Towards a general framework for fast and feasible k-space trajectories for MRI based on projection methods

Sharma, Shubham; Coutino, Mario; Chepuri, Sundeep Prabhakar; Leus, Geert; Hari, K. V.S.

DOI
10.1016/j.mri.2020.06.016

Publication date
2020

Document Version
Final published version

Published in
Magnetic Resonance Imaging

Magnetic Resonance Imaging 72 (2020) 122–134

Contents lists available at ScienceDirect

Magnetic Resonance Imaging

journal homepage: www.elsevier.com/locate/mri

Original Contribution

Towards a general framework for fast and feasible *k*-space trajectories for MRI based on projection methods

Shubham Sharma[a,*], Mario Coutino[b], Sundeep Prabhakar Chepuri[a], Geert Leus[b], K.V.S. Hari[a]

[a] Department of Electrical Communication Engineering, Indian Institute of Science, Bangalore, India
[b] Department of Microelectronics, Delft University of Technology, Netherlands

ARTICLE INFO

Keywords:
compressed sensing
k-space sampling
MRI
projection methods
trajectory design

ABSTRACT

The design of feasible trajectories to traverse the *k*-space for sampling in magnetic resonance imaging (MRI) is important while considering ways to reduce the scan time. Over the recent years, non-Cartesian trajectories have been observed to result in benign artifacts and being less sensitive to motion. In this paper, we propose a generalized framework that encompasses projection-based methods to generate feasible non-Cartesian *k*-space trajectories. This framework allows to construct feasible trajectories from both random or structured initial trajectories, e.g., based on the traveling salesman problem (TSP). We evaluate the performance of the proposed methods by simulating the reconstruction of 128 × 128 and 256 × 256 phantom and brain MRI images in terms of structural similarity (SSIM) index and peak signal-to-noise ratio (PSNR) using compressed sensing techniques. It is observed that the TSP-based trajectories from the proposed projection method with constant acceleration parameterization (CAP) result in better reconstruction compared to the projection method with constant velocity parameterization (CVP) and this for a similar read-out time. Further, random-like trajectories are observed to be better than TSP-based trajectories as they reduce the read-out time while providing better reconstruction quality. A reduction in read-out time by upto 67% is achieved using the proposed projection with permutation (PP) method.

1. Introduction

Magnetic resonance imaging (MRI) is a non-invasive imaging modality that provides detailed soft tissue images without exposing the subject to any harmful radiation. The MR signal is frequency encoded and can be directly mapped to the spatial frequency domain called the *k*-space [1]. The image is constructed by taking the inverse Fourier transform of the *k*-space signal. The *k*-space is traversed along a trajectory, $\mathbf{s}(t) : \mathbb{R} \rightarrow \mathbb{R}^2$,

$$\mathbf{s}(t) = [s_x(t), s_y(t)]^T, \quad t \in [0, T], \tag{1}$$

where the parameter t corresponds to time. Here, we consider 2D scanning, i.e., we assume that a slice has been selected and the *k*-space is traversed in the k_x-k_y plane. The trajectory has as initial and final points $\mathbf{s}(0) = \mathbf{s}_{start}$ and $\mathbf{s}(T) = \mathbf{s}_{end}$, respectively, and it is governed by the magnetic gradients $\mathbf{g}(t) = [g_x(t), g_y(t)]^T$, $t \in [0, T]$ as

$$s_x(t) = \gamma \int_0^t g_x(\tau) d\tau, \quad s_y(t) = \gamma \int_0^t g_x(\tau) d\tau, . \tag{2}$$

where γ is the gyromagnetic ratio, which is $\gamma = 42.58$MHz/T for hydrogen. The most commonly used trajectory is the Cartesian trajectory in which the *k*-space is traversed line-by-line. Non-Cartesian trajectories such as the spiral [2,3] and the radial [4] trajectories have also been used extensively in practice. Non-Cartesian trajectories are advantageous over Cartesian trajectories as they can better utilize the hardware of an MRI scanner and provide incoherent artifacts. However, long acquisition times in MRI are a limitation and also pose a problem for claustrophobic patients. Therefore, to reduce the acquisition time in MRI, the design of faster pulse sequences [5–7], parallel imaging [8–11] and compressed sensing (CS) techniques [12,13] have been proposed. More recently, deep learning-based methods for MRI have been proposed in the literature as well [14–19]. In addition to providing an improvement in the reconstruction quality, the main advantage observed in these methods is the speed of reconstruction compared to iterative reconstruction methods.

CS theory allows undersampling the *k*-space data based on the fact that MR images are sparse in transform domains such as wavelet and finite differences. The sparse signal is then recovered by means of non-linear optimization methods [20–22]. However, there are two main considerations that need to be accounted for when designing sampling

* Corresponding author.
E-mail address: shubham@iisc.ac.in (S. Sharma).

https://doi.org/10.1016/j.mri.2020.06.016
Received 20 February 2020; Received in revised form 11 June 2020; Accepted 24 June 2020

schemes for MRI. The first is related to variable density sampling (VDS) in which low frequency region of the k-space is more densely sampled as compared to the high frequency region. The use of VDS has become an integral part of the system owing to the theory of level sparsity as advocated in [23–25]. The second is the fact that the magnetic gradients are physically limited in their maximum magnitude and slew rate. Therefore, to achieve feasible trajectories, the design must satisfy these gradient constraints. Although there has been a lot of research to find optimal density functions [26,27], to design sparsifying transforms [28,29] and to develop CS image reconstruction methods [30,31] for a Cartesian trajectory, less attention has been given to the design of feasible non-Cartesian k-space trajectories satisfying the magnetic gradient constraints.

A recent approach to obtain a trajectory from randomly undersampled points in the k-space using VDS is proposed in [32], in which the shortest path through these points is computed by solving a traveling salesman problem (TSP). However, TSP-based trajectories are not feasible for implementation in an MRI machine since the physical constraints as discussed before are generally not satisfied here. In view of this, various methods have been proposed in the literature to obtain feasible trajectories. The design of feasible trajectories satisfying the magnetic gradient constraints is generally formulated as a constrained convex optimization problem. For instance, in [33], the authors explore various convex optimization problems to find multi-dimensional time-optimal gradient waveforms. In [34], the time-optimal control (TOC) method gives an optimal reparameterization of any curve such that it can be traversed in the fastest way possible. This method, being the most commonly used, suffers from the drawback that it requires long read-out times if the curve has sharp turns. A different approach for designing feasible trajectories was proposed in [35]. Here, trajectories are designed by finding optimal interpolation points between two consecutive sample points. Despite that these methods obtain feasible trajectories, they distort the sampling density of the trajectory. To compensate for this, the projection method [36] projects some parameterized trajectory onto the set of feasible trajectories. By doing so, the density of the trajectory is preserved at the cost of smoothing near the center of the k-space which results in a poor sampling coverage.

In this paper, we provide a general framework for projection-based methods to obtain feasible trajectories. The original projection method with constant velocity parameterization [36] turns out to be a special case of the proposed general framework. In addition, we show that other parameterizations as discussed later outperform the projection method as it was originally proposed.

2. Material and methods

In this section, we discuss various aspects to be considered while designing a feasible k-space trajectory, the proposed methods, and the experimental setup. To set the notations, we denote a 2D continuous curve as $\mathbf{f}(t)$. $\mathbf{f}_x$ and $\mathbf{f}_y$ denote the discretized x- and y-dimesions of $\mathbf{f}(t)$, repectively. $\mathbf{f}$ denotes the complete discretized curve with one dimension appended below the other. $\mathbf{f}_i$ denotes the i^{th} 2D point of $\mathbf{f}$. $[\mathbf{f}_x]_i$ and $[\mathbf{f}_y]_i$ denote the i^{th} entry of the vectors $\mathbf{f}_x$ and $\mathbf{f}_y$, repectively.

2.1. System considerations

The linear magnetic gradients mentioned earlier are obtained by using gradient coils in the three directions. The current in these coils produces different magnetic fields in different spatial locations of the body. The amplitude and slew rate (rate of change of the current) of these gradients are limited by the gradient amplifiers, heating in the amplifiers and coils, and by uncomfortable nerve stimulation. The maximum magnetic gradient amplitude and slew rate are denoted as $G_{\max}$ and $S_{\max}$, respectively, and are machine specific. Hence, the k-space trajectory needs to satisfy these gradient constraints. For details of gradient coils and other hardware aspects, we refer to [1]. The effects of off-resonance [37], T_2^* decay [1], imperfect magnetic gradients due to eddy currents [38,39] and other irregularities [40] are assumed to be negligible in this work.

In this paper, we use the following definitions of a curve and a trajectory. A *trajectory* in 2D is a function $\mathbf{s}(t) : \mathbb{R}_+ \to \mathbb{R}^2$ mapping time, $t > 0$, to $\mathbb{R}^2$. A *curve* in $\mathbb{R}^2$ is the image of this trajectory. Thus, the function $\mathbf{s}(t)$ describes a parameterization of the curve, where time t is the parameter. For example, the function $\mathbf{s}(t)$ given by

$$\mathbf{s}(t) = (t\sin(\omega t), t\cos(\omega t)) \tag{3}$$

describes the trajectory of a point that moves in $\mathbb{R}^2$ with t. The curve described by (3) is a spiral. We use $\mathbf{s}$ to denote both the curve and the trajectory while the difference will be clear from the context.

Next, we discuss the notions of distance and velocity with respect to sampling in the k-space for MRI. Consider an arbitrary 2D trajectory $\mathbf{s}(t) = [s_x(t), s_y(t)]$. Assuming a fixed sampling interval t_s, this trajectory can be described by an m-point discrete 2D trajectory $\mathbf{s} \in \mathbb{R}^{2m} = [\mathbf{s}_x{}^T \ \mathbf{s}_y{}^T]^T$ where $\mathbf{s}_x \in \mathbb{R}^m$ and $\mathbf{s}_y \in \mathbb{R}^m$ denote the x- and y-coordinates, respectively. We denote the i^{th} point along the trajectory as $\mathbf{s}_i = [[\mathbf{s}_x]_i \ \ [\mathbf{s}_y]_i]^T$, $1 \le i \le m$. Throughout this work, we assume that the sampling interval t_s is small compared to the overall scanning time. In practice, the minimum t_s is driven by the machine. Under these settings, (2) can be discretized (in 2D) as

$$\mathbf{s}_i = \gamma \sum_{j=1}^{i} \mathbf{g}_j t_s, \qquad 1 \le i \le m, \tag{4}$$

where $\mathbf{g}_i = [[\mathbf{g}_x]_i \ \ [\mathbf{g}_y]_i]^T$, $1 \le i \le m$ is the magnetic gradient magnitude at the i^{th} point.

