros. In doing so, the NFFT in Algorithm 2.1 can be formulated in matrix-vector notation such that we receive the approximation $\boldsymbol{A} \approx \boldsymbol{B}\boldsymbol{F}\boldsymbol{D}$ of (2.7), cf. [53, p. 383]. In other words, the NFFT uses the approximation

$$\mathrm{e}^{2\pi \mathrm{i}\boldsymbol{k}\boldsymbol{x}_j} \approx \frac{1}{|\mathcal{I}_{\boldsymbol{M}_{\boldsymbol{\sigma}}}|\cdot\hat{w}(\boldsymbol{k})} \sum_{\boldsymbol{\ell}\in\mathcal{I}_{\boldsymbol{M}_{\boldsymbol{\sigma}},m}(\boldsymbol{x}_j)} \mathrm{e}^{2\pi \mathrm{i}\boldsymbol{k}(\boldsymbol{M}_{\boldsymbol{\sigma}}^{-1}\odot\boldsymbol{\ell})}\, \tilde{w}_m\!\left(\boldsymbol{x}_j - \boldsymbol{M}_{\boldsymbol{\sigma}}^{-1}\odot\boldsymbol{\ell}\right).$$

Remark 2.3. *It has to be pointed out that because of consistency the factor $|\mathcal{I}_{\boldsymbol{M}_{\boldsymbol{\sigma}}}|^{-1}$ is here not located in the matrix $\boldsymbol{F}$ as usual but in the matrix $\boldsymbol{D}$.* $\qquad\square$

2.2 The adjoint NFFT

Now we proceed with the adjoint problem (2.6). As already seen, this can be written as $\boldsymbol{h} = \boldsymbol{A}^*\boldsymbol{f}$ with the adjoint matrix $\boldsymbol{A}^*$ of (2.7). Thus, using the matrices (2.13), (2.14) and (2.15) we receive the approximation $\boldsymbol{A}^* \approx \boldsymbol{D}^*\boldsymbol{F}^*\boldsymbol{B}^*$, such that a fast algorithm for the adjoint problem can be denoted as follows.

Algorithm 2.4 (adjoint NFFT)**.**

For $d,N\in\mathbb{N}$ let $\boldsymbol{x}_j\in\mathbb{T}^d$, $j=1,\dots,N$, be given nodes as well as $f_j\in\mathbb{C}$ given coefficients. Further, we are given the oversampling factor $\sigma\geq 1$, $2\mathbb{N}\ni M_\sigma := 2\lceil\lceil\sigma M\rceil/2\rceil$, $\boldsymbol{M}_{\boldsymbol{\sigma}} := M_\sigma\cdot\boldsymbol{1}_d$, as well as the window function w, the truncated function w_m with truncation parameter $m\ll M_\sigma$, and their 1-periodic versions $\tilde{w}$ and $\tilde{w}_m$.

1. *Compute the sparse sums*

$$g_{\boldsymbol{\ell}} := \sum_{j=1}^{N} f_j\,\tilde{w}_m\!\left(\boldsymbol{x}_j - \boldsymbol{M}_{\boldsymbol{\sigma}}^{-1}\odot\boldsymbol{\ell}\right), \quad \boldsymbol{\ell}\in\mathcal{I}_{\boldsymbol{M}_{\boldsymbol{\sigma}}}.$$

$\mathcal{O}(N)$

2. *Compute*

$$\hat{g}_{\boldsymbol{k}} := \frac{1}{|\mathcal{I}_{\boldsymbol{M}_{\boldsymbol{\sigma}}}|} \sum_{\boldsymbol{\ell}\in\mathcal{I}_{\boldsymbol{M}_{\boldsymbol{\sigma}}}} g_{\boldsymbol{\ell}}\,\mathrm{e}^{-2\pi \mathrm{i}\boldsymbol{k}(\boldsymbol{M}_{\boldsymbol{\sigma}}^{-1}\odot\boldsymbol{\ell})}, \quad \boldsymbol{k}\in\mathcal{I}_{\boldsymbol{M}},$$

   by means of a d-variate FFT.

$\mathcal{O}(|\mathcal{I}_{\boldsymbol{M}}|\log(|\mathcal{I}_{\boldsymbol{M}}|))$

3. Set

$$\tilde{h}_{\boldsymbol{k}} := \frac{\hat{g}_{\boldsymbol{k}}}{\hat{w}(\boldsymbol{k})}, \quad \boldsymbol{k} \in \mathcal{I}_{\boldsymbol{M}}.$$

$$\mathcal{O}(|\mathcal{I}_{\boldsymbol{M}}|)$$

Output: $\tilde{h}_{\boldsymbol{k}} \approx h_{\boldsymbol{k}}, \boldsymbol{k} \in \mathcal{I}_{\boldsymbol{M}}$, *cf.* (2.6).　　　　　**Complexity:** $\mathcal{O}(|\mathcal{I}_{\boldsymbol{M}}| \log(|\mathcal{I}_{\boldsymbol{M}}|) + N)$

The algorithms presented in this section (Algorithms 2.1 and 2.4) are part of the NFFT software [35]. For algorithmic details we refer to [36].

3　Direct inversion using density compensation

Having introduced the fast methods for nonequispaced data, we remark that various applications such as MRI, solution of PDEs, etc. are interested in the inverse problem, i. e., instead of the evaluation of (2.5) the aim is computing Fourier coefficients $\hat{f}_{\boldsymbol{k}}$, $\boldsymbol{k} \in \mathcal{I}_{\boldsymbol{M}}$, from given nonequispaced data $f(\boldsymbol{x}_j)$, $j = 1, \dots, N$. Therefore, this section shall be dedicated to this task.

To clarify the major dissimilarity between equispaced and nonequispaced data, we start considering the equispaced case. When evaluating at the points $x_{\boldsymbol{j}} = \frac{1}{n}\boldsymbol{j} \in \mathbb{T}^d$, $\boldsymbol{j} \in \mathcal{I}_{\boldsymbol{n}}$, with $\boldsymbol{n} := n \cdot \mathbf{1}_d$ and $|\mathcal{I}_{\boldsymbol{n}}| = N$, the nonequispaced Fourier matrix $\boldsymbol{A} \in \mathbb{C}^{N \times |\mathcal{I}_{\boldsymbol{M}}|}$ in (2.7) turns into the equispaced Fourier matrix $\boldsymbol{F} \in \mathbb{C}^{|\mathcal{I}_{\boldsymbol{M}_{\boldsymbol{\sigma}}}| \times |\mathcal{I}_{\boldsymbol{M}}|}$ from (2.14) with $|\mathcal{I}_{\boldsymbol{M}_{\boldsymbol{\sigma}}}| = N$. Thereby, it results from the geometric sum formula that

$$\boldsymbol{F}^*\boldsymbol{F} = \left(\sum_{\boldsymbol{j} \in \mathcal{I}_{\boldsymbol{n}}} \mathrm{e}^{2\pi\mathrm{i}(\boldsymbol{k}-\boldsymbol{\ell})\boldsymbol{j}/n} \right)_{\boldsymbol{k},\boldsymbol{\ell} \in \mathcal{I}_{\boldsymbol{M}}} = N\boldsymbol{I}_{|\mathcal{I}_{\boldsymbol{M}}|}, \quad \text{if } |\mathcal{I}_{\boldsymbol{M}}| \leq N, \tag{3.1}$$

as well as

$$\boldsymbol{F}\boldsymbol{F}^* = \left(\sum_{\boldsymbol{k} \in \mathcal{I}_{\boldsymbol{M}}} \mathrm{e}^{2\pi\mathrm{i}\boldsymbol{k}(\boldsymbol{j}-\boldsymbol{h})/n} \right)_{\boldsymbol{j},\boldsymbol{h} \in \mathcal{I}_{\boldsymbol{n}}} = |\mathcal{I}_{\boldsymbol{M}}| \cdot \boldsymbol{I}_N, \quad \text{if } |\mathcal{I}_{\boldsymbol{M}}| \geq N \tag{3.2}$$

and $|\mathcal{I}_{\boldsymbol{M}}|$ is divisible by N. Thus, in the equispaced setting a one-sided inverse is given by the (scaled) adjoint matrix. However, when considering arbitrary points $\boldsymbol{x}_j \in \mathbb{T}^d$, $j = 1, \dots, N$, this property is lost, i. e., for the nonequispaced Fourier matrix we have

$$\boldsymbol{A}^*\boldsymbol{A} \neq N\boldsymbol{I}_{|\mathcal{I}_{\boldsymbol{M}}|} \quad \text{and} \quad \boldsymbol{A}\boldsymbol{A}^* \neq |\mathcal{I}_{\boldsymbol{M}}| \cdot \boldsymbol{I}_N. \tag{3.3}$$

Because of this, more effort is needed in the nonequispaced setting.

In general, we face the following two problems.

(1) Solve the linear system

$$\boldsymbol{A}\hat{\boldsymbol{f}} = \boldsymbol{f}, \tag{3.4}$$

i. e., reconstruct the Fourier coefficients $\hat{\boldsymbol{f}} = (\hat{f}_{\boldsymbol{k}})_{\boldsymbol{k} \in \mathcal{I}_{\boldsymbol{M}}}$ from given function values $\boldsymbol{f} = (f(\boldsymbol{x}_j))_{j=1}^N$. This problem is referred to as *inverse NDFT (iNDFT)* and an efficient solver shall be called *inverse NFFT (iNFFT)*.

(2) Solve the linear system

$$\boldsymbol{A}^* \boldsymbol{f} = \boldsymbol{h}, \tag{3.5}$$

i.e., reconstruct the coefficients $\boldsymbol{f} = (f_j)_{j=1}^N$ from given data $\boldsymbol{h} = (h_{\boldsymbol{k}})_{\boldsymbol{k} \in \mathcal{I}_{\boldsymbol{M}}}$. This problem is referred to as *inverse adjoint NDFT (iNDFT*)* and an efficient solver shall be called *inverse adjoint NFFT (iNFFT*)*.

Note that in both problems the numbers $|\mathcal{I}_{\boldsymbol{M}}|$ and N are independent, such that the nonequispaced Fourier matrix $\boldsymbol{A} \in \mathbb{C}^{N \times |\mathcal{I}_{\boldsymbol{M}}|}$ in (2.7) is generally rectangular.

At first, we restrict our attention to problem (3.4). When considering iterative inversion procedures as those mentioned in the introduction, these methods require multiple iteration steps by definition. Therefore, multiple matrix vector multiplications with the system matrix $\boldsymbol{A}$, or rather multiple applications of the NFFT (see Algorithm 2.1), are needed to compute a solution. To reduce the computational effort, we now proceed, in contrast to this iterated procedures, with so-called direct methods. In the setting of problem (3.4) we hereby mean methods, where for a fixed set of points $\boldsymbol{x}_j$, $j = 1, \ldots, N$, the reconstruction of $\hat{\boldsymbol{f}}$ from given $\boldsymbol{f}$ can be realized with the same number of arithmetic operations as a single application of an adjoint NFFT (see Algorithm 2.4). To achieve this, a certain precomputational step is compulsory, since the adjoint NFFT does not yield an inversion of the NFFT per se, see (3.3). Although this precomputations might be rather costly, they need to be done only once for a given set of points $\boldsymbol{x}_j$, $j = 1, \ldots, N$. In fact, the actual reconstruction step is very efficient. Therefore, direct methods are especially beneficial in case we are given fixed nodes for several measurement vectors $\boldsymbol{f}$.

In this section we focus on a direct inversion method for solving problem (3.4) that utilizes so-called *sampling density compensation*. To this end, we consider the integral (2.2) and introduce a corresponding quadrature formula. In contrast to the already known equispaced approximation (2.3) we now assume given arbitrary, nonequispaced points $\boldsymbol{x}_j \in \mathbb{T}^d$, $j = 1, \ldots, N$. Thereby, the Fourier coefficients (2.2) are approximated by a general quadrature rule using quadrature weights $w_j \in \mathbb{C}$, $j = 1, \ldots, N$, which are needed for sampling density compensation due to the nonequispaced sampling. Thus, for a trigonometric polynomial (2.8) we have

$$\hat{f}_{\boldsymbol{k}} = c_{\boldsymbol{k}}(f) \approx h_{\boldsymbol{k}}^{\mathrm{w}} := \sum_{j=1}^N w_j \, f(\boldsymbol{x}_j) \, \mathrm{e}^{-2\pi \mathrm{i} \boldsymbol{k} \boldsymbol{x}_j}, \quad \boldsymbol{k} \in \mathcal{I}_{\boldsymbol{M}}. \tag{3.6}$$

Using the nonequispaced Fourier matrix $\boldsymbol{A} \in \mathbb{C}^{N \times |\mathcal{I}_{\boldsymbol{M}}|}$ in (2.7), the diagonal matrix of weights $\boldsymbol{W} := \mathrm{diag}(w_j)_{j=1}^N \in \mathbb{C}^{N \times N}$ as well as the vector $\boldsymbol{h}^{\mathrm{w}} := (h_{\boldsymbol{k}}^{\mathrm{w}})_{\boldsymbol{k} \in \mathcal{I}_{\boldsymbol{M}}}$, the nonequispaced quadrature rule (3.6) can be written as $\hat{\boldsymbol{f}} \approx \boldsymbol{h}^{\mathrm{w}} := \boldsymbol{A}^* \boldsymbol{W} \boldsymbol{f}$. For achieving a fast computation method we make use of the approximation of the adjoint NFFT, cf. Section 2.2, i.e., the final approximation is given by

$$\hat{\boldsymbol{f}} \approx \tilde{\boldsymbol{h}}^{\mathrm{w}} := \boldsymbol{D}^* \boldsymbol{F}^* \boldsymbol{B}^* \boldsymbol{W} \boldsymbol{f}, \tag{3.7}$$

with the matrices $\boldsymbol{D} \in \mathbb{C}^{|\mathcal{I}_M| \times |\mathcal{I}_M|}$, $\boldsymbol{F} \in \mathbb{C}^{|\mathcal{I}_{M_\sigma}| \times |\mathcal{I}_M|}$ and $\boldsymbol{B} \in \mathbb{R}^{N \times |\mathcal{I}_{M_\sigma}|}$ defined in (2.13), (2.14) and (2.15). In other words, for density compensation methods the already mentioned precomputations consist of computing the quadrature weights $w_j \in \mathbb{C}$, $j = 1, \ldots, N$, while the actual reconstruction step includes only one adjoint NFFT (see Algorithm 2.4) applied to the scaled measurement vector $\boldsymbol{W}\boldsymbol{f}$.

The aim of all density compensation techniques is then to choose appropriate weights $w_j \in \mathbb{C}$, $j = 1, \ldots, N$, such that the underlying quadrature (3.6) is preferably exact. In the following we have a look at the specific choice of the so-called *density compensation factors* w_j.

An intuitive approach for density compensation is based on geometry, where each sample is considered as representative of a certain surrounding area, as in numerical integration. The weights for each sample can be obtained for instance by constructing a Voronoi diagram and calculating the area of each cell, see e. g. [58]. This approach of *Voronoi weights* is well-known and widely used in practice. However, it does not necessarily yield a good approximation (3.7), which is why we examine some more sophisticated approaches in the remainder of this section.

To this end, this section is organized as follows. Firstly, in Section 3.1 we introduce density compensation factors w_j, $j = 1, \ldots, N$, that lead to an exact reconstruction formula (3.6) for all trigonometric polynomials (2.8) of degree $\boldsymbol{M}$. In addition to the theoretical results, we also discuss methods for the numerical computation. Secondly, in Section 3.2 we show that it is reasonable to consider the inversion problem (3.4) and density compensation via (3.7) for bandlimited functions $f \in L_1(\mathbb{R}^d) \cap C_0(\mathbb{R}^d)$ as well. Subsequently, we summarize our previous findings by presenting a general error bound on density compensation factors in Section 3.3. Finally, in Section 3.4 we reconsider certain approaches from literature and illustrate their connection among each other as well as to the method introduced in Section 3.1.

Remark 3.1. *Recapitulating, we have a closer look at some possible interpretation perspectives on the reconstruction (3.7).*

(i) *If we define $\boldsymbol{g} := \boldsymbol{W}\boldsymbol{f}$, i. e., each entry of $\boldsymbol{f}$ is scaled with respect to the nodes $\boldsymbol{x}_j$, $j = 1, \ldots, N$, the approximation (3.7) can be written as $\hat{\boldsymbol{f}} \approx \boldsymbol{D}^* \boldsymbol{F}^* \boldsymbol{B}^* \boldsymbol{g}$. As mentioned before, this coincides with an ordinary adjoint NFFT applied to a modified coefficient vector $\boldsymbol{g}$.*

(ii) *By defining the matrix $\tilde{\boldsymbol{B}} := \boldsymbol{W}^* \boldsymbol{B}$, i. e., scaling the rows of $\boldsymbol{B}$ with respect to the nodes $\boldsymbol{x}_j$, $j = 1, \ldots, N$, the approximation (3.7) can be written as $\hat{\boldsymbol{f}} \approx \boldsymbol{D}^* \boldsymbol{F}^* \tilde{\boldsymbol{B}}^* \boldsymbol{f}$. In this sense, density compensation can also be seen as a modification of the adjoint NFFT and its application to the original coefficient vector.*

Note that (i) is the common viewpoint. However, we keep (ii) in mind, since this allows treating density compensation methods as an optimization of the sparse matrix $\boldsymbol{B} \in \mathbb{R}^{N \times |\mathcal{I}_{M_\sigma}|}$ in (2.15), as it shall be done in Section 4. We remark that density compensation methods allow only N degrees of freedom. $\qquad\square$

3.1 Exact quadrature weights for trigonometric polynomials

Similar to [28], we aim to introduce density compensation factors w_j, $j = 1, \ldots, N$, that lead to an exact reconstruction formula (3.6) for all trigonometric polynomials (2.8) of degree $\boldsymbol{M}$. To this end, we firstly examine certain properties that arise from (3.6) being exact.

Theorem 3.2. *Let a polynomial degree* $\boldsymbol{M} \in (2\mathbb{N})^d$, *nonequispaced nodes* $\boldsymbol{x}_j \in \mathbb{T}^d$, $j = 1, \ldots, N$, *and quadrature weights* $w_j \in \mathbb{C}$ *be given. Then an exact reconstruction formula (3.6) for trigonometric polynomials (2.8) with maximum degree* $\boldsymbol{M}$ *satisfying*

$$\hat{f}_{\boldsymbol{k}} = c_{\boldsymbol{k}}(f) = h_{\boldsymbol{k}}^{\mathrm{w}}, \quad \boldsymbol{k} \in \mathcal{I}_{\boldsymbol{M}}, \tag{3.8}$$

implies the following equivalent statements.

(i) *The quadrature rule*

$$\int_{\mathbb{T}^d} f(\boldsymbol{x}) \, \mathrm{d}\boldsymbol{x} = \sum_{j=1}^{N} w_j f(\boldsymbol{x}_j) \tag{3.9}$$

is exact for all trigonometric polynomials (2.8) with maximum degree $\boldsymbol{M}$.

(ii) *The linear system of equations*

$$\left[\boldsymbol{A}^T \boldsymbol{w}\right]_{\boldsymbol{k}} = \sum_{j=1}^{N} w_j \, \mathrm{e}^{2\pi\mathrm{i}\boldsymbol{k}\boldsymbol{x}_j} = \delta_{\boldsymbol{0}, \boldsymbol{k}}, \quad \boldsymbol{k} \in \mathcal{I}_{\boldsymbol{M}}, \tag{3.10}$$

is fulfilled with the matrix $\boldsymbol{A} \in \mathbb{C}^{N \times |\mathcal{I}_{\boldsymbol{M}}|}$ *in (2.7) and* $\boldsymbol{w} := (w_j)_{j=1}^{N}$.

Proof.
$(3.8) \Rightarrow (i)$: By inserting the definition (2.8) of a trigonometric polynomial of degree $\boldsymbol{M}$ into the integral considered in (3.9) we have

$$\int_{\mathbb{T}^d} f(\boldsymbol{x}) \, \mathrm{d}\boldsymbol{x} = \sum_{\boldsymbol{k} \in \mathcal{I}_{\boldsymbol{M}}} \hat{f}_{\boldsymbol{k}} \cdot \int_{\mathbb{T}^d} \mathrm{e}^{2\pi\mathrm{i}\boldsymbol{k}\boldsymbol{x}} \, \mathrm{d}\boldsymbol{x} = \sum_{\boldsymbol{k} \in \mathcal{I}_{\boldsymbol{M}}} \hat{f}_{\boldsymbol{k}} \cdot \delta_{\boldsymbol{0}, \boldsymbol{k}} = \hat{f}_{\boldsymbol{0}}. \tag{3.11}$$

Now using the property (3.8) as well as definition (3.6) of $h_{\boldsymbol{k}}^{\mathrm{w}}$ we proceed with

$$\hat{f}_{\boldsymbol{0}} = h_{\boldsymbol{0}}^{\mathrm{w}} = \sum_{j=1}^{N} w_j f(\boldsymbol{x}_j) \sum_{\boldsymbol{k} \in \mathcal{I}_{\boldsymbol{M}}} \mathrm{e}^0 = \sum_{j=1}^{N} w_j f(\boldsymbol{x}_j), \tag{3.12}$$

such that (3.11) combined with (3.12) yields the assertion (3.9).
$(i) \Rightarrow (ii)$: Inserting the definition (2.8) of a trigonometric polynomial of degree $\boldsymbol{M}$ into the right-hand side of (3.9) it holds

$$\sum_{j=1}^{N} w_j f(\boldsymbol{x}_j) = \sum_{j=1}^{N} w_j \sum_{\boldsymbol{k} \in \mathcal{I}_{\boldsymbol{M}}} \hat{f}_{\boldsymbol{k}} \, \mathrm{e}^{2\pi\mathrm{i}\boldsymbol{k}\boldsymbol{x}_j} = \sum_{\boldsymbol{k} \in \mathcal{I}_{\boldsymbol{M}}} \hat{f}_{\boldsymbol{k}} \sum_{j=1}^{N} w_j \, \mathrm{e}^{2\pi\mathrm{i}\boldsymbol{k}\boldsymbol{x}_j}. \tag{3.13}$$

This together with the property (i) and (3.11) leads to

$$\hat{f}_{\mathbf{0}} = \sum_{\mathbf{k} \in \mathcal{I}_M} \hat{f}_{\mathbf{k}} \sum_{j=1}^{N} w_j \, \mathrm{e}^{2\pi\mathrm{i}\mathbf{k}\mathbf{x}_j}$$

and thus to assertion (3.10).

(ii) $\Rightarrow$ (i): Combining (3.11), (3.10) and (3.13) yields the assertion via

$$\int_{\mathbb{T}^d} f(\mathbf{x}) \, \mathrm{d}\mathbf{x} = \sum_{\mathbf{k} \in \mathcal{I}_M} \hat{f}_{\mathbf{k}} \cdot \delta_{\mathbf{0},\mathbf{k}} = \sum_{\mathbf{k} \in \mathcal{I}_M} \hat{f}_{\mathbf{k}} \sum_{j=1}^{N} w_j \, \mathrm{e}^{2\pi\mathrm{i}\mathbf{k}\mathbf{x}_j} = \sum_{j=1}^{N} w_j f(\mathbf{x}_j).$$

■

Remark 3.3. *Comparable results can also be found in literature. A fundamental theorem in numerical integration, see [70], states that for any integral $\int_{\mathbb{T}^d} f(\mathbf{x}) \, \mathrm{d}\mathbf{x}$ there exists an exact quadrature rule (3.9), i.e., optimal points $\mathbf{x}_j \in \mathbb{T}^d$ and weights $w_j \in \mathbb{C}$, $j = 1, \dots, N$, such that (3.9) is fulfilled. In [31, Lemma 2.6] it was shown that for given points $\mathbf{x}_j \in \mathbb{T}^d$, $j = 1, \dots, N$, certain quadrature weights w_j can be stated by means of frame theoretical considerations which lead to an exact quadrature rule (3.9) by definition. Moreover, it was shown (cf. [31, Lemma 3.6]) that these weights are the ones with minimal (weighted) ℓ_2-norm, which are already known under the name "least squares quadrature", see [34]. According to [34, Sec. 2.1] these quadrature weights w_j, $j = 1, \dots, N$, can be found by solving a linear system of equations $\mathbf{\Phi}\mathbf{w} = \mathbf{v}$, where for a set of basis functions $\{\phi_{\mathbf{k}}\}_{\mathbf{k} \in \mathcal{I}_M}$ it holds $\Phi_{\mathbf{k},j} = \phi_{\mathbf{k}}(\mathbf{x}_j)$ and $v_{\mathbf{k}} = \int_{\mathbb{T}^d} \phi_{\mathbf{k}}(\mathbf{x}) \, \mathrm{d}\mathbf{x}$. In our setting we have $\phi_{\mathbf{k}}(\mathbf{x}) = \mathrm{e}^{2\pi\mathrm{i}\mathbf{k}\mathbf{x}}$ and therefore*

$$\mathbf{\Phi} = \left(\mathrm{e}^{2\pi\mathrm{i}\mathbf{k}\mathbf{x}_j} \right)_{\mathbf{k},j} = \mathbf{A}^T \quad \text{and} \quad v_{\mathbf{k}} = \int_{\mathbb{T}^d} 1 \cdot \mathrm{e}^{2\pi\mathrm{i}\mathbf{k}\mathbf{x}} \, \mathrm{d}\mathbf{x} = \delta_{\mathbf{0},\mathbf{k}},$$

i.e., the same linear system of equations as in (3.10). We remark that both [31] and [34] state the results in the case $d = 1$, a generalization to $d > 1$, however, is straight-forward.

□

By means of Theorem 3.2 we can now give a condition that guaranties (3.6) being exact for all trigonometric polynomials (2.8) with maximum degree $\mathbf{M}$.

Corollary 3.4. *The two statements (i) and (ii) in Theorem 3.2 are not equivalent to property (3.8), since (3.10) does not imply an exact reconstruction in (3.6).*

However, an augmented variant of (3.10), namely

$$\sum_{j=1}^{N} w_j \, \mathrm{e}^{2\pi\mathrm{i}\mathbf{k}\mathbf{x}_j} = \delta_{\mathbf{0},\mathbf{k}}, \quad \mathbf{k} \in \mathcal{I}_{2M}, \tag{3.14}$$

yields an exact reconstruction $\hat{f}_{\mathbf{k}} = h_{\mathbf{k}}^{\mathrm{w}}$ in (3.6) for trigonometric polynomials (2.8) with maximum degree $\mathbf{M}$. Additionally, (3.14) implies the matrix equation $\mathbf{A}^* \mathbf{W} \mathbf{A} = \mathbf{I}_{|\mathcal{I}_M|}$ with $\mathbf{A} \in \mathbb{C}^{N \times |\mathcal{I}_M|}$ in (2.7) and the identity matrix $\mathbf{I}_{|\mathcal{I}_M|}$ of size $|\mathcal{I}_M|$.

