Research Article

A Note on the Iterative MRI Reconstruction from Nonuniform $k$-Space Data

Tobias Knopp,1 Stefan Kunis,2 and Daniel Potts2

1 Institute of Mathematics, University of Lübeck, 23538 Lübeck, Germany
2 Department of Mathematics, Chemnitz University of Technology, 09107 Chemnitz, Germany

Received 18 January 2006; Revised 7 November 2006; Accepted 16 January 2007

Recommended by David L. Wilson

In magnetic resonance imaging (MRI), methods that use a non-Cartesian grid in $k$-space are becoming increasingly important. In this paper, we use a recently proposed implicit discretisation scheme which generalises the standard approach based on gridding. While the latter succeeds for sufficiently uniform sampling sets and accurate estimated density compensation weights, the implicit method further improves the reconstruction quality when the sampling scheme or the weights are less regular. Both approaches can be solved efficiently with the nonequispaced FFT. Due to several new techniques for the storage of an involved sparse matrix, our examples include also the reconstruction of a large 3D data set. We present four case studies and report on efficient implementation of the related algorithms.

Copyright © 2007 Tobias Knopp et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

1. INTRODUCTION

The raw data for magnetic resonance imaging (MRI) is measured in $k$-space, the domain of spatial frequencies, where non-Cartesian sampling schemes like spiral or radial scans have received much attention. In contrast to the use of the computationally efficient fast Fourier transform (FFT) for the reconstruction from Cartesian grids, the more general sampling trajectories ask for the so-called nonequispaced FFTs. On the other hand, iterative image reconstruction algorithms play an important role in modern tomographic systems [1]. Recently, iterative image reconstruction in combination with the nonequispaced FFT has been applied to data on spiral $k$-space trajectories [2] and in the presence of field inhomogeneities [3]. Efficient reconstruction procedures for sensitivity encoding with arbitrary $k$-space trajectories were proposed in [4]. Its authors present methods that combine the gridding principles with the conjugate gradient scheme, but mention the long computation times due to their nonoptimised preliminary software.

In this paper, we use a similar method, compare the gridding approach [5] and an approach based on an implicit discretisation [6], where we focus for the reader’s convenience on a simplified signal equation. It turns out that the iterative solution of the latter approach resembles the gridding method in its first iteration. Both reconstruction problems are easily solved by our mature software package [7] for the nonequispaced fast Fourier transform (NFFT) [8, 9]. For readers not familiar with the NFFT, we suggest to read the appendix of this paper first. The publicly available implementation easily allows for the efficient treatment of large 3D data sets by the use of new sparse matrix storage techniques [10]. Moreover, we compare different density compensation weights arising from the discretisation of the underlying integrals. We present numerical results, based on the Shepp-Logan phantom as well as on data acquired by an MR scanner.

The outline of this paper is as follows: Section 2 gives a brief introduction to the theory of Fourier transform image reconstruction and unifies the two considered approaches in MRI. We suggest the conjugate gradient method for the reconstruction problem and show that the solution is efficiently computed by the iterative use of the nonequispaced FFT. Subsequently, Section 3 presents the used $k$-space trajectories, considers sampling density compensation, and introduces the tested simulated and acquired data sets. Section 4 shows our numerical tests emphasising the reconstruction quality with respect to the number of iterations and different density compensation weights. Moreover, we give detailed information on the computational requirements of the suggested scheme. Finally, a short discussion of
our results is contained in Section 5 and an introduction to
the nonequispaced FFT might be found in the appendix.

2. THEORY

Given a trajectory \( \mathbf{k} = \mathbf{k}(t) \), the relation between the MR
signal \( s(t) \) during the readout and the object \( p \) can be modeled by
the simplified signal equation

\[
s(k) = \int_{\Omega} p(r) e^{2\pi i r \cdot k} dr. \tag{1}
\]

In the following, we describe two different approaches.
For convenience, let the available samples in \( k \)-space be con-
tained in the shifted unit cube, that is, \( \mathbf{k} \in [-1/2, 1/2]^3 \),
and the field of view be restricted to \( \Omega_N \subset [-N_1/2, N_1/2) \times
[-N_2/2, N_2/2) \times [-N_3/2, N_3/2) \), where \( \mathbf{N} = (N_1, N_2, N_3)^T \in
2\mathbb{N}^3 \). Then, the discretisation of integral (1) on equispaced
points leads to

\[
s(k) \approx \tilde{s}(k) := \sum_{\mathbf{r} \in I_N^k} p(r) e^{2\pi i r \cdot k}, \tag{2}
\]

where \( I_N^k \) is \((N_1/2, ..., N_1/2 - 1) \times (N_2/2, ..., N_2/2 - 1) \times
(N_3/2, ..., N_3/2 - 1) \). Thus, the unknown object \( p \) is
given implicitly by (2). The authors of [6] call this the “in-
verse model.”