Definition 1(Read-out time) The read-out time is the amount of time required to collect the MR signal by traversing the k-space. It is directly proportional to the length of the trajectory. The discretization (4) assumes that two consecutive points in the trajectory $\mathbf{s}$ are traversed in the sampling time, t_s, hence, the i^{th} point is reached at time it_s. The total read-out time required to traverse the trajectory will be $T = mt_s$.

Definition 2 (Distance) The distance traveled up to the i^{th} point is given as

$$l_i = \sum_{j=1}^{i} \|\mathbf{s}_j - \mathbf{s}_{j-1}\|_2 \ , \qquad 2 \le i \le m. \tag{5}$$

Definition 3 (Instantaneous velocity) The instantaneous velocity at the i^{th} point on the trajectory can be obtained from (4) and (5) as

$$v_i = \frac{l_i - l_{i-1}}{t_s} = \frac{\|\mathbf{s}_i - \mathbf{s}_{i-1}\|_2}{t_s} = \gamma\|\mathbf{g}_i\|_2, \qquad 2 \le i \le m. \tag{6}$$

The instantaneous velocity is proportional to the magnitude of the magnetic gradient at that time instant. Hence, the gradient constraints restrict the distance traversed per second. Note that the maximum velocity with which the trajectory can be traversed is $v_{\max} = \gamma G_{\max}$.

Definition 4 (Instantaneous acceleration) The instantaneous acceleration is given as

$$a_i = \frac{v_i - v_{i-1}}{t_s} = \gamma\frac{\|\mathbf{g}_i\|_2 - \|\mathbf{g}_{i-1}\|_2}{t_s}, \qquad 2 \le i \le m. \tag{7}$$

The maximum acceleration with which the trajectory can be traversed is $a_{\max} = \gamma S_{\max}$.

Definition 5(Feasible trajectories) The maximum magnitude and slew rate constraints on the magnetic gradients $\mathbf{g}$ are equivalent to velocity and acceleration constraints on the trajectory $\mathbf{s}$. A trajectory is feasible if it satisfies these constraints. To formalize this, we introduce the block diagonal matrix notation

$$\mathbf{A}^{(2)} = \text{blkdiag}_2(\mathbf{A}) = \begin{bmatrix} \mathbf{A} & \mathbf{0} \\ \mathbf{0} & \mathbf{A} \end{bmatrix}.$$

The set of m-point feasible curves $\mathbf{s} \in \mathbb{R}^{2m} = [\mathbf{s}_x{}^T \ \ \mathbf{s}_y{}^T]^T$ can then be defined as

$$\mathscr{S}^m = \{\mathbf{s} \in \mathbb{R}^{2m}: \|\mathbf{D}_1^{(2)}\mathbf{s}\|_\infty \le t_s \gamma G_{\max}, \|\mathbf{D}_2^{(2)}\mathbf{s}\|_\infty \le t_s^2 \gamma S_{\max}\}, \quad (8)$$

where $\mathbf{D}_1{}^{(2)} = \text{blkdiag}_2(\mathbf{D}_1)$, $\mathbf{D}_2{}^{(2)} = \text{blkdiag}_2(\mathbf{D}_2)$ with

$$\mathbf{D}_1 = \begin{bmatrix} 0 & 0 & 0 & 0 & \dots & 0 \\ -1 & 1 & 0 & 0 & \dots & 0 \\ 0 & -1 & 1 & 0 & \dots & 0 \\ & \vdots & & & & \\ 0 & 0 & 0 & 0 & \dots & 1 \end{bmatrix}_{m\times m}$$

being the first-order difference matrix, and $\mathbf{D}_2 = -\mathbf{D}_1{}^T\mathbf{D}_1 \in \mathbb{R}^{m\times m}$ the second-order difference matrix. Here, the gradient constraints are taken to be rotation invariant, i.e., the gradient coil in each direction is assumed to work independent of the other coils and each of them can reach the maximum constraints.

2.2. Design of feasible k-space trajectories

A common heuristical method for designing feasible trajectories is to start from an arbitrarily parameterized curve. The curve is then re-parameterized to obtain gradients which satisfy the magnetic gradient constraints using optimal control theory [33,34,41]; or the trajectory is approximated by a feasible trajectory using the projection method [36]. We briefly discuss two significant and relevant methods, the time-optimal control [34] and the projection method [36] as they will be used to compare the performance of the proposed methods.

2.2.1. Time-optimal control method

If the trajectory is described by a closed-form expression such as a spiral, rosette and others, the magnetic gradients can be obtained using (2). However, the time-optimal contol (TOC) method provides the fastest gradient waveforms to traverse the given trajectory $\mathbf{s}(t)$ with start and end points given by $\mathbf{s}_{\text{start}}$ and $\mathbf{s}_{\text{end}}$, respectively. For details of the method, we refer to [34]. This method suffers from the major drawback that the resultant trajectory takes more time near sharp turns, thereby increasing the overall read-out time. The method therefore works well for some classical (smooth) trajectories but not for the TSP-based trajectory which has many sharp turns.

2.2.2. Projection method

A projection-based method was proposed in [36] to obtain a feasible trajectory, where instead of finding the fastest reparameterization to traverse the given arbitrary curve, the constraint to follow the curve exactly is relaxed, providing a smoother curve without sharp edges. This overcomes the drawback of the TOC method and introduces points in the *k*-space that are not present in the initial trajectory. In this method, an initial arbitrary curve $\mathbf{c}(t)$, discretized as $\mathbf{c} \in \mathbb{R}^{2n}$, is interpolated by parameterising $\mathbf{c}$ at a constant velocity (a fraction of $v_{\max} = \gamma G_{\max}$ due to hardware restrictions). We denote the velocity parameterized curve as $\mathbf{c}_{\text{vpar}} \in \mathbb{R}^{2m}$ (assuming that the parameterized curve has m points). Method to obtain $\mathbf{c}_{vpar}$ is discussed below. It is then projected onto the set $\mathscr{S}^m$ of feasible curves satisfying various magnetic gradient constraints [36]. The cost function for the projection method is

$$(\text{P}_{\text{proj}}): \quad \min_{\mathbf{s}\in\mathscr{S}^m} \frac{1}{2}\|\mathbf{s} - \mathbf{c}_{\text{vpar}}\|_2^2. \quad (9)$$

here, the distance between $\mathbf{s}$ and $\mathbf{c}_{\text{vpar}}$ is minimized to keep the trajectory as close as possible to the original. The velocity of the parameterization can be varied to attain different read-out times. The reduction in readout time of this method is limited by the gradient constraints as the maximum velocity at which the curve can be traversed is $\gamma G_{\max}$. It is important to note that the initial parameterization of the curve $\mathbf{c}$ in (9) significantly affects the resulting feasible trajectory $\mathbf{s}$ in the projection method.

2.2.2.1. Constant velocity parameterization (CVP).

CVP means that the input trajectory $\mathbf{c}$ is essentially linearly interpolated by introducing new sample points on the trajectory such that consecutive points attain a velocity proportional to a fraction α of the maximum magnetic gradient amplitude $v_{\max}$. In this work, we propose to describe the CVP operation using a linear operator $\mathbf{A}_{\text{v}}(\alpha, \mathbf{c}) \in \mathbb{R}^{m\times n}$ that is applied to $\mathbf{c}$ such that $\mathbf{c}_{\text{vpar}} = \mathbf{A}_{\text{v}}(\alpha, \mathbf{c})\mathbf{c}$. This notation is to indicate that the linear CVP operator is dependent on α and $\mathbf{c}$ and the subscript 'v' is to denote CVP. Hence, the problem in (9) becomes

$$(\text{P}_{\text{proj-CVP}}): \quad \min_{\mathbf{s}\in\mathscr{S}^m} \frac{1}{2}\|\mathbf{s} - \mathbf{A}_{\text{v}}^{(2)}(\alpha, \mathbf{c})\mathbf{c}\|_2^2 \quad (10)$$

where as before $\mathbf{A}_{\text{v}}{}^{(2)}(\alpha, \mathbf{c}) = \text{blkdiag}_2(\mathbf{A}_{\text{v}}(\alpha, \mathbf{c}))$. The linear CVP operator can be decomposed into two operators:

1. a linear interpolation matrix $\mathbf{L}$.
2. a selection matrix $\Pi_{\alpha,\ \mathbf{c}}$ such that the consecutive points are traversed with a constant velocity.

Hence, $\mathbf{A}_{\text{v}}(\alpha, \mathbf{c}) = \Pi_{\alpha,\ \mathbf{c}}\ \mathbf{L}$. The details for the design method for the two matrices is provided in Appendix A.

2.2.2.2. Other parameterizations.

Although constant velocity parameterization (CVP) is used here as an initial parameterization, other parameterizations can be used. One of the simplest parameterizations of a curve is the arc-length parameterization. Following the convention, we can represent a trajectory $\mathbf{c}$ by samples in the arc-length parameterization as $\mathbf{c}(l_i) = (\mathbf{c}_{\text{x}}(l_i), \mathbf{c}_{\text{y}}(l_i))$, $1 \le i \le m$, where l_i is as described before. When a trajectory is uniformly sampled in the arc-length parameterization, all consecutive sample points are equi-distant on the trajectory. Hence, traversing such a trajectory at a fixed sampling time interval t_s makes arc-length parameterization equivalent to CVP, which is used in the projection method [36]. This is the most common parameterization in the literature. Similar to CVP, we propose another possible parameterization: Constant acceleration parameterization (CAP). In this method, the trajectory is to be traversed at a constant acceleration a as a fraction of the maximum acceleration possible $a_{\max} = \gamma S_{\max}$. The details are discussed in Section 2.4.1.