Proof. Utilizing definitions (3.6) and (2.8) we have

$$h_{\boldsymbol{k}}^{\mathrm{w}} = \sum_{j=1}^{N} w_j \left(\sum_{\boldsymbol{\ell} \in \mathcal{I}_{\boldsymbol{M}}} \hat{f}_{\boldsymbol{\ell}} \, \mathrm{e}^{2\pi \mathrm{i} \boldsymbol{\ell} \boldsymbol{x}_j} \right) \mathrm{e}^{-2\pi \mathrm{i} \boldsymbol{k} \boldsymbol{x}_j} = \sum_{\boldsymbol{\ell} \in \mathcal{I}_{\boldsymbol{M}}} \hat{f}_{\boldsymbol{\ell}} \sum_{j=1}^{N} w_j \, \mathrm{e}^{2\pi \mathrm{i} (\boldsymbol{\ell} - \boldsymbol{k}) \boldsymbol{x}_j}$$

$$= \sum_{\substack{\boldsymbol{\ell} \in \mathcal{I}_{\boldsymbol{M}} \\ (\boldsymbol{\ell} - \boldsymbol{k}) \in \mathcal{I}_{\boldsymbol{M}}}} \hat{f}_{\boldsymbol{\ell}} \sum_{j=1}^{N} w_j \, \mathrm{e}^{2\pi \mathrm{i} (\boldsymbol{\ell} - \boldsymbol{k}) \boldsymbol{x}_j} + \sum_{\substack{\boldsymbol{\ell} \in \mathcal{I}_{\boldsymbol{M}} \\ (\boldsymbol{\ell} - \boldsymbol{k}) \notin \mathcal{I}_{\boldsymbol{M}}}} \hat{f}_{\boldsymbol{\ell}} \sum_{j=1}^{N} w_j \, \mathrm{e}^{2\pi \mathrm{i} (\boldsymbol{\ell} - \boldsymbol{k}) \boldsymbol{x}_j}, \quad \boldsymbol{k} \in \mathcal{I}_{\boldsymbol{M}}.$$

Since (3.10) only holds for $\boldsymbol{k}, \boldsymbol{\ell} \in \mathcal{I}_{\boldsymbol{M}}$ with $(\boldsymbol{\ell} - \boldsymbol{k}) \in \mathcal{I}_{\boldsymbol{M}}$, this implies

$$h_{\boldsymbol{k}}^{\mathrm{w}} = \hat{f}_{\boldsymbol{k}} + \sum_{\substack{\boldsymbol{\ell} \in \mathcal{I}_{\boldsymbol{M}} \\ (\boldsymbol{\ell} - \boldsymbol{k}) \notin \mathcal{I}_{\boldsymbol{M}}}} \hat{f}_{\boldsymbol{\ell}} \sum_{j=1}^{N} w_j \, \mathrm{e}^{2\pi \mathrm{i} (\boldsymbol{\ell} - \boldsymbol{k}) \boldsymbol{x}_j}, \quad \boldsymbol{k} \in \mathcal{I}_{\boldsymbol{M}},$$

where for all $\boldsymbol{k} \in \mathcal{I}_{\boldsymbol{M}} \setminus \{\boldsymbol{0}\}$ there exists an $\boldsymbol{\ell} \in \mathcal{I}_{\boldsymbol{M}}$ with $(\boldsymbol{\ell} - \boldsymbol{k}) \in \mathcal{I}_{\boldsymbol{2M}} \setminus \mathcal{I}_{\boldsymbol{M}}$.

As $(\boldsymbol{\ell} - \boldsymbol{k}) \in \mathcal{I}_{\boldsymbol{2M}}$ for $\boldsymbol{k}, \boldsymbol{\ell} \in \mathcal{I}_{\boldsymbol{M}}$, the augmented variant (3.14) yields

$$h_{\boldsymbol{k}}^{\mathrm{w}} = \sum_{\boldsymbol{\ell} \in \mathcal{I}_{\boldsymbol{M}}} \hat{f}_{\boldsymbol{\ell}} \sum_{j=1}^{N} w_j \, \mathrm{e}^{2\pi \mathrm{i} (\boldsymbol{\ell} - \boldsymbol{k}) \boldsymbol{x}_j} = \sum_{\boldsymbol{\ell} \in \mathcal{I}_{\boldsymbol{M}}} \hat{f}_{\boldsymbol{k}} \cdot \delta_{\boldsymbol{0}, \boldsymbol{k}} = \hat{f}_{\boldsymbol{k}}, \quad \boldsymbol{k} \in \mathcal{I}_{\boldsymbol{M}}.$$

Moreover, since $\delta_{(\boldsymbol{\ell} - \boldsymbol{k}), \boldsymbol{0}} = \delta_{\boldsymbol{k}, \boldsymbol{\ell}}$, the condition (3.14) implies

$$\delta_{\boldsymbol{k}, \boldsymbol{\ell}} = \sum_{j=1}^{N} w_j \, \mathrm{e}^{2\pi \mathrm{i} (\boldsymbol{\ell} - \boldsymbol{k}) \boldsymbol{x}_j} = \sum_{j=1}^{N} \mathrm{e}^{-2\pi \mathrm{i} \boldsymbol{k} \boldsymbol{x}_j} \left(w_j \, \mathrm{e}^{2\pi \mathrm{i} \boldsymbol{\ell} \boldsymbol{x}_j} \right), \quad \boldsymbol{k}, \boldsymbol{\ell} \in \mathcal{I}_{\boldsymbol{M}}.$$

In matrix-vector notation this can be written as $\boldsymbol{A}^* \boldsymbol{W} \boldsymbol{A} = \boldsymbol{I}_{|\mathcal{I}_{\boldsymbol{M}}|}$ with $\boldsymbol{A} \in \mathbb{C}^{N \times |\mathcal{I}_{\boldsymbol{M}}|}$ in (2.7) and the identity matrix $\boldsymbol{I}_{|\mathcal{I}_{\boldsymbol{M}}|}$ of size $|\mathcal{I}_{\boldsymbol{M}}|$. We remark that this matrix equation immediately shows that we have an exact reconstruction of the form (3.8), since if $\boldsymbol{A}^* \boldsymbol{W} \boldsymbol{A} = \boldsymbol{I}_{|\mathcal{I}_{\boldsymbol{M}}|}$ is fulfilled, it holds by (3.4) that $\hat{\boldsymbol{f}} = \boldsymbol{A}^* \boldsymbol{W} \boldsymbol{A} \hat{\boldsymbol{f}} = \boldsymbol{A}^* \boldsymbol{W} \boldsymbol{f}$. ∎

Remark 3.5. *Let $f \in L_2(\mathbb{T}^d)$ be an arbitrary function (2.1). Then (3.14) yields*

$$h_{\boldsymbol{k}}^{\mathrm{w}} = \sum_{\boldsymbol{\ell} \in \mathbb{Z}^d} c_{\boldsymbol{\ell}}(f) \sum_{j=1}^{N} w_j \, \mathrm{e}^{2\pi \mathrm{i} (\boldsymbol{\ell} - \boldsymbol{k}) \boldsymbol{x}_j}$$

$$= \sum_{\substack{\boldsymbol{\ell} \in \mathbb{Z}^d \\ (\boldsymbol{\ell} - \boldsymbol{k}) \in \mathcal{I}_{\boldsymbol{2M}}}} c_{\boldsymbol{\ell}}(f) \sum_{j=1}^{N} w_j \, \mathrm{e}^{2\pi \mathrm{i} (\boldsymbol{\ell} - \boldsymbol{k}) \boldsymbol{x}_j} + \sum_{\substack{\boldsymbol{\ell} \in \mathbb{Z}^d \\ (\boldsymbol{\ell} - \boldsymbol{k}) \notin \mathcal{I}_{\boldsymbol{2M}}}} c_{\boldsymbol{\ell}}(f) \sum_{j=1}^{N} w_j \, \mathrm{e}^{2\pi \mathrm{i} (\boldsymbol{\ell} - \boldsymbol{k}) \boldsymbol{x}_j}$$

$$= c_{\boldsymbol{k}}(f) + \sum_{\substack{\boldsymbol{\ell} \in \mathbb{Z}^d \\ (\boldsymbol{\ell} - \boldsymbol{k}) \notin \mathcal{I}_{\boldsymbol{2M}}}} c_{\boldsymbol{\ell}}(f) \sum_{j=1}^{N} w_j \, \mathrm{e}^{2\pi \mathrm{i} (\boldsymbol{\ell} - \boldsymbol{k}) \boldsymbol{x}_j}, \quad \boldsymbol{k} \in \mathcal{I}_{\boldsymbol{M}},$$

i. e., for a function $f \in L_2(\mathbb{T}^d)$ we only have a good approximation in case the coefficients $c_{\boldsymbol{\ell}}(f)$ are small for $\boldsymbol{\ell} \notin \mathcal{I}_{\boldsymbol{M}}$, whereas this reconstruction can only be exact for f being a trigonometric polynomial (2.8). □

3.1.1 Practical computation in the underdetermined setting $|\mathcal{I}_{2M}| \leq N$

So far, we have seen in Corollary 3.4 that an exact solution $\boldsymbol{w} = (w_j)_{j=1}^N$ to the linear system (3.14) leads to an exact reconstruction formula (3.6) for all trigonometric polynomials (2.8) with maximum degree $\boldsymbol{M}$. Therefore, we aim to use this condition (3.14) to numerically find optimal density compensation factors $w_j \in \mathbb{C}$, $j = 1, \ldots, N$.

Having a closer look at the condition (3.14) we recognize that it can be written as the linear system of equations $\boldsymbol{A}_{|\mathcal{I}_{2M}|}^T \, \boldsymbol{w} = \boldsymbol{e_0}$ with the matrix $\boldsymbol{A}_{|\mathcal{I}_{2M}|} \in \mathbb{C}^{N \times |\mathcal{I}_{2M}|}$, cf. (2.7), and right side $\boldsymbol{e_0} \coloneqq (\delta_{\boldsymbol{0},\boldsymbol{k}})_{\boldsymbol{k} \in \mathcal{I}_{2M}}$. We remark that in contrast to $\boldsymbol{A} \in \mathbb{C}^{N \times |\mathcal{I}_M|}$ we now deal with the enlarged matrix $\boldsymbol{A}_{|\mathcal{I}_{2M}|} \in \mathbb{C}^{N \times |\mathcal{I}_{2M}|}$, such that single matrix operations are more costly. Nevertheless, Corollary 3.4 yields a direct inversion method for (3.4), where the system $\boldsymbol{A}_{|\mathcal{I}_{2M}|}^T \, \boldsymbol{w} = \boldsymbol{e_0}$ needs to be solved only once for fixed nodes $\boldsymbol{x}_j \in \mathbb{T}^d$, $j = 1, \ldots, N$. Its solution $\boldsymbol{w}$ can then be used to efficiently approximate $\hat{\boldsymbol{f}}$ for multiple measurement vectors $\boldsymbol{f}$, whereas iterative methods for (3.4) need to solve $\boldsymbol{A}\hat{\boldsymbol{f}} = \boldsymbol{f}$ each time.

As already mentioned in [34, Sec. 3.1] an exact solution to (3.14) can only be found if $|\mathcal{I}_{2M}| \leq N$, i.e., in case $\boldsymbol{A}_{|\mathcal{I}_{2M}|}^T \, \boldsymbol{w} = \boldsymbol{e_0}$ is an underdetermined system of equations. By [34, Lem. 3.1] this system has at least one solution, which is why we may choose the one with minimal ℓ_2-norm. If $\mathrm{rank}(\boldsymbol{A}_{|\mathcal{I}_{2M}|}) = |\mathcal{I}_{2M}|$, then the system $\boldsymbol{A}_{|\mathcal{I}_{2M}|}^T \, \boldsymbol{w} = \boldsymbol{e_0}$ is consistent and the unique solution is given by the normal equations of second kind

$$\boldsymbol{A}_{|\mathcal{I}_{2M}|}^T \, \overline{\boldsymbol{A}_{|\mathcal{I}_{2M}|}} \, \boldsymbol{v} = \boldsymbol{e_0}, \quad \overline{\boldsymbol{A}_{|\mathcal{I}_{2M}|}} \, \boldsymbol{v} = \boldsymbol{w}. \tag{3.15}$$

More precisely, we may compute the vector $\boldsymbol{v}$ using an iterative procedure such as PCG, such that only matrix multiplications with $\boldsymbol{A}_{|\mathcal{I}_{2M}|}^T$ and $\overline{\boldsymbol{A}_{|\mathcal{I}_{2M}|}}$ are needed. Since fast multiplication with $\boldsymbol{A}_{|\mathcal{I}_{2M}|}^T$ and $\overline{\boldsymbol{A}_{|\mathcal{I}_{2M}|}}$ can easily be realized by means of an adjoint NFFT (see Algorithm 2.4) and an NFFT (see Algorithm 2.1), respectively, computing the solution $\boldsymbol{w}$ to (3.15) is of complexity $\mathcal{O}(|\mathcal{I}_{2M}| \log(|\mathcal{I}_{2M}|) + N)$, where

$$|\mathcal{I}_{2M}| = (2M)^d = 2^d M^d = 2^d \, |\mathcal{I}_M|.$$

Thus, in order to receive exact quadrature weights $w_j \in \mathbb{C}$, $j = 1, \ldots, N$, via (3.15) we need to satisfy the full rank condition $\mathrm{rank}(\boldsymbol{A}_{|\mathcal{I}_{2M}|}) = |\mathcal{I}_{2M}|$. In case of a low rank matrix $\boldsymbol{A}_{|\mathcal{I}_{2M}|} \in \mathbb{C}^{N \times |\mathcal{I}_{2M}|}$ for $|\mathcal{I}_{2M}| \leq N$, we may still use (3.15) to obtain a least squares approximation to (3.14).

3.1.2 Practical computation in the overdetermined setting $|\mathcal{I}_{2M}| > N$

In the setting $|\mathcal{I}_{2M}| > N$, we cannot expect finding an exact solution $\boldsymbol{w}$ to (3.14), since we have to deal with an overdetermined system possessing more conditions than variables. However, we still aim to numerically find optimal density compensation factors $w_j \in \mathbb{C}$, $j = 1, \ldots, N$, by considering a least squares approximation to (3.14) that minimizes $\left\| \boldsymbol{A}_{|\mathcal{I}_{2M}|}^T \, \boldsymbol{w} - \boldsymbol{e_0} \right\|_2$. In [8, Thm. 1.1.2] it was shown that every least squares solution satisfies the normal equations of first kind

$$\overline{\boldsymbol{A}_{|\mathcal{I}_{2M}|}} \, \boldsymbol{A}_{|\mathcal{I}_{2M}|}^T \, \boldsymbol{w} = \overline{\boldsymbol{A}_{|\mathcal{I}_{2M}|}} \, \boldsymbol{e_0}. \tag{3.16}$$

By means of definitions of $\boldsymbol{A}_{|\mathcal{I}_{\boldsymbol{2M}}|} \in \mathbb{C}^{N \times |\mathcal{I}_{\boldsymbol{2M}}|}$, cf. (2.7), and $\boldsymbol{e_0} = (\delta_{\boldsymbol{0,k}})_{\boldsymbol{k} \in \mathcal{I}_{\boldsymbol{2M}}}$ we simplify the right hand side via

$$\overline{\boldsymbol{A}_{|\mathcal{I}_{\boldsymbol{2M}}|}} \, \boldsymbol{e_0} = \left(\sum_{\boldsymbol{k} \in \mathcal{I}_{\boldsymbol{2M}}} \delta_{\boldsymbol{0,k}} \, \mathrm{e}^{-2\pi \mathrm{i} \boldsymbol{k} \boldsymbol{x}_j} \right)_{j=1}^{N} = \boldsymbol{1}_N.$$

Since fast multiplication with $\boldsymbol{A}_{|\mathcal{I}_{\boldsymbol{2M}}|}^{T}$ and $\overline{\boldsymbol{A}_{|\mathcal{I}_{\boldsymbol{2M}}|}}$ can easily be realized by means of an adjoint NFFT (see Algorithm 2.4) and an NFFT (see Algorithm 2.1), respectively, the solution $\boldsymbol{w}$ to (3.16) can be computed iteratively by means of PCG in $\mathcal{O}(|\mathcal{I}_{\boldsymbol{2M}}| \log(|\mathcal{I}_{\boldsymbol{2M}}|) + N)$ arithmetic operations. Note that the solution to (3.16) is only unique if the full rank condition $\mathrm{rank}(\boldsymbol{A}_{|\mathcal{I}_{\boldsymbol{2M}}|}) = N$ is satisfied, cf. [8, p. 7]. We remark that the computed weight matrix $\boldsymbol{W} = \mathrm{diag}(\boldsymbol{w})$ can further be used in an iterative procedure as in [53, Alg. 7.27] to improve the approximation of $\hat{\boldsymbol{f}}$.

The previous considerations lead to the following algorithms.

Algorithm 3.6 (Computation of the optimal density compensation factors)**.**

For $d, N \in \mathbb{N}$ let $\boldsymbol{x}_j \in \mathbb{T}^d$, $j = 1, \ldots, N$, as well as $M \in 2\mathbb{N}$ and $\boldsymbol{M} := M \cdot \boldsymbol{1}_d$ be given.

   1. Compute $|\mathcal{I}_{\boldsymbol{2M}}| = (2M)^d$. $\mathcal{O}(1)$

   2. If $|\mathcal{I}_{\boldsymbol{2M}}| \leq N$

       Compute the solution $\boldsymbol{v}$ to (3.15) iteratively using the NFFT.

                                         $\mathcal{O}(|\mathcal{I}_{\boldsymbol{2M}}| \log(|\mathcal{I}_{\boldsymbol{2M}}|) + N)$

       Compute the solution $\boldsymbol{w} = \overline{\boldsymbol{A}_{|\mathcal{I}_{\boldsymbol{2M}}|}} \, \boldsymbol{v}$, see (3.15), using an NFFT.

                                         $\mathcal{O}(|\mathcal{I}_{\boldsymbol{2M}}| \log(|\mathcal{I}_{\boldsymbol{2M}}|) + N)$

     elseif $|\mathcal{I}_{\boldsymbol{2M}}| > N$

       Compute the solution $\boldsymbol{w}$ to (3.16) iteratively using the NFFT.

                                         $\mathcal{O}(|\mathcal{I}_{\boldsymbol{2M}}| \log(|\mathcal{I}_{\boldsymbol{2M}}|) + N)$

   3. Compose $\boldsymbol{W} = \mathrm{diag}(\boldsymbol{w}) \in \mathbb{R}^{N \times N}$. $\mathcal{O}(N)$

Output: *weights matrix $\boldsymbol{W}$*   **Complexity:** $\mathcal{O}(|\mathcal{I}_{\boldsymbol{2M}}| \log(|\mathcal{I}_{\boldsymbol{2M}}|) + N)$

Algorithm 3.7 (iNFFT – density compensation approach)**.**

For $d, N \in \mathbb{N}$ let $\boldsymbol{x}_j \in \mathbb{T}^d$, $j = 1, \ldots, N$, as well as $\boldsymbol{f} \in \mathbb{C}^N$, $M \in 2\mathbb{N}$ and $\boldsymbol{M} := M \cdot \boldsymbol{1}_d$ be given.

   0. Precompute the weights matrix $\boldsymbol{W}$ using Algorithm 3.6.

   1. Compute $\tilde{\boldsymbol{h}}^{\mathrm{w}} := \boldsymbol{D}^* \boldsymbol{F}^* \boldsymbol{B}^* \boldsymbol{W} \boldsymbol{f}$, cf. (3.7), by means of an adjoint NFFT.

Output: $\tilde{\boldsymbol{h}}^{\mathrm{w}} \approx \hat{\boldsymbol{f}} \in \mathbb{C}^{|\mathcal{I}_M|}$, *cf. (3.4).*   **Complexity:** $\mathcal{O}(|\mathcal{I}_M| \log(|\mathcal{I}_M|) + N)$

3.2 Bandlimited functions

In some numerical examples, such as in MRI, we are concerned with bandlimited functions $f \in L_1(\mathbb{R}^d) \cap C_0(\mathbb{R}^d)$ instead of trigonometric polynomials $f \in L_2(\mathbb{T}^d)$ in (2.8), cf. [15]. In the following we show that it is reasonable to consider the inversion problem (3.4) as well as density compensation via (3.7) for bandlimited functions as well.

To this end, let $f \in L_1(\mathbb{R}^d) \cap C_0(\mathbb{R}^d)$ be a bandlimited function with bandwidth $\boldsymbol{M}$, i. e., its (continuous) Fourier transform

$$\hat{f}(\boldsymbol{v}) := \int\limits_{\mathbb{R}^d} f(\boldsymbol{x}) \, \mathrm{e}^{-2\pi \mathrm{i} \boldsymbol{v} \boldsymbol{x}} \, \mathrm{d}\boldsymbol{x}, \quad \boldsymbol{v} \in \mathbb{R}^d, \tag{3.17}$$

is supported on $\left[-\frac{M}{2}, \frac{M}{2}\right)^d$. Utilizing this fact, we have $\hat{f} \in L_1(\mathbb{R}^d)$ and thus by the Fourier inversion theorem [53, Thm. 2.10] the inverse Fourier transform of f can be written as

$$f(\boldsymbol{x}) = \int\limits_{\mathbb{R}^d} \hat{f}(\boldsymbol{v}) \, \mathrm{e}^{2\pi \mathrm{i} \boldsymbol{v} \boldsymbol{x}} \, \mathrm{d}\boldsymbol{v} = \int\limits_{\left[-\frac{M}{2}, \frac{M}{2}\right)^d} \hat{f}(\boldsymbol{v}) \, \mathrm{e}^{2\pi \mathrm{i} \boldsymbol{v} \boldsymbol{x}} \, \mathrm{d}\boldsymbol{v}, \quad \boldsymbol{x} \in \mathbb{R}^d. \tag{3.18}$$

Analogous to (2.3), the approximation using equispaced quadrature points $\boldsymbol{k} \in \mathcal{I}_{\boldsymbol{M}}$ yields

$$f(\boldsymbol{x}) \approx \frac{\left(\frac{M}{2} - \left(-\frac{M}{2}\right)\right)^d}{|\mathcal{I}_{\boldsymbol{M}}|} \sum_{\boldsymbol{k} \in \mathcal{I}_{\boldsymbol{M}}} \hat{f}(\boldsymbol{k}) \, \mathrm{e}^{2\pi \mathrm{i} \boldsymbol{k} \boldsymbol{x}} = \sum_{\boldsymbol{k} \in \mathcal{I}_{\boldsymbol{M}}} \hat{f}(\boldsymbol{k}) \, \mathrm{e}^{2\pi \mathrm{i} \boldsymbol{k} \boldsymbol{x}}, \quad \boldsymbol{x} \in \mathbb{R}^d, \tag{3.19}$$

such that evaluation at the given nonequispaced nodes $\boldsymbol{x}_j \in \left[-\frac{1}{2}, \frac{1}{2}\right)^d$, $j = 1, \dots, N$, leads to

$$f(\boldsymbol{x}_j) \approx \sum_{\boldsymbol{k} \in \mathcal{I}_{\boldsymbol{M}}} \hat{f}(\boldsymbol{k}) \, \mathrm{e}^{2\pi \mathrm{i} \boldsymbol{k} \boldsymbol{x}_j}.$$

By means of the definition (2.7) of the matrix $\boldsymbol{A} \in \mathbb{C}^{N \times |\mathcal{I}_{\boldsymbol{M}}|}$ this can be written as $\boldsymbol{f} \approx \boldsymbol{A}\hat{\boldsymbol{f}}$, where we used the notation $\hat{\boldsymbol{f}} := (\hat{f}(\boldsymbol{k}))_{\boldsymbol{k} \in \mathcal{I}_{\boldsymbol{M}}}$ in this setting. Thus, also for bandlimited functions f its evaluations at points $\boldsymbol{x}_j$ can be approximated in the form (2.5), such that it is reasonable to consider the inversion problem (3.4) for bandlimited functions as well.

Considering (3.17) we are given an exact formula for the evaluation of the Fourier transform $\hat{f}$. However, in practical applications, such as MRI, this is only a hypothetical case, since f cannot be sampled on whole $\mathbb{R}^d$, cf. [15]. Due to a limited coverage of space by the acquisition, the function f is typically only known on a bounded domain, w.l.o.g. for $\boldsymbol{x} \in \left[-\frac{1}{2}, \frac{1}{2}\right)^d$. Thus, we have to deal with the approximation

$$\hat{f}(\boldsymbol{v}) \approx \int\limits_{\left[-\frac{1}{2}, \frac{1}{2}\right)^d} f(\boldsymbol{x}) \, \mathrm{e}^{-2\pi \mathrm{i} \boldsymbol{v} \boldsymbol{x}} \, \mathrm{d}\boldsymbol{x}, \quad \boldsymbol{v} \in \left[-\frac{M}{2}, \frac{M}{2}\right)^d. \tag{3.20}$$

Using the nonequispaced quadrature rule in (3.6), we find that evaluation at uniform grid points $\boldsymbol{k} \in \mathcal{I}_{\boldsymbol{M}}$ can be approximated via

$$\hat{f}(\boldsymbol{k}) \approx \tilde{h}(\boldsymbol{k}) := \sum_{j=1}^{N} w_j \, f(\boldsymbol{x}_j) \, \mathrm{e}^{-2\pi \mathrm{i} \boldsymbol{k} \boldsymbol{x}_j}, \quad \boldsymbol{k} \in \mathcal{I}_{\boldsymbol{M}}.$$

This is to say, equispaced samples of the Fourier transform of a bandlimited function may be approximated in the same form $(\hat{f}(\boldsymbol{k}))_{\boldsymbol{k} \in \mathcal{I}_{\boldsymbol{M}}} = \hat{\boldsymbol{f}} \approx \boldsymbol{h}^{\mathrm{w}} := \boldsymbol{A}^* \boldsymbol{W} \boldsymbol{f}$ as in (3.6), where we used the notation $\boldsymbol{h}^{\mathrm{w}} := (\tilde{h}(\boldsymbol{k}))_{\boldsymbol{k} \in \mathcal{I}_{\boldsymbol{M}}}$ in this setting. Moreover, we extend this approximation onto the whole interval $\left[-\frac{M}{2}, \frac{M}{2}\right)^d$, i.e., we consider

$$\hat{f}(\boldsymbol{v}) \approx \tilde{h}(\boldsymbol{v}) := \sum_{j=1}^{N} w_j \, f(\boldsymbol{x}_j) \, \mathrm{e}^{-2\pi \mathrm{i} \boldsymbol{v} \boldsymbol{x}_j}, \quad \boldsymbol{v} \in \left[-\tfrac{M}{2}, \tfrac{M}{2}\right)^d. \tag{3.21}$$

So all in all, we have seen that it is reasonable to study the inversion problem (3.4) and the associated density compensation via (3.7) for bandlimited functions as well. Analogous to Section 3.1 we now aim to find a numerical method for computing suitable weights $w_j \in \mathbb{C}$, $j = 1, \ldots, N$, such that the reconstruction formula (3.21) is preferably exact. To this end, we have a closer look at (3.21) being exact and start with analogous considerations as in Theorem 3.2.