A second approach uses the Fourier inversion theorem
\( p(r) = \int_{\Omega_N} s(k) e^{-2\pi i k \cdot r} \, dk \). The discretisation of this inte-
rnal leads to

\[
p(r) \approx \tilde{p}(r) := \sum_{j=0}^{M-1} s(k_j) e^{-2\pi i r \cdot k_j}, \tag{3}
\]

where \( w_j \) is weights, which compensate for local variations
of the sampling density. Here, the unknown object \( p \approx \tilde{p} \) can be
computed explicitly.

The important difference between (2) and (3) is that the
former is discretised in the image domain with pixels on a
uniform grid and hence with unit weighting coefficients and
the latter is an integral discretised in \( k \)-space with nonuni-
form samples and specific weights. We reformulate problem
(2) and (3) in matrix vector notation and denote the vector
of the given measurements by \( \mathbf{s} := (s(k_j))_{j=0, ..., M-1} \in
\mathbb{C}^M \). the reshaped of the unknown object by \( \mathbf{p} :=
(p(r))_{r \in I_N^k} \in \mathbb{C}^{N_1 \times N_2 \times N_3} \), the density compensation matrix by
\( \mathbf{W} := \text{diag}(w_j)_{j=0, ..., M-1} \), and the nonequispaced Fourier ma-
trix by

\[
\mathbf{A} := (e^{2\pi i n k_j})_{j=0, ..., M-1; \mathbf{r} \in I_N^k}, \tag{4}
\]

where \( \mathbf{A}^\dagger \) denotes its adjoint (conjugate transpose).

The gridding approximation (3) is easily computed by
one matrix vector multiplication

\[
\tilde{\mathbf{p}} = \mathbf{A}^\dagger \mathbf{W} \mathbf{s}. \tag{5}
\]

The adjoint NFFT takes \( \mathcal{O}(|I_N^k| \log |I_N^k| + M) \) floating point
operations for this task.

\[
\begin{align*}
\text{Input} & \quad M \in \mathbb{N} \\
& \quad N_1, N_2, N_3 \in \mathbb{N} \\
& \quad (k_j, s_j)_{j=0, ..., M-1} \\
& \quad \mathbf{W} = \text{diag}(w_j)_{j=0, ..., M-1} \\
\text{number of given samples} & \quad \text{number of unknowns} \\
\text{number of sampling points} & \quad \text{values} \\
\text{diagonal matrix, density} & \quad \text{compensation weights} \\
\rho_0 & = 0 \\
\mathbf{r}_0 & = \mathbf{s} \\
\tilde{\mathbf{p}}_0 & = \tilde{\mathbf{z}}_0 = \mathbf{A}^\dagger \mathbf{W} \mathbf{r}_0 \\
& \text{for } l = 0, 1, \ldots \ 	ext{do} \\
& \quad \mathbf{v}_l = \mathbf{A}^\dagger \mathbf{W} \mathbf{v}_{l-1} \\
& \quad \mathbf{p}_{l+1} = \mathbf{p}_l + \mathbf{v}_l \\
& \quad \mathbf{r}_{l+1} = \mathbf{r}_l - \mathbf{v}_l \\
& \quad \tilde{\mathbf{z}}_{l+1} = \mathbf{A}^\dagger \mathbf{W} \mathbf{r}_{l+1} \\
& \quad \tilde{\mathbf{p}}_{l+1} = \tilde{\mathbf{z}}_{l+1} + \mathbf{v}_l \\
& \text{end for} \\
\text{Output} & \quad \mathbf{p}_l \approx \text{approximate solution}
\end{align*}
\]

\textbf{Algorithm 1:} Conjugate gradients for normal equations (CGNR).