There are no strict or clear measures to find a good parameterization. Hence, in this paper, we generalize the projection method and propose further possible parameterizations that might result in a shorter read-out time and/or better reconstruction performance.

2.3. Proposed generalized framework for projection-based trajectory design

The CVP of $\mathbf{c}$ as used in the projection method [36] is in fact a linear operation on $\mathbf{c}$ as shown in the previous section. Therefore, it is natural to ask if there are other such operations on $\mathbf{c}$ that could result in a feasible trajectory that provides a better reconstruction quality or reduces the read-out time with similar reconstruction performance. Here, we explore such possible linear operations on $\mathbf{c}$ under a generalized framework. This approach can be expressed as the following optimization problem

$$(\text{P}_{gen}): \quad \min_{\mathbf{s}\in\mathscr{S}^m, \mathbf{A}^{(2)}\in\mathscr{A}} \frac{1}{2}\|\mathbf{s} - \mathbf{A}^{(2)}(\mathbf{c})\|_2^2 \quad (11)$$

where $\mathbf{A}^{(2)}(\mathbf{c})$ is a linear or non-linear function of the known vector $\mathbf{c}$ and $\mathscr{A}$ is a constraint set (more details in Section 2.4) associated to the parameterization required to avoid a trivial solution to the problem (11). When this transformation is linear, i.e., $\mathbf{A}^{(2)}(\mathbf{c}) = \mathbf{A}^{(2)}\mathbf{c}$ with, $\mathbf{A}^{(2)} \in \mathbb{R}^{2m\times 2n}$, the optimization problem could be defined as

$$(\text{P}_{\text{gen-proj}}): \quad \min_{\mathbf{s}\in\mathscr{S}^m, \mathbf{A}^{(2)}\in\mathscr{A}} \frac{1}{2}\|\mathbf{s} - \mathbf{A}^{(2)}\mathbf{c}\|_2^2 \quad (12)$$

where $\mathbf{A}^{(2)} = \text{blkdiag}_2(\mathbf{A})$. Here, $\mathbf{A}^{(2)}$ may or may not depend on $\mathbf{c}$. We aim to obtain a non-trivial solution to the above problems. In the

previous section, we had provided a method to fit the original projection method into the proposed generalized framework. For $\mathbf{A}^{(2)} = \mathbf{A}_v{}^{(2)}(\alpha, \mathbf{c})$ (the linear CVP operator), ($P_{gen-proj}$) becomes the same as (P_{proj}). In this case, $\mathbf{A}^{(2)}$ depends on $\mathbf{c}$. The advantage of the generalized framework is that it provides opportunities to explore different parameterizations using different structures on $\mathbf{A}^{(2)}$ which may lead to better projection-based feasible trajectories.

2.4. Proposed parameterizations

In the following subsections, we explore some possible variations of $\mathbf{A}^{(2)}$.

2.4.1. Constant acceleration parameterization (CAP)

As already mentioned in Section 2.2.2, we can consider a parameterization method using constant acceleration. Just as the CVP curve has successive points that have constant velocity (i.e., equidistant points for fixed sampling time t_s), CAP will have successive points that have constant acceleration (i.e., the distance between points increases with the progression of the trajectory). Acceleration naturally occurs in curves like the spiral where the trajectory starts from the center of the k-space and moves towards the boundary. The CAP operator can also be described by a linear operator which will be denoted by $\mathbf{A}_a(\beta, \mathbf{c}) \in \mathbb{R}^{m \times n}$ where β is the fraction of a_{max} used for the parameterization. The trajectory design problem now becomes:

$$(P_{proj-CAP}): \quad \min_{\mathbf{s} \in \mathscr{S}^m} \quad \frac{1}{2}\|\mathbf{s} - \mathbf{A}_a^{(2)}(\beta, \mathbf{c})\mathbf{c}\|_2^2 \tag{13}$$

where $\mathbf{A}_a{}^{(2)}(\beta, \mathbf{c}) = \text{blkdiag}_2(\mathbf{A}_a(\beta, \mathbf{c}))$. The linear CAP operator can be computed similar to the constant velocity matrix by first interpolating $\mathbf{c}$ using the linear interpolation matrix $\mathbf{L}$ and then picking points with the selection matrix $\Pi_{\beta,\,\mathbf{c}}$ such that the resultant points have constant acceleration. These two operations can be combined as $\mathbf{A}_a(\beta, \mathbf{c}) = \Pi_{\beta,\,\mathbf{c}}\mathbf{L}$. The design method for these two matrices is provided in Appendix B.

One should also note that when a curve is parameterized with constant velocity, the acceleration constraints ($\|\mathbf{D}_2{}^{(2)}\mathbf{s}\|_\infty \leq t_s{}^2\gamma S_{max}$) are generally not taken care of. This results in a distortion of the original trajectory after projection onto the feasible curves, which in certain cases, leads to a poor reconstruction performance. For example, in case of a spiral curve, the CVP curve might not satisfy the acceleration/slew rate constraint near the center of the curve as the curvature is high in that region. Similarly, if the spiral is constant acceleration parameterized, the velocity/gradient magnitude constraint ($\|\mathbf{D}_1{}^{(2)}\mathbf{s}\|_\infty \leq t_s\gamma G_{max}$) might not be satisfied near the boundary as the curvature is low in that region. This will result in distortion of the trajectory after projection (near the center for CVP and near the boundary for CAP). Hence, proper care needs to be taken for either parameterization. One should properly parameterize the curve by adjusting the velocity and acceleration such that the spiral is not distorted after projection. However, this may result in a significantly higher read-out time [42].

2.4.2. $\boldsymbol{A}$ as a general banded matrix

The structure of $\mathbf{A}_v(\alpha, \mathbf{c})$ and $\mathbf{A}_a(\beta, \mathbf{c})$ is similar to a banded matrix. The structure is so because these matrices work on a few consecutive points in $\mathbf{c}$. Note that the number of columns of the matrix remains the same for all values of α and β. However, with increasing α (or β), the number of rows increases as there are fewer points to be chosen with higher velocity (or acceleration).

We now adapt the structure of the CVP/CAP matrices by making $\mathbf{A}$ a square matrix with a banded structure, denoted as $\mathbf{A}_b$. $\mathbf{A}_b$ is made a square matrix instead of a tall matrix because we do not want to specify a certain parameterization. Remember that the number of rows of $\mathbf{A}$ in CVP/CAP depends on the parameterization. Further, the varying bandwidth of the CVP/CAP operators is changed to have a constant bandwidth, i.e., a diagonal band of fixed bandwidth is used. The design problem then becomes

$$(P_{GBP}): \quad \min_{\mathbf{s} \in \mathscr{S}^m} \quad \frac{1}{2}\|\mathbf{s} - \mathbf{A}_b^{(2)}\mathbf{c}\|_2^2 \tag{14}$$

where $\mathbf{A}_b{}^{(2)} = \text{blkdiag}_2(\mathbf{A}_b)$, $\mathbf{A}_b \in \mathscr{D}_{l,u}$ with $\mathscr{D}_{l,u}$ being the set of banded matrices with l and u as the lower and upper bandwidths, respectively. This method will be called general banded projection (GBP) method henceforth. We consider two specific cases:

- GBP (Type-1): when $\mathbf{A}_b$ is taken as the identity matrix.
- GBP (Type-2): when $\mathbf{A}_b$ is taken as a banded matrix with normal random entries and rows normalized to add up to 1.

In the GBP (Type-1) method, $\mathbf{c}$ is directly projected onto the set of feasible trajectories $\mathscr{S}$. In the GBP (Type-2) method, a random linear combination of a few consecutive points in $\mathbf{c}$ are projected onto $\mathscr{S}$. The effect of the two cases is discussed in Section 3.

2.4.3. Other variations with $\boldsymbol{A} = \boldsymbol{I}$

It is assumed that each sample is taken in the sampling time t_s. Hence, to reduce the read-out time we need to limit the number of sample points in the trajectory. However, fewer sample points might not be enough for a good image reconstruction [25,43]. Hence, there is a trade-off between the read-out time and image quality. We aim to reduce the read-out time such that the resultant trajectory provides a minimal compromise on the image quality. The main idea in the following two methods is that we project $\mathbf{c}$ as it is without doing any reparameterization. Then to obtain feasible trajectories with more sample points such that the reconstruction quality is improved, we use interpolation methods.

Here, we discuss two variations of this idea.

(a) Constrained Length Trajectory (COLT)

In this variation, $\mathbf{c}$ is not parameterized as was done in the projection method, i.e., $\mathbf{A} = \mathbf{I}$ ($n = m$). In addition to just the projection function in (12), we here also add a cost on the length of the segments of $\mathbf{s}$. This method to find a feasible trajectory is hereby called the constrained length trajectory (COLT) method [44]. It can be formulated as the following constrained convex optimization problem

$$(P_{COLT}): \quad \min_{\mathbf{s} \in \mathscr{S}^m} \quad \frac{1}{2}\|\mathbf{s} - \mathbf{c}\|_2^2 + \frac{\lambda}{2}\|\mathbf{D}_1^{(2)}\mathbf{s}\|_2^2 \tag{15}$$

where $\lambda \in \mathbb{R}^+$ is a weighting parameter, $\|\mathbf{D}_1{}^{(2)}\mathbf{s}\|_2{}^2$ is the sum of squares of the Euclidean distances between consecutive points of $\mathbf{s}$. This imposes a cost on the segments of the trajectory $\mathbf{s}$ which in turn decides the overall length of the trajectory. Hence, we project the given curve $\mathbf{c}$ with arbitrary parameterization (i.e., it contains the original randomly sampled points in any order, for instance the TSP order to construct the TSP curve) onto the set of feasible curves and include a cost on the length of the segments of the trajectory. The resultant trajectory $\mathbf{s}_{COLT} \in \mathbb{R}^{2m}$ will have the same number of sample points as that of $\mathbf{c}$. As the final step, to reduce the variation of the velocity in consecutive points, $\mathbf{s}_{COLT}$ is parameterized using a constant velocity, denoted $\mathbf{s}_{COLT}{}' \in \mathbb{R}^{2m'}$. A variable number of sample points, and hence a variable read-out time is achieved by this parameterization. This is because the higher the value of the weighting parameter, the smaller the number of sample points after CVP, and thus the shorter the read-out time. So, in effect, the second term in the cost function provides control over the read-out time which provides the motivation to include this additional term in the cost function. The gradient constraints in this case will still be satisfied as the velocity chosen for parameterization is feasible and the curve is smooth without sharp edges. Fig. 1 shows the effect of varying λ for a 4098 point TSP trajectory with initial points taken from the density π as before. For larger values of λ, the consecutive points come closer to each other, resulting in a reduction of the overall length