Theorem 3.8. *Let a bandwidth $\boldsymbol{M} \in \mathbb{N}^d$, nonequispaced nodes $\boldsymbol{x}_j \in \left[-\frac{1}{2}, \frac{1}{2}\right)^d$ as well as quadrature weights $w_j \in \mathbb{C}$, $j = 1, \ldots, N$, be given. Then an exact reconstruction formula (3.21) for bandlimited functions $f \in L_1(\mathbb{R}^d) \cap C_0(\mathbb{R}^d)$ with bandwidth $\boldsymbol{M}$, i.e.,*

$$\hat{f}(\boldsymbol{v}) = \tilde{h}(\boldsymbol{v}) = \sum_{j=1}^{N} w_j f(\boldsymbol{x}_j) \, \mathrm{e}^{-2\pi \mathrm{i} \boldsymbol{v} \boldsymbol{x}_j}, \quad \boldsymbol{v} \in \left[-\tfrac{M}{2}, \tfrac{M}{2}\right)^d, \tag{3.22}$$

implies that the quadrature rule

$$\int_{\mathbb{R}^d} f(\boldsymbol{x}) \, \mathrm{d}\boldsymbol{x} = \sum_{j=1}^{N} w_j f(\boldsymbol{x}_j)$$

is exact for all bandlimited functions $f \in L_1(\mathbb{R}^d) \cap C_0(\mathbb{R}^d)$ with bandwidth $\boldsymbol{M}$.

Proof. By (3.17) the assumption (3.22) can be written as

$$\int_{\mathbb{R}^d} f(\boldsymbol{x}) \, \mathrm{e}^{-2\pi \mathrm{i} \boldsymbol{v} \boldsymbol{x}} \, \mathrm{d}\boldsymbol{x} = \hat{f}(\boldsymbol{v}) = \sum_{j=1}^{N} w_j f(\boldsymbol{x}_j) \, \mathrm{e}^{-2\pi \mathrm{i} \boldsymbol{v} \boldsymbol{x}_j}, \quad \boldsymbol{v} \in \left[-\tfrac{M}{2}, \tfrac{M}{2}\right)^d. \tag{3.23}$$

Especially, for $\boldsymbol{v} = \boldsymbol{0}$ evaluation of (3.23) yields the assertion

$$\int_{\mathbb{R}^d} f(\boldsymbol{x}) \, \mathrm{d}\boldsymbol{x} = \hat{f}(\boldsymbol{0}) = \sum_{j=1}^{N} w_j f(\boldsymbol{x}_j).$$

■

However, in contrast to Theorem 3.2, this Theorem 3.8 does not yield an explicit condition for computing suitable weights $w_j \in \mathbb{C}$, $j = 1, \ldots, N$. To derive a numerical procedure anyway, we generalize the notion of an exact reconstruction $\tilde{h}$ of f and have a look at the theory of tempered distributions. To this end, let $\mathscr{S}(\mathbb{R}^d)$ be the Schwartz space of rapidly decaying functions, cf. [53, Sec. 4.2.1]. The tempered Dirac distribution δ shall be defined by $\langle \delta, \varphi \rangle := \int_{\mathbb{R}^d} \varphi(\boldsymbol{v}) \, \delta(\boldsymbol{v}) \, \mathrm{d}\boldsymbol{v} = \varphi(\boldsymbol{0})$ for all $\varphi \in \mathscr{S}(\mathbb{R}^d)$, cf. [53, Ex. 4.36]. For a slowly increasing function $f : \mathbb{R}^d \to \mathbb{C}$ satisfying $|f(\boldsymbol{x})| \leq c(1 + \|\boldsymbol{x}\|_2)^n$ almost everywhere with $c > 0$ and $n \in \mathbb{N}_0$, the induced distribution T_f shall be defined by $\langle T_f, \varphi \rangle := \int_{\mathbb{R}^d} \varphi(\boldsymbol{x}) \, f(\boldsymbol{x}) \, \mathrm{d}\boldsymbol{x}$ for all $\varphi \in \mathscr{S}(\mathbb{R}^d)$. For a detailed introduction to the topic we refer to [53, Sec. 4.2.1 and Sec. 4.3].

Then the following property can be shown.

Theorem 3.9. *Let nonequispaced nodes $\boldsymbol{x}_j \in \left[-\frac{1}{2}, \frac{1}{2}\right)^d$, $j = 1, \ldots, N$, and quadrature weights $w_j \in \mathbb{C}$ be given. Further let T_f be the distribution induced by some bandlimited function $f \in L_1(\mathbb{R}^d) \cap C_0(\mathbb{R}^d)$ with bandwidth $\boldsymbol{M}$. Then*

$$\langle \delta, \varphi \rangle = \langle T_\xi, \varphi \rangle, \quad \varphi \in \mathscr{S}(\mathbb{R}^d), \tag{3.24}$$

with

$$\xi(\boldsymbol{v}) := \sum_{j=1}^{N} w_j \, \mathrm{e}^{2\pi \mathrm{i} \boldsymbol{v} \boldsymbol{x}_j}, \quad \boldsymbol{v} \in \mathbb{R}^d, \tag{3.25}$$

implies

$$\langle \hat{T}_f, \varphi \rangle = \langle T_{\tilde{h}}, \varphi \rangle, \quad \varphi \in \mathscr{S}(\mathbb{R}^d), \tag{3.26}$$

with the function $\tilde{h}$ defined in (3.21).

Proof. Using the definition of the function $\tilde{h}$ in (3.21) as well as the fact that the inversion formula (3.18) holds for all $\boldsymbol{x} \in \mathbb{R}^d$, we have

$$\langle T_{\tilde{h}}, \varphi \rangle = \int_{\mathbb{R}^d} \varphi(\boldsymbol{v}) \sum_{j=1}^{N} w_j f(\boldsymbol{x}_j) \, \mathrm{e}^{-2\pi \mathrm{i} \boldsymbol{v} \boldsymbol{x}_j} \, \mathrm{d}\boldsymbol{v}$$

$$= \int_{\mathbb{R}^d} \varphi(\boldsymbol{v}) \sum_{j=1}^{N} w_j \left(\int_{\mathbb{R}^d} \hat{f}(\boldsymbol{u}) \, \mathrm{e}^{2\pi \mathrm{i} \boldsymbol{u} \boldsymbol{x}_j} \, \mathrm{d}\boldsymbol{u} \right) \mathrm{e}^{-2\pi \mathrm{i} \boldsymbol{v} \boldsymbol{x}_j} \, \mathrm{d}\boldsymbol{v}$$

$$= -\int_{\mathbb{R}^d} \hat{f}(\boldsymbol{u}) \int_{\mathbb{R}^d} \varphi(\boldsymbol{u} - \boldsymbol{v}) \sum_{j=1}^{N} w_j \, \mathrm{e}^{2\pi \mathrm{i} \boldsymbol{v} \boldsymbol{x}_j} \, \mathrm{d}\boldsymbol{v} \, \mathrm{d}\boldsymbol{u}.$$

Hence, by (3.24) this implies

$$\langle T_{\tilde{h}}, \varphi \rangle = -\int_{\mathbb{R}^d} \hat{f}(\boldsymbol{u}) \int_{\mathbb{R}^d} \varphi(\boldsymbol{u} - \boldsymbol{v}) \, \delta(\boldsymbol{v}) \, \mathrm{d}\boldsymbol{v} \, \mathrm{d}\boldsymbol{u}$$

$$= \int_{\mathbb{R}^d} \hat{f}(\boldsymbol{u}) \int_{\mathbb{R}^d} \varphi(\boldsymbol{v}) \, \delta(\boldsymbol{u} - \boldsymbol{v}) \, \mathrm{d}\boldsymbol{v} \, \mathrm{d}\boldsymbol{u} = \int_{\mathbb{R}^d} \hat{f}(\boldsymbol{u}) \, \varphi(\boldsymbol{u}) \, \mathrm{d}\boldsymbol{u}.$$

∎

Considering the property (3.26), we remark that this indeed states an exact reconstruction $\hat{f} = \tilde{h}$ in the sense of tempered distributions, as distinct from (3.22). Since it is known by Corollary 3.4 that the condition (3.14) yields an exact reconstruction for trigonometric polynomials, we aim to use this result to compute suitable weights $w_j \in \mathbb{C}$, $j = 1, \ldots, N$, for bandlimited functions as well. To this end, suppose we have (3.14), i.e.,

$$\sum_{j=1}^{N} w_j \, e^{2\pi i \boldsymbol{k} \boldsymbol{x}_j} = \delta_{\boldsymbol{0}, \boldsymbol{k}}, \quad \boldsymbol{k} \in \mathcal{I}_{\boldsymbol{2M}},$$

then this implies

$$\varphi(\boldsymbol{0}) = \sum_{\boldsymbol{k} \in \mathcal{I}_{\boldsymbol{2M}}} \varphi(\boldsymbol{k}) \sum_{j=1}^{N} w_j \, e^{2\pi i \boldsymbol{k} \boldsymbol{x}_j}, \quad \varphi \in \mathscr{S}(\mathbb{R}^d). \tag{3.27}$$

Having a look at Theorem 3.9, an exact reconstruction (3.26) is implied by (3.24), i.e.,

$$\varphi(\boldsymbol{0}) = \langle \delta, \varphi \rangle = \langle T_\xi, \varphi \rangle = \int_{\mathbb{R}^d} \varphi(\boldsymbol{v}) \sum_{j=1}^{N} w_j \, e^{2\pi i \boldsymbol{v} \boldsymbol{x}_j} \, d\boldsymbol{v}, \quad \varphi \in \mathscr{S}(\mathbb{R}^d).$$

Thus, the property (3.27) that is fulfilled by (3.14) could be interpreted as an equispaced quadrature of (3.24) at integer frequencies $\boldsymbol{k} \in \mathcal{I}_{\boldsymbol{2M}}$.

Remark 3.10. *We remark that for deriving the quadrature rule (3.27) from (3.24), we implicitly truncate the integral bounds in (3.24) as*

$$\langle T_\xi, \varphi \rangle = \int_{\mathbb{R}^d} \varphi(\boldsymbol{v}) \sum_{j=1}^{N} w_j \, e^{2\pi i \boldsymbol{v} \boldsymbol{x}_j} \, d\boldsymbol{v} \approx \int_{[-M,M)^d} \varphi(\boldsymbol{v}) \sum_{j=1}^{N} w_j \, e^{2\pi i \boldsymbol{v} \boldsymbol{x}_j} \, d\boldsymbol{v}$$

$$= \int_{\mathbb{R}^d} \varphi(\boldsymbol{v}) \sum_{j=1}^{N} w_j \, e^{2\pi i \boldsymbol{v} \boldsymbol{x}_j} \, \chi_{[-M,M)^d}(\boldsymbol{v}) \, d\boldsymbol{v},$$

i.e., instead of (3.25) we rather deal with a distribution induced by

$$\tilde{\xi}(\boldsymbol{v}) := \xi(\boldsymbol{v}) \, \chi_{[-M,M)^d}(\boldsymbol{v}), \quad \boldsymbol{v} \in \mathbb{R}^d. \tag{3.28}$$

However, an analogous implication as in Theorem 3.9 cannot be shown when using the function (3.28) instead of (3.25). □

As seen in Remark 3.10, our numerical method for computing suitable weights $w_j \in \mathbb{C}$, $j = 1, \ldots, N$, can also be derived by means of a quadrature formula applied to the property $\langle \delta, \hat{\varphi} \rangle = \langle T_{\tilde{\xi}}, \hat{\varphi} \rangle$ with $\tilde{\xi}$ defined in (3.28). Having a closer look at this property, the following equivalent characterization can be shown.

Theorem 3.11. *Let a bandwidth $\boldsymbol{M} \in \mathbb{N}^d$, nonequispaced nodes $\boldsymbol{x}_j \in \left[-\frac{1}{2}, \frac{1}{2}\right)^d$ and quadrature weights $w_j \in \mathbb{C}$, $j = 1, \dots, N$, be given. Then the following two statements are equivalent.*

(i) *For all $\varphi \in \mathscr{S}(\mathbb{R}^d)$ it holds $\langle \delta, \hat{\varphi} \rangle = \langle T_{\tilde{\xi}}, \hat{\varphi} \rangle$ with $\tilde{\xi}$ defined in (3.28).*

(ii) *We have $\langle 1, \varphi \rangle = \langle T_\psi, \varphi \rangle$ for all $\varphi \in \mathscr{S}(\mathbb{R}^d)$, where*

$$\psi(\boldsymbol{x}) := \sum_{j=1}^{N} w_j \cdot |\mathcal{I}_{\boldsymbol{2M}}| \operatorname{sinc}\left(2M\pi\left(\boldsymbol{x}_j - \boldsymbol{x}\right)\right), \quad \boldsymbol{x} \in \mathbb{R}^d,$$

with the d-variate sinc function $\operatorname{sinc}(\boldsymbol{x}) := \prod_{t=1}^{d} \operatorname{sinc}(x_t)$ and

$$\operatorname{sinc}(x) := \begin{cases} \frac{\sin x}{x} & x \in \mathbb{R} \setminus \{0\}, \\ 1 & x = 0. \end{cases}$$

Proof. By the definition $\langle \hat{T}, \varphi \rangle = \langle T, \hat{\varphi} \rangle$ of the Fourier transform of a tempered distribution $T \in \mathscr{S}'(\mathbb{R}^d)$ we have $\langle 1, \varphi \rangle = \langle \delta, \hat{\varphi} \rangle$, cf. [53, Ex. 4.46]. Moreover, the distribution induced by (3.28) can be rewritten using the Fourier transform (3.17) as

$$\langle T_{\tilde{\xi}}, \hat{\varphi} \rangle = \int_{\mathbb{R}^d} \hat{\varphi}(\boldsymbol{v})\, \tilde{\xi}(\boldsymbol{v})\, \mathrm{d}\boldsymbol{v} = \sum_{j=1}^{N} w_j \int_{[-M,M)^d} \hat{\varphi}(\boldsymbol{v})\, \mathrm{e}^{2\pi \mathrm{i} \boldsymbol{v} \boldsymbol{x}_j}\, \mathrm{d}\boldsymbol{v}$$

$$= \sum_{j=1}^{N} w_j \int_{[-M,M)^d} \left(\int_{\mathbb{R}^d} \varphi(\boldsymbol{x})\, \mathrm{e}^{-2\pi \mathrm{i} \boldsymbol{v} \boldsymbol{x}}\, \mathrm{d}\boldsymbol{x} \right) \mathrm{e}^{2\pi \mathrm{i} \boldsymbol{v} \boldsymbol{x}_j}\, \mathrm{d}\boldsymbol{v}$$

$$= \sum_{j=1}^{N} w_j \int_{\mathbb{R}^d} \varphi(\boldsymbol{x}) \int_{[-M,M)^d} \mathrm{e}^{2\pi \mathrm{i} \boldsymbol{v} (\boldsymbol{x}_j - \boldsymbol{x})}\, \mathrm{d}\boldsymbol{v}\, \mathrm{d}\boldsymbol{x}. \tag{3.29}$$

The inner integral can be determined by

$$\int_{[-M,M)^d} \mathrm{e}^{2\pi \mathrm{i} \boldsymbol{v}(\boldsymbol{x}_j - \boldsymbol{x})}\, \mathrm{d}\boldsymbol{v} = |\mathcal{I}_{\boldsymbol{2M}}| \operatorname{sinc}\left(2M\pi\left(\boldsymbol{x}_j - \boldsymbol{x}\right)\right), \tag{3.30}$$

such that (3.29) shows the equality $\langle T_{\tilde{\xi}}, \hat{\varphi} \rangle = \langle T_\psi, \varphi \rangle$. Hence, the assertions (i) and (ii) are equivalent. ∎

Thus, since the statements (i) and (ii) of Theorem 3.11 are equivalent, one could also consider the property (ii) for deriving a numerical method to compute suitable weights $w_j \in \mathbb{C}$, $j = 1, \dots, N$. To this end, we have a closer look at

$$\int_{\mathbb{R}^d} \varphi(\boldsymbol{x})\, \mathrm{d}\boldsymbol{x} = \langle 1, \varphi \rangle = \langle T_\psi, \varphi \rangle = \int_{\mathbb{R}^d} \varphi(\boldsymbol{x}) \sum_{j=1}^{N} w_j \cdot |\mathcal{I}_{\boldsymbol{2M}}| \operatorname{sinc}\left(2M\pi\left(\boldsymbol{x}_j - \boldsymbol{x}\right)\right)\, \mathrm{d}\boldsymbol{x}$$

$$\tag{3.31}$$

for all $\varphi \in \mathscr{S}(\mathbb{R}^d)$. Due to the integrals on both sides of (3.31), we need to discretize twice and therefore use the same quadrature rule on both sides of (3.31). For better comparability to (3.27) we utilize the same number $|\mathcal{I}_{\mathbf{2M}}|$ of equispaced quadrature points $\boldsymbol{y}_{\boldsymbol{\ell}} := (\mathbf{2M})^{-1} \odot \boldsymbol{\ell}$, $\boldsymbol{\ell} \in \mathcal{I}_{\mathbf{2M}}$, as in (3.27), i.e., we consider

$$\sum_{\boldsymbol{\ell} \in \mathcal{I}_{\mathbf{2M}}} \varphi(\boldsymbol{y}_{\boldsymbol{\ell}}) = \sum_{\boldsymbol{\ell} \in \mathcal{I}_{\mathbf{2M}}} \varphi(\boldsymbol{y}_{\boldsymbol{\ell}}) \sum_{j=1}^{N} w_j \cdot |\mathcal{I}_{\mathbf{2M}}| \operatorname{sinc}\left(2M\pi\left(\boldsymbol{x}_j - \boldsymbol{y}_{\boldsymbol{\ell}}\right)\right).$$

In order that this applies for all $\varphi \in \mathscr{S}(\mathbb{R}^d)$, we need to satisfy

$$1 = \sum_{j=1}^{N} w_j \cdot |\mathcal{I}_{\mathbf{2M}}| \operatorname{sinc}\left(2M\pi\left(\boldsymbol{x}_j - \boldsymbol{y}_{\boldsymbol{\ell}}\right)\right), \quad \boldsymbol{\ell} \in \mathcal{I}_{\mathbf{2M}}, \tag{3.32}$$

i.e., one could also compute weights $w_j \in \mathbb{C}$, $j = 1, \dots, N$, as a least squares solution to the linear system of equations (3.32). Hence, it merely remains the comparison of the two computation schemes.

Remark 3.12. *Since we derived discretizations out of both statements of Theorem 3.11, we examine if also the two linear systems (3.14) and (3.32) are related. Considering the statements in Theorem 3.11 we notice that in some sense they are the Fourier transformed versions of each other. To this end, we need to Fourier transform one of the linear systems for better comparability. More precisely, we apply an iFFT of length $|\mathcal{I}_{\mathbf{2M}}|$, cf. (2.4), to both sides of equation (3.32). Since the left side transforms to*

$$\sum_{\boldsymbol{\ell} \in \mathcal{I}_{\mathbf{2M}}} 1 \cdot \mathrm{e}^{2\pi \mathrm{i} \boldsymbol{k} \boldsymbol{y}_{\boldsymbol{\ell}}} = |\mathcal{I}_{\mathbf{2M}}| \cdot \delta_{\mathbf{0}, \boldsymbol{k}}, \quad \boldsymbol{k} \in \mathcal{I}_{\mathbf{2M}},$$

we obtain the transformed system

$$\delta_{\mathbf{0}, \boldsymbol{k}} = \sum_{j=1}^{N} w_j \sum_{\boldsymbol{\ell} \in \mathcal{I}_{\mathbf{2M}}} \operatorname{sinc}\left(2M\pi\left(\boldsymbol{x}_j - \boldsymbol{y}_{\boldsymbol{\ell}}\right)\right) \mathrm{e}^{2\pi \mathrm{i} \boldsymbol{k} \boldsymbol{y}_{\boldsymbol{\ell}}}, \quad \boldsymbol{k} \in \mathcal{I}_{\mathbf{2M}}. \tag{3.33}$$

Comparing this linear system of equations to (3.14), we recognize an identical structure. Hence, we have a closer look at the connection between the expressions $\mathrm{e}^{2\pi \mathrm{i} \boldsymbol{k} \boldsymbol{x}_j}$ and $\sum_{\boldsymbol{\ell} \in \mathcal{I}_{\mathbf{2M}}} \operatorname{sinc}\left(2M\pi\left(\boldsymbol{x}_j - \boldsymbol{y}_{\boldsymbol{\ell}}\right)\right) \mathrm{e}^{2\pi \mathrm{i} \boldsymbol{k} \boldsymbol{y}_{\boldsymbol{\ell}}}$.

For this purpose, we consider the function $f(\boldsymbol{t}) = \mathrm{e}^{2\pi \mathrm{i} \boldsymbol{t} \boldsymbol{x}}$, $\boldsymbol{t} \in [-M, M)^d$, for fixed $\boldsymbol{x} \in \mathbb{C}^d$. By means of $\tilde{f}(\boldsymbol{t}) := \sum_{\boldsymbol{k} \in \mathbb{Z}^d} f(\boldsymbol{t} + 2M\boldsymbol{k})$ we extend it into a $(2M)$-periodic function. This periodized version then possesses the Fourier coefficients

$$c_{\boldsymbol{\ell}}(\tilde{f}) = \frac{1}{|\mathcal{I}_{\mathbf{2M}}|} \int_{[-M,M)^d} f(\boldsymbol{t}) \, \mathrm{e}^{-2\pi \mathrm{i} \boldsymbol{t} \boldsymbol{y}_{\boldsymbol{\ell}}} \, \mathrm{d}\boldsymbol{t}$$

$$= \frac{1}{|\mathcal{I}_{\mathbf{2M}}|} \int_{[-M,M)^d} \mathrm{e}^{2\pi \mathrm{i} \boldsymbol{t} (\boldsymbol{x} - \boldsymbol{y}_{\boldsymbol{\ell}})} \, \mathrm{d}\boldsymbol{t} = \operatorname{sinc}\left(2M\pi\left(\boldsymbol{x} - \boldsymbol{y}_{\boldsymbol{\ell}}\right)\right), \quad \boldsymbol{\ell} \in \mathbb{Z}^d,$$

cf. (2.2), i.e., the Fourier expansion of $\tilde{f}(\boldsymbol{t}) = \mathrm{e}^{2\pi\mathrm{i}\boldsymbol{t}\boldsymbol{x}}$, $\boldsymbol{t} \in [-M, M)^d$, for fixed $\boldsymbol{x}$ is given by

$$\mathrm{e}^{2\pi\mathrm{i}\boldsymbol{t}\boldsymbol{x}} = \sum_{\boldsymbol{\ell}\in\mathbb{Z}^d} \mathrm{e}^{2\pi\mathrm{i}\boldsymbol{t}\boldsymbol{y}_{\boldsymbol{\ell}}} \operatorname{sinc}\left(2M\pi\left(\boldsymbol{x} - \boldsymbol{y}_{\boldsymbol{\ell}}\right)\right), \quad \boldsymbol{x} \in \mathbb{C}^d,$$

cf. (2.1). Since $\tilde{f}(\boldsymbol{t}) = \mathrm{e}^{2\pi\mathrm{i}\boldsymbol{t}\boldsymbol{x}}$ is continuous and piecewise differentiable, this Fourier series converges absolutely and uniformly, cf. [49]. Thereby, we may consider the point evaluations at $\boldsymbol{x} = \boldsymbol{x}_j$, $j = 1, \dots, N$, and $\boldsymbol{t} = \boldsymbol{k} \in \mathcal{I}_{2\boldsymbol{M}}$, such that we obtain the representation

$$\mathrm{e}^{2\pi\mathrm{i}\boldsymbol{k}\boldsymbol{x}_j} = \sum_{\boldsymbol{\ell}\in\mathbb{Z}^d} \mathrm{e}^{2\pi\mathrm{i}\boldsymbol{k}\boldsymbol{y}_{\boldsymbol{\ell}}} \operatorname{sinc}\left(2M\pi\left(\boldsymbol{x}_j - \boldsymbol{y}_{\boldsymbol{\ell}}\right)\right). \tag{3.34}$$

Thus, we recognize that (3.33) is a truncated version of (3.34). In other words, this implies that the linear system (3.14) is equivalent to a discretization of (3.31) incorporating infinitely many points $\boldsymbol{y}_{\boldsymbol{\ell}} \in \mathbb{R}^d$ in (3.32). $\qquad\square$

Remark 3.13. For bandlimited functions several fast evaluation methods including the sinc function are known. The classical sampling theorem of Shannon-Whittaker-Kotelnikov, see [72, 64, 42], states that any bandlimited function $f \in L_1(\mathbb{R}^d) \cap C_0(\mathbb{R}^d)$ with maximum bandwidth $\boldsymbol{M}$ can be recovered from its uniform samples $f(\boldsymbol{L}^{-1} \odot \boldsymbol{\ell})$, $\boldsymbol{\ell} \in \mathbb{Z}^d$, with $L \geq M$, $\boldsymbol{L} := L \cdot \boldsymbol{1}_d$, and it holds

$$f(\boldsymbol{x}) = \sum_{\boldsymbol{\ell}\in\mathbb{Z}^d} f(\boldsymbol{L}^{-1} \odot \boldsymbol{\ell}) \operatorname{sinc}\left(L\pi\left(\boldsymbol{x} - \boldsymbol{L}^{-1} \odot \boldsymbol{\ell}\right)\right), \quad \boldsymbol{x} \in \mathbb{R}^d. \tag{3.35}$$

Since the practical use of this sampling theorem is limited due to the infinite number of samples, which is impossible in practice, and the very slow decay of the sinc function, various authors such as [57, 68, 50, 47, 40] considered the regularized Shannon sampling formula with localized sampling

$$f(\boldsymbol{x}) \approx \sum_{\boldsymbol{\ell}\in\mathbb{Z}^d} f(\boldsymbol{L}^{-1} \odot \boldsymbol{\ell}) \operatorname{sinc}\left(L\pi\left(\boldsymbol{x} - \boldsymbol{L}^{-1} \odot \boldsymbol{\ell}\right)\right) \varphi_m\left(\boldsymbol{x} - \boldsymbol{L}^{-1} \odot \boldsymbol{\ell}\right), \quad \boldsymbol{x} \in \mathbb{R}^d, \tag{3.36}$$

instead. Here $\varphi_m \colon \mathbb{R}^d \to [0, 1]$ is a compactly supported window function with truncation parameter $m \in \mathbb{N} \setminus \{1\}$, such that for φ_m with small support the direct evaluation of (3.36) is efficient, see [40] for the relation to the NFFT window functions.