Slightly more involved, the reconstruction problem (2) is
solved by the method of least squares and hence consists in
solving the weighted normal equation of first kind

\[
\mathbf{A}^\dagger \mathbf{W} \mathbf{A} \mathbf{p} = \mathbf{A}^\dagger \mathbf{W} \mathbf{s} \tag{6}
\]

for the unknown vector \( \mathbf{p} \). In contrast to [6], we include den-
sity compensation weights also for the implicit discretisation
since this is more natural with respect to the “continuous
residual” in \( k \)-space and has been proven to be better con-
ditioned in [11]. From the mathematical point of view, (6)
is solved most efficiently by the conjugate gradients (CG)
method (cf. [12, page 288]). We prefer to solve (6) by a fac-
torised variant of conjugated gradients, where the two multi-
plies with the (adjoint) nonequispaced Fourier matrix
per iteration are computed by the NFFT. This scheme is de-
noted by CGNR, \( \mathbf{N} \) for “Normal equation,” and \( \mathbf{R} \) for “Resid-
ual minimisation” (cf. [13]). Note that the CG method ap-
pied directly to (6) as suggested in [4] generates the same se-
quence of approximations in exact arithmetic, but the CGNR
approach is considered to be more stable with respect to round-off errors (cf. [14, Section 7.1]). In summary, we sug-
gest Algorithm 1.

Remarkably, this algorithm resembles an optimised gridding
solution after one iteration. More formally, let the weighted residue norm \( \| \mathbf{s} - \mathbf{A} \mathbf{p} \|_w =: r^w \mathbf{W} \mathbf{r} \) be given. Then, the
first iteration obeys

\[
\mathbf{p}_1 = \arg \min_{\mathbf{p} = \mathbf{p}_0} \| \mathbf{s} - \mathbf{A} \mathbf{p} \|_w, \tag{7}
\]

since for the first residual \( \mathbf{r}_1 = \mathbf{s} - \mathbf{A} \mathbf{p}_1 \) holds the perpendicular
condition

\[
r^w_1 \mathbf{W} \mathbf{A} \mathbf{p}_1 - (\mathbf{A}^\dagger \mathbf{W} \mathbf{r}_1)^\dagger \times - \frac{(\mathbf{A}^\dagger \mathbf{W} \mathbf{r}_1)^\dagger \times}{(\mathbf{A}^\dagger \mathbf{W} \mathbf{A})^\dagger \times} (\mathbf{A}^\dagger \mathbf{W} \mathbf{A})^\dagger \times \mathbf{W} \mathbf{A} \mathbf{p}_1 = 0. \tag{8}
\]

In other words, the gridding solution \( \tilde{\mathbf{p}} \) in (5) is scaled such
that its residual is minimised and we do not consider a gridding
approach separately anymore.
Moreover, note that the proposed scheme should be stopped as soon as the current residual $\|s - Ap\|_W$ drops below the discretisation error or the level of noise in the measurements $s$. For our first evaluation, we terminate the suggested algorithm after a fixed number of iterations.

### 3. METHODS

We are concerned with the reconstruction from data acquired by an MR scanner as well as the reconstruction quality and the usage of time and memory resources for simulated data. Algorithm 1 has been tested with MR measurements of a physical phantom by the Philips Achieva 1.5T device. Here, the sampling scheme consists of 36 equidistant radial trajectories with $M = 7,557,120$ points in total, whereas the reconstructed image contains $256 \times 256 \times 36 = 2,359,296$ voxels.

Moreover, we compare different reconstructions for simulated MR data. We use two 3D-Shepp-Logan phantoms of sizes $256 \times 256 \times 36$ and $128 \times 128 \times 128$ (2,097,152 voxels) as shown in Figure 1. Comparison is done with respect to the number of iterations, sampling schemes, and density compensation weights. The simulated MR data is computed by a 3D-NFFT using the following approaches.

- **(SPIRAL)** This 2D- k-space trajectory is given by
  \[
  \tilde{k}_j = \sqrt{\frac{j}{M}} \begin{pmatrix} \cos \omega_j \sin \omega_j \\ \sin \omega_j \cos \omega_j \end{pmatrix},
  \]
  where $\omega_j = (8\pi/5)\sqrt{j}$, and $j = 0, \ldots, M - 1$. We have chosen $M = 65,536$ yielding a total number of $M = 36M_1 = 2,359,296$ points in k-space.

- **(RADIAL)** Popular also within computer tomography is the 2D-radial trajectory
  \[
  \tilde{k}_{p,r} = (-1)^r \left( \frac{r}{R} - 0.5 \right) \begin{pmatrix} \cos \frac{\pi p}{P} \sin \frac{\pi r}{R} \\ \sin \frac{\pi p}{P} \cos \frac{\pi r}{R} \end{pmatrix},
  \]
  with $p = 0, \ldots, P - 1$, $r = 0, \ldots, R - 1$. We set $P = 410$ and $R = 512$ yielding a total number of $M = 36M_1 = 36PR = 7,557,120$ points in k-space.