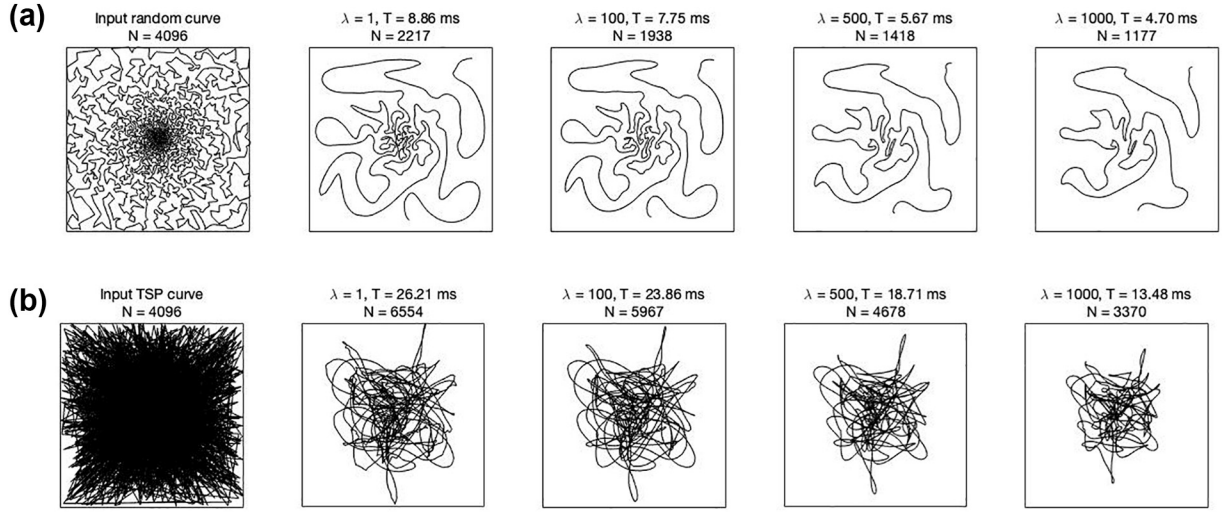


Fig. 1. Trajectories for different values of the weighting parameter λ in the COLT method for (a) a TSP curve and (b) a random curve. The read-out time and the number of sample points are denoted by T and N, respectively. $\alpha = 0.5$.

of the trajectory. Subsequently, the additional sample points introduced by CVP will also be less for larger λ. Also note that with increasing λ, the trajectories tend to become smoother.

(b) Spline Interpolation-based Projection (SIP) method

This variation uses spline interpolation instead of CVP to increase the number of sample points in the trajectory. First, we solve (12) with $\mathbf{A} = \mathbf{I}$ $(n = m)$.

$$(\mathrm{P_{SIP}}): \quad \min_{\mathbf{s}\in\mathscr{S}^m} \quad \frac{1}{2}\|\mathbf{s}-\mathbf{c}\|_2^2. \tag{16}$$

The main difference between this variation and the projection method is that in (9), instead of $\mathbf{c}$, a parameterized $\mathbf{c}$ is used. In this method, we introduce new points in the trajectory or resample the trajectory by interpolating the trajectory obtained instead of interpolating the trajectory before projection. This reduces the computational complexity and allows us to choose the interpolation factor by doing multiple simulations without having to project the trajectory for each trial. Hence, interpolating the trajectory after projection reduces the complexity overall. Note that although the trajectory $\mathbf{s}$ obtained from (16) is feasible and can be traversed, the number of sample points is not enough to result in a good reconstruction. Although linear interpolation is generally used for reparameterization, here we use splines for interpolation as they provide piece-wise polynomial interpolation [45]. Although any spline could be used, we have used cubic splines as they are smooth ($\mathscr{C}^2$ continuous). We rewrite the 2D trajectory $\mathbf{s} \in \mathbb{R}^{2m}$ as a complex trajectory, denoted as $\mathbf{s}_c \in \mathbb{C}^m$. The trajectory $\mathbf{s}_c$ can be assumed to be sampled from a trajectory $s_c(p)$, $p \in [0, m-1]$ such that $[\mathbf{s}_c]_i = s_c(i)$, $i = \{0,1,2,\ldots,m-1\}$ where $[\mathbf{s}_c]_i$ are the original points from (16) and $s_c(i)$ are uniformly sampled points on $s_c(p)$. We want to approximate $s_c(p)$ by piece-wise cubic polynomial curves, i.e., between every two consecutive points $[i, i+1]$, we have $s_c(p) = P_i^3(p)$, $p \in [i, i+1]$ where $P_i^3(p) = c_{i,0} + c_{i,1}p + c_{i,2}p^2 + c_{i,3}p^3$ with $c_{i,k} \in \mathbb{C}$, $k \in \{0,1,2,3\}$. Interpolation is done by evaluating the trajectory locations using the obtained piece-wise polynomials at points finer than the given i's depending on the oversampling factor (OSF). The interpolated trajectory is rewritten as a $2m \cdot \mathrm{OSF}$ length vector by concatenating the real and imaginary parts, denoted as $\mathbf{s}_{\mathrm{SIP}}' \in \mathbb{R}^{2m\cdot\mathrm{OSF}}$. Such interpolation using cubic splines provides enough sample points to ensure good image recovery. This also provides another advantage of being able to control the read-out time by varying the interpolation factor. The formulation of the SIP method in (16) is mathematically similar to the one of the projection method in the discrete domain.

In case we use the SIP method with a random-like input, certain permutations of the points might result in a better performance than others. Hence, the method can be modified to permute the sequence of points until a permutation with a desired reconstruction quality is obtained. Note that this does not have to be the permutation leading to the TSP sequence in order to perform well. This also holds for all the methods described previously.

2.4.4. *A designed as a permutation matrix*

When $\mathbf{c}$ is a TSP solution (denoted as $\mathbf{c}_{TSP}$) to the randomly sampled points, we can consider it as a special permutation (Π_{TSP}) of the sampled points such that the path covers the shortest distance. This may or may not be a feasible trajectory. Thus, we propose to find a different permutation matrix $\Pi \in \mathbb{R}^{m\times m}$ that is not necessarily a TSP solution but leads to a feasible trajectory. The idea is to obtain a random-like trajectory by alternatively optimizing over Π and $\mathbf{s}$. This is done by first initializing $\mathbf{c}$ by randomly sampling points from the density function $\pi \propto 1/|k|^2$ [27,46]. Then we initialize $\mathbf{s}$ similar to $\mathbf{c}$ by randomly sampling points from the same density function π and then find a permutation matrix Π such that points in $\mathbf{c}$ that are close to points in $\mathbf{s}$ are paired together. By doing this, a feasible $\mathbf{s}$ is obtained using the resultant Π. This is repeated until updates of $\mathbf{s}$ converge $(e = \|\mathbf{s}_{it} - \mathbf{s}_{it-1}\|^2 < \varepsilon$, where it is the current iteration and ε is a small positive number). The related problem can be written as

$$\begin{aligned} \min_{\mathbf{s}\in\mathscr{S}^m,\ \prod^{(2)}=\mathrm{blkdiag}_2(\prod)} \quad & \frac{1}{2}\|\mathbf{s}-\prod^{(2)}\mathbf{c}\|_2^2 \\ \text{subject to} \quad & \prod \mathbf{1} = \mathbf{1} \\ & \mathbf{1}^T\prod = \mathbf{1}^T \\ & \prod_{i,j} \in \{0,1\} \end{aligned} \tag{17}$$

Note that (17) is convex in $\mathbf{s}$ for fixed $\Pi^{(2)}$ as it is the projection of $\Pi^{(2)}\mathbf{c}$ onto the convex set $\mathscr{S}^m$. On the other hand, although for fixed $\mathbf{s}$ the problem is not convex in $\Pi^{(2)}$ due to the Boolean constraints in the entries of $\Pi^{(2)}$, the solution for $\Pi^{(2)}$ can be obtained efficiently by solving a bipartite matching problem[1] [47]. This method will be called the projection with permutation (PP) method henceforth.

[1] https://in.mathworks.com/matlabcentral/fileexchange/24134-gaimc-graph-algorithms-in-matlab-code

2.5. Experiments

We discuss the methods used to test and compare the performance of the various methods discussed previously in the following subsections.

2.5.1. k-space data estimation

The k-space data for an image is generated by taking the non-uniform Fourier transform of the image at the trajectory points. The non-uniform fast Fourier transform (NUFFT) is used because the k-space points do not necessarily lie on the Cartesian grid. The NUFFT is implemented using Fessler's Michigan image reconstruction toolbox (MIRT) [48]. It is assumed that the data thus obtained is the under-sampled data from an MRI machine and is then used to reconstruct the image as described below.

2.5.2. Reconstruction of image

An MRI image is sparse in various transform domains such as the wavelet domain, finite difference domain or discrete cosine transform (DCT) domain [12]. Let $\mathbf{X}$ be the desired image to be estimated. Then, by CS theory, the image can be non-linearly reconstructed from a sparsely and incoherently sampled k-space using, for example, [12]:

$$\widehat{\mathbf{X}} = \arg\min_{\mathbf{x}} \|\text{NUFFT}(\mathbf{X}) - \mathbf{Y}\|_2^2 + \lambda_1 \|\mathscr{W}(\mathbf{X})\|_1 + \lambda_2 \|\mathbf{X}\|_{\text{TV}} \tag{18}$$

where $\mathbf{Y}$ is the observed k-space data, $\mathscr{W}(\cdot)$ is the wavelet transform and $\|\cdot\|_{\text{TV}}$ is the total variation (TV) norm:

$$\|\mathbf{X}\|_{\text{TV}} = \sum_i \sum_j \sqrt{|\mathbf{X}_{i+1,j} - \mathbf{X}_{i,j}|^2 + |\mathbf{X}_{i,j+1} - \mathbf{X}_{i,j}|^2} \tag{19}$$

Numerous sparsity penalties have been used in MRI such as wavelets and finite differences. Note that (18) is a convex optimization problem which can be solved using a non-linear conjugate gradient method with a fast and cheap backtracking line-search [12,49].