On the other hand, in the one-dimensional setting a fast sinc transform was introduced in [39], which is based on the Clenshaw-Curtis quadrature

$$\operatorname{sinc}\left(L\pi\left(x - \tfrac{\ell}{L}\right)\right) \approx \sum_{k=0}^{n} w_k \, \mathrm{e}^{-\pi\mathrm{i}L(x-\frac{\ell}{L})z_k}$$

using Chebyshev points $z_k = \cos(\frac{k\pi}{n}) \in [-1, 1]$, $k = 0, \dots, n$, and corresponding Clenshaw-Curtis weights $w_k > 0$. Thereby, sums of the form

$$h(x) = \sum_{\ell\in\mathcal{I}_T} f\left(\tfrac{\ell}{L}\right) \operatorname{sinc}\left(L\pi\left(x - \tfrac{\ell}{L}\right)\right) \approx \sum_{k=0}^{n} w_k \left(\sum_{\ell\in\mathcal{I}_T} f\left(\tfrac{\ell}{L}\right) \mathrm{e}^{\pi\mathrm{i}\ell z_k}\right) \mathrm{e}^{-\pi\mathrm{i}Lxz_k}$$

with uniform truncation parameter $T \in 2\mathbb{N}$, can efficiently be approximated by means of fast Fourier transforms. More precisely, for the term in brackets one may utilize an NFFT, cf. (2.5). Then the resulting outer sum can be computed using an NNFFT, also referred to as NFFT of type, see [17, 20, 46] or [53, pp. 394–397]. □

3.3 General error bound

In this section we summarize our previous findings by presenting a general error bound on density compensation factors computed by means of (3.14), that applies to trigonometric polynomials, 1-periodic functions $f \in L_2(\mathbb{T}^d) \cap C(\mathbb{T}^d)$ and bandlimited functions $f \in L_1(\mathbb{R}^d) \cap C_0(\mathbb{R}^d)$ as well.

Theorem 3.14. *Let $p, q \in \{1, 2, \infty\}$ with $\frac{1}{p} + \frac{1}{q} = 1$. For given $d, N \in \mathbb{N}$, $\boldsymbol{M} \in (2\mathbb{N})^d$ and nonequispaced nodes $\boldsymbol{x}_j \in \left[-\frac{1}{2}, \frac{1}{2}\right)^d$, $j = 1, \ldots, N$, let $\boldsymbol{A} \in \mathbb{C}^{N \times |\mathcal{I}_M|}$ be the nonequispaced Fourier matrix in (2.7). Further assume we can compute density compensation factors $\boldsymbol{W} = \operatorname{diag}(w_j)_{j=1}^N \in \mathbb{C}^{N \times N}$ by means of Algorithm 3.6, such that*

$$\sum_{j=1}^{N} w_j \, \mathrm{e}^{2\pi \mathrm{i} \boldsymbol{k} \boldsymbol{x}_j} = \delta_{\boldsymbol{0}, \boldsymbol{k}} + \varepsilon_{\boldsymbol{k}}, \quad \boldsymbol{k} \in \mathcal{I}_{\boldsymbol{2M}}, \tag{3.37}$$

with small $\varepsilon_{\boldsymbol{k}} \in \mathbb{R}$ for all $\boldsymbol{k} \in \mathcal{I}_{\boldsymbol{2M}}$.

Then there exists an $\varepsilon \geq 0$ such that the corresponding density compensation procedure with $\boldsymbol{W} = \operatorname{diag}(w_j)_{j=1}^N$ satisfies the following error bounds.

(i) *For any trigonometric polynomial $f \in L_2(\mathbb{T}^d)$ of degree $\boldsymbol{M}$ given in (2.8) we have*

$$\left\| \hat{\boldsymbol{f}} - \boldsymbol{A}^* \boldsymbol{W} \boldsymbol{f} \right\|_p \leq |\mathcal{I}_{\boldsymbol{M}}| \, \varepsilon \cdot \left\| \hat{\boldsymbol{f}} \right\|_p, \tag{3.38}$$

where $\hat{\boldsymbol{f}} := (\hat{f}_{\boldsymbol{k}})_{\boldsymbol{k} \in \mathcal{I}_M}$ are the coefficients given in (2.8).

(ii) *For any 1-periodic function $f \in L_2(\mathbb{T}^d) \cap C(\mathbb{T}^d)$ we have*

$$\left\| \hat{\boldsymbol{f}} - \boldsymbol{A}^* \boldsymbol{W} \boldsymbol{f} \right\|_p \leq |\mathcal{I}_{\boldsymbol{M}}| \, \varepsilon \cdot \left\| \hat{\boldsymbol{f}} \right\|_p + (N \, |\mathcal{I}_{\boldsymbol{M}}|)^{1/p} \, \|\boldsymbol{w}\|_q \cdot \|f - p_{\boldsymbol{M}}\|_{C(\mathbb{T}^d)}, \tag{3.39}$$

where $\hat{\boldsymbol{f}} := (c_{\boldsymbol{k}}(f))_{\boldsymbol{k} \in \mathcal{I}_M}$ are the first $|\mathcal{I}_{\boldsymbol{M}}|$ coefficients given in (2.1) and $p_{\boldsymbol{M}}$ is the best approximating trigonometric polynomial of degree $\boldsymbol{M}$ of f.

(iii) *For any bandlimited function $f \in L_1(\mathbb{R}^d) \cap C_0(\mathbb{R}^d)$ with bandwidth $\boldsymbol{M}$ we have*

$$\left\| \hat{\boldsymbol{f}} - \boldsymbol{A}^* \boldsymbol{W} \boldsymbol{f} \right\|_p \leq |\mathcal{I}_{\boldsymbol{M}}| \, \varepsilon \cdot \left\| \hat{\boldsymbol{f}} \right\|_p + (N \, |\mathcal{I}_{\boldsymbol{M}}|)^{1/p} \, \|\boldsymbol{w}\|_q \cdot \|Q\|_{C(\mathbb{T}^d)}, \tag{3.40}$$

where $\hat{\boldsymbol{f}} := (\hat{f}(\boldsymbol{k}))_{\boldsymbol{k} \in \mathcal{I}_M}$ are the integer evaluations of (3.17) and Q in (3.46) is the pointwise quadrature error of the equispaced quadrature rule (3.19).

Proof. We start with some general considerations that are independent of the function f. By (3.37) we can find some $\varepsilon \geq 0$ with $|\varepsilon_{\boldsymbol{k}}| \leq \varepsilon$, $\boldsymbol{k} \in \mathcal{I}_{\boldsymbol{2M}}$, satisfying

$$\left| \sum_{j=1}^{N} w_j \, \mathrm{e}^{2\pi \mathrm{i} \boldsymbol{k} \boldsymbol{x}_j} - \delta_{\boldsymbol{0},\boldsymbol{k}} \right| \leq \varepsilon, \quad \boldsymbol{k} \in \mathcal{I}_{\boldsymbol{2M}}.$$

Then for all $\boldsymbol{k}, \boldsymbol{\ell} \in \mathcal{I}_{\boldsymbol{M}}$ with $(\boldsymbol{\ell} - \boldsymbol{k}) \in \mathcal{I}_{\boldsymbol{2M}}$ this yields

$$\left| [\boldsymbol{E}_{\mathrm{r}}]_{\boldsymbol{k},\boldsymbol{\ell}} \right| = \left| \sum_{j=1}^{N} w_j \, \mathrm{e}^{2\pi \mathrm{i} (\boldsymbol{\ell}-\boldsymbol{k}) \boldsymbol{x}_j} - \delta_{\boldsymbol{k},\boldsymbol{\ell}} \right| \leq \varepsilon,$$

where $\boldsymbol{E}_{\mathrm{r}} \coloneqq \boldsymbol{A}^* \boldsymbol{W} \boldsymbol{A} - \boldsymbol{I}_{|\mathcal{I}_{\boldsymbol{M}}|}$. Hence, we have

$$\left\| \boldsymbol{A}^* \boldsymbol{W} \boldsymbol{A} - \boldsymbol{I}_{|\mathcal{I}_{\boldsymbol{M}}|} \right\|_1 = \max_{\boldsymbol{\ell} \in \mathcal{I}_{\boldsymbol{M}}} \sum_{\boldsymbol{k} \in \mathcal{I}_{\boldsymbol{M}}} \left| [\boldsymbol{E}_{\mathrm{r}}]_{\boldsymbol{k},\boldsymbol{\ell}} \right| \leq \max_{\boldsymbol{\ell} \in \mathcal{I}_{\boldsymbol{M}}} \sum_{\boldsymbol{k} \in \mathcal{I}_{\boldsymbol{M}}} \varepsilon = |\mathcal{I}_{\boldsymbol{M}}| \, \varepsilon, \qquad (3.41)$$

$$\left\| \boldsymbol{A}^* \boldsymbol{W} \boldsymbol{A} - \boldsymbol{I}_{|\mathcal{I}_{\boldsymbol{M}}|} \right\|_\infty = \max_{\boldsymbol{k} \in \mathcal{I}_{\boldsymbol{M}}} \sum_{\boldsymbol{\ell} \in \mathcal{I}_{\boldsymbol{M}}} \left| [\boldsymbol{E}_{\mathrm{r}}]_{\boldsymbol{k},\boldsymbol{\ell}} \right| \leq \max_{\boldsymbol{k} \in \mathcal{I}_{\boldsymbol{M}}} \sum_{\boldsymbol{\ell} \in \mathcal{I}_{\boldsymbol{M}}} \varepsilon = |\mathcal{I}_{\boldsymbol{M}}| \, \varepsilon, \qquad (3.42)$$

and

$$\left\| \boldsymbol{A}^* \boldsymbol{W} \boldsymbol{A} - \boldsymbol{I}_{|\mathcal{I}_{\boldsymbol{M}}|} \right\|_{\mathrm{F}} = \sqrt{\sum_{\boldsymbol{k} \in \mathcal{I}_{\boldsymbol{M}}} \sum_{\boldsymbol{\ell} \in \mathcal{I}_{\boldsymbol{M}}} \left| [\boldsymbol{E}_{\mathrm{r}}]_{\boldsymbol{k},\boldsymbol{\ell}} \right|^2} \leq \sqrt{\sum_{\boldsymbol{k} \in \mathcal{I}_{\boldsymbol{M}}} \sum_{\boldsymbol{\ell} \in \mathcal{I}_{\boldsymbol{M}}} \varepsilon^2} = |\mathcal{I}_{\boldsymbol{M}}| \, \varepsilon. \qquad (3.43)$$

Considering the approximation error of (3.6) it holds

$$\begin{aligned}
\left\| \hat{\boldsymbol{f}} - \boldsymbol{A}^* \boldsymbol{W} \boldsymbol{f} \right\|_p &\leq \left\| \hat{\boldsymbol{f}} - \boldsymbol{A}^* \boldsymbol{W} \boldsymbol{A} \hat{\boldsymbol{f}} \right\|_p + \left\| \boldsymbol{A}^* \boldsymbol{W} \boldsymbol{A} \hat{\boldsymbol{f}} - \boldsymbol{A}^* \boldsymbol{W} \boldsymbol{f} \right\|_p \\
&= \left\| \left(\boldsymbol{A}^* \boldsymbol{W} \boldsymbol{A} - \boldsymbol{I}_{|\mathcal{I}_{\boldsymbol{M}}|} \right) \hat{\boldsymbol{f}} \right\|_p + \left\| \boldsymbol{A}^* \boldsymbol{W} \left(\boldsymbol{A} \hat{\boldsymbol{f}} - \boldsymbol{f} \right) \right\|_p \\
&\leq \left\| \boldsymbol{A}^* \boldsymbol{W} \boldsymbol{A} - \boldsymbol{I}_{|\mathcal{I}_{\boldsymbol{M}}|} \right\|_p \cdot \left\| \hat{\boldsymbol{f}} \right\|_p + \left\| \boldsymbol{A}^* \boldsymbol{W} \right\|_p \cdot \left\| \boldsymbol{A} \hat{\boldsymbol{f}} - \boldsymbol{f} \right\|_p. \qquad (3.44)
\end{aligned}$$

Using $\boldsymbol{A} \in \mathbb{C}^{N \times |\mathcal{I}_{\boldsymbol{M}}|}$ from (2.7) as well as $\boldsymbol{W} = \mathrm{diag}(w_j)_{j=1}^{N} = \mathrm{diag}(\boldsymbol{w})$ it holds

$$\left\| \boldsymbol{A}^* \boldsymbol{W} \right\|_1 = \max_{j=1,\dots,N} \sum_{\boldsymbol{k} \in \mathcal{I}_{\boldsymbol{M}}} |w_j| \cdot \left| \mathrm{e}^{-2\pi \mathrm{i} \boldsymbol{k} \boldsymbol{x}_j} \right| \leq \max_{j=1,\dots,N} |w_j| \cdot \sum_{\boldsymbol{k} \in \mathcal{I}_{\boldsymbol{M}}} 1 = |\mathcal{I}_{\boldsymbol{M}}| \cdot \|\boldsymbol{w}\|_\infty,$$

$$\left\| \boldsymbol{A}^* \boldsymbol{W} \right\|_\infty = \max_{\boldsymbol{k} \in \mathcal{I}_{\boldsymbol{M}}} \sum_{j=1}^{N} |w_j| \cdot \left| \mathrm{e}^{-2\pi \mathrm{i} \boldsymbol{k} \boldsymbol{x}_j} \right| \leq \sum_{j=1}^{N} |w_j| \cdot \max_{\boldsymbol{k} \in \mathcal{I}_{\boldsymbol{M}}} 1 = \|\boldsymbol{w}\|_1,$$

and

$$\left\| \boldsymbol{A}^* \boldsymbol{W} \right\|_{\mathrm{F}} = \sqrt{\sum_{\boldsymbol{k} \in \mathcal{I}_{\boldsymbol{M}}} \sum_{j=1}^{N} |w_j|^2 \cdot \left| \mathrm{e}^{-2\pi \mathrm{i} \boldsymbol{k} \boldsymbol{x}_j} \right|^2} \leq \sqrt{\sum_{j=1}^{N} |w_j|^2 \cdot |\mathcal{I}_{\boldsymbol{M}}|} = \sqrt{|\mathcal{I}_{\boldsymbol{M}}|} \cdot \|\boldsymbol{w}\|_2.$$

26

Hence, from (3.41) – (3.44) and $\|\cdot\|_2 \leq \|\cdot\|_F$ it follows that

$$\left\|\hat{\boldsymbol{f}} - \boldsymbol{A}^* \boldsymbol{W} \boldsymbol{f}\right\|_p \leq |\mathcal{I}_{\boldsymbol{M}}| \varepsilon \cdot \left\|\hat{\boldsymbol{f}}\right\|_p + \left\|\boldsymbol{A}\hat{\boldsymbol{f}} - \boldsymbol{f}\right\|_p \cdot |\mathcal{I}_{\boldsymbol{M}}|^{1/p} \|\boldsymbol{w}\|_q \tag{3.45}$$

for $p \in \{1, 2, \infty\}$ with $\frac{1}{p} + \frac{1}{q} = 1$. Now it merely remains to estimate $\left\|\boldsymbol{A}\hat{\boldsymbol{f}} - \boldsymbol{f}\right\|_p$ for the specific choice of f.

(i): Since for a trigonometric polynomial (2.8) of degree $\boldsymbol{M}$ it holds $\boldsymbol{A}\hat{\boldsymbol{f}} = \boldsymbol{f}$, the second error term in (3.45) vanishes and we obtain the assertion (3.38).

(ii): When considering a general 1-periodic function $f \in L_2(\mathbb{T}^d) \cap C(\mathbb{T}^d)$ in (2.1) we have

$$\left|\left[\boldsymbol{A}\hat{\boldsymbol{f}} - \boldsymbol{f}\right]_j\right| = \left|f(\boldsymbol{x}_j) - \sum_{\boldsymbol{k} \in \mathcal{I}_{\boldsymbol{M}}} c_{\boldsymbol{k}}(f) \, \mathrm{e}^{2\pi \mathrm{i} \boldsymbol{k} \boldsymbol{x}_j}\right|$$

$$\leq \max_{\boldsymbol{x} \in \mathbb{T}^d} \left|\sum_{\boldsymbol{k} \in \mathbb{Z}^d \setminus \mathcal{I}_{\boldsymbol{M}}} c_{\boldsymbol{k}}(f) \, \mathrm{e}^{2\pi \mathrm{i} \boldsymbol{k} \boldsymbol{x}}\right| = \|f - p_{\boldsymbol{M}}\|_{C(\mathbb{T}^d)} \quad j = 1, \ldots, N,$$

with the best approximating trigonometric polynomial $p_{\boldsymbol{M}}$ of degree $\boldsymbol{M}$ of f. Thus, this yields $\left\|\boldsymbol{A}\hat{\boldsymbol{f}} - \boldsymbol{f}\right\|_p \leq N^{1/p} \|f - p_{\boldsymbol{M}}\|_{C(\mathbb{T}^d)}$ and by (3.45) the assertion (3.39).

(iii): For a bandlimited function $f \in L_1(\mathbb{R}^d) \cap C_0(\mathbb{R}^d)$ with bandwidth $\boldsymbol{M}$ we may use the notation $\hat{\boldsymbol{f}} := (\hat{f}(\boldsymbol{k}))_{\boldsymbol{k} \in \mathcal{I}_{\boldsymbol{M}}}$ as well as the inverse Fourier transform (3.18) to estimate

$$\left|\left[\boldsymbol{A}\hat{\boldsymbol{f}} - \boldsymbol{f}\right]_j\right| = \left|f(\boldsymbol{x}_j) - \sum_{\boldsymbol{k} \in \mathcal{I}_{\boldsymbol{M}}} \hat{f}(\boldsymbol{k}) \, \mathrm{e}^{2\pi \mathrm{i} \boldsymbol{k} \boldsymbol{x}_j}\right| \leq \max_{\boldsymbol{x} \in \mathbb{T}^d} |Q(\boldsymbol{x})| = \|Q\|_{C(\mathbb{T}^d)}, \quad j = 1, \ldots, N,$$

with the pointwise quadrature error

$$Q(\boldsymbol{x}) := \int_{\left[-\frac{M}{2}, \frac{M}{2}\right)^d} \hat{f}(\boldsymbol{v}) \, \mathrm{e}^{2\pi \mathrm{i} \boldsymbol{v} \boldsymbol{x}} \, \mathrm{d}\boldsymbol{v} - \sum_{\boldsymbol{k} \in \mathcal{I}_{\boldsymbol{M}}} \hat{f}(\boldsymbol{k}) \, \mathrm{e}^{2\pi \mathrm{i} \boldsymbol{k} \boldsymbol{x}} \tag{3.46}$$

of the uniform quadrature rule (3.19). For detailed investigations of quadrature errors for bandlimited functions we refer to [39, 27]. Hence, we obtain $\left\|\boldsymbol{A}\hat{\boldsymbol{f}} - \boldsymbol{f}\right\|_p \leq N^{1/p} \|Q\|_{C(\mathbb{T}^d)}$ and by (3.45) the assertion (3.40). ∎

By Corollary 3.4 it is known that in the setting of trigonometric polynomials there is a linkage between an exact reconstruction (3.8) and the matrix product $\boldsymbol{A}^* \boldsymbol{W} \boldsymbol{A}$ being equal to identity $\boldsymbol{I}_{|\mathcal{I}_{\boldsymbol{M}}|}$. The following theorem shows that the error of the reconstruction (3.6) also affects the condition of the matrix $\boldsymbol{A}^* \boldsymbol{W} \boldsymbol{A}$.

Theorem 3.15. *Let $\boldsymbol{A} \in \mathbb{C}^{N \times |\mathcal{I}_{\boldsymbol{M}}|}$ from (2.7), $\boldsymbol{W} = \mathrm{diag}(w_j)_{j=1}^N$ and $\varepsilon > 0$ be given as in Theorem 3.14. If additionally $\varepsilon |\mathcal{I}_{\boldsymbol{M}}| < 1$ is fulfilled, then it holds*

$$1 \leq \kappa_2(\boldsymbol{A}^* \boldsymbol{W} \boldsymbol{A}) \leq \frac{1 + \varepsilon |\mathcal{I}_{\boldsymbol{M}}|}{1 - \varepsilon |\mathcal{I}_{\boldsymbol{M}}|} \tag{3.47}$$

for the condition number $\kappa_2(\boldsymbol{X}) := \|\boldsymbol{X}\|_2 \|\boldsymbol{X}^{-1}\|_2$.

Proof. To estimate the condition number $\kappa_2(\boldsymbol{A}^*\boldsymbol{W}\boldsymbol{A})$ we need to determine the norms $\left\|\boldsymbol{A}^*\boldsymbol{W}\boldsymbol{A}\right\|_2$ and $\left\|(\boldsymbol{A}^*\boldsymbol{W}\boldsymbol{A})^{-1}\right\|_2$. By (3.37) it is known that $\boldsymbol{A}^*\boldsymbol{W}\boldsymbol{A} = \boldsymbol{I}_{|\mathcal{I}_M|} + \boldsymbol{\mathcal{E}}$, where $\boldsymbol{\mathcal{E}} \coloneqq (\varepsilon_{\boldsymbol{\ell}-\boldsymbol{k}})_{\boldsymbol{\ell},\boldsymbol{k}\in\mathcal{I}_M}$, and therefore we have

$$\left\|\boldsymbol{A}^*\boldsymbol{W}\boldsymbol{A}\right\|_2 = \left\|\boldsymbol{I}_{|\mathcal{I}_M|} + \boldsymbol{\mathcal{E}}\right\|_2 \leq \left\|\boldsymbol{I}_{|\mathcal{I}_M|}\right\|_2 + \left\|\boldsymbol{\mathcal{E}}\right\|_2. \tag{3.48}$$

Moreover, it is known by the theory of Neumann series, cf. [67, Thm. 4.20], that if $\left\|\boldsymbol{I}_{|\mathcal{I}_M|} - \boldsymbol{T}\right\|_2 < 1$ holds for a matrix $\boldsymbol{T} \in \mathbb{C}^{|\mathcal{I}_M|\times|\mathcal{I}_M|}$, then $\boldsymbol{T}$ is invertible and its inverse is given by

$$\boldsymbol{T}^{-1} = \sum_{n=0}^{\infty} \left(\boldsymbol{I}_{|\mathcal{I}_M|} - \boldsymbol{T}\right)^n.$$

Using this property for $\boldsymbol{T} = \boldsymbol{A}^*\boldsymbol{W}\boldsymbol{A}$ we have

$$\left\|(\boldsymbol{A}^*\boldsymbol{W}\boldsymbol{A})^{-1}\right\|_2 = \left\|\sum_{n=0}^{\infty} \left(\boldsymbol{I}_{|\mathcal{I}_M|} - \boldsymbol{A}^*\boldsymbol{W}\boldsymbol{A}\right)^n\right\|_2 = \left\|\sum_{n=0}^{\infty} \boldsymbol{\mathcal{E}}^n\right\|_2 \leq \sum_{n=0}^{\infty} \left\|\boldsymbol{\mathcal{E}}^n\right\|_2, \tag{3.49}$$

in case that $\left\|\boldsymbol{I}_{|\mathcal{I}_M|} - \boldsymbol{A}^*\boldsymbol{W}\boldsymbol{A}\right\|_2 = \|\boldsymbol{\mathcal{E}}\|_2 < 1$. Hence, by (3.48) and (3.49) it holds

$$\kappa_2(\boldsymbol{A}^*\boldsymbol{W}\boldsymbol{A}) \leq (1 + \|\boldsymbol{\mathcal{E}}\|_2) \cdot \left(\sum_{n=0}^{\infty} \left\|\boldsymbol{\mathcal{E}}^n\right\|_2\right). \tag{3.50}$$

Additionally, we know that $|\varepsilon_{\boldsymbol{k}}| \leq \varepsilon$, $\boldsymbol{k} \in \mathcal{I}_{\boldsymbol{2M}}$, with some $\varepsilon > 0$ and therefore

$$\|\boldsymbol{\mathcal{E}}\|_2 \leq \|\boldsymbol{\mathcal{E}}\|_{\mathrm{F}} = \sqrt{\sum_{\boldsymbol{k}\in\mathcal{I}_M} \sum_{\boldsymbol{\ell}\in\mathcal{I}_M} |\varepsilon_{\boldsymbol{\ell}-\boldsymbol{k}}|^2} \leq \sqrt{\sum_{\boldsymbol{k}\in\mathcal{I}_M} \sum_{\boldsymbol{\ell}\in\mathcal{I}_M} \varepsilon^2} = \varepsilon\,|\mathcal{I}_{\boldsymbol{M}}|. \tag{3.51}$$

In other words, the correctness of (3.49) is ensured if $\varepsilon\,|\mathcal{I}_{\boldsymbol{M}}| < 1$. Since the spectral norm is a sub-multiplicative norm, (3.51) also implies $\|\boldsymbol{\mathcal{E}}^n\|_2 \leq \|\boldsymbol{\mathcal{E}}\|_2^n \leq (\varepsilon\,|\mathcal{I}_{\boldsymbol{M}}|)^n$. Consequently, we have

$$\sum_{n=0}^{\infty} \left\|\boldsymbol{\mathcal{E}}^n\right\|_2 \leq \sum_{n=0}^{\infty} (\varepsilon\,|\mathcal{I}_{\boldsymbol{M}}|)^n = \frac{1}{1 - \varepsilon\,|\mathcal{I}_{\boldsymbol{M}}|}. \tag{3.52}$$

Thus, combining (3.50), (3.51) and (3.52) yields the assertion (3.47). ∎

3.4 Connection to certain density compensation approaches from literature

In literature a variety of density compensation approaches can be found that are concerned with the setting of bandlimited functions and make use of a sinc transform

$$\boldsymbol{C} \coloneqq \left(|\mathcal{I}_{\boldsymbol{M}}|\operatorname{sinc}\left(M\pi\left(\boldsymbol{x}_j - \boldsymbol{M}^{-1} \odot \boldsymbol{\ell}\right)\right)\right)_{j=1,\,\boldsymbol{\ell}\in\mathcal{I}_{\boldsymbol{M}}}^{N} \in \mathbb{R}^{N\times|\mathcal{I}_{\boldsymbol{M}}|} \tag{3.53}$$

instead of the Fourier transform (3.6). Namely, instead of directly using the quadrature (3.21) for reconstruction, in these methods it is inserted into the inverse Fourier transform (3.18), i.e.,

$$f(\boldsymbol{x}) = \int\limits_{[-\frac{M}{2},\frac{M}{2})^d} \hat{f}(\boldsymbol{v}) \, \mathrm{e}^{2\pi\mathrm{i}\boldsymbol{v}\boldsymbol{x}} \, \mathrm{d}\boldsymbol{v} \approx \sum_{j=1}^N w_j \, f(\boldsymbol{x}_j) \int\limits_{[-\frac{M}{2},\frac{M}{2})^d} \mathrm{e}^{-2\pi\mathrm{i}\boldsymbol{v}(\boldsymbol{x}_j-\boldsymbol{x})} \, \mathrm{d}\boldsymbol{v}$$

$$= \sum_{j=1}^N w_j \, f(\boldsymbol{x}_j) \cdot |\mathcal{I_M}| \operatorname{sinc}(M\pi(\boldsymbol{x}_j - \boldsymbol{x})), \quad \boldsymbol{x} \in \mathbb{R}^d. \tag{3.54}$$

By using the sinc matrix $\boldsymbol{C} \in \mathbb{R}^{N \times |\mathcal{I_M}|}$ from (3.53), the weight matrix $\boldsymbol{W} = \operatorname{diag}(w_j)_{j=1}^N$ as well as the vectors $\boldsymbol{f} = (f(\boldsymbol{x}_j))_{j=1}^N$ and $\tilde{\boldsymbol{f}} = (f(\boldsymbol{M}^{-1} \odot \boldsymbol{\ell}))_{l \in \mathcal{I_M}}$, the evaluation of (3.54) at equispaced points $\boldsymbol{M}^{-1} \odot \boldsymbol{\ell}$, $\boldsymbol{\ell} \in \mathcal{I_M}$, can be denoted as $\tilde{\boldsymbol{f}} \approx \boldsymbol{C}^* \boldsymbol{W} \boldsymbol{f}$. Using the equispaced quadrature rule in (2.3), we find that evaluations $\hat{f}(\boldsymbol{k})$ of (3.17) at the uniform grid points $\boldsymbol{k} \in \mathcal{I_M}$ can be approximated by (3.20) by means of a simple FFT. In matrix-vector notation this can be written as $\hat{\boldsymbol{f}} \approx \tilde{\boldsymbol{D}}^* \boldsymbol{F}^*_{|\mathcal{I_M}|} \tilde{\boldsymbol{f}}$ where $\hat{\boldsymbol{f}} = (\hat{f}(\boldsymbol{k}))_{\boldsymbol{k} \in \mathcal{I_M}}$, $\boldsymbol{F}_{|\mathcal{I_M}|} := (\mathrm{e}^{2\pi\mathrm{i}\boldsymbol{k}(\boldsymbol{M}^{-1}\odot\boldsymbol{\ell})})_{\boldsymbol{\ell},\boldsymbol{k}\in\mathcal{I_M}}$, cf. (2.14), and $\tilde{\boldsymbol{D}} := \frac{1}{|\mathcal{I_M}|} \boldsymbol{I}_{|\mathcal{I_M}|}$. Thus, all in all one obtains an approximation of the form $\hat{\boldsymbol{f}} \approx \tilde{\boldsymbol{D}}^* \boldsymbol{F}^*_{|\mathcal{I_M}|} \boldsymbol{C}^* \boldsymbol{W} \boldsymbol{f}$.