### 4. NUMERICAL RESULTS

Algorithm 1 was implemented as part of the NFFT library [7]. All tests were implemented in Matlab&C and tested on an AMD Athlon XP 2700+, 2GB memory, SuSe-Linux, kernel 2.4.20-4GB-athlon, FFTW3.0.1, and NFFT2 (Kaiser-Bessel window function with cut-off parameter $m = 6$ and oversampling factor $\sigma \geq 1.5$). Besides a couple of representative examples, our web page on this project [http://www.tu-chemnitz.de/~potts/projects/mri](http://www.tu-chemnitz.de/~potts/projects/mri) collects a more thorough set of tests as pointed out in the subsequent examples. In particular, this includes animated graphics showing the progress during the iterations or slicing through the 3D data set.
Numerical Example 1. First of all, we apply Algorithm 1 to MR measurements taken by the Philips Achieva 1.5T device. Figure 3 shows the result after one iteration, where we used Voronoi weights for sampling density compensation.

Numerical Example 2. We compare the reconstruction quality with respect to the three different $k$-space trajectories and sampling density compensation weights. For the 2D $\otimes$ 1D type of the trajectories, we reconstruct each slice separately and compute the overall reconstruction by 256$^2$ regular 1D-FFTs of length $N_3 = 36$ within the third component. Table 1 shows the normalized root-mean-square error

$$\text{RMS}(\hat{p}, p) := \frac{\|p - \hat{p}\|_2}{\|p\|_2},$$

(14)

where $\hat{p}$ is our reconstruction and $p$ denotes the original image of size $128 \times 128 \times 128$ for the 3D-RADIAL trajectory and of size $256 \times 256 \times 36$ for all others.

| Weights \ iterations | 1     | 2     | 5     | 10    |
|---------------------|-------|-------|-------|-------|
| 3D RADIAL           |       |       |       |       |
| None                | 0.8281| 0.7491| 0.6482| 0.5509|
| Analytic            | 0.6981| 0.5685| 0.3047| 0.2146|
| RADIAL              |       |       |       |       |
| None                | 0.6458| 0.5276| 0.3025| 0.1170|
| Approximation       | 0.1597| 0.0773| 0.0767| 0.0764|
| Voronoi             | 0.0776| 0.0775| 0.0772| 0.0769|
| SPIRAL              |       |       |       |       |
| None                | 0.1658| 0.0908| 0.0769| 0.0767|
| Approximation       | 0.1686| 0.0864| 0.0773| 0.0768|
| Voronoi             | 0.1360| 0.0812| 0.0781| 0.0779|
| SPIRAL with noise   |       |       |       |       |
| Voronoi             | 0.1444| 0.0937| 0.0971| 0.1010|

Additionally, we computed 30 iterations for the 3D-RADIAL example which yields a RMS of 0.3309 and 0.2104 for no weights and analytic weights, respectively. A first hint on the stability of the reconstruction process with respect to noisy data is given in the last line of Table 1. We show the RMS for the reconstruction from simulated SPIRAL data which is perturbed by 20 percent standard normally distributed noise.

Numerical Example 3. We present some reconstructions our algorithm achieves. The main purpose of the 3D-RADIAL example is to demonstrate that the NFFT-based reconstruction is straightforward. Figure 4 shows the 64th slice and the profile of the 64th row of this slice after one, five, and ten iterations for analytic weights. The same experiment using no weights is presented on our web page.
Furthermore, we show the 18th slice and the profile of the 128th row of this slice after one and ten iterations for different weights and the 2D $\otimes$ 1D $k$-space trajectory RADIAL in Figure 5. The same experiment using the SPIRAL trajectory and an interleaved spiral trajectory is presented on our web-page.

**Numerical Example 4.** In the last test, we measured CPU time and used memory of our algorithms with respect to the size of the reconstruction problem $N$, that is, the size of the reconstructed image is $N \times N \times 36$ and the total number of SPIRAL $k$-space samples is $36N^2$. We compare Algorithm 1 exploiting the 2D $\otimes$ 1D type of the trajectory with Algorithm 1 using the full 3D NFFT. Both algorithms produce almost the same sequence of reconstructions.

The optimisation of the computing time is done by different NFFT flags which affect the last step of the NFFT. In this last step, we compute by $O(M)$ floating point operations the matrix vector product with the sparse matrix $B$ as defined in the appendix, (A.2). We propose different methods for the precomputation and storage of this matrix that basically trade main memory for computation speed. We were able to store all nonzero entries of the matrix $B$ together with their row and column index by the flag PRE_FULL_PSI in all 2D $\otimes$ 1D tests; see [18] for a similar approach. A lossless compressed form, flag PRE_PSI, is used in the 3D-tests up to $N = 256$. Finally, we use a lossy compressed version which is independent of the actual trajectory, uses a lookup table, and linear interpolation for the entries of the sparse matrix (PRE_LIN_PSI) in the 3D-tests for $N = 512$, see the manual to [7, 10] for details. Table 2 shows our results for the Shepp-Logan phantom, the SPIRAL trajectory, and Voronoi weights. The measured CPU-times grow as expected like $N^2 \log N$. Clearly, the method based on the 2D $\otimes$ 1D model is faster, however our algorithms can handle arbitrary scattered data in $d$ dimensions by storing a heavily compressed form of the sparse matrix $B$.