2.5.3. Performance metrics

To compare the reconstruction performances of different methods, we use structural similarity index (SSIM) and peak signal-to-noise ratio (PSNR) as measures. SSIM provides a comparison in perception of the two images using local statistics over windows $\hat{\mathbf{x}}$ and $\mathbf{x}$ of $\widehat{\mathbf{X}}$ and $\mathbf{X}$, respectively as

$$\text{SSIM}(\mathbf{x}, \hat{\mathbf{x}}) = \frac{(2\mu_x\mu_{\hat{x}} + c_1)(2\sigma_{x\hat{x}} + c_2)}{(\mu_x^2 + \mu_{\hat{x}}^2 + c_1)(\sigma_x^2 + \sigma_{\hat{x}}^2 + c_2)} \tag{20}$$

where, μ_x, $\mu_{\hat{x}}$, σ_x, $\sigma_{\hat{x}}$, $\sigma_{x\hat{x}}$ are the local means, standard deviations, and cross-covariance. $c_1 = 10^{-4}$ and $c_2 = 9 \times 10^{-4}$ are constants. The SSIM of the whole image is calculated by taking the mean of SSIMs over all windows. PSNR is a measure of the error in intensity values and is given by

$$\text{PSNR} = 10\log_{10} \frac{(\max_{i,j} \mathbf{X}(i,j))^2}{\text{MSE}} \tag{21}$$

where, $\text{MSE} = \frac{1}{N^2} \sum_{i,j=0}^{N} [\mathbf{X}(i,j) - \widehat{\mathbf{X}}(i,j)]^2$. Mean SSIM and mean PSNR are calculated from 100 different trials for each method for better comparison.

2.5.4. T_2 decay

T_2 decay is the decay of the transverse magnetization (M_{xy}) due to interactions among the close by spins. This is an exponential decay with time-constant T_2 which depends on the tissue. Long read-out times result in blurring and loss of SNR due to T_2 decay [50]. Since most of the trajectories discussed here have long read-out times, we study the effect of T_2 decay by comparing the amplitude loss (AL) of the peak of the point spread function (PSF) [50]. To obtain the PSF, the NUFFT $S(k_x, k_y)$ of a 256 × 256 point image $I(x,y)$ is calculated first using trajectories by different methods. The PSF is then obtained by

$$\text{PSF} = \text{NUFFT}^{-1}(S(k_x, k_y)),$$

where NUFFT^{-1} is the inverse NUFFT operator. To include the effect of T_2 decay, $\text{PSF}_{\text{decay}}$ is calculated by weighting the k-space signal $S(k_x, k_y)$ by the exponential function $e^{(-t/T_2)}$, $0 \le t \le T$ and then taking the inverse NUFFT.

$$\text{PSF}_{\text{decay}} = \text{NUFFT}^{-1}(S(k_x, k_y)e^{-t/T_2}), \quad 0 \le t \le T,$$

where T is the total read-out time. The percentage amplitude loss is then calculated as

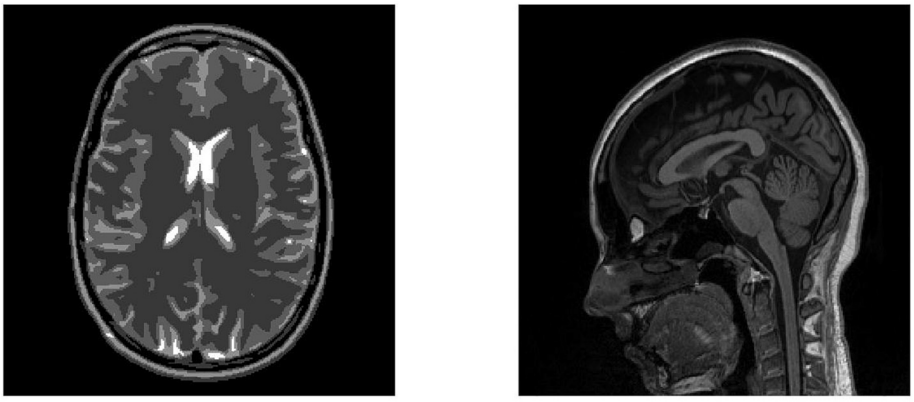
$$\text{AL} = \frac{\max(\text{PSF}) - \max(\text{PSF}_{\text{decay}})}{\max(\text{PSF})} \times 100\%.$$

2.5.5. Algorithm to solve the general projection problem

A proximal gradient descent based iterative algorithm is used to solve the general projection problem (11) and is described in Appendix C. The algorithm

2.5.6. Simulation framework

In this section, we test the performance of the methods discussed in the previous sections for the reconstruction of a 128 × 128 Shepp-Logan phantom, a 256 × 256 realistic analytical head phantom [51] and a T1-weighted sagittal brain MRI image (obtained using Cartesian imaging) as shown in Fig. 2:. All the simulations are performed in



(a) (b)

Fig. 2. Reference images (256 × 256) for simulation results: (a) realistic analytical phantom, (b) an MRI image.

MATLAB 2018b in a 64 bit UBUNTU system using an Intel Core i5-4460 CPU, with 16 GB RAM. The test images are made complex by adding complex Gaussian random noise. The gradient constraints are taken as G_{max} = 40mT/m and S_{max} = 150mT/m/ms to be consistent with the TOC and projection methods. The sampling frequency is taken to be 250 kHz. The variable density function $\pi \propto 1/|k|^2$ is used for all trajectories. The k-space data for the test images is then obtained by using NUFFT [48] along the feasible trajectories. The results are computed over multiple trials (each trial consists of a new selection of reference curve **c**). We define the sampling factor in the same way as in [36] as $N^2/m \times 100\%$. Here, N corresponds to the image size and m is the number of sample points in the k-space trajectory. The initial reference trajectory **c** in the TOC and projection methods are taken such that the sampling factor is about 50%.

3. Results and discussion

In this section, we compare and discuss the performance of the proposed methods for different test images.

3.1. Simulation results

3.1.1. 128 × 128 Shepp-Logan phantom

Simulation results of all methods are summarized in Table 1. For the TOC method, **c** is considered a 400 point TSP curve to get a sampling factor of about 50%. For the projection-CVP method, **c** is a TSP curve with 128 × 128/4 = 4096 points. The same initial trajectory **c** is used for the projection-CAP, COLT-TSP and SIP-TSP methods. A velocity of $0.5v_{max}$ and an acceleration of $0.3a_{max}$ are used for parameterizing **c** to get a sampling factor of about 50% for the projection-CVP and projection-CAP methods, respectively. The projection-CAP provides a better reconstruction quality compared to its CVP counterpart. An improvement of about 0.1951 in SSIM is observed on average. In case of the COLT-TSP method, with increasing velocity v of the parameterization and λ, the mean read-out time reduces and so does the reconstruction performance as shown in Fig. 4:(a). A decline in performance is expected since the number of sample points in the trajectory reduces. As can be observed from the figure, the read-out time reduces drastically with α but the decline in performance is not as drastic. Hence, this method can be used as a method to reduce read-out time with λ = 1. The same is observed for the COLT-random method as well, however, the COLT-random method provides a better reconstruction compared to COLT-TSP. A mean improvement of about 0.2757 in SSIM is observed with a 50% reduction in read-out time over the COLT-TSP method. Similarly, the SIP-random provides a much better reconstruction performance than the SIP-TSP with an improvement of 0.2545 in SSIM. The GBP methods (Type-1 and Type-2) are used to obtain TSP-based and random-like trajectories with 8192 = 128 × 128/2 points. In both types of methods, random-like trajectories perform better than TSP-based trajectories. For random-like trajectories, the GBP (Type-2) method performs better than the GBP (Type-1) method. For TSP-based trajectories, the two types of GBP methods show very similar performance. The PP method with 5461 points outperforms all other methods with just 21.84 ms read-out time. In Fig. 3, the simulation results for one trial of all methods are shown. The feasible TSP-based trajectories for these methods are shown in Fig. 3(a), (b), (c), (d), (e), (h) and (j). Trajectories obtained from the COLT, SIP, GBP (Type-1) and GBP (Type-2) methods when random points are used as **c** are shown in Fig. 3(f), (g), (i) and (k), respectively. The trajectory from the PP method is shown in Fig. 3(l).

Table 1
Mean read-out time and SSIM over 100 trials for the 128 × 128 phantom image reconstruction using different methods under a single-shot scheme to obtain feasible trajectories.

Method	Read-out time (ms)	Sampling factor	SSIM	PSNR (dB)
TOC	33.11	50.50%	0.7607	25.46
Projection-CVP ($v = 0.5v_{max}$)	33.95	51.80 %	0.6813	28.25
Projection-CAP ($a = 0.3a_{max}$)	31.19	47.59 %	0.8764	32.20
COLT-TSP ($v = 0.2v_{max}$)	16.76	25.57 %	0.5620	22.70
COLT-random ($v = 0.8v_{max}$)	15.46	23.57 %	0.8377	27.43
SIP-TSP (OSF = 2)	32.77	50 %	0.6726	25.81
SIP-random (OSF = 2)	32.77	50 %	0.9271	32.22
GBP (Type-1) **c** = random	32.77	50 %	0.9113	29.25
GBP (Type-1) **c** = TSP	32.77	50 %	0.8845	34.08
GBP (Type-2) **c** = random	32.77	50 %	0.9362	28.96
GBP (Type-2) **c** = TSP	32.77	50 %	0.8800	33.63
PP method	21.84	33.33 %	0.9942	42.54

3.1.2. 256 × 256 analytical phantom and brain MRI images

For higher resolution images, we compare the methods for single-shot and multi-shot (2-shot and 4-shot) trajectories. Multi-shot trajectories are constructed only for TSP-based trajectories. To obtain the multi-shot trajectories, we divide the points sampled from the density function $\pi \propto 1/|k|^2$ into two halves and four quadrants for 2-shot and 4-shot trajectories, respectively with some overlap between the regions. A TSP-based trajectory is obtained for each region and a corresponding feasible TSP-based trajectory is to be traversed in one RF excitation. The performance of all methods is summarized in Table 2 (also see Fig. 6:). For the TOC method, 2500 points are sampled on the k-space such that the feasible single-shot trajectory takes a read-out time of about 133 ms. For projection-CVP, projection-CAP, COLT-TSP and SIP-TSP methods, 16384 (256 × 256/4) points are sampled to get **c**. The velocity and acceleration used for the parameterization are taken as $0.5v_{max}$, $0.25v_{max}$ and $0.8a_{max}$ for the projection-CVP, COLT-TSP and projection-CAP methods, respectively. The projection-CAP method performs better than the projection-CVP method and comparable with the TOC method. Similar to the lower resolution phantom image, random-like trajectories using the COLT and SIP methods perform better than the TSP-based trajectories obtained by these methods. The GBP method is used to create 21845 (256 × 256/3) point TSP-based and random-like trajectories. GBP (Type-1) random-like trajectories perform comparable to the GBP (Type-2) TSP-based trajectories on average with a read-out time of nearly 87 ms. The PP method provides a good reconstruction with an SSIM of 0.8191 and a significantly lower read-out time of nearly 43 ms in a single shot.