Here some of these approaches, cf. [19], shall be reconsidered in the context of the Fourier transform (3.6). We especially focus on the connection of the approaches among each other as well as to our new method introduced in Section 3.1.

3.4.1 Density compensation using the pseudoinverse

Since (3.4) is in general not exactly solvable, we study the corresponding least squares problem, instead, i.e., we look for the approximant that minimizes the residual norm $\|\boldsymbol{f} - \boldsymbol{A}\hat{\boldsymbol{f}}\|_2$. It is known (e.g. [8, p. 15]) that this problem always has the unique solution

$$\hat{\boldsymbol{f}} \approx \tilde{\boldsymbol{h}}^{\mathrm{pinv}} := \boldsymbol{A}^\dagger \boldsymbol{f} \tag{3.55}$$

with the Moore-Penrose pseudoinverse $\boldsymbol{A}^\dagger$. Comparing (3.55) to the density compensation approach (3.6), the weights w_j should be chosen such that the matrix product $\boldsymbol{A}^* \boldsymbol{W}$ approximates the pseudoinverse $\boldsymbol{A}^\dagger$ as best as possible, i.e., we study the optimization problem

$$\operatorname*{Minimize}_{\boldsymbol{W}=\operatorname{diag}(w_j)_{j=1}^N} \left\| \boldsymbol{A}^* \boldsymbol{W} - \boldsymbol{A}^\dagger \right\|_{\mathrm{F}}^2, \tag{3.56}$$

where $\|\cdot\|_\mathrm{F}$ denotes the Frobenius norm of a matrix. It was shown in [62] that the solution to this least squares problem can be computed as

$$w_j = \frac{[\boldsymbol{A}\boldsymbol{A}^\dagger]_{j,j}}{[\boldsymbol{A}\boldsymbol{A}^*]_{j,j}} = \frac{1}{|\mathcal{I_M}|} \cdot [\boldsymbol{A}\boldsymbol{A}^\dagger]_{j,j}, \quad j = 1, \dots N. \tag{3.57}$$

However, since a singular value decomposition is necessary for the calculations in (3.57), we obtain a high complexity of $\mathcal{O}(N^2 |\mathcal{I_M}| + |\mathcal{I_M}|^3)$. Therefore, we study some more sophisticated least squares approaches in the following.

3.4.2 Density compensation using weighted normal equations of first kind

It is known, that every least squares solution to (3.4) satisfies the weighted normal equations of first kind $\boldsymbol{A}^*\boldsymbol{W}\boldsymbol{A}\hat{\boldsymbol{f}} = \boldsymbol{A}^*\boldsymbol{W}\boldsymbol{f}$, see e.g. [8, Thm. 1.1.2]. As already mentioned in Corollary 3.4, we have an exact reconstruction formula (3.6) for all trigonometric polynomials (2.8) of degree $\boldsymbol{M}$, if $\boldsymbol{A}^*\boldsymbol{W}\boldsymbol{A} = \boldsymbol{I}_{|\mathcal{I}_{\boldsymbol{M}}|}$ is fulfilled. Thus, we aim to compute optional weights w_j, $j = 1, \ldots, N$, by considering the optimization problem

$$\underset{\boldsymbol{W}=\mathrm{diag}(w_j)_{j=1}^N}{\mathrm{Minimize}} \ \left\| \boldsymbol{A}^*\boldsymbol{W}\boldsymbol{A} - \boldsymbol{I}_{|\mathcal{I}_{\boldsymbol{M}}|} \right\|_{\mathrm{F}}^2. \tag{3.58}$$

Analogous to [62] this could also be derived from (3.56) by introducing a right-hand scaling in the domain of measured data and minimizing the Frobenius norm of the weighted error matrix $\boldsymbol{E}_{\mathrm{r}} := \boldsymbol{E} \cdot \boldsymbol{A}$, where $\boldsymbol{E} := \boldsymbol{A}^*\boldsymbol{W} - \boldsymbol{A}^\dagger$ is the error matrix in (3.56).

In [59] it was shown that a solution $\boldsymbol{W} = \mathrm{diag}(\boldsymbol{w})$ to (3.58) can be obtained by solving $\boldsymbol{S}\boldsymbol{w} = \boldsymbol{b}$, where

$$\boldsymbol{S} := \left(\left| [\boldsymbol{A}\boldsymbol{A}^*]_{j,h} \right|^2 \right)_{j,h=1}^N \quad \text{and} \quad \boldsymbol{b} = |\mathcal{I}_{\boldsymbol{M}}| \cdot \boldsymbol{1}_N. \tag{3.59}$$

However, since $\boldsymbol{S}\boldsymbol{w} = \boldsymbol{b}$ is not separable for single w_j, $j = 1, \ldots, N$, computing these weights is of complexity $\mathcal{O}(N^3)$. This is why the authors in [62] restricted themselves to a maximal image size of 64×64 pixels, which corresponds to setting $M = 64$.

3.4.3 Density compensation using weighted normal equations of second kind

Another approach for density compensation factors is based on the weighted normal equations of second kind

$$\boldsymbol{A}\boldsymbol{A}^*\boldsymbol{W}\boldsymbol{y} = \boldsymbol{f}, \quad \boldsymbol{A}^*\boldsymbol{W}\boldsymbol{y} = \hat{\boldsymbol{f}}. \tag{3.60}$$

We recognize that by (3.60) we are given an exact approximation $\hat{\boldsymbol{f}} = \boldsymbol{A}^*\boldsymbol{W}\boldsymbol{f}$ of the Fourier coefficients in (3.6), in case it holds $\boldsymbol{y} = \boldsymbol{f}$ and thereby $\boldsymbol{A}\boldsymbol{A}^*\boldsymbol{W} = \boldsymbol{I}_N$. To this end, we consider the optimization problem

$$\underset{\boldsymbol{W}=\mathrm{diag}(w_j)_{j=1}^N}{\mathrm{Minimize}} \ \|\boldsymbol{A}\boldsymbol{A}^*\boldsymbol{W} - \boldsymbol{I}_N\|_{\mathrm{F}}^2. \tag{3.61}$$

As in Section 3.4.2, we remark that this optimization problem (3.61) could also be derived from (3.56) by introducing an additional left-hand scaling in the Fourier domain and minimizing the Frobenius norm of the weighted error matrix $\boldsymbol{E}_{\mathrm{l}} := \boldsymbol{A} \cdot \boldsymbol{E}$.

Remark 3.16. *An analogous approach considering the* sinc *transform (3.53) instead of the Fourier transform (3.6) was already studied in [52]. Another version using a sinc transform evaluated at pointwise differences of the nonequispaced nodes instead of (3.53) was studied in [12, 30], where it was claimed that this approach coincides with the one*

in [52]. However, we remark that due to the sampling theorem of Shannon-Whittaker-Kotelnikov, see (3.35), applied to the function $f(\boldsymbol{x}) = \operatorname{sinc}(M\pi(\boldsymbol{x}_j - \boldsymbol{x}))$, i.e.,

$$\operatorname{sinc}(M\pi(\boldsymbol{x}_j - \boldsymbol{x})) = \sum_{\boldsymbol{\ell} \in \mathbb{Z}^d} \operatorname{sinc}(M\pi(\boldsymbol{x}_j - \boldsymbol{M}^{-1} \odot \boldsymbol{\ell})) \operatorname{sinc}\left(M\pi\left(\boldsymbol{x} - \boldsymbol{M}^{-1} \odot \boldsymbol{\ell}\right)\right)$$

and its evaluation at $\boldsymbol{x} = \boldsymbol{x}_h$, $h = 1, \dots, N$, this claim only holds asymptotically for $|\mathcal{I}_{\boldsymbol{M}}| \to \infty$ in the setting of the sinc transform.

In contrast, when using the Fourier transform (3.6) this equality can directly be seen. Then the analog to [30] utilizes an approximation of the form $\boldsymbol{f} \approx \boldsymbol{HWf}$, where the matrix $\boldsymbol{H}$ is defined as the system matrix of (3.4) evaluated at pointwise differences of the nonequispaced nodes, i.e.,

$$\boldsymbol{H} := \left(\sum_{\boldsymbol{k} \in \mathcal{I}_{\boldsymbol{M}}} \mathrm{e}^{2\pi\mathrm{i}\boldsymbol{k}(\boldsymbol{x}_j - \boldsymbol{x}_h)} \right)_{j,h=1}^{N}. \tag{3.62}$$

Since by (2.7) it holds $\boldsymbol{H} = \boldsymbol{A}\boldsymbol{A}^*$, minimizing the approximation error

$$\|\boldsymbol{H}^*\boldsymbol{W}\boldsymbol{f} - \boldsymbol{f}\|_2^2 = \|\boldsymbol{A}\boldsymbol{A}^*\boldsymbol{W}\boldsymbol{f} - \boldsymbol{f}\|_2^2 \le \|\boldsymbol{A}\boldsymbol{A}^*\boldsymbol{W} - \boldsymbol{I}_N\|_{\mathrm{F}}^2 \cdot \|\boldsymbol{f}\|_2^2,$$

leads to the optimization problem (3.61) as well. □

It was shown in [52] that the minimizer of (3.61) is given by

$$w_j = \frac{|\mathcal{I}_{\boldsymbol{M}}|}{\sum_{h=1}^{N} \left|[\boldsymbol{A}\boldsymbol{A}^*]_{j,h}\right|^2}, \quad j = 1, \dots, N. \tag{3.63}$$

Since for fixed j the computation of $[\boldsymbol{A}\boldsymbol{A}^*]_{j,h}$, $h = 1, \dots, N$, is of complexity $\mathcal{O}(N|\mathcal{I}_{\boldsymbol{M}}|)$, the weights (3.63) can be computed in $\mathcal{O}(N^2|\mathcal{I}_{\boldsymbol{M}}|)$ arithmetic operations. However, due to the explicit representation (3.62) the computation of $[\boldsymbol{A}\boldsymbol{A}^*]_{j,h}$, $h = 1, \dots, N$, for fixed j can be accelerated by means of the NFFT (see Algorithm 2.1). Then this step takes $\mathcal{O}(|\mathcal{I}_{\boldsymbol{M}}|\log(|\mathcal{I}_{\boldsymbol{M}}|) + N)$ arithmetic operations and the overall complexity is given by $\mathcal{O}(N \cdot |\mathcal{I}_{\boldsymbol{M}}|\log(|\mathcal{I}_{\boldsymbol{M}}|) + N^2)$.

As mentioned in [52] one could also consider a simplified version of the optimization problem (3.61) by reducing the number of conditions, e.g. by summing over the columns of both sides of $\boldsymbol{A}\boldsymbol{A}^*\boldsymbol{W} = \boldsymbol{I}_N$ as

$$\sum_{j=1}^{N} w_j \sum_{\boldsymbol{k} \in \mathcal{I}_{\boldsymbol{M}}} \mathrm{e}^{2\pi\mathrm{i}\boldsymbol{k}(\boldsymbol{x}_h - \boldsymbol{x}_j)} = \sum_{j=1}^{N} \delta_{j,h} = 1, \quad h = 1, \dots, N. \tag{3.64}$$

By means of (2.7) this can be written as $\boldsymbol{A}\boldsymbol{A}^*\boldsymbol{w} = \boldsymbol{1}_N$. Since fast multiplication with $\boldsymbol{A}$ and $\boldsymbol{A}^*$ can be realized using the NFFT (see Algorithm 2.1) and the adjoint NFFT (see Algorithm 2.4), respectively, a solution to the linear system of equations $\boldsymbol{A}\boldsymbol{A}^*\boldsymbol{w} = \boldsymbol{1}_N$ can be computed iteratively with arithmetic complexity $\mathcal{O}(|\mathcal{I}_{\boldsymbol{M}}|\log(|\mathcal{I}_{\boldsymbol{M}}|) + N)$.

Finally, we investigate the connection of this approach to our method introduced in Section 3.1. To this end, suppose the linear system (3.10) is fulfilled for given

$\boldsymbol{w} \in \mathbb{C}^N$, i.e., by $\boldsymbol{A}^* = \overline{\boldsymbol{A}^T}$ we have $(\delta_{\mathbf{0},\boldsymbol{k}})_{\boldsymbol{k} \in \mathcal{I}_M} = \overline{\boldsymbol{A}^T \boldsymbol{w}} = \boldsymbol{A}^* \overline{\boldsymbol{w}}$. Then multiplication with $\boldsymbol{A} \in \mathbb{C}^{N \times |\mathcal{I}_M|}$ in (2.7) yields

$$\boldsymbol{A}\boldsymbol{A}^*\overline{\boldsymbol{w}} = \boldsymbol{A} \cdot (\delta_{\mathbf{0},\boldsymbol{k}})_{\boldsymbol{k} \in \mathcal{I}_M} = \left(\sum_{\boldsymbol{k} \in \mathcal{I}_M} \delta_{\mathbf{0},\boldsymbol{k}} \cdot \mathrm{e}^{2\pi \mathrm{i} \boldsymbol{k} \boldsymbol{x}_j} \right)_{j=1}^N = \mathbf{1}_N.$$

In other words, an exact solution $\boldsymbol{w}$ to the linear system (3.10) implies that the conjugate complex weights $\overline{\boldsymbol{w}}$ exactly solve the system (3.64). However, the reversal does not hold true and therefore (3.64) is not equivalent to (3.10). Moreover, we have seen in Corollary 3.4 that an augmented variant of (3.10), namely (3.14), is necessary to obtain an exact reconstruction $\hat{f}_{\boldsymbol{k}} = h_{\boldsymbol{k}}^{\mathrm{w}}$ in (3.6) for trigonometric polynomials (2.8) with maximum degree $\boldsymbol{M}$.

4 Direct inversion using matrix optimization

As seen in Remark 3.1, the previously considered density compensation techniques can be regarded as an optimization of the sparse matrix $\boldsymbol{B} \in \mathbb{R}^{N \times |\mathcal{I}_{M_\sigma}|}$ from the NFFT, cf. Section 2.1. Since density compensation allows only N degrees of freedom, this limitation shall now be softened, i.e., instead of searching for optimal scaling factors for the rows of $\boldsymbol{B}$, we now study the optimization of each nonzero entry of the sparse matrix $\boldsymbol{B}$, cf. [38]. To this end, we firstly have another look at the equispaced setting. It is known by (3.1) and (3.2), that for equispaced nodes and appropriately chosen parameters a one-sided inversion is given by composition of the Fourier matrix and its adjoint. Hence, we aim to use this result to find a good approximation of the inverse in the general setting.

Considering problem (3.4) we seek to find an appropriate matrix $\boldsymbol{X}$ such that we have $\boldsymbol{X}\boldsymbol{A} \approx \boldsymbol{I}_{|\mathcal{I}_M|}$, since then we can simply compute an approximation of the Fourier coefficients by means of $\boldsymbol{X}\boldsymbol{f} = \boldsymbol{X}\boldsymbol{A}\hat{\boldsymbol{f}} \approx \hat{\boldsymbol{f}}$. To find this left-inverse $\boldsymbol{X}$, we utilize the fact that in the equispaced case it is known that (3.1) holds in the overdetermined setting $|\mathcal{I}_M| \leq N$. In addition, we also incorporate the approximate factorization $\boldsymbol{A}^* \approx \boldsymbol{D}^* \boldsymbol{F}^* \boldsymbol{B}^*$ of the adjoint NFFT, cf. Section 2.2, with the matrices $\boldsymbol{D} \in \mathbb{C}^{|\mathcal{I}_M| \times |\mathcal{I}_M|}$, $\boldsymbol{F} \in \mathbb{C}^{|\mathcal{I}_{M_\sigma}| \times |\mathcal{I}_M|}$ and $\boldsymbol{B} \in \mathbb{R}^{N \times |\mathcal{I}_{M_\sigma}|}$ defined in (2.13), (2.14) and (2.15). Combining both ingredients we aim for an approximation of the form $\boldsymbol{D}^* \boldsymbol{F}^* \boldsymbol{B}^* \boldsymbol{A} \approx \boldsymbol{I}_{|\mathcal{I}_M|}$. To achieve an approximation like this, we aim to modify the matrix $\boldsymbol{B}$ such that its sparse structure with at most $(2m+1)^d$ entries per row and consequently the arithmetic complexity of its evaluation is preserved. A matrix satisfying this property we call $(2m+1)^d$-*sparse*.

Remark 4.1. *We remark that similar considerations can also be deduced from the density compensation approach in Section 3 as follows. By Corollary 3.4 it is known that an exact reconstruction needs to satisfy $\boldsymbol{A}^* \boldsymbol{W} \boldsymbol{A} = \boldsymbol{I}_{|\mathcal{I}_M|}$. Since the reconstruction shall be realized efficiently by means of an adjoint NFFT, one rather has to consider $\boldsymbol{D}^* \boldsymbol{F}^* \boldsymbol{B}^* \boldsymbol{W} \boldsymbol{A} \approx \boldsymbol{I}_{|\mathcal{I}_M|}$. Using the definition $\tilde{\boldsymbol{B}} := \boldsymbol{W}^* \boldsymbol{B}$ as in Remark 3.1, we end up with an approximation of the form $\boldsymbol{D}^* \boldsymbol{F}^* \tilde{\boldsymbol{B}}^* \boldsymbol{A} \approx \boldsymbol{I}_{|\mathcal{I}_M|}$. Thus, it is a natural choice to*

consider an approximation of the same form for the optimization of each nonzero entry of the sparse matrix $\boldsymbol{B}$. $\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\square$

Let $\tilde{\boldsymbol{B}}$ denote such a modified matrix. By defining $\tilde{\boldsymbol{h}} := \boldsymbol{D}^*\boldsymbol{F}^*\tilde{\boldsymbol{B}}^*\boldsymbol{f}$, we recognize that the minimization of the approximation error

$$\left\|\tilde{\boldsymbol{h}} - \hat{\boldsymbol{f}}\right\|_2 = \left\|\boldsymbol{D}^*\boldsymbol{F}^*\tilde{\boldsymbol{B}}^*\boldsymbol{f} - \hat{\boldsymbol{f}}\right\|_2 = \left\|\boldsymbol{D}^*\boldsymbol{F}^*\tilde{\boldsymbol{B}}^*\boldsymbol{A}\hat{\boldsymbol{f}} - \hat{\boldsymbol{f}}\right\|_2$$
$$= \left\|\left(\boldsymbol{D}^*\boldsymbol{F}^*\tilde{\boldsymbol{B}}^*\boldsymbol{A} - \boldsymbol{I}_{|\mathcal{I}_M|}\right)\hat{\boldsymbol{f}}\right\|_2 \leq \left\|\boldsymbol{D}^*\boldsymbol{F}^*\tilde{\boldsymbol{B}}^*\boldsymbol{A} - \boldsymbol{I}_{|\mathcal{I}_M|}\right\|_{\mathrm{F}}\left\|\hat{\boldsymbol{f}}\right\|_2 \qquad (4.1)$$

implies the optimization problem

$$\underset{\tilde{\boldsymbol{B}} \in \mathbb{R}^{N \times |\mathcal{I}_{\boldsymbol{M}_{\boldsymbol{\sigma}}}|} : \; \tilde{\boldsymbol{B}} \; (2m+1)^d\text{-sparse}}{\text{Minimize}} \left\|\boldsymbol{D}^*\boldsymbol{F}^*\tilde{\boldsymbol{B}}^*\boldsymbol{A} - \boldsymbol{I}_{|\mathcal{I}_M|}\right\|_{\mathrm{F}}^2. \qquad (4.2)$$

Note that a similar idea for the forward problem, i. e., the evaluation of (2.5), was already studied in [51]. By the definition of the Frobenius norm we have $\|\boldsymbol{Z}\|_{\mathrm{F}} = \|\boldsymbol{Z}^*\|_{\mathrm{F}}$, such that (4.2) is equivalent to its adjoint

$$\underset{\tilde{\boldsymbol{B}} \in \mathbb{R}^{N \times |\mathcal{I}_{\boldsymbol{M}_{\boldsymbol{\sigma}}}|} : \; \tilde{\boldsymbol{B}} \; (2m+1)^d\text{-sparse}}{\text{Minimize}} \left\|\boldsymbol{A}^*\tilde{\boldsymbol{B}}\boldsymbol{F}\boldsymbol{D} - \boldsymbol{I}_{|\mathcal{I}_M|}\right\|_{\mathrm{F}}^2. \qquad (4.3)$$

Since it is known by (2.14) that $\boldsymbol{F}^*\boldsymbol{F} = |\mathcal{I}_{\boldsymbol{M}_{\boldsymbol{\sigma}}}|\,\boldsymbol{I}_{|\mathcal{I}_M|}$ and $\boldsymbol{D} \in \mathbb{R}^{|\mathcal{I}_M| \times |\mathcal{I}_M|}$ is diagonal by (2.13), we have $\frac{1}{|\mathcal{I}_{\boldsymbol{M}_{\boldsymbol{\sigma}}}|}\boldsymbol{D}^{-1}\boldsymbol{F}^*\boldsymbol{F}\boldsymbol{D} = \boldsymbol{I}_{|\mathcal{I}_M|}$. Thus, due to the fact that the Frobenius norm is a submultiplicative norm, it holds

$$\left\|\boldsymbol{A}^*\tilde{\boldsymbol{B}}\boldsymbol{F}\boldsymbol{D} - \boldsymbol{I}_{|\mathcal{I}_M|}\right\|_{\mathrm{F}} = \left\|\left(\boldsymbol{A}^*\tilde{\boldsymbol{B}} - \tfrac{1}{|\mathcal{I}_{\boldsymbol{M}_{\boldsymbol{\sigma}}}|}\boldsymbol{D}^{-1}\boldsymbol{F}^*\right)\boldsymbol{F}\boldsymbol{D}\right\|_{\mathrm{F}}$$
$$\leq \left\|\boldsymbol{A}^*\tilde{\boldsymbol{B}} - \tfrac{1}{|\mathcal{I}_{\boldsymbol{M}_{\boldsymbol{\sigma}}}|}\boldsymbol{D}^{-1}\boldsymbol{F}^*\right\|_{\mathrm{F}}\left\|\boldsymbol{F}\boldsymbol{D}\right\|_{\mathrm{F}}. \qquad (4.4)$$

Hence, we consider the optimization problem

$$\underset{\tilde{\boldsymbol{B}} \in \mathbb{R}^{N \times |\mathcal{I}_{\boldsymbol{M}_{\boldsymbol{\sigma}}}|} : \; \tilde{\boldsymbol{B}} \; (2m+1)^d\text{-sparse}}{\text{Minimize}} \left\|\boldsymbol{A}^*\tilde{\boldsymbol{B}} - \tfrac{1}{|\mathcal{I}_{\boldsymbol{M}_{\boldsymbol{\sigma}}}|}\boldsymbol{D}^{-1}\boldsymbol{F}^*\right\|_{\mathrm{F}}^2. \qquad (4.5)$$

Based on the definition of the Frobenius norm of a matrix $\boldsymbol{Z} \in \mathbb{R}^{k \times n}$ and the definition of the Euclidean norm of a vector $\boldsymbol{y} \in \mathbb{R}^n$, we obtain for $\boldsymbol{z}_j$ being the columns of $\boldsymbol{Z} \in \mathbb{R}^{k \times n}$ that

$$\|\boldsymbol{Z}\|_F^2 = \sum_{i=1}^k \sum_{j=1}^n |z_{ij}|^2 = \sum_{j=1}^n \|\boldsymbol{z}_j\|_2^2. \qquad (4.6)$$