5. DISCUSSION

Magnetic resonance signals are measured in $k$-space, where magnetic field gradients determine the specific trajectories that form the sampling points. We have shown that the reconstruction from non-Cartesian grids is easily computed by means of the nonequispaced FFT where at least an approximate compensation of the non-uniform sampling density proves necessary for highly nonuniform sampling sets.

The gridding approach, based on an explicit discretisation, and the “inverse model” [6], based on an implicit discretisation, are unified and solved by Algorithm 1. In particular, the gridding solution is optimised in one iteration to minimise the residual, see (7), and further refined in subsequent iterations. We see that already the gridding method

---

**Figure 4:** Sixty-fourth slice of the phantom (top) and the profile of the Sixty fourth row of this slice (bottom) with 3D-RADIAL data and analytic weights (from left to right after one, five, and ten iterations).
may lead to very good results in Table 1 and in Figure 5. However, using no density compensation weights gives reasonable results only if the trajectory covers the k-space uniformly as for the SPIRAL trajectory, see Table 1 "weights: none" and our web-page. Gridding without density compensation gives non satisfactory results for 3D-RADIAL and RADIAL trajectories as shown in the leftmost images of Figures 4 and 5. Including approximate weights improves the reconstruction quality, the best gridding results are obtained using the more expensive Voronoi weights. However, substantial improvement of the reconstruction quality is achieved during a small number of iterations for all used trajectories and weights. Furthermore, only the iterative method in conjunction with a reasonable approximation of weights gives an acceptable reconstruction from the 3D-RADIAL trajectory (cf. Figure 4 and our web page). Finally, we have compared the computation time and memory usage of the proposed algorithm for the 2D ⊗ 1D model and a full 3D model.

In summary, the implicit discretisation and its iterative solution generalises the gridding approach in a natural way and refines the image quality. Particularly, a poor gridding solution from a highly nonuniform trajectory with approximated density compensation weights is improved substantially by a few iterations of our scheme. We have shown that very efficient methods are available, which can be generalised to reconstruction methods for sensitivity encoding as proposed in [4] and to reconstruction methods in presence of field inhomogeneities [3, 19].

Note, however, that the computation times reported in Table 2 depend on the stopping criterion. Directions for our future research include the development of reliable stopping
criteria for the iterative reconstruction and proven convergence rates when the Nyquist criteria is violated.

**APPENDIX**

In recent years, the nonequispaced fast Fourier transform (NFFT) has attracted much attention due to the fact that it generalises the FFT for arbitrary sampling geometries. Common names for these algorithms are *nonequispaced FFT [8, 9, 20], nonuniform FFT [21], generalised FFT [22], or unequally spaced FFT [23].* Its accuracy is adjusted to the practical requirements by an oversampling factor and a cut-off parameter, whereas no dependency on the sampling points occurs. In contrast, the inverse NFFT can be computed with a CG-type algorithm utilising one NFFT and one adjoint NFFT per iteration and its reconstruction error depends strongly on the sampling geometry and used compensation weights. In what follows, we briefly describe the NFFT, the adjoint NFFT, and the inverse NFFT. These methods are implemented in our public software package [7].

**Nonequispaced FFT**

The first problem to be addressed here can be regarded as a matrix vector multiplication. For a finite number of given Fourier coefficients \( f_\ell \in \mathbb{C}, \ell \in I_N^d := \{-N_1/2, \ldots, N_1/2 - 1\} \times \{-N_2/2, \ldots, N_2/2 - 1\} \times \cdots \times \{-N_d/2, \ldots, N_d/2 - 1\} \) with \( N := (N_1, \ldots, N_d)^\top \in 2\mathbb{N}^d \), one wants to evaluate the trigonometric polynomial

\[
  f(x) := \sum_{\ell \in I_N^d} \hat{f}_\ell e^{2\pi i \ell x},
\]

at given points \( x_j \in [-1/2, 1/2]^d, \ j = 0, \ldots, M - 1 \). In matrix vector notation, see also (4), this reads as \( f = \mathbf{A} \hat{f} \).