Fig. 4(b) compares the performance of the projection-CVP and COLT-TSP methods for the 256 × 256 Shepp-Logan phantom image reconstruction with the same TSP-based trajectories as discussed above. The mean read-out times, PSNRs and SSIMs are plotted as functions of the fraction α of v_{max} used for CVP in both methods. The variation with respect to λ in the COLT-TSP method is also depicted. For lower values of α (< 0.5), the projection method performs really well with SSIMs greater than 0.9 and high PSNRs. However, these values are achieved at very high read-out times. The effect of α is not observed to be as prominent for the COLT-TSP method. For a similar mean PSNR of 29dB, the read-out time for the COLT-TSP method is 37.60 ms as compared to 137.13 ms for the projection-CVP method.

3.1.3. 256 × 256 phantom MRI images

Fully sampled k-space data using Cartesian trajectory was acquired by scanning a spherical (10496625) and a cylindrical (08624186) Siemens phantom placed side-by-side on a 3T Siemens Prisma machine. The scan was done by placing the arrangement inside a 20 channel head coil. A single image is obtained by combining the reconstructed images from each channel by root-sum-of-squares (RSS) method. In the RSS method, firstly, inverse FT is applied to k-space data of each coil and then the individual images are combined as

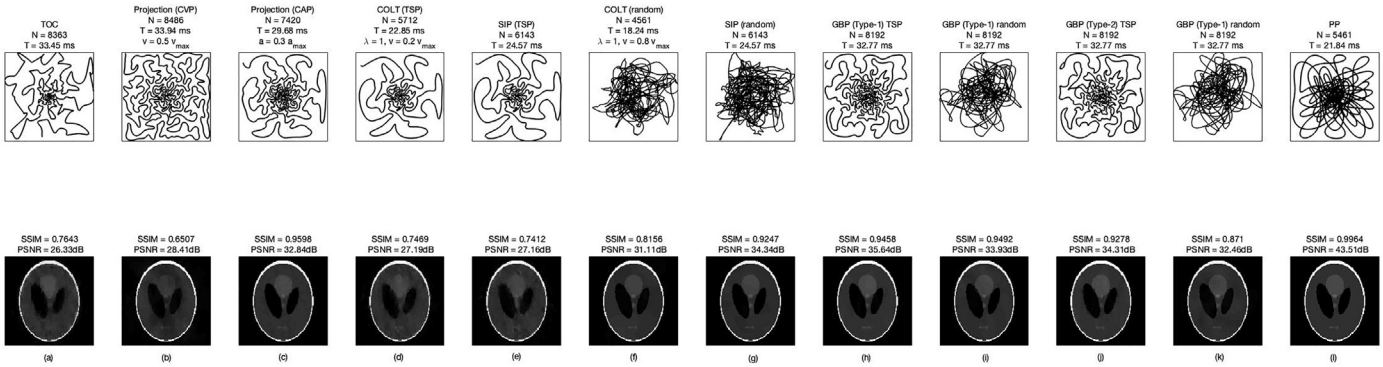


Fig. 3. Comparison of performance of TOC, projection-CVP, projection-CAP, COLT, SIP, GBP and PP method for 128 × 128 phantom image. $\lambda = 1$ for COLT method. OSF = 1.5 for SIP method.

$$X_{rss} = \left(\sum_{i=1}^{n_c} |X_i|^2 \right)^{1/2} \tag{22}$$

where X_i is the image reconstructed using the i^{th} coil and n_c is the total number of coils (here, $n_c = 20$). We obtain 256 × 256 images of resolution 1 × 1 × 4 mm using TSE sequence with TR/TE of 600/6.4 ms. We use the single reconstructed image X_{rss} to get the *k*-space data at the proposed non-Cartesian trajectories and reconstruct the images back using (18). The two images considered here are shown in Fig. 5:. Fig. 5: (a) is a sagittal image of the arrangement of the two phantoms and (b) is a coronal image. The trajectories used here are the same as the ones used for 256 × 256 images previously for the analytical phantom and brain MRI images. The results are summarized in in Table 3.

3.2. Robustness of methods

To test the robustness of the methods discussed in this paper, we compare the histograms of SSIMs of the reconstructed 256 × 256 analytical realistic brain phantom image over 100 trials for different reference curves c as shown in Fig. 7:. It is observed that the TSP-based trajectories obtained using the TOC and GBP (Type-2) methods and random-like trajectories obtained using COLT, SIP and PP methods provide consistent performances over trials. SSIMs from other methods vary a lot where some trials result in a really poor performance with an SSIM less than 0.5 and some have a really good performance with SSIM greater than 0.95. Similar to most methods that require an initialization, the proposed methods are also susceptible to such initializations, such as the sample points on the *k*-space, choice of **c** from the sampled points (in random-like trajectory case). Since selection of a trajectory is done offline, it is advised to do multiple simulations and finally choose a trajectory that leads to the best reconstruction performance on the test images: phantom and MRI of different anatomical structures.

3.3. Effect of T_2 decay

Table 4 shows the percentage AL for $T_2 = 90$ ms and 400 ms (values typical for gray matter and cerebrospinal fluid, respectively at 1.5T field strength [1]) for trajectories obtained by different methods. Methods resulting in shorter trajectories suffer less from T_2 decay. Hence, proposed methods like COLT, SIP, GBP and PP with shorter read-out times are expected to be robust to the effects of T_2 decay, especially in single-shot trajectories.

3.4. Discussion

The problem of finding short yet feasible trajectories for MRI sampling has been addressed here. The results indicate that random-like trajectories are better than TSP-based trajectories under the proposed projection-based method. For 128 × 128 images, the PP method is observed to be better than other existing and proposed methods in terms of reconstruction performance and read-out time. This method is suggested to be used in lower resolution scans. For 256 × 256 images, the PP method provides the shortest trajectories that can be implemented in practice. Although COLT-random and SIP-random provide better reconstruction than other proposed methods, these cannot be implemented in the multi-shot scenario. The single-shot scheme of COLT-random and SIP-random may not be practical in all cases because of longer read-out times. The same is true for the GBP methods. Hence, for high resolution single-shot imaging, the PP method and COLT-random methods are recommended to be used if the read-out time can

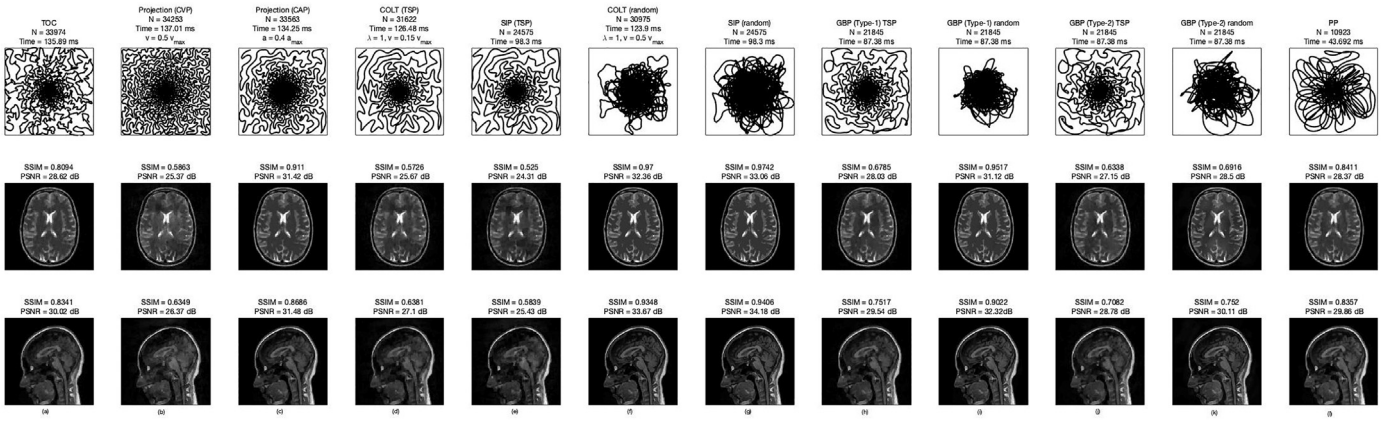


Fig. 6. Trajectories and reconstructed 256 × 256 analytical realistic brain phantom and brain MRI images for the TOC, projection-CVP, projection-CAP, COLT, SIP, GBP and PP methods.

Table 2
Mean read-out time and SSIM over 100 trials for the 256 × 256 analytical realistic brain phantom and brain MRI image reconstruction using different methods under single-shot and multi-shot schemes to obtain feasible trajectories. Results in bold indicate best reconstruction performance in a single-shot scheme.