Since we aim to preserve the property that $\boldsymbol{B}$ is a $(2m+1)^d$-sparse matrix, we rewrite the norm in (4.5) by (4.6) in terms of the columns of $\tilde{\boldsymbol{B}}$ considering only the nonzero entries of each column. To this end, analogously to (2.12) we define the index set

$$\mathcal{I}_{\boldsymbol{M}_{\boldsymbol{\sigma}},m}(\boldsymbol{\ell}) := \left\{j \in \{1,\dots,N\} : \exists\, \boldsymbol{z} \in \mathbb{Z}^d \text{ with } -m\mathbf{1} \leq \boldsymbol{M}_{\boldsymbol{\sigma}} \odot \boldsymbol{x}_j - \boldsymbol{\ell} + \boldsymbol{z} \leq m\mathbf{1}\right\} \quad (4.7)$$

of the nonzero entries of $\boldsymbol{\ell}$-th column of $\boldsymbol{B} \in \mathbb{R}^{N \times |\mathcal{I}_{\boldsymbol{M}_{\boldsymbol{\sigma}}}|}$. Thus, it holds

$$\left\|\boldsymbol{A}^*\tilde{\boldsymbol{B}} - \tfrac{1}{|\mathcal{I}_{\boldsymbol{M}_{\boldsymbol{\sigma}}}|}\boldsymbol{D}^{-1}\boldsymbol{F}^*\right\|_{\mathrm{F}}^2 = \sum_{\boldsymbol{\ell} \in \mathcal{I}_{\boldsymbol{M}_{\boldsymbol{\sigma}}}} \left\|\boldsymbol{H}_{\boldsymbol{\ell}}\tilde{\boldsymbol{b}}_{\boldsymbol{\ell}} - \tfrac{1}{|\mathcal{I}_{\boldsymbol{M}_{\boldsymbol{\sigma}}}|}\boldsymbol{D}^{-1}\boldsymbol{f}_{\boldsymbol{\ell}}\right\|_2^2, \qquad (4.8)$$

where $\tilde{\boldsymbol{b}}_{\boldsymbol{\ell}} \in \mathbb{R}^{|\mathcal{I}_{\boldsymbol{M}_{\boldsymbol{\sigma}},m}(\boldsymbol{\ell})|}$ denotes the vectors of the nonzeros of each column of $\tilde{\boldsymbol{B}}$,

$$\boldsymbol{H}_{\boldsymbol{\ell}} := \left(\mathrm{e}^{-2\pi\mathrm{i}\boldsymbol{k}\boldsymbol{x}_j}\right)_{\boldsymbol{k}\in\mathcal{I}_{\boldsymbol{M}},\, j\in\mathcal{I}_{\boldsymbol{M}_{\boldsymbol{\sigma}},m}(\boldsymbol{\ell})} \in \mathbb{C}^{|\mathcal{I}_{\boldsymbol{M}}|\times|\mathcal{I}_{\boldsymbol{M}_{\boldsymbol{\sigma}},m}(\boldsymbol{\ell})|} \tag{4.9}$$

are the corresponding submatrices of $\boldsymbol{A}^* \in \mathbb{C}^{|\mathcal{I}_{\boldsymbol{M}}|\times N}$, cf. (2.7), and $\boldsymbol{f}_{\boldsymbol{\ell}} \in \mathbb{C}^{|\mathcal{I}_{\boldsymbol{M}}|}$ are the columns of $\boldsymbol{F}^* \in \mathbb{C}^{|\mathcal{I}_{\boldsymbol{M}}|\times|\mathcal{I}_{\boldsymbol{M}_{\boldsymbol{\sigma}}}|}$, cf. (2.14). Hence, we receive the equivalent optimization problems

$$\underset{\tilde{\boldsymbol{b}}_{\boldsymbol{\ell}}\in\mathbb{R}^{|\mathcal{I}_{\boldsymbol{M}_{\boldsymbol{\sigma}},m}(\boldsymbol{\ell})|}}{\text{Minimize}} \left\| \boldsymbol{H}_{\boldsymbol{\ell}}\tilde{\boldsymbol{b}}_{\boldsymbol{\ell}} - \tfrac{1}{|\mathcal{I}_{\boldsymbol{M}_{\boldsymbol{\sigma}}}|}\boldsymbol{D}^{-1}\boldsymbol{f}_{\boldsymbol{\ell}} \right\|_2^2, \quad \boldsymbol{\ell}\in\mathcal{I}_{\boldsymbol{M}_{\boldsymbol{\sigma}}}. \tag{4.10}$$

Thus, if the matrix (4.9) has full column rank, the solution of the least squares problem (4.10) can be computed by means of the pseudoinverse $\boldsymbol{H}_{\boldsymbol{\ell}}^{\dagger}$ as

$$\boldsymbol{b}_{\boldsymbol{\ell}}^{\mathrm{opt}} := \tfrac{1}{|\mathcal{I}_{\boldsymbol{M}_{\boldsymbol{\sigma}}}|}\left(\boldsymbol{H}_{\boldsymbol{\ell}}^*\boldsymbol{H}_{\boldsymbol{\ell}}\right)^{-1}\boldsymbol{H}_{\boldsymbol{\ell}}^*\boldsymbol{D}^{-1}\boldsymbol{f}_{\boldsymbol{\ell}}, \quad \boldsymbol{\ell}\in\mathcal{I}_{\boldsymbol{M}_{\boldsymbol{\sigma}}}. \tag{4.11}$$

Having these vectors $\boldsymbol{b}_{\boldsymbol{\ell}}^{\mathrm{opt}}$ we compose the optimized matrix $\boldsymbol{B}_{\mathrm{opt}}$, observing that $\boldsymbol{b}_{\boldsymbol{\ell}}^{\mathrm{opt}}$ only consist of the nonzero entries of $\boldsymbol{B}_{\mathrm{opt}}$. Then the approximation of the Fourier coefficients is given by

$$\hat{\boldsymbol{f}} \approx \boldsymbol{h}_{\mathrm{opt}} := \boldsymbol{D}^*\boldsymbol{F}^*\boldsymbol{B}_{\mathrm{opt}}^*\boldsymbol{f}. \tag{4.12}$$

In other words, this approach yields an inverse NFFT by modifying the adjoint NFFT.

Remark 4.2. *To achieve an efficient algorithm we now have a closer look at the computation scheme* (4.11). *We start with the computation of the matrix* $\boldsymbol{H}_{\boldsymbol{\ell}}^*\boldsymbol{H}_{\boldsymbol{\ell}}$. *By introducing the d-dimensional Dirichlet kernel*

$$D_{\boldsymbol{M}}(\boldsymbol{x}) := \sum_{k_1=-\frac{M}{2}+1}^{\frac{M}{2}-1} \cdots \sum_{k_d=-\frac{M}{2}+1}^{\frac{M}{2}-1} \mathrm{e}^{2\pi\mathrm{i}\boldsymbol{k}\boldsymbol{x}} = \prod_{t=1}^{d} D_{\frac{M}{2}-1}(x_t) = \prod_{t=1}^{d} \frac{\sin((M-1)\pi x_t)}{\sin(\pi x_t)},$$

the matrix $\boldsymbol{H}_{\boldsymbol{\ell}}^*\boldsymbol{H}_{\boldsymbol{\ell}}$ *in* (4.11) *can explicitly be stated via*

$$\begin{aligned}
\boldsymbol{H}_{\boldsymbol{\ell}}^*\boldsymbol{H}_{\boldsymbol{\ell}} &= \left[\sum_{\boldsymbol{k}\in\mathcal{I}_{\boldsymbol{M}}} \mathrm{e}^{2\pi\mathrm{i}\boldsymbol{k}(\boldsymbol{x}_h-\boldsymbol{x}_j)} \right]_{h,j\in\mathcal{I}_{\boldsymbol{M}_{\boldsymbol{\sigma}},m}(\boldsymbol{\ell})} \\
&= \left[\prod_{t=1}^{d}\left(D_{\frac{M}{2}-1}(x_{h_t}-x_{j_t}) + \mathrm{e}^{-M\pi\mathrm{i}(x_{h_t}-x_{j_t})} \right) \right]_{h,j\in\mathcal{I}_{\boldsymbol{M}_{\boldsymbol{\sigma}},m}(\boldsymbol{\ell})},
\end{aligned} \tag{4.13}$$

i. e., for given index set $\mathcal{I}_{\boldsymbol{M}_{\boldsymbol{\sigma}},m}(\boldsymbol{\ell})$ *the matrix* $\boldsymbol{H}_{\boldsymbol{\ell}}^*\boldsymbol{H}_{\boldsymbol{\ell}}$ *can be determined in* $\mathcal{O}(|\mathcal{I}_{\boldsymbol{M}_{\boldsymbol{\sigma}},m}(\boldsymbol{\ell})|)$ *operations. Considering the right hand sides of* (4.11), *by definitions* (4.9), (2.13) *and* (2.14) *we have*

$$\boldsymbol{v}_{\boldsymbol{\ell}} := \tfrac{1}{|\mathcal{I}_{\boldsymbol{M}_{\boldsymbol{\sigma}}}|}\boldsymbol{H}_{\boldsymbol{\ell}}^*\boldsymbol{D}^{-1}\boldsymbol{f}_{\boldsymbol{\ell}} = \left(\sum_{\boldsymbol{k}\in\mathcal{I}_{\boldsymbol{M}}} \hat{w}(\boldsymbol{k})\,\mathrm{e}^{2\pi\mathrm{i}\boldsymbol{k}\left(\boldsymbol{x}_j-\boldsymbol{M}_{\boldsymbol{\sigma}}^{-1}\odot\boldsymbol{\ell}\right)} \right)_{j\in\mathcal{I}_{\boldsymbol{M}_{\boldsymbol{\sigma}},m}(\boldsymbol{\ell})}, \quad \boldsymbol{\ell}\in\mathcal{I}_{\boldsymbol{M}_{\boldsymbol{\sigma}}}. \tag{4.14}$$

Thus, since $\tfrac{1}{|\mathcal{I}_{\boldsymbol{M}_{\boldsymbol{\sigma}}}|}\boldsymbol{D}^{-1} = \mathrm{diag}(\hat{w}(\boldsymbol{k}))_{\boldsymbol{k}\in\mathcal{I}_{\boldsymbol{M}}}$, *the computation of* $\boldsymbol{v}_{\boldsymbol{\ell}}$ *involves neither multiplication with nor division by the (possibly) huge number* $|\mathcal{I}_{\boldsymbol{M}_{\boldsymbol{\sigma}}}|$ *and is therefore numerically stable.* $\qquad\square$

This leads to the following algorithm.

Algorithm 4.3 (Optimization of the sparse matrix $\boldsymbol{B}$)**.**

For $d, N \in \mathbb{N}$ let $\boldsymbol{x}_j \in \mathbb{T}^d$, $j = 1, \dots, N$, be given nodes. Further let $M \in 2\mathbb{N}$, $\boldsymbol{M} := M \cdot \mathbf{1}_d$, an oversampling factor $\sigma \geq 1$ with $2\mathbb{N} \ni M_\sigma := 2\lceil \lceil \sigma M \rceil / 2 \rceil$ and $\boldsymbol{M_\sigma} := M_\sigma \cdot \mathbf{1}_d$ as well as a truncation parameter $m \ll M_\sigma$ be given.

1. *For $\boldsymbol{\ell} \in \mathcal{I}_{\boldsymbol{M_\sigma}}$:*

Determine the index set $\mathcal{I}_{\boldsymbol{M_\sigma}, m}(\boldsymbol{\ell})$, cf. (4.7).	$\mathcal{O}(\lvert \mathcal{I}_{\boldsymbol{M_\sigma}, m}(\boldsymbol{\ell}) \rvert)$
Compute the right side $\boldsymbol{v}_{\boldsymbol{\ell}}$ via (4.14).	$\mathcal{O}(\lvert \mathcal{I}_{\boldsymbol{M_\sigma}, m}(\boldsymbol{\ell}) \rvert)$
Determine $\boldsymbol{H}_{\boldsymbol{\ell}}^* \boldsymbol{H}_{\boldsymbol{\ell}}$ via (4.13).	$\mathcal{O}(\lvert \mathcal{I}_{\boldsymbol{M_\sigma}, m}(\boldsymbol{\ell}) \rvert^2)$
Solve $(\boldsymbol{H}_{\boldsymbol{\ell}}^* \boldsymbol{H}_{\boldsymbol{\ell}}) \, \boldsymbol{b}_{\boldsymbol{\ell}}^{\mathrm{opt}} = \boldsymbol{v}_{\boldsymbol{\ell}}$, i. e., compute $\boldsymbol{b}_{\boldsymbol{\ell}}^{\mathrm{opt}}$, cf. (4.11).	$\mathcal{O}(\lvert \mathcal{I}_{\boldsymbol{M_\sigma}, m}(\boldsymbol{\ell}) \rvert^3)$

2. *Compose $\boldsymbol{B}_{\mathrm{opt}} \in \mathbb{R}^{N \times \lvert \mathcal{I}_{\boldsymbol{M_\sigma}} \rvert}$ columnwise of the $\boldsymbol{b}_{\boldsymbol{\ell}}^{\mathrm{opt}} \in \mathbb{R}^{\lvert \mathcal{I}_{\boldsymbol{M_\sigma}, m}(\boldsymbol{\ell}) \rvert}$.* $\qquad \mathcal{O}(\lvert \mathcal{I}_{\boldsymbol{M}} \rvert)$

Output: *optimized matrix $\boldsymbol{B}_{\mathrm{opt}}$* $\qquad\qquad$ **Complexity:** $\mathcal{O}(\lvert \mathcal{I}_{\boldsymbol{M}} \rvert \cdot \lvert \mathcal{I}_{\boldsymbol{M_\sigma}, m}(\boldsymbol{\ell}) \rvert^3)$

Note that a general statement about the dimensions of $\boldsymbol{H}_{\boldsymbol{\ell}} \in \mathbb{C}^{\lvert \mathcal{I}_{\boldsymbol{M}} \rvert \times \lvert \mathcal{I}_{\boldsymbol{M_\sigma}, m}(\boldsymbol{\ell}) \rvert}$ is not possible, since the size of the set $\mathcal{I}_{\boldsymbol{M_\sigma}, m}(\boldsymbol{\ell})$ heavily depends on the distribution of the nodes. To visualize this circumstance, we depicted some exemplary patterns of the nonzero entries of the original matrix $\boldsymbol{B} \in \mathbb{R}^{N \times \lvert \mathcal{I}_{\boldsymbol{M_\sigma}} \rvert}$ in Figure 4.1. It can easily be seen that for all choices of the nodes each row contains the same number of nonzero entries, i. e., all index sets (2.12) are of the same size of maximum $(2m + 1)^d$. However, when considering the columns instead, we recognize an evident mismatch in the number of nonzero entries. We remark that because of the fact that each row of $\boldsymbol{B} \in \mathbb{R}^{N \times \lvert \mathcal{I}_{\boldsymbol{M_\sigma}} \rvert}$ contains at most $(2m + 1)^d$ entries, each column contains $\frac{N}{\lvert \mathcal{I}_{\boldsymbol{M_\sigma}} \rvert} (2m + 1)^d$ entries on average. A general statement about the maximum size of the index sets (4.7) cannot be made. Roughly speaking, the more irregular the distribution of the nodes is, the larger the index sets (4.7) can be. Nevertheless, in general $\lvert \mathcal{I}_{\boldsymbol{M_\sigma}, m}(\boldsymbol{\ell}) \rvert$ is a small constant compared to $\lvert \mathcal{I}_{\boldsymbol{M}} \rvert$, such that Algorithm 4.3 ends up with total arithmetic costs of approximately $\mathcal{O}(\lvert \mathcal{I}_{\boldsymbol{M}} \rvert)$.

In conclusion, our approach for an inverse NFFT can be summarized as follows.

Algorithm 4.4 (iNFFT – optimization approach)**.**

For $d, N \in \mathbb{N}$ let $\boldsymbol{x}_j \in \mathbb{T}^d$, $j = 1, \dots, N$, be given nodes as well as $\boldsymbol{f} \in \mathbb{C}^N$. Further let $M \in 2\mathbb{N}$, $\boldsymbol{M} := M \cdot \mathbf{1}_d$, an oversampling factor $\sigma \geq 1$ with $2\mathbb{N} \ni M_\sigma := 2\lceil \lceil \sigma M \rceil / 2 \rceil$ and $\boldsymbol{M_\sigma} := M_\sigma \cdot \mathbf{1}_d$ as well as a truncation parameter $m \ll M_\sigma$ be given.

0. *Precompute the optimal sparse matrix $\boldsymbol{B}_{\mathrm{opt}}$ using Algorithm 4.3.*

1. *Compute $\boldsymbol{h}_{\mathrm{opt}} := \boldsymbol{D}^* \boldsymbol{F}^* \boldsymbol{B}_{\mathrm{opt}}^* \boldsymbol{f}$, cf. (4.12), by means of a modified adjoint NFFT.*

Output: $\boldsymbol{h}_{\mathrm{opt}} \approx \hat{\boldsymbol{f}} \in \mathbb{C}^{\lvert \mathcal{I}_{\boldsymbol{M}} \rvert}$, *cf. (3.4).* $\qquad\qquad$ **Complexity:** $\mathcal{O}(\lvert \mathcal{I}_{\boldsymbol{M}} \rvert \log(\lvert \mathcal{I}_{\boldsymbol{M}} \rvert) + N)$
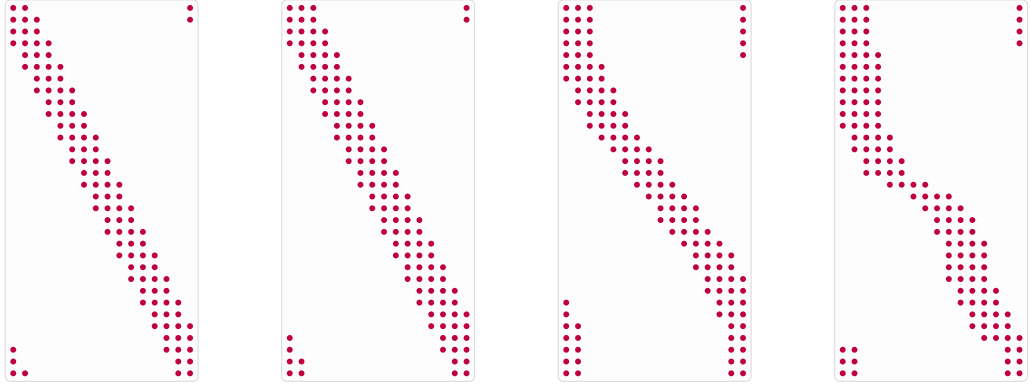
(a) Equispaced points (b) Jittered points (c) Chebyshev points (d) Random points

Figure 4.1: Nonzero entries of the matrix $\boldsymbol{B} \in \mathbb{R}^{N \times |\mathcal{I}_{M_{\sigma}}|}$ for several choices of the nodes $\boldsymbol{x}_j \in \mathbb{T}^d$, $j = 1, \ldots, N$, with $d = 1$, $M_{\sigma} = M = 16$, $N = 2M$ and $m = 2$.

Theorem 4.5. *Let $\boldsymbol{B}_{\mathrm{opt}} \in \mathbb{R}^{N \times |\mathcal{I}_{M_{\sigma}}|}$ be the optimized matrix computed by means of Algorithm 4.3 and let $\boldsymbol{h}_{\mathrm{opt}} = \boldsymbol{D}^* \boldsymbol{F}^* \boldsymbol{B}_{\mathrm{opt}}^* \boldsymbol{f} \in \mathbb{C}^{|\mathcal{I}_M|}$ be the corresponding approximation of $\hat{\boldsymbol{f}}$ computed by means of Algorithm 4.4. Further assume that each column $\boldsymbol{b}_{\boldsymbol{\ell}}^{\mathrm{opt}} \in \mathbb{R}^{|\mathcal{I}_{M_{\sigma},m}|}$ of $\boldsymbol{B}_{\mathrm{opt}} \in \mathbb{R}^{N \times |\mathcal{I}_{M_{\sigma}}|}$ as solution to (4.10) possesses a small residual*

$$\left\| \boldsymbol{H}_{\boldsymbol{\ell}} \boldsymbol{b}_{\boldsymbol{\ell}}^{\mathrm{opt}} - \tfrac{1}{|\mathcal{I}_{M_{\sigma}}|} \boldsymbol{D}^{-1} \boldsymbol{f}_{\boldsymbol{\ell}} \right\|_2^2 = \varepsilon_{\boldsymbol{\ell}} > 0, \quad \boldsymbol{\ell} \in \mathcal{I}_{M_{\sigma}}. \tag{4.15}$$

Then there exists an $\varepsilon > 0$ such that

$$\left\| \boldsymbol{h}_{\mathrm{opt}} - \hat{\boldsymbol{f}} \right\|_2^2 \leq \varepsilon \sum_{\boldsymbol{k} \in \mathcal{I}_M} \frac{1}{\hat{w}(\boldsymbol{k})^2} \cdot \left\| \hat{\boldsymbol{f}} \right\|_2^2. \tag{4.16}$$

Moreover, the (asymmetric) Dirichlet kernel

$$w_{\mathrm{D}} := \sum_{\boldsymbol{k} \in \mathcal{I}_M} \mathrm{e}^{2\pi \mathrm{i} \boldsymbol{k} \boldsymbol{x}} = \prod_{t=1}^d \left(D_{\frac{M}{2}-1}(x_t) + \mathrm{e}^{-M\pi \mathrm{i} x_t} \right) \tag{4.17}$$

is the optimal window function for the inverse NFFT in Algorithm 4.4.

Proof. As in (4.1) the approximation error can be estimated by

$$\left\| \boldsymbol{h}_{\mathrm{opt}} - \hat{\boldsymbol{f}} \right\|_2^2 = \left\| \left(\boldsymbol{D}^* \boldsymbol{F}^* \boldsymbol{B}_{\mathrm{opt}}^* \boldsymbol{A} - \boldsymbol{I}_{|\mathcal{I}_M|} \right) \hat{\boldsymbol{f}} \right\|_2^2 \leq \left\| \boldsymbol{D}^* \boldsymbol{F}^* \boldsymbol{B}_{\mathrm{opt}}^* \boldsymbol{A} - \boldsymbol{I}_{|\mathcal{I}_M|} \right\|_{\mathrm{F}}^2 \left\| \hat{\boldsymbol{f}} \right\|_2^2. \tag{4.18}$$

Using the same arguments as for (4.3) and (4.4) we proceed with

$$\left\| \boldsymbol{D}^* \boldsymbol{F}^* \boldsymbol{B}_{\mathrm{opt}}^* \boldsymbol{A} - \boldsymbol{I}_{|\mathcal{I}_M|} \right\|_{\mathrm{F}}^2 = \left\| \boldsymbol{A}^* \boldsymbol{B}_{\mathrm{opt}} \boldsymbol{F} \boldsymbol{D} - \boldsymbol{I}_{|\mathcal{I}_M|} \right\|_{\mathrm{F}}^2$$
$$\leq \left\| \boldsymbol{A}^* \boldsymbol{B}_{\mathrm{opt}} - \tfrac{1}{|\mathcal{I}_{M_{\sigma}}|} \boldsymbol{D}^{-1} \boldsymbol{F}^* \right\|_{\mathrm{F}}^2 \left\| \boldsymbol{F} \boldsymbol{D} \right\|_{\mathrm{F}}^2. \tag{4.19}$$

To estimate the first Frobenius norm in (4.19), we rewrite it analogously to (4.8) columnwise as

$$\left\| \boldsymbol{A}^* \boldsymbol{B}_{\mathrm{opt}} - \tfrac{1}{|\mathcal{I}_{\boldsymbol{M_\sigma}}|} \boldsymbol{D}^{-1} \boldsymbol{F}^* \right\|_{\mathrm{F}}^2 = \sum_{\boldsymbol{\ell} \in \mathcal{I}_{\boldsymbol{M_\sigma}}} \left\| \boldsymbol{H}_{\boldsymbol{\ell}} \boldsymbol{b}_{\boldsymbol{\ell}}^{\mathrm{opt}} - \tfrac{1}{|\mathcal{I}_{\boldsymbol{M_\sigma}}|} \boldsymbol{D}^{-1} \boldsymbol{f}_{\boldsymbol{\ell}} \right\|_2^2,$$

where $\boldsymbol{b}_{\boldsymbol{\ell}}^{\mathrm{opt}} \in \mathbb{R}^{|\mathcal{I}_{\boldsymbol{M_\sigma},m}(\boldsymbol{\ell})|}$ are the nonzeros of the columns of $\boldsymbol{B}_{\mathrm{opt}}$, $\boldsymbol{H}_{\boldsymbol{\ell}} \in \mathbb{C}^{|\mathcal{I}_{\boldsymbol{M}}| \times |\mathcal{I}_{\boldsymbol{M_\sigma},m}(\boldsymbol{\ell})|}$ in (4.9) are the corresponding submatrices of $\boldsymbol{A}^* \in \mathbb{C}^{|\mathcal{I}_{\boldsymbol{M}}| \times N}$, cf. (2.7), and $\boldsymbol{f}_{\boldsymbol{\ell}} \in \mathbb{C}^{|\mathcal{I}_{\boldsymbol{M}}|}$ are the columns of $\boldsymbol{F}^* \in \mathbb{C}^{|\mathcal{I}_{\boldsymbol{M}}| \times |\mathcal{I}_{\boldsymbol{M_\sigma}}|}$, cf. (2.14). Since $\boldsymbol{b}_{\boldsymbol{\ell}}^{\mathrm{opt}} \in \mathbb{R}^{|\mathcal{I}_{\boldsymbol{M_\sigma},m}|}$ as solutions to the least squares problems (4.10) satisfy (4.15), we can find some $\varepsilon > 0$ with $\varepsilon_{\boldsymbol{\ell}} \leq \varepsilon$, $\boldsymbol{\ell} \in \mathcal{I}_{\boldsymbol{M_\sigma}}$, such that

$$\sum_{\boldsymbol{\ell} \in \mathcal{I}_{\boldsymbol{M_\sigma}}} \left\| \boldsymbol{H}_{\boldsymbol{\ell}} \boldsymbol{b}_{\boldsymbol{\ell}}^{\mathrm{opt}} - \tfrac{1}{|\mathcal{I}_{\boldsymbol{M_\sigma}}|} \boldsymbol{D}^{-1} \boldsymbol{f}_{\boldsymbol{\ell}} \right\|_2^2 \leq \sum_{\boldsymbol{\ell} \in \mathcal{I}_{\boldsymbol{M_\sigma}}} \varepsilon_{\boldsymbol{\ell}} \leq \varepsilon \, |\mathcal{I}_{\boldsymbol{M_\sigma}}|.$$

Therefore, we may write (4.19) as

$$\left\| \boldsymbol{D}^* \boldsymbol{F}^* \boldsymbol{B}_{\mathrm{opt}}^* \boldsymbol{A} - \boldsymbol{I}_{|\mathcal{I}_{\boldsymbol{M}}|} \right\|_{\mathrm{F}}^2 \leq \varepsilon \, |\mathcal{I}_{\boldsymbol{M_\sigma}}| \cdot \left\| \boldsymbol{F} \boldsymbol{D} \right\|_{\mathrm{F}}^2. \tag{4.20}$$

Thus, it remains to estimate the Frobenius norm $\left\| \boldsymbol{F} \boldsymbol{D} \right\|_{\mathrm{F}}^2$. By the definitions of the Frobenius norm and the trace $\mathrm{tr}(\boldsymbol{Z})$ of a matrix $\boldsymbol{Z}$, it holds that $\|\boldsymbol{Z}\|_{\mathrm{F}}^2 = \mathrm{tr}(\boldsymbol{Z}^* \boldsymbol{Z})$. Since by (2.14) we have that $\boldsymbol{F}^* \boldsymbol{F} = |\mathcal{I}_{\boldsymbol{M_\sigma}}| \, \boldsymbol{I}_{|\mathcal{I}_{\boldsymbol{M}}|}$, this yields

$$\left\| \boldsymbol{F} \boldsymbol{D} \right\|_{\mathrm{F}}^2 = \mathrm{tr}(\boldsymbol{D}^* \boldsymbol{F}^* \boldsymbol{F} \boldsymbol{D}) = |\mathcal{I}_{\boldsymbol{M_\sigma}}| \cdot \mathrm{tr}(\boldsymbol{D}^* \boldsymbol{D}) = |\mathcal{I}_{\boldsymbol{M_\sigma}}| \cdot \left\| \boldsymbol{D} \right\|_{\mathrm{F}}^2. \tag{4.21}$$

Using the definition (2.13) of the diagonal matrix $\boldsymbol{D} \in \mathbb{R}^{|\mathcal{I}_{\boldsymbol{M}}| \times |\mathcal{I}_{\boldsymbol{M}}|}$, we obtain

$$\left\| \boldsymbol{D} \right\|_{\mathrm{F}}^2 = \frac{1}{|\mathcal{I}_{\boldsymbol{M_\sigma}}|^2} \sum_{\boldsymbol{k} \in \mathcal{I}_{\boldsymbol{M}}} \frac{1}{\hat{w}(\boldsymbol{k})^2}. \tag{4.22}$$

Then combination of (4.20), (4.21) and (4.22) implies

$$\left\| \boldsymbol{D}^* \boldsymbol{F}^* \boldsymbol{B}_{\mathrm{opt}}^* \boldsymbol{A} - \boldsymbol{I}_{|\mathcal{I}_{\boldsymbol{M}}|} \right\|_{\mathrm{F}}^2 \leq \varepsilon \sum_{\boldsymbol{k} \in \mathcal{I}_{\boldsymbol{M}}} \frac{1}{\hat{w}(\boldsymbol{k})^2},$$

such that (4.18) yields the assertion (4.16).