Fast approximate matrix vector multiplication with \( \mathbf{A} \) and its adjoint matrix \( \mathbf{A}^\dagger \) are known as NFFT and adjoint NFFT, respectively. A unified approach for this task was suggested in [8, 9] and is based on the factorisation \( \mathbf{A} \approx \mathbf{BFD} \) and the use of a simultaneously in space and frequency localised window function \( \varphi \). The diagonal matrix \( \mathbf{D} \) collects the inverse of the sampled Fourier transform of the window function \( \mathbf{D} = [\text{diag}(1/\varphi(x))_{x \in I_N^d}, 0]^\top \). Subsequently, we apply an oversampled Fourier matrix \( \mathbf{F} = \mathbf{F}_{mN} \), where \( \sigma > 1 \) denotes a chosen oversampling factor. Finally, we multiply with the sparse matrix \( \mathbf{B} = (b_{ij})_{i=0,\ldots,M-1, j \in I_N^d} \) having for some cut-off parameter \( m \in \mathbb{N} \) at most \((2m+1)^d \) nonzero entries per row, that is,

\[
  b_{ij} = \varphi(x_j - \frac{1}{\sigma N}), \quad -m \leq x_j - \frac{1}{\sigma N} \leq m.
\]

The relation between speed and accuracy of the algorithm was initially investigated in [8, 9, 22, 23]. The accuracy of the NFFT increases exponentially with the cut-off parameter \( m \) for each fixed oversampling factor \( \sigma > 1 \). The total number of floating point operations is of order \( \mathcal{O}(|\sigma I_N^d| \log |\sigma I_N^d| + m^d M) \).

In various papers, different window functions \( \varphi \) for the NFFT were considered, for example, a Gaussian pulse tapered with a Hanning window in [24], Gaussian kernels combined with sinc kernels in [25], and special optimised windows in [24, 26]. Furthermore, special approaches based on scaling vectors [27], based on minimising a discrete norm of certain error matrices [28], or based on min-max interpolation [21] are proposed. We remark that the NFFT and its adjoint can be realised with a broad class of these window functions yielding the same accuracy by changing the cut-off parameter \( m \). In particular, the obtained reconstruction quality in MRI is fairly independent of the particularly chosen window function. However, numerical results in [20, 21, 28] show that the Kaiser-Bessel window function provides the best accuracy for a fixed cut-off parameter \( m \), that is, number of nonzeros per row of the matrix \( \mathbf{B} \). Moreover, the library adapts over a wide class of available memory through the so-called flags; see, for example, Numerical Example 4. The manual to [7, 10] present details for a suitable choice of the window function, the cut-off parameter, and the oversampling factor with respect to accuracy, speed, and memory usage.

The adjoint algorithm, that is, the evaluation of the sums \( \hat{f}_\ell := \sum_{j=0}^{M-1} f_j e^{-2\pi i \ell x_j}, \ell \in I_M^d \), can be computed by the matrix vector multiplication with \( \mathbf{A}^\dagger \), where we obtain a fast algorithm by using the same factorisation \( \mathbf{A}^\dagger \approx \mathbf{D}^\dagger \mathbf{F}^\dagger \mathbf{B}^\dagger \). It was already pointed out in [8, 29], that the gridding method is simply a fast algorithm for the multiplication of the matrix \( \mathbf{A}^\dagger \) with a vector. Including a sampling density compensation yields the following gridding algorithm, see also [25, 26]:

(i) sampling density compensation, that is, multiplication with a diagonal matrix \( \mathbf{W} \);
(ii) approximation to an oversampled Cartesian grid, that is, multiplication with the matrix \( \mathbf{B}^\dagger \);
(iii) inverse fast Fourier transform, that is, multiplication with \( \mathbf{F}^\dagger \);
(iv) roll-off correction, that is, multiplication with \( \mathbf{D}^\dagger \).

Note that the role of the window function, appearing in the matrices \( \mathbf{D} \) and \( \mathbf{B} \), and the sampling density compensation, appearing in \( \mathbf{W} \), should not be mixed up.

| \( N \) no. | CPU time in seconds | MByte |
|---|---|---|
| 1 | 2 | 5 | 10 |
| 2D \( \otimes \) 1D trajectory |
| 512 | 118.95 | 117.94 | 355.08 | 650.55 | 691 |
| 256 | 29.24 | 43.55 | 87.21 | 159.71 | 172 |
| 128 | 6.56 | 9.84 | 19.73 | 35.78 | 43 |
| 64 | 1.45 | 2.16 | 4.31 | 8.00 | 10 |
| 3D trajectory |
| 512 | 114.22 | 17.11 | 34.18 | 62.60 | 1952 |
| 256 | 1865.9 | 280.04 | 560.33 | 10.27 | 1244 |
| 128 | 452.31 | 679.28 | 1344.9 | 2454.4 | 311 |
| 64 | 89.24 | 133.72 | 267.18 | 489.59 | 77 |
We stress again that the NFFT and the adjoint NFFT (gridding) are algorithms that realise the matrix vector multiplications in a fast and efficient way.