Method	#shots	Read-out	Sampling	Phantom image		MRI image	
		time (ms)	Factor	SSIM	PSNR (dB)	SSIM	PSNR (dB)
	1	133.35	50.86 %	0.8131	28.59	0.8325	29.94
TOC	2	89.95	68.62 %	0.8628	29.86	0.8566	30.72
	4	37.27	65.86 %	0.8357	37.27	0.8427	30.24
Projection-CVP	1	137.13	52.31 %	0.5352	23.53	0.6105	25.82
($v = 0.5v_{max}$)	2	86.51	66.00 %	0.6492	26.30	0.6892	27.15
	4	37.88	57.80 %	0.3738	17.55	0.4492	21.22
Projection-CAP	1	136.05	51.89 %	0.8133	29.67	0.8134	30.36
($a = 0.4a_{max}$)	2	86.22	65.78 %	0.9395	31.66	0.8912	29.87
	4	34.27	52.29 %	0.4291	19.67	0.4594	21.90
COLT-TSP	1	75.21	28.69 %	0.5742	25.53	0.6557	26.85
($v = 0.25v_{max}$)	2	56.86	43.38 %	0.7274	27.25	0.7453	27.59
	4	21.15	32.27 %	0.4236	19.47	0.5088	23.13
SIP-TSP (OSF = 1.5)	1	98.30	37.49 %	0.5726	24.55	0.6641	26.98
	2	66.66	50.85 %	0.7348	27.65	0.7592	27.61
	4	27.90	42.57 %	0.4187	19.26	0.5172	23.13
COLT-random ($v = v_{max}$)	1	61.59	23.49 %	0.8893	30.11	0.8728	29.55
SIP-random (OSF = 1.5)	1	98.30	37.49 %	**0.9624**	**31.52**	**0.9350**	**31.21**
GBP (Type-1)	1 (c = TSP)	87.38	33.33 %	0.6994	27.39	0.7484	28.96
	1 (c = random)	87.38	33.33 %	0.8095	29.58	0.8278	31.16
GBP (Type-2)	1 (c = TSP)	87.38	33.33 %	0.7919	28.35	0.8115	29.76
	1 (c = random)	87.38	33.33 %	0.6792	26.93	0.7164	28.32
PP	1	43.69	16.66 %	0.8535	28.73	0.8191	27.10

be close to 40 ms and 60 ms, respectively. In the multi-shot scheme, the 4-shot TSP trajectory with TOC method performs the best.

A drawback of the TSP and random-like trajectories from projection-based methods is that the sampled points do not follow the density function. This is solved using the PP method which provides a trajectory with points close to the original density. The better performance of the PP method compared to other existing and proposed methods proves that the TSP-based trajectory is not necessarily the best trajectory to follow the density.

The proposed trajectories have not yet been tested on an MRI scanner. Since the trajectories are designed to satisfy the physical constraints and are feasible, they should be implementable in a scanner without any difficulty. Since the trajectories are smooth, there should not be very loud acoustic noise. Some possible sources of image artifacts might occur due to eddy currents and deviation from the desired trajectory which are common in non-Cartesian trajectories. There could also be motion-related errors in case of trajectories with longer read-out times. Various methods exist in literature to address these potential issues [38,52].

4. Conclusion

We focus on a recent method for obtaining feasible trajectories which uses the projection of a CVP curve onto a convex set and we propose a generalization of this method to obtain faster trajectories. We have discussed some special instances of this generalized method starting from some TSP-based and random-like trajectories. We have proposed a new parameterization scheme based on constant acceleration which is observed to perform better than CVP. Using the COLT, SIP, GBP and PP methods, we have shown the effectiveness of random-like trajectories in reducing the read-out time while maintaining a good reconstruction performance. The general framework opens many more possibilities for trajectory design. The considered methods consist of solving a constrained convex optimization problem for which an iterative algorithm in the dual space has been provided. They also give the designer the freedom to choose from various design parameters (the weighting parameter, OSF) in order to tune the trade-off between read-out time and reconstruction performance. The choice of method in practice would be determined according to the application at hand. The trajectories designed here are feasible under the gradient constraints and hence can be implemented in a real scanner. However, the results discussed here are preliminary and may vary in practice due to imprecisions in trajectories because of eddy currents and other scanner related errors. Faster acquisition using the proposed methods would be useful in applications such as dynamic imaging, cardiac imaging, etc.

Acknowledgment

The authors would like to thank the Ministry of Electronics and Information Technology (MeitY), India for the financial support to conduct this work. We are also grateful to the IISc-TU Delft collaboration agreement.

Appendix A. Design of the CVP operator

As discussed in Section 2.2.2, the CVP matrix is the product of an interpolation matrix **L** and a selection matrix $\Pi_{\alpha,\,\mathbf{c}}$. The design method for the two matrices is described below.

Interpolation matrix **L**:

Let $\mathbf{x} \in \mathbb{R}^n$ be a vector of n points $x_1, x_2, , x_n$. To interpolate this by a factor of N, i.e., to have N points between all two consecutive points, we design the interpolated points

$$x_{i,j} = \frac{j}{N}(x_{i+1} - x_i) + x_i \tag{A.1}$$

where $x_{i,\,j}$ denotes the $(j+1)^{th}$ interpolated point between x_i and x_{i+1}, $1 \le i \le N-1$, $0 \le j \le N-1$. This operation on all points of $\mathbf{x}$ can be written in matrix form as

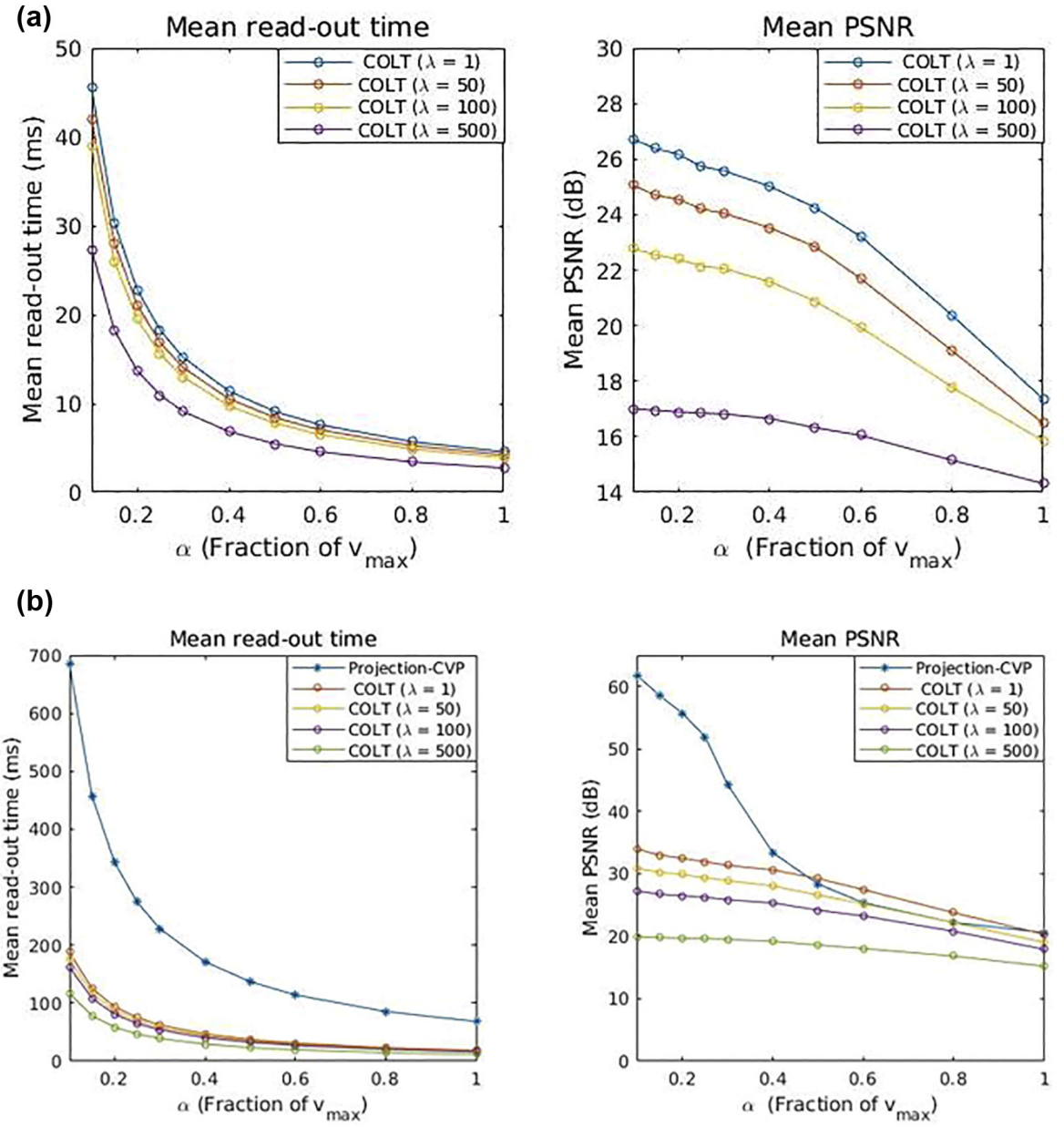


Fig. 4. Effect of velocity and λ in COLT-TSP method for (a) 128 × 128 and (b) 256 × 256 Shepp-Logan phantom and its comparison with projection-CVP method.

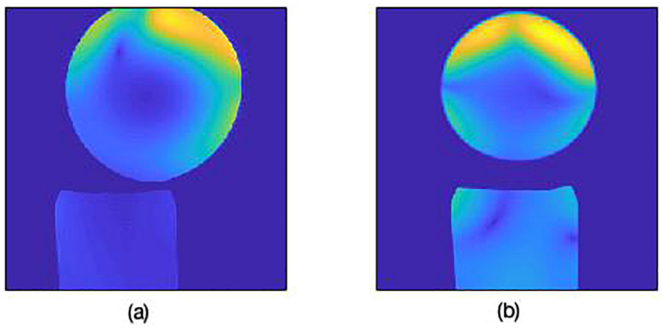


Fig. 5. (a) Phantom MRI image-1: Sagittal image and (b) Phantom MRI image-2: Coronal image.

Table 3
Mean read-out time and SSIM over 100 trials for the 256 × 256 phantom MRI image reconstruction using different methods under single-shot scheme to obtain feasible trajectories.