Since it is known that $0 \leq \hat{w}(\boldsymbol{k}) \leq 1$, $\boldsymbol{k} \in \mathcal{I}_{\boldsymbol{M}}$, for suitable window functions of the NFFT, cf. [55], we have

$$1 \leq \frac{1}{\hat{w}(\boldsymbol{k})} \leq \frac{1}{\hat{w}(\boldsymbol{k})^2}$$

and therefore

$$\sum_{\boldsymbol{k} \in \mathcal{I}_{\boldsymbol{M}}} \frac{1}{\hat{w}(\boldsymbol{k})^2} \geq \sum_{\boldsymbol{k} \in \mathcal{I}_{\boldsymbol{M}}} 1 = |\mathcal{I}_{\boldsymbol{M}}|.$$

Hence, the smallest constant is achieved in (4.16) when $\hat{w}(\boldsymbol{k}) = 1$, $\boldsymbol{k} \in \mathcal{I}_{\boldsymbol{M}}$, i.e., the (asymmetric) Dirichlet kernel (4.17) is the optimal window function for the inverse NFFT in Algorithm 4.4. ∎

Note that for trigonometric polynomials (2.8) the error bound of Theorem 4.5 with the optimal window function (4.17) is the same as the error bound (3.38) from Theorem 3.14.

Remark 4.6. *Up to now, we only focused on the problem (3.4). Finally, considering the inverse adjoint NFFT in (3.5), we remark that this problem can also be solved by means of the optimization procedure in Algorithm 4.3. Assuming again $|\mathcal{I}_M| \leq N$, this is the underdetermined setting for the adjoint problem (3.5). Therefore, the minimum norm solution of (3.5) is given by the normal equations of second kind*

$$A^*Ay = h, \quad f = Ay.$$

Incorporating the matrix decomposition of the NFFT, cf. Section 2.1, we recognize that a modification of the matrix $B \in \mathbb{R}^{N \times |\mathcal{I}_{M_\sigma}|}$ such that $A^*BFD \approx I_{|\mathcal{I}_M|}$ implies $y \approx h$ and hence $f \approx BFDh$. Thus, the optimization problem (4.3) is also the one to consider for (3.5). In other words, our approach provides both, an inverse NFFT as well as an inverse adjoint NFFT. □

5 Numerics

Concluding, we have a look at some numerical examples. Besides comparing the density compensation approach from Section 3 to the optimization approach from Section 4, for both trigonometric polynomials (2.8) and bandlimited functions, we also demonstrate the accuracy of these approaches.

Remark 5.1. *At first we introduce some exemplary grids. For visualization we restrict ourselves to the two-dimensional setting $d = 2$.*

(i) *We start with a sampling scheme that is somehow "close" to the Cartesian grid, but also possesses a random part. To this end, we consider the two-dimensional Cartesian grid and add a two-dimensional perturbation, i. e.,*

$$x_{t,j} := \left(-\frac{1}{2} + \frac{2t-1}{N_1}, -\frac{1}{2} + \frac{2j-1}{N_2}\right)^T + \left(\frac{1}{N_1}\eta_1, \frac{1}{N_2}\eta_2\right)^T, \quad (5.1)$$

$t = 1, \ldots, N_1$, $j = 1, \ldots, N_2$, and $\eta_1, \eta_2 \sim U(-1,1)$, where $U(-1,1)$ denotes the uniform distribution on the interval $(-1,1)$. A visualization of this jittered grid can be found in Figure 5.1a. Additionally, we also consider the random grid $x_{t,j} := \frac{1}{2}(\eta_1, \eta_2)^T$, see Figure 5.1b.

(ii) *Moreover, we examine grids of polar kind, as mentioned in [23]. For $R, T \in 2\mathbb{N}$ the points of the polar grid are given by a signed radius $r_j := \frac{j}{R} \in \left[-\frac{1}{2}, \frac{1}{2}\right)$ and an angle $\theta_t := \frac{\pi t}{T} \in \left[-\frac{\pi}{2}, \frac{\pi}{2}\right)$ as*

$$x_{t,j} := r_j (\cos\theta_t, \sin\theta_t)^T, \quad (j,t)^T \in \mathcal{I}_R \times \mathcal{I}_T. \quad (5.2)$$

Since it is known that the inversion problem is ill-conditioned for this grid we consider a modification, the modified polar grid

$$x_{t,j} := r_j (\cos\theta_t, \sin\theta_t)^T, \quad (j,t)^T \in \mathcal{I}_{\sqrt{2}R} \times \mathcal{I}_T, \quad (5.3)$$

(a) Jittered grid   (b) Random grid
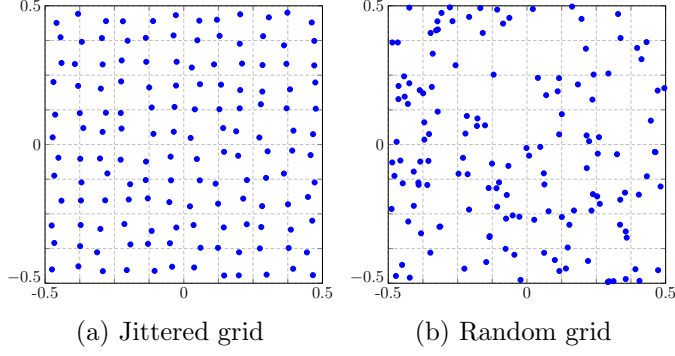
Figure 5.1: Exemplary randomized grids of size $N_1 = N_2 = 12$.

i. e., we added more concentric circles and excluded the points outside the unit square, see Figure 5.2a. Another sampling scheme which is known to lead to more stable results than the polar grid is the linogram or pseudo-polar grid, where the nodes lie on concentric squares instead of concentric circles, see Figure 5.2b. There we distinguish two sets of nodes, i. e.,

$$x_{t,j}^{BH} := \left(\frac{j}{R}, \frac{4t}{T}\frac{j}{R} \right)^T, \quad x_{t,j}^{BV} := \left(-\frac{4t}{T}\frac{j}{R}, \frac{j}{R} \right)^T, \quad (j,t)^T \in \mathcal{I}_R \times \mathcal{I}_{\frac{T}{2}}. \quad (5.4)$$

(a) Polar (blue) and   (b) Linogram /   (c) Golden angle linogram
modified polar (red) grid   pseudo-polar grid   grid
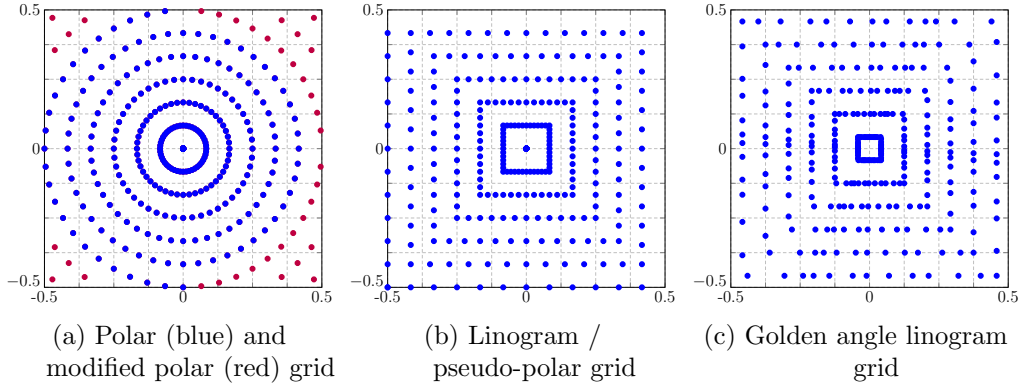
Figure 5.2: Polar grids of size $R = 12$ and $T = 2R$.

(iii) *Another modification of these polar grids was introduced in [33], where the angles θ_t are not chosen equidistantly but are obtained by golden angle increments. For the golden angle polar grid we only exchange the equispaced angles of the polar grid to*

$$\theta_t = \mod\left(\frac{\pi}{2} + t\frac{2\pi}{1+\sqrt{5}}, \pi \right) - \frac{\pi}{2}, \quad t = 0, \dots, T-1. \quad (5.5)$$

The golden angle linogram grid is given by

$$x_{t,j} := \begin{cases} \left(\frac{2j+1}{2R}, \frac{2j+1}{2R} \tan\left(\theta_t - \frac{\pi}{4}\right) \right)^T & : \theta_t \in \left[0, \frac{\pi}{2}\right) \\ \left(-\frac{2j+1}{2R} \cot\left(\theta_t - \frac{\pi}{4}\right), \frac{2j+1}{2R} \right)^T & : \theta_t \in \left[-\frac{\pi}{2}, 0\right) \end{cases}, \quad j \in \mathcal{I}_R,$$

with θ_t in (5.5), as illustrated in Figure 5.2c. $\quad\square$

Before comparing the different approaches from Sections 3 and 4, we study the quality of our methods for the grids mentioned in Remark 5.1. More specifically, in Example 5.2 we investigate the accuracy of the density compensation method from Algorithm 3.7 with the weights introduced in Section 3.1, and in Example 5.3 we check if the norm minimization targeted in Section 4 is successful.

Example 5.2. *Firstly, we examine the quality of our density compensation method in Algorithm 3.7 for a trigonometric polynomial f as in (2.8) with given Fourier coefficients $\hat{f}_{\boldsymbol{k}} \in [1, 10]$, $\boldsymbol{k} \in \mathcal{I}_{\boldsymbol{M}}$. For the corresponding function evaluations of (2.8) at given points $\boldsymbol{x}_j \in \left[-\frac{1}{2}, \frac{1}{2}\right)^2$, $j = 1, \ldots, N$, we test how well these Fourier coefficients can be approximated. More precisely, we consider the estimate $\tilde{\boldsymbol{h}}^{\mathrm{w}} = \boldsymbol{D}^* \boldsymbol{F}^* \boldsymbol{B}^* \boldsymbol{W} \boldsymbol{f}$, cf. (3.7), with the matrix $\boldsymbol{W} = \mathrm{diag}\,(w_j)_{j=1}^N$ of density compensation factors computed by means of (3.15), in case $|\mathcal{I}_{2\boldsymbol{M}}| \leq N$, or by (3.16), if $|\mathcal{I}_{2\boldsymbol{M}}| > N$, and compute the relative errors*

$$e_r := \frac{\|\tilde{\boldsymbol{h}}^{\mathrm{w}} - \hat{\boldsymbol{f}}\|_r}{\|\hat{\boldsymbol{f}}\|_r}, \quad r \in \{2, \infty\}. \tag{5.6}$$

In our experiment we firstly choose random nodes $\boldsymbol{x}_j$ with $N_1 = N_2 = 512$, cf. Figure 5.1b, and then, for several problem sizes $\boldsymbol{M} = (M, M)^T$, $M = 2^c$ with $c = 1, \ldots, 11$, we choose random Fourier coefficients $\hat{f}_{\boldsymbol{k}} \in [1, 10]$, $\boldsymbol{k} \in \mathcal{I}_{\boldsymbol{M}}$. Afterwards, we compute the evaluations of the trigonometric polynomial (2.8) by means of an NFFT and use the resulting vector $\boldsymbol{f}$ as input for the reconstruction. Due to the randomness we repeat this 10 times and then consider the mean error over all runs. The corresponding results are displayed in Figure 5.3. It can clearly be seen that up to the point where we reach $|\mathcal{I}_{2\boldsymbol{M}}| = N$, i.e., when $M = \frac{N_1}{4} = 2^7$, the weights computed by means of (3.15) lead to an exact reconstruction of the given Fourier coefficients. However, as soon as we are in the setting $|\mathcal{I}_{2\boldsymbol{M}}| > N$ the least squares approximation via (3.16) does not yield good results anymore. $\quad\square$

Example 5.3. *For studying the quality of our optimization method in Section 4, we consider the Frobenius norms*

$$n(w, m, \sigma) := \left\| \boldsymbol{A}^* \boldsymbol{B} \boldsymbol{F} \boldsymbol{D} - \boldsymbol{I}_{|\mathcal{I}_{\boldsymbol{M}}|} \right\|_{\mathrm{F}}, \quad n^{opt}(w, m, \sigma) := \left\| \boldsymbol{A}^* \boldsymbol{B}_{opt} \boldsymbol{F} \boldsymbol{D} - \boldsymbol{I}_{|\mathcal{I}_{\boldsymbol{M}}|} \right\|_{\mathrm{F}}, \tag{5.7}$$

where $\boldsymbol{B}$ denotes the original matrix from the NFFT in (2.15) and $\boldsymbol{B}_{\mathrm{opt}}$ the optimized matrix generated by Algorithm 4.3. For the original matrix we utilize the common B-Spline window function

$$w_{\mathrm{B}} := B_{2m}(\boldsymbol{M}_{\boldsymbol{\sigma}} \boldsymbol{x}) \tag{5.8}$$

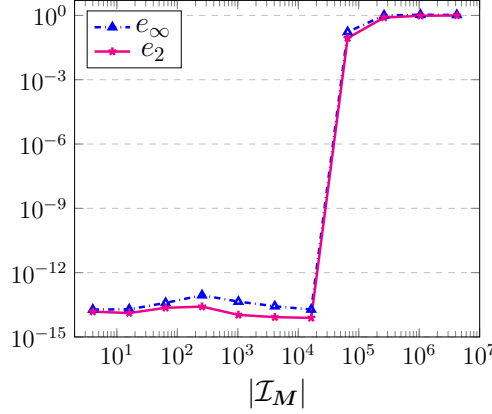Figure 5.3: Relative errors (5.6) of the reconstruction of the Fourier coefficients of a trigonometric polynomial (2.8) with given $\hat{f}_{\boldsymbol{k}} \in [1, 10]$, $\boldsymbol{k} \in \mathcal{I}_{\boldsymbol{M}}$, computed via the density compensation method from Algorithm 3.7, for random grids with $N_1 = N_2 = 512$ and $\boldsymbol{M} = (M, M)^T$, $M = 2^c$ with $c = 1, \ldots, 11$.

with the centered B-Spline of order $2m$, cf. [53, p. 388], and compare it to the optimized matrix computed by means of the B-Spline (5.8) as well as the Dirichlet window function (4.17), which is the optimal window by Theorem 4.5. For the grids mentioned in Remark 5.1 we choose $N_1 = N_2 = R = 2^{\mu}$, $\mu \in \{2, \ldots, 7\}$, and $T = 2R$ as well as $\boldsymbol{M} = (12, 12)^T$. In other words, we test Algorithm 4.3 in the underdetermined setting $|\mathcal{I}_{\boldsymbol{M}}| > N$ as well as for the overdetermined setting $|\mathcal{I}_{\boldsymbol{M}}| \leq N$. Indeed, these are very small problems due to memory limitations in the computation of the Frobenius norm, nevertheless they show the functionality of Algorithm 4.3. For larger examples we refer to the remainder of this section.

Having a look at the results for the grids in Remark 5.1, it becomes apparent that they separate into two groups. Figure 5.4a displays the results for the polar grid (5.2), which are the same as for the golden angle polar grid, cf. (5.5). In these cases there is only a slight improvement by the optimization. However, for all other mentioned grids the minimization procedure in Algorithm 4.3 is very effective. The results for these grids are depicted in Figure 5.4b exemplarily for the modified polar grid (5.3). Moreover, it can be seen that our optimization procedure in Algorithm 4.3 is most effective in the overdetermined setting $|\mathcal{I}_{\boldsymbol{M}}| \leq N$.

One reason for the different behavior of polar and modified polar grid could be the ill-posedness of the inversion problem for the polar grid, which becomes evident in huge condition numbers of $\boldsymbol{H}_{\boldsymbol{\ell}}^* \boldsymbol{H}_{\boldsymbol{\ell}}$, whereas the problem for modified polar grids is well-posed. Another reason can be found in the optimization procedure itself. Having a closer look at the polar grid, see Figure 5.2a, there are no grid points in the corners of the unit square. Therefore, some of the index sets $\mathcal{I}_{\boldsymbol{M}_{\boldsymbol{\sigma}}, m}(\boldsymbol{\ell})$, cf. (4.7), are empty and no optimization can be done for the corresponding matrix columns. This could also cause the worsened minimization properties of the polar grid. □

(a) Polar grid (5.2) (b) Modified polar grid (5.3)
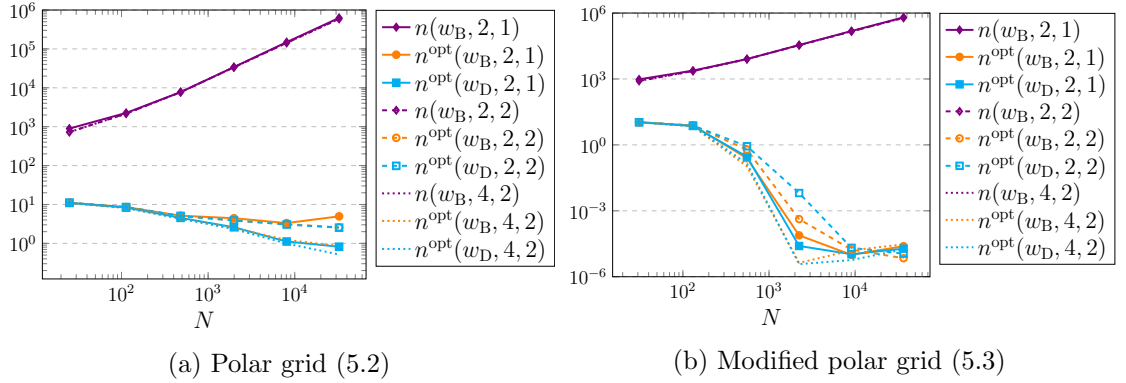
Figure 5.4: Frobenius norms (5.7) of the original matrix $\boldsymbol{B}$ (violet) and the optimized matrix $\boldsymbol{B}_{\mathrm{opt}}$ generated by Algorithm 4.3 using the B-Spline w_{B} (orange) as well as the Dirichlet window w_{D} (cyan) with $R = 2^{\mu}$, $\mu \in \{2, \ldots, 7\}$, and $T = 2R$ as well as $\boldsymbol{M} = (12, 12)^T$, $m \in \{2, 4\}$ and $\sigma \in \{1, 2\}$.

Next, we proceed with comparing the density compensation approach from Section 3 using the weights w_j introduced in Section 3.1 to the optimization approach for modifying the matrix $\boldsymbol{B}$ from Section 4. To this end, we show an example concerning trigonometric polynomials (2.8) of degree $\boldsymbol{M}$ and a second one that deals with bandlimited functions of bandwidth $\boldsymbol{M}$.

Example 5.4. *Similar to [3, 38] we have a look at the reconstruction of the Shepp-Logan phantom, see Figure 5.5a. Here we treat the phantom data as given Fourier coefficients $\hat{\boldsymbol{f}} := (\hat{f}_{\boldsymbol{k}})_{\boldsymbol{k} \in \mathcal{I}_{\boldsymbol{M}}}$ of a trigonometric polynomial (2.8). For given points $\boldsymbol{x}_j \in \left[-\frac{1}{2}, \frac{1}{2}\right)^2$, $j = 1, \ldots, N$, we then compute the evaluations of the trigonometric polynomial (2.8) by means of an NFFT and use the resulting vector as input for the reconstruction.*

In a first experiment, we test the inversion methods from Sections 3 and 4 as in [3] for increasing input sizes. To this end, we choose $\boldsymbol{M} = (M, M)^T$, $M = 2^c$ with $c = 3, \ldots, 9$, and linogram grids (5.4) of size $R = 2M$, $T = 2R$, i.e., we have $|\mathcal{I}_{2\boldsymbol{M}}| < N$. For using Algorithm 4.4 we choose the oversampling factor $\sigma = 1.0$ and the truncation parameter $m = 4$. For each input size we measure the computation time of the precomputational steps, i.e., the computation of the weight matrix $\boldsymbol{W}$ or the computation of the optimized sparse matrix $\boldsymbol{B}_{\mathrm{opt}}$, as well as the time needed for the reconstruction, i.e., the corresponding adjoint NFFT, see Algorithms 3.7 and 4.4. Moreover, for the reconstruction $\tilde{\boldsymbol{h}} \in \{\tilde{\boldsymbol{h}}^{\mathrm{w}}, \boldsymbol{h}_{\mathrm{opt}}\}$, cf. (3.7) and (4.12), we consider the relative errors

$$e_2 := \frac{\|\tilde{\boldsymbol{h}} - \hat{\boldsymbol{f}}\|_2}{\|\hat{\boldsymbol{f}}\|_2}. \tag{5.9}$$

The corresponding results can be found in Table 5.1. We remark that since we are in the setting $|\mathcal{I}_{2\boldsymbol{M}}| < N$, the density compensation method in Algorithm 3.7 with weights computed by (3.15) indeed produces nearly exact results. Although, our optimization

procedure from Algorithm 4.4 achieves small errors as well, this reconstruction is not as good as the one by means of our density compensation method.

Note that in comparison to [3] our method in Algorithm 3.7 using density compensation produces errors of the same order, but is much more effective for solving several problems using the same nodes $\boldsymbol{x}_j$ for different input values $\boldsymbol{f}$. Since in our case the precomputations have to be done only once in this setting, we strongly profit from the fact that as reconstruction we only need to perform an adjoint NFFT, which is very fast, whereas in [3] they would need to execute their whole method each time again.

	Relative error e_2		Precomputation time		Reconstruction time	
M	Alg. 3.7	Alg. 4.4	Alg. 3.7	Alg. 4.4	Alg. 3.7	Alg. 4.4
8	1.6655e-15	4.4864e-14	2.3736e-02	1.2092e+00	9.2000e-04	2.2000e-04
16	7.0937e-15	1.7715e-07	6.7233e-02	6.8401e+00	8.0200e-04	4.2800e-04
32	2.3645e-14	2.7870e-07	4.1536e-01	4.1215e+01	4.0310e-03	1.0960e-03
64	3.2201e-14	3.2994e-07	1.9256e+00	2.4294e+02	1.2215e-02	3.5880e-03
128	7.6913e-14	4.5987e-07	1.1164e+01	1.3798e+03	3.3879e-02	1.4178e-02
256	3.6838e-13	1.5145e-06	6.4293e+01	8.0902e+03	1.6462e-01	6.0529e-02
512	9.9041e-13	1.6428e-06	4.1066e+02	1.0722e+05	5.5705e-01	5.7574e+01

Table 5.1: Relative errors (5.9) of the reconstruction of the Shepp-Logan phantom of size M as well as the runtime in seconds for the density compensation method from Algorithm 3.7 compared to Algorithm 4.4 with $\sigma = 1.0$ and $m = 4$, using linogram grids (5.4) of size $R = 2M$, $T = 2R$.

As a second experiment we aim to decrease the amount overdetermination, i. e., we want to keep the size $|\mathcal{I}_M|$ of the phantom, but reduce the number N of the points $\boldsymbol{x}_j$, $j = 1, \ldots, N$. To this end, we now consider linogram grids (5.4) of the smaller size $R = M$, $T = 2R$, i. e., now we have $|\mathcal{I}_{2M}| > N$. The reconstruction of the phantom of size 1024×1024 is presented in Figure 5.5 (top) including a detailed view of the 832nd row of this reconstruction (bottom). Despite the reconstruction via Algorithm 4.4 as well as the density compensation method using weights computed by means of (3.16), we also considered the result using Voronoi weights. For all approaches we added the corresponding relative errors (5.9) to Figure 5.5 as well.