**Inverse nonequispaced FFT**

The aim of the inverse NFFT is to construct a trigonometric polynomial \( f \), respectively, its Fourier coefficients \( \hat{f} \), \( r \in \mathbb{Z}^N \), such that \( f(x_j) \) approximates given values \( y_j \in \mathbb{C} \), that is,

\[
y_j \approx f(x_j) = \sum_{r \in \mathbb{Z}^N} \hat{f}_r e^{2\pi i r x_j}, \quad j = 0, \ldots, M - 1. \quad (A.3)
\]

In matrix vector notation, this reads as \( \mathbf{A}^{\dagger} \approx \mathbf{y} \). We propose to solve this system of linear equations by the method of least squares with sampling density compensation weights \( w_j > 0 \), that is,

\[
\min_{\mathbf{f}} ||\mathbf{y} - \mathbf{A}^{\dagger}||_W^2 = \min_{\mathbf{f}} \sum_{j=0}^{M-1} w_j |y_j - f(x_j)|^2, \quad (A.4)
\]

where \( W := \text{diag}(w_j)_{j=0}^{M-1} \). The iterative solution of this problem is addressed in [11, 30] by solving the explicitly formed Toeplitz system \( \mathbf{A}^{\dagger} \mathbf{A} \mathbf{W} = \mathbf{A}^{\dagger} \mathbf{W} \mathbf{y} \). This method is known as ACT (adaptive weights, conjugate gradients, Toeplitz) and has been proven to be superior to previous established methods like POCS (projection onto convex sets) or the Landweber iteration (cf. [30]). The inverse NFFT applies the conjugate gradient method as well, but uses the NFFT instead of methods for Toeplitz matrices.

**ACKNOWLEDGMENTS**

The second author is grateful for partial support of this work by the German Academic Exchange Service (DAAD). The authors would like to thank the referees for their valuable suggestions. Moreover, they would like to thank Michael Schacht Hansen, Steffen Ringgaard (MR Research Centre, University of Aarhus, Denmark), and Holger Eggers (Philips Research, Sector Technical Systems, Hamburg, Germany) for numerous fruitful discussions and for provided data sets.

**REFERENCES**

[1] S. Matej, J. A. Fessler, and I. G. Kazantsev, “Iterative tomographic image reconstruction using Fourier-based forward and back-projectors,” *IEEE Transactions on Medical Imaging*, vol. 23, no. 4, pp. 401–412, 2004.

[2] 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 *Proceedings of the 9th Annual Meeting of the International Society for Magnetic Resonance in Medicine (ISMRM ’01)*, p. 763, Scotland, UK, April 2001.

[3] B. P. Sutton, D. C. Noll, and J. A. Fessler, “Fast, iterative image reconstruction for MRI in the presence of field inhomogeneities,” *IEEE Transactions on Medical Imaging*, vol. 22, no. 2, pp. 178–188, 2003.

[4] K. P. Pruessmann, M. Weiger, P. Bornert, and P. Boesiger, “Advances in sensitivity encoding with arbitrary k-space trajectories,” *Magnetic Resonance in Medicine*, vol. 46, no. 4, pp. 638–651, 2001.

[5] J. D. O’Sullivan, “A fast sinc function gridding algorithm for Fourier inversion in computer tomography,” *IEEE Transactions on Medical Imaging*, vol. 4, no. 4, pp. 200–207, 1985.

[6] B. Desplanques, D. J. Cornelis, E. Achten, R. Van De Walle, and I. Lemahieu, “Iterative reconstruction of magnetic resonance images from arbitrary samples in k-space,” *IEEE Transactions on Nuclear Science*, vol. 49, no. 5, part 1, pp. 2268–2273, 2002.

[7] S. Kunis and D. Potts, “NFFT, Softwarepackage, C subroutine library,” 2002–2006, http://www-user.tu-chemnitz.de/~potts/nfft/.

[8] D. Potts, G. Steidl, and M. Tasche, “Fast Fourier transforms for nonequispaced data: a tutorial,” in *Modern Sampling Theory: Mathematics and Applications*, J. J. Benedetto and P. I. S. G. Ferreira, Eds., pp. 247–270, Birkhäuser, Boston, Mass, USA, 2001.