Method	Read-out time (ms)	MRI image-1		MRI image-2	
		SSIM	PSNR	SSIM	PSNR
TOC	133.35	0.9589	37.54	0.9611	39.13
Projection-CVP ($v = 0.5v_{max}$)	137.13	0.6826	26.35	0.6608	24.49
Projection-CAP ($a = 0.4a_{max}$)	136.05	0.8871	34.99	0.8884	35.32
COLT-TSP ($v = 0.25v_{max}$)	75.21	0.7720	30.10	0.7601	28.60
SIP-TSP (OSF = 1.5)	98.30	0.7813	30.21	0.7680	28.78
COLT-random ($v = v_{max}$)	61.59	0.9022	33.50	0.8994	33.87
SIP-random (OSF = 1.5)	98.30	0.9510	35.11	0.9493	35.61
GBP (Type-1) (c = TSP)	87.38	0.8240	32.05	0.7986	30.31
GBP (Type-1) (c = random)	87.38	0.8482	32.11	0.8508	32.22
GBP (Type-2) (c = TSP)	87.38	0.9118	34.77	0.9093	34.71
GBP (Type-2) (c = random)	87.38	0.7333	28.49	0.7204	27.31
PP	43.69	0.8932	33.57	0.9056	34.12

Table 4
Effect of T_2 decay.

Method	Amplitude Loss	
	T_2 = 90 ms	T_2 = 400 ms
TOC	47.85 %	14.96 %
Projection-CVP	48.67 %	15.34 %
Projection-CAP	48.44 %	15.23 %
COLT-TSP	32.22 %	8.84 %
COLT-random	27.58 %	7.32 %
SIP	39.16 %	11.34 %
GBP	36.01 %	10.17 %
PP	30.22 %	5.27 %

$$\mathbf{L} = \frac{1}{N}\mathbf{D}\mathbf{I}_E\mathbf{D}_1 + \mathbf{I}_E \tag{A.2}$$

where $\mathbf{D} \in \mathbb{R}^{nN\times nN} = \mathbf{I} \otimes$ diag (0,1,2,…,$N-1$), and $\mathbf{I}_E \in \mathbb{R}^{nN\times n} = \mathbf{I} \otimes \mathbf{1}_N$. We denote $\mathbf{c_L} = \mathbf{L}^{(2)}\mathbf{c} \in \mathbb{R}^{2nN}$, where $\mathbf{L}^{(2)} = \text{blkdiag}_2(\mathbf{L})$.

Selection matrix, $\Pi_{\alpha,\,\mathbf{c}}$:

As discussed in Section 2.1, consecutive points in $\mathbf{c}$ are assumed to be traversed in time t_s. Since the distance between points is not the same, this results in varying velocity through the trajectory. To achieve CVP, we need the distance between consecutive points to be the same. To do this, let us introduce t_i as the time taken to traverse two consecutive points in $\mathbf{c_L}$ in terms of instantaneous velocity, which is given by

$$t_i = \frac{l_{i+1} - l_i}{\alpha v_{\max}}, \quad 1 \le i \le (n-1)N - 1 \tag{A.3}$$

where $l_i = \Sigma_{j=1}{}^{i}\|\mathbf{c_L}_{j+1} - \mathbf{c_L}_j\|_2$ as defined in (5) with $\mathbf{c_L}_j = [[\mathbf{c}_{\mathrm{Lx}}]_j \quad [\mathbf{c}_{\mathrm{Ly}}]_j]^T$. Points from $\mathbf{c_L}$ are then chosen using a selection matrix $\Pi_{\alpha,\,\mathbf{c}}$ such that the total time between consecutive selected points is t_s. For this, we define the set

$$\mathscr{C} = \{j\colon t_{j+1} - t_j \simeq t_s, 1 \le j \le n-1\}$$

and the selection matrix $\Pi_{\alpha,\,\mathbf{c}} = [\mathbf{e}_1\ \mathbf{0}\ \mathbf{e}_2\ \mathbf{0}\ \mathbf{e}_3\ \mathbf{0}\ \ldots\ \mathbf{0}\ \mathbf{e}_m]$ with $\mathbf{e}_i$'s at the column positions given by the set $\mathscr{C}$. Here, $\{\mathbf{e}_i\}_{i=1}{}^{m}$ are the columns of an $m \times m$ identity matrix.

The procedure to obtain $\Pi_{\alpha,\,\mathbf{c}}$ is summarized in Algorithm 1.

Algorithm 1: Algorithm to construct CVP matrix $\mathbf{A}_v(\alpha, \mathbf{c})$

Input: $\mathbf{c}$, α, N

Result: $\mathbf{A}_v(\alpha, \mathbf{c})\mathbf{c} = \Pi_{\alpha,\,\mathbf{c}}\mathbf{L}$

1. Construct $\mathbf{L} = \frac{1}{N}\mathbf{D}\mathbf{I}_E\mathbf{D}_1 + \mathbf{I}_E$.
2. Construct $\mathbf{c_L} = \mathbf{L}\mathbf{c}$.
3. Construct $t_i = \frac{l_{i+1}-l_i}{\alpha v_{\max}}$, where $l_i = \sum_{j=1}^{i}\|\mathbf{c_L}_{j+1} - \mathbf{c_L}_j\|_2, \quad 1 \le i \le (n-1)N$.
4. Obtain set $\mathscr{C} = \{j\colon t_{j+1} - t_j \simeq t_s, 1 \le j \le n-1\}$.
5. Construct $\Pi_{\alpha,\,\mathbf{c}}$ as a selection matrix to select indices in $\mathscr{C}$.

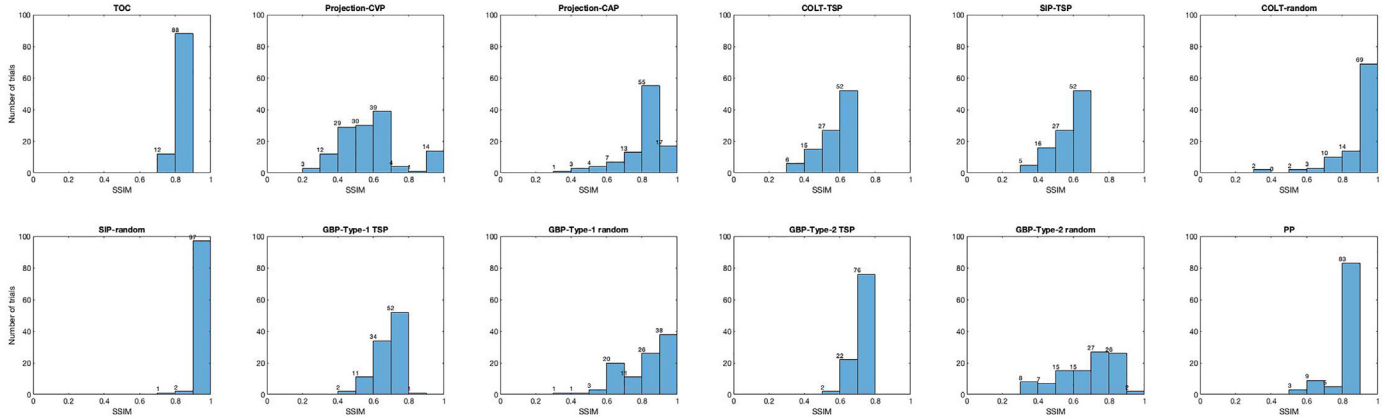


Fig. 7. Histograms with bin size of 0.05 of SSIMs of 256 × 256 analytical realistic brain phantom image using different methods.

Appendix B. Design of the CAP operator

The first step for the construction of the linear CAP operator is the same as that for the CVP operator, i.e., interpolation of $\mathbf{c}$ using $\mathbf{L}$ as given in (A.2). The difference is in the second step, i.e., in picking the points from the interpolated $\mathbf{c}$ using the selection matrix $\Pi_{\beta,\,\mathbf{c}}$. To obtain $\Pi_{\beta,\,\mathbf{c}}$ for this case, we do the following. First, find the time taken between consecutive points in $\mathbf{c}_\mathbf{L} = \mathbf{L}\mathbf{c}$ assuming acceleration $a = \beta a_{\max}$, where β is the fraction of the maximum acceleration $a_{\max} = \gamma S_{\max}$ needed for the parameterization. This results in

$$t_i = \sqrt{\frac{v_{i+1} - v_i}{\beta a_{\max}}}, \qquad 1 \le i \le (n-1)N - 1 \tag{B.1}$$

where $v_i = \frac{\|\mathbf{c}_{\mathbf{L}i+1} - \mathbf{c}_{\mathbf{L}i}\|_2}{t_s}$ as defined in (6) with $\mathbf{c}_{\mathbf{L}i} = [[\mathbf{c}_{\mathbf{L}\mathrm{x}}]_i \quad [\mathbf{c}_{\mathbf{L}\mathrm{y}}]_i]^T$. Then, a set $\mathscr{C}$ is defined as before. Finally, define $\Pi_{\beta,\,\mathbf{c}}$ as the selection matrix $\Pi_{\beta,\,\mathbf{c}} = [\mathbf{e}_1 \quad \mathbf{0} \quad \mathbf{e}_2 \quad \mathbf{0} \quad \mathbf{e}_3 \quad \mathbf{0} \ \ldots \ \mathbf{0} \quad \mathbf{e}_m]$ with $\mathbf{e}_i$'s at the column positions given by the set $\mathscr{C}$. This procedure is summarized in Algorithm 2.

Algorithm 2: Algorithm to construct CAP matrix $\mathbf{A}_\mathrm{a}(\beta, \mathbf{c})$

Input: $\mathbf{c}$, β, N

Result: $\mathbf{A}_\mathrm{a}(\beta, \mathbf{c}) = \Pi_{\beta,\,\mathbf{c}}\mathbf{L}$

1. Construct $\mathbf{L} = \frac{1}{N}\mathbf{D}\mathbf{I}_E\mathbf{D}_1 + \mathbf{I}_E$.
2. Construct $\mathbf{c}_\mathbf{L} = \mathbf{L}\mathbf{c}$.
3. Construct $t_i = \sqrt{\frac{v_{i+1} - v_i}{\beta a_{\max}}}, \qquad 1 \le i \le (n-1)N - 1$.
4. Obtain set $\mathscr{C} = \{j\colon t_{j+1} - t_j \simeq t_s, 1 \le j \le n-1\}$.
5. Construct $\Pi_{\beta,\,\mathbf{