Due to the fact that the exactness condition $|\mathcal{I}_{2M}| < N$ (cf. Section 3.1.1) is violated, it can be seen in Figure 5.5c that the density compensation method using weights computed by means of (3.16) does not yield an exact reconstruction is this setting. On the contrary, we recognize that our optimization method, see Figure 5.5d, achieves a huge improvement in comparison to the density compensation techniques in Figure 5.5b and 5.5c since no artifacts are visible. Presumably, this arises because there are more degrees of freedom in the optimization of the matrix $\boldsymbol{B}$ for method from Section 4 than with the density compensation techniques from Section 3, cf. Remark 3.1. We remark that although the errors are not as small as in Table 5.1, by comparing Figure 5.5a and 5.5d it becomes apparent that the differences are not even visible anymore. Note that for this result the number N of nodes is ca. 4 times lower as for the results in depicted in Table 5.1, i. e.,

we only needed twice as much function values as Fourier coefficients, whereas e.g. in [3] they worked with a factor of more than 4. □
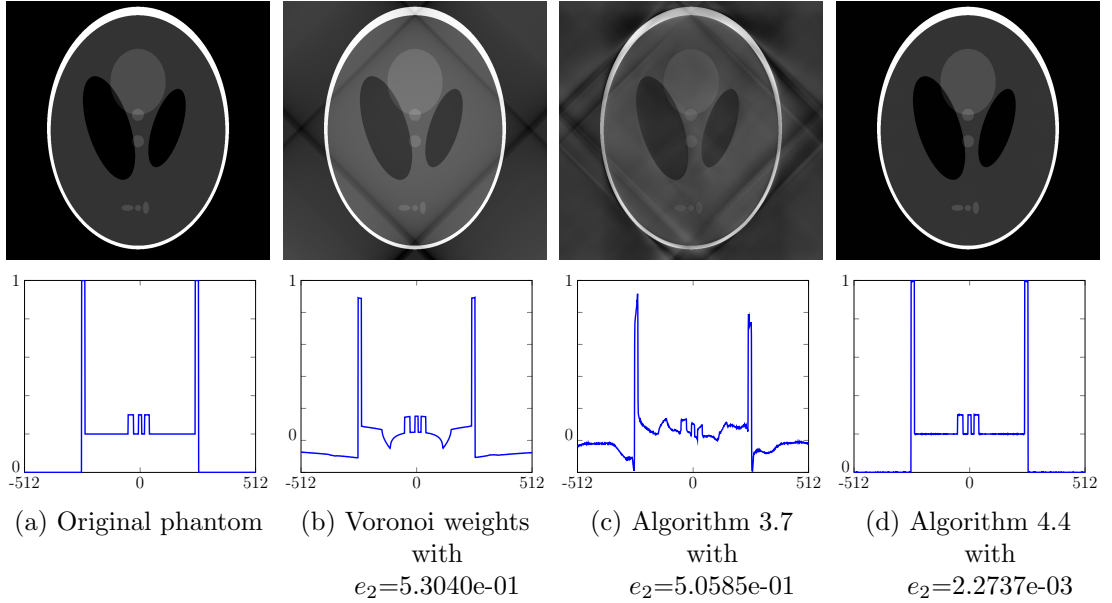

(a) Original phantom

(b) Voronoi weights with $e_2=$5.3040e-01

(c) Algorithm 3.7 with $e_2=$5.0585e-01

(d) Algorithm 4.4 with $e_2=$2.2737e-03

Figure 5.5: Reconstruction of the Shepp-Logan phantom of size 1024×1024 (top) via the density compensation method from Section 3.1 using Voronoi weights and Algorithm 3.7 with weights computed by (3.16) compared to Algorithm 4.4 for the linogram grid (5.4) of size $R = M = 1024$, $T = 2R$; as well as a detailed view of the 832nd row each (bottom).

Example 5.5. *Finally, we examine the reconstruction properties for bandlimited functions $f \in L_1(\mathbb{R}^d) \cap C_0(\mathbb{R}^d)$ with maximum bandwidth $\boldsymbol{M}$. To this end, we firstly specify a compactly supported function $\hat{f}$ and consequently compute its inverse Fourier transform f, such that its samples $f(\boldsymbol{x}_j)$ for given $\boldsymbol{x}_j \in \left[-\frac{1}{2}, \frac{1}{2}\right]^2$, $j = 1, \dots, N$, can be used for the reconstruction of the samples $\hat{f}(\boldsymbol{k})$, $\boldsymbol{k} \in \mathcal{I}_{\boldsymbol{M}}$. Here we consider the tensorized function $\hat{f}(\boldsymbol{v}) = g(v_1) \cdot g(v_2)$, where $g(v)$ is the one-dimensional triangular pulse $g(v) := (1 - \left|\frac{v}{b}\right|) \cdot \chi_{[-b,b]}(v)$. Then for all $b \in \mathbb{N}$ with $b \leq \frac{M}{2}$ the associated inverse Fourier transform*

$$f(\boldsymbol{x}) = \int_{\mathbb{R}^2} \hat{f}(\boldsymbol{v}) \, e^{2\pi i \boldsymbol{v} \boldsymbol{x}} \, d\boldsymbol{v} = b^2 \operatorname{sinc}^2(b\pi\boldsymbol{x}) = b^2 \operatorname{sinc}^2(b\pi x_1) \operatorname{sinc}^2(b\pi x_2), \quad \boldsymbol{x} \in \mathbb{R}^2,$$

is bandlimited with bandwidth $\boldsymbol{M}$. In this case, we consider $M = 64$ and $b = 24$ as well as the jittered grid (5.1) of size $N_1 = N_2 = 144$, i.e., we study the setting $|\mathcal{I}_{\boldsymbol{2M}}| \leq N$.

Now the aim is comparing the different density compensation methods considered in Section 3 and the optimization approach from Section 4. More precisely, we consider the reconstruction using Voronoi weights, the weights computed via (3.59), the weights in

(3.63) *and Algorithm 3.7 with weights computed via* (3.15), *as well as Algorithm 4.4. For the reconstruction* $\tilde{\boldsymbol{h}} \in \{\tilde{\boldsymbol{h}}^{\mathrm{w}}, \boldsymbol{h}_{\mathrm{opt}}\}$, *cf.* (3.7) *and* (4.12), *we then compute the pointwise absolute errors* $\left|\tilde{\boldsymbol{h}} - \hat{\boldsymbol{f}}\right|$. *The corresponding results are displayed in Figure 5.6. It can easily be seen that Voronoi weights, see Figure 5.6a, and the weights in* (3.63), *see Figure 5.6d, do not yield a good reconstruction, as expected. The other three approaches produce nearly the same reconstruction error, which is also obtained by reconstruction on an equispaced grid and therefore is the best possible. In other words, in case of bandlimited functions the truncation error in* (3.20) *is dominating and thus reconstruction errors smaller than the ones shown in Figure 5.6 cannot be expected.*

Note that the comparatively small choice of $M = 64$ *was made in order that the computation of the weights via* (3.59), *see Figure 5.6c, as well as the weights in* (3.63), *see Figure 5.6d, is affordable, cf. Section 3.4.2. In contrast, our new methods using Algorithm 3.7, see Figure 5.6b, or Algorithm 4.4, see Figure 5.6e, are much more effective and therefore better suited for the given problem.* □

(a) Voronoi weights   (b) Algorithm 3.7   (c) Weights via (3.59)

(d) Weights (3.63)   (e) Algorithm 4.4
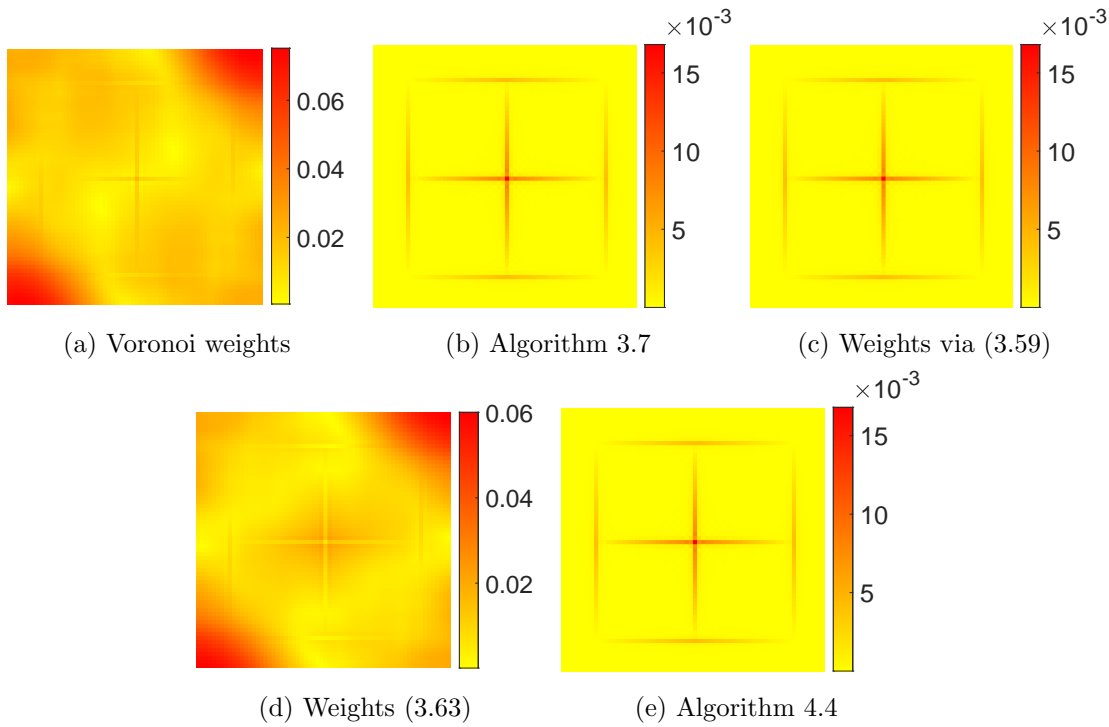
Figure 5.6: Pointwise absolute error $\left|\tilde{\boldsymbol{h}} - \hat{\boldsymbol{f}}\right|$ of the reconstruction of samples $\hat{f}(\boldsymbol{k})$ of the tensorized triangular pulse function with $M = 64$ and $b = 24$, using the density compensation methods considered in Section 3 as well as the optimization approach from Algorithm 4.4 for the jittered grid (5.1) of size $N_1 = N_2 = 144$.

6 Conclusion

In the present paper we considered several direct methods for computing an inverse NFFT, i.e., reconstructing the Fourier coefficients $\hat{f}_{\boldsymbol{k}}$, $\boldsymbol{k} \in \mathcal{I}_{\boldsymbol{M}}$, from given nonequispaced data $f(\boldsymbol{x}_j)$, $j = 1, \ldots, N$. Being a direct method here means, that for a fixed set of points $\boldsymbol{x}_j$, $j = 1, \ldots, N$, the reconstruction can be realized with the same number of arithmetic operations as a single application of an adjoint NFFT (see Algorithm 2.4). As we have seen in (3.3), a certain precomputational step is compulsory, since the adjoint NFFT does not yield an inversion by itself. Although this precomputations might be rather costly, they need to be done only once for a given set of points $\boldsymbol{x}_j$, $j = 1, \ldots, N$. Therefore, direct methods are especially beneficial in case of fixed nodes. For this reason, we studied two different approaches of this kind and especially focused on methods for the multidimensional setting $d \geq 1$ that are applicable for general sampling patterns.

Firstly, in Section 3 we examined the well known approach of sampling density compensation, where suitable weights $w_j \in \mathbb{C}$, $j = 1, \ldots, N$, are precomputed, such that the reconstruction can be realized by means of an adjoint NFFT applied to the scaled data $w_j f(\boldsymbol{x}_j)$. We started our investigations with trigonometric polynomials in Section 3.1. In Corollary 3.4 we introduced the main formula (3.14), that yields exact reconstruction for all trigonometric polynomials of degree $\boldsymbol{M}$. In addition to this theoretical considerations, we also discussed practical computation schemes for the overdetermined as well as the underdetermined setting, as summarized in Algorithm 3.6. Afterwards, in Section 3.2 we studied the case of bandlimited functions, which often occurs in the context of MRI, and discussed that the same numerical procedures as in Section 3.1 can be used in this setting as well. In Section 3.3 we then summarized the previous findings by presenting a general error bound on density compensation factors computed by means of Algorithm 3.6 in Theorem 3.14. In addition, this also yields an estimate on the condition number of the matrix product $\boldsymbol{A}^* \boldsymbol{W} \boldsymbol{A}$, as shown in Theorem 3.15. In Section 3.4 we surveyed certain approaches from literature and commented on their connection among each other as well as to the method presented in Section 3.1.

Subsequently, in Section 4 we studied another direct inversion method, where the matrix representation $\boldsymbol{A} \approx \boldsymbol{B} \boldsymbol{F} \boldsymbol{D}$ of the NFFT is used to modify the sparse matrix $\boldsymbol{B}$, such that a reconstruction is given by $\hat{\boldsymbol{f}} \approx \boldsymbol{D}^* \boldsymbol{F}^* \boldsymbol{B}^*_{\text{opt}} \boldsymbol{f}$. In other words, the inversion is done by a modified adjoint NFFT, while the optimization of the matrix $\boldsymbol{B}$ can be realized in a precomputational step, see Algorithm 4.4.

Finally, in Section 5 we had a look at some numerical examples to investigate the accuracy of the previously introduced methods. We have seen that our approaches are best-suited for the overdetermined setting $|\mathcal{I}_{\boldsymbol{M}}| \leq N$ and work for many different sampling patterns. More specifically, in the highly overdetermined case $|\mathcal{I}_{\boldsymbol{2M}}| \leq N$ we have theoretically proven as well as numerically verified in several examples that the density compensation technique in Algorithm 3.7 leads to an exact reconstruction for trigonometric polynomials. In case not that much data is available and we have to reduce the amount of overdetermination such that $|\mathcal{I}_{\boldsymbol{2M}}| > N$, we have shown that the optimization approach from Algorithm 4.4 is preferable, since the higher number of degrees of freedom in the optimization (see Remark 3.1) yields better results. In addition,

also for the setting of bandlimited functions we demonstrated that our methods are much more efficient than existing ones.

Acknowledgments

References

[1] R. Archibald, A. Gelb, and R. B. Platte. Image reconstruction from undersampled Fourier data using the polynomial annihilation transform. *J. Sci. Comput.*, 67(2):432–452, 2016.

[2] R. Archibald, A. Gelb, and J. Yoon. Polynomial fitting for edge detection in irregularly sampled signals and images. *SINUM*, 43(1):259–279, 2005.

[3] A. Averbuch, R. Coifman, D. Donoho, M. Israeli, and Y. Shkolnisky. A framework for discrete integral transformations I – the pseudopolar Fourier transform. *SIAM J. Sci. Comput.*, 30:764–784, 2008.

[4] A. Averbuch, R. Coifman, D. L. Donoho, M. Elad, and M. Israeli. Fast and accurate polar Fourier transform. *Appl. Comput. Harmon. Anal.*, 21:145–167, 2006.

[5] A. Averbuch, G. Shabat, and Y. Shkolnisky. Direct inversion of the three-dimensional pseudo-polar Fourier transform. *SIAM J. Sci. Comput.*, 38(2):A1100–A1120, 2016.

[6] R. F. Bass and K. Gröchenig. Random sampling of multivariate trigonometric polynomials. *SIAM J. Math. Anal.*, 36:773–795, 2004.

[7] G. Beylkin. On the fast Fourier transform of functions with singularities. *Appl. Comput. Harmon. Anal.*, 2:363–381, 1995.

[8] Å. Björck. *Numerical Methods for Least Squares Problems.* SIAM, Philadelphia, PA, USA, 1996.

[9] A. Böttcher and D. Potts. Probability against condition number and sampling of multivariate trigonometric random polynomials. *Electron. Trans. Numer. Anal.*, 26:178–189, 2007.

[10] E. J. Candès, M. B. Wakin, and S. P. Boyd. Enhancing sparsity by reweighted ℓ_1 minimization. *J. Fourier Anal. Appl.*, 14(5):877–905, 2008.

[11] R. Chartrand and W. Yin. Iteratively reweighted algorithms for compressive sensing. In *IEEE International Conference on Acoustics, Speech and Signal Processing*, pages 3869–3872. IEEE, 2008.

[12] H. Choi and D. C. Munson. Analysis and design of minimax-optimal interpolators. *IEEE Trans. Signal Process.*, 46(6):1571–1579, 1998.

[13] V. Churchill, R. Archibald, and A. Gelb. Edge-adaptive ℓ_2 regularization image reconstruction from non-uniform Fourier data. *Inverse Probl. Imaging*, 13(5):931–958, 2019.

[14] I. Daubechies, R. DeVore, M. Fornasier, and C. S. Güntürk. Iteratively reweighted least squares minimization for sparse recovery. *Commun. Pure Appl. Math*, 63(1):1–38, 2010.

[15] M. Doneva, M. Akcakaya, and C. Prieto, editors. *Magnetic Resonance Image Reconstruction: Theory, Methods and Applications*, volume 6. Academic Press, 2022.

[16] A. J. W. Duijndam and M. A. Schonewille. Nonuniform fast Fourier transform. *Geophysics*, 64:539–551, 1999.

[17] A. Dutt and V. Rokhlin. Fast Fourier transforms for nonequispaced data. *SIAM J. Sci. Stat. Comput.*, 14:1368–1393, 1993.

[18] A. Dutt and V. Rokhlin. Fast Fourier transforms for nonequispaced data II. *Appl. Comput. Harmon. Anal.*, 2:85–100, 1995.

[19] H. Eggers, M. Kircheis, and D. Potts. Non-Cartesian MRI reconstruction. In M. Doneva, M. Akcakaya, and C. Prieto, editors, *Magnetic Resonance Image Reconstruction: Theory, Methods and Applications*, volume 6. Academic Press, 2022.

[20] B. Elbel and G. Steidl. Fast Fourier transform for nonequispaced data. In C. K. Chui and L. L. Schumaker, editors, *Approximation Theory IX*, pages 39–46, Nashville, 1998. Vanderbilt University Press.

[21] G. E. Fasshauer. *Meshfree approximation methods with MATLAB*. World Scientific Publishers, 2007.

[22] H. G. Feichtinger, K. Gröchenig, and T. Strohmer. Efficient numerical methods in non-uniform sampling theory. *Numer. Math.*, 69:423–440, 1995.

[23] M. Fenn, S. Kunis, and D. Potts. On the computation of the polar FFT. *Appl. Comput. Harmon. Anal.*, 22:257–263, 2007.

[24] J. A. Fessler and B. P. Sutton. Nonuniform fast Fourier transforms using min-max interpolation. *IEEE Trans. Signal Process.*, 51:560–574, 2003.

[25] K. Fourmont. Non equispaced fast Fourier transforms with applications to tomography. *J. Fourier Anal. Appl.*, 9:431–450, 2003.

[26] A. Gelb and G. Song. A frame theoretic approach to the nonuniform fast Fourier transform. *SIAM J. Numer. Anal.*, 52(3):1222–1242, 2014.

[27] A. Gopal and V. Rokhlin. A fast procedure for the construction of quadrature formulas for bandlimited functions. *Technical Report*, YALEU/DCS/TR-1563, 2022.

[28] M. Gräf, S. Kunis, and D. Potts. On the computation of nonnegative quadrature weights on the sphere. *Appl. Comput. Harmon. Anal.*, 27:124–132, 2009.

[29] L. Greengard and J.-Y. Lee. Accelerating the nonuniform fast Fourier transform. *SIAM Rev.*, 46:443–454, 2004.

[30] L. Greengard, J.-Y. Lee, and S. Inati. The fast sinc transform and image reconstruction from nonuniform samples in k-space. *Commun. Appl. Math. Comput. Sci.*, 1:121–131, 2006.

[31] K. Gröchenig. Sampling, Marcinkiewicz–Zygmund inequalities, approximation, and quadrature rules. *J. Approx. Theory*, 257:105455, 2020.

[32] G. Heinig and K. Rost. *Algebraic methods for Toeplitz-like matrices and operators*, volume 19 of *Mathematical Research*. Akademie-Verlag, Berlin, 1984.

[33] E. S. Helou et al. The discrete Fourier transform for golden angle linogram sampling. *Inverse Problems*, 35(125004), 2019.

[34] D. Huybrechs. Stable high-order quadrature rules with equidistant points. *J. Comput. Appl. Math.*, 231(2):933–947, 2009.

[35] J. Keiner, S. Kunis, and D. Potts. NFFT 3.5, C subroutine library. `http://www.tu-chemnitz.de/~potts/nfft`. Contributors: F. Bartel, M. Fenn, T. Görner, M. Kircheis, T. Knopp, M. Quellmalz, M. Schmischke, T. Volkmer, A. Vollrath.

[36] J. Keiner, S. Kunis, and D. Potts. Using NFFT3 - a software library for various nonequispaced fast Fourier transforms. *ACM Trans. Math. Software*, 36:Article 19, 1–30, 2009.

[37] M. Kircheis and D. Potts. Direct inversion of the nonequispaced fast Fourier transform. *Linear Algebra Appl.*, 575:106–140, 2019.

[38] M. Kircheis and D. Potts. Efficient multivariate inversion of the nonequispaced fast Fourier transform. *PAMM*, 20(1):e202000120, 2021.

[39] M. Kircheis, D. Potts, and M. Tasche. Nonuniform fast Fourier transforms with nonequispaced spatial and frequency data and fast sinc transforms. *Numer. Algor.*, 2022.

[40] M. Kircheis, D. Potts, and M. Tasche. On regularized Shannon sampling formulas with localized sampling. *Sampl. Theory Signal Process. Data Anal.*, 20(20), 2022.

[41] T. Knopp, S. Kunis, and D. Potts. A note on the iterative MRI reconstruction from nonuniform k-space data. *Int. J. Biomed. Imag.*, 2007, 2007. ID 24727.

[42] V. A. Kotelnikov. On the transmission capacity of the "ether" and wire in electro-communications. In *Modern Sampling Theory: Mathematics and Application*, pages 27–45. Birkhäuser, Boston, 2001. Translated from Russian.

[43] S. Kunis and D. Nagel. On the condition number of Vandermonde matrices with pairs of nearly-colliding nodes. *Numer. Algor.*, 87:473–496, 2021.

[44] S. Kunis and D. Potts. Stability results for scattered data interpolation by trigono-metric polynomials. *SIAM J. Sci. Comput.*, 29:1403–1419, 2007.

[45] S. Kunis and D. Potts. Time and memory requirements of the nonequispaced FFT. *Sampl. Theory Signal Image Process.*, 7:77–100, 2008.

[46] J.-Y. Lee and L. Greengard. The type 3 nonuniform FFT and its applications. *J. Comput. Physics*, 206:1–5, 2005.

[47] R. Lin and H. Zhang. Convergence analysis of the Gaussian regularized Shannon sampling formula. *Numer. Funct. Anal. Optim.*, 38(2):224–247, 2017.

[48] Y. Liu, J. Ma, Y. Fan, and Z. Liang. Adaptive-weighted total variation minimization for sparse data toward low-dose x-ray computed tomography image reconstruction. *Phys. Med. Biol.*, 57(23):7923, 2012.

[49] J. Lund and K. L. Bowers. *Sinc Methods for Quadrature and Differential Equations.* Society for Industrial and Applied Mathematics, 1992.

[50] C. Micchelli, Y. Xu, and H. Zhang. Optimal learning of bandlimited functions from localized sampling. *J. Complexity*, 25(2):85–114, 2009.

[51] A. Nieslony and G. Steidl. Approximate factorizations of Fourier matrices with nonequispaced knots. *Linear Algebra Appl.*, 266:337–351, 2003.

[52] J. G. Pipe and P. Menon. Sampling density compensation in MRI: rationale and an iterative numerical solution. *Magn. Reson. Med.*, 41:179–186, 1999.

[53] G. Plonka, D. Potts, G. Steidl, and M. Tasche. *Numerical Fourier Analysis.* Applied and Numerical Harmonic Analysis. Birkhäuser, 2018.

[54] D. Potts and M. Tasche. Continuous window functions for NFFT. *Adv. Comput. Math.*, 47(53):1–34, 2021.

[55] D. Potts and M. Tasche. Uniform error estimates for nonequispaced fast Fourier transforms. *Sampl. Theory Signal Process. Data Anal.*, 19(17):1–42, 2021.

[56] K. P. Pruessmann and F. T. A. W. Wayer. Major speedup of reconstruction for sensitivity encoding with arbitrary trajectories. In *Proc. Intl. Soc. Mag. Reson. Med. 9*, page 767. Glasgow, Scotland, 2001.

[57] L. Qian. On the regularized Whittaker–Kotelnikov–Shannon sampling formula. *Proc. Amer. Math. Soc.*, 131(4):1169–1176, 2003.

[58] V. Rasche, R. Proksa, R. Sinkus, P. Börnert, and H. Eggers. Resampling of data between arbitrary grids using convolution interpolation. *IEEE Trans. Med. Imag.*, 18:385–392, 1999.

[59] D. Rosenfeld. An optimal and efficient new gridding algorithm using singular value decomposition. *Magn. Reson. Med.*, 40(1):14–23, 1998.

[60] D. Ruiz-Antolin and A. Townsend. A nonuniform fast Fourier transform based on low rank approximation. *SIAM J. Sci. Comput.*, 40(1):A529–A547, 2018.

[61] T. Scarnati and A. Gelb. Accelerated variance based joint sparsity recovery of images from Fourier data. arXiv, 2019.

[62] H. Sedarat and D. G. Nishimura. On the optimality of the gridding reconstruction algorithm. *IEEE Trans. Med. Imaging*, 19(4):306–317, 2000.

[63] J. Selva. Efficient type-4 and type-5 non-uniform FFT methods in the one-dimensional case. *IET Signal Processing*, 12(1):74–81, 2018.

[64] C. E. Shannon. Communication in the presence of noise. *Proc. I.R.E.*, 37:10–21, 1949.

[65] W. Stefan, R. A. Renaut, and A. Gelb. Improved total variation-type regularization using higher order edge detectors. *SIIMS*, 3(2):232–251, 2010.

[66] G. Steidl. A note on fast Fourier transforms for nonequispaced grids. *Adv. Comput. Math.*, 9:337–353, 1998.

[67] G. Stewart. *Matrix Algorithms: Basic decompositions*. SIAM, Philadelphia, 1998.

[68] T. Strohmer and J. Tanner. Fast reconstruction methods for bandlimited functions from periodic nonuniform sampling. *SIAM J. Numer. Anal.*, 44(3):1071–1094, 2006.

[69] B. P. Sutton, J. A. Fessler, and D. C. Noll. A min-max approach to the nonuniform N-dimensional FFT for rapid iterative reconstruction of MR images. In *Proc. ISMRM 9th Scientific Meeting*, page 763, 2001.

[70] V. Tchakaloff. Formules de cubature mécaniques à coefficients non négatifs. *Bull. Sci. Math.*, 81:123–134, 1957.

[71] G. Wasserman, R. Archibald, and A. Gelb. Image reconstruction from Fourier data using sparsity of edges. *J. Sci. Comput.*, 65(2):533–552, 2015.

[72] E. T. Whittaker. On the functions which are represented by the expansions of the interpolation theory. *Proc. R. Soc. Edinb.*, 35:181–194, 1915.