[9] G. Steidl, “A note on fast Fourier transforms for nonequispaced grids,” *Advances in Computational Mathematics*, vol. 9, no. 3–4, pp. 337–352, 1998.

[10] S. Kunis and D. Potts, “Time and memory requirements of the nonequispaced FFT,” preprint 2006-01, Chemnitz University of Technology.

[11] R. F. Bass and K. Gröchenig, “Random sampling of multivariate trigonometric polynomials,” *SIAM Journal on Mathematical Analysis*, vol. 36, no. 3, pp. 773–795, 2004.

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

[13] Y. Saad, *Iterative Methods for Sparse Linear Systems*, PWS, Boston, Mass, USA, 1st edition, 1996.

[14] C. C. Paige and M. A. Saunders, “LSQR: an algorithm for sparse linear equations and sparse least squares,” *ACM Transactions on Mathematical Software*, vol. 8, no. 1, pp. 43–71, 1982.

[15] C. B. Barber and H. Huhdanpaa, “Qhull, Softwarepackage,” http://www.qhull.org/.

[16] V. Rasche, R. Proksa, R. Sinkus, P. Börnert, and H. Eggers, “Resampling of data between arbitrary grids using convolution interpolation,” *IEEE Transactions on Medical Imaging*, vol. 18, no. 5, pp. 385–392, 1999.

[17] J. G. Pipe, “Reconstructing MR images from undersampled data: data-weighing considerations,” *Magnetic Resonance in Medicine*, vol. 43, no. 6, pp. 867–875, 2000.

[18] Y. M. Kadah, A. S. Fahmy, R. E. Gabr, K. Heberlein, and X. P. Hu, “Progressive magnetic resonance image reconstruction based on iterative solution of a sparse linear system,” *International Journal of Biomedical Imaging*, vol. 2006, Article ID 49378, 9 pages, 2006.

[19] H. Eggers, T. Knopp, and D. Potts, “Field inhomogeneity correction based on gridding reconstruction,” to appear in *IEEE Transactions on Medical Imaging*.

[20] K. Fourmont, “Non-equispaced fast Fourier transforms with applications to tomography,” *Journal of Fourier Analysis and Applications*, vol. 9, no. 5, pp. 431–450, 2003.

[21] J. A. Fessler and B. P. Sutton, “Nonuniform fast Fourier transforms using min-max interpolation,” *IEEE Transactions on Signal Processing*, vol. 51, no. 2, pp. 560–574, 2003.

[22] A. Dutt and V. Rokhlin, “Fast Fourier transforms for nonequispaced data,” *SIAM Journal on Scientific Computing*, vol. 14, no. 6, pp. 1368–1393, 1993.

[23] G. Beylkin, “On the fast Fourier transform of functions with singularities,” *Applied and Computational Harmonic Analysis*, vol. 2, no. 4, pp. 363–381, 1995.

[24] A. J. W. Duijndam and M. A. Schonewille, “Nonuniform fast Fourier transform,” *Geophysics*, vol. 64, no. 2, pp. 539–551, 1999.
[25] J. Pelt, “Fast computation of trigonometric sums with applications to frequency analysis of astronomical data,” in Proceedings of the Florence and George Wise Observatory, 25th Anniversary Symposium, D. Maoz, A. Sternberg, and E. Leibowitz, Eds., Astronomical Time Series, pp. 179–182, Kluwer Academic, Tel-Aviv, Israel, January 1997.

[26] J. I. Jackson, C. H. Meyer, D. G. Nishimura, and A. Macovski, “Selection of a convolution function for Fourier inversion using gridding,” IEEE Transactions on Medical Imaging, vol. 10, no. 3, pp. 473–478, 1991.

[27] N. Nguyen and Q. H. Liu, “The regular Fourier matrices and nonuniform fast Fourier transforms,” SIAM Journal of Scientific Computing, vol. 21, no. 1, pp. 283–293, 1999.

[28] A. Nieslony and G. Steidl, “Approximate factorizations of Fourier matrices with nonequispaced knots,” Linear Algebra and Its Applications, vol. 366, pp. 337–351, 2003.

[29] G. E. Sarty, R. Bennett, and R. W. Cox, “Direct reconstruction of non-cartesian k-space data using a nonuniform fast Fourier transform,” Magnetic Resonance in Medicine, vol. 45, no. 5, pp. 908–915, 2001.

[30] H. G. Feichtinger, K. Gröchenig, and T. Strohmer, “Efficient numerical methods in non-uniform sampling theory,” Numerische Mathematik, vol. 69, no. 4, pp. 423–440, 1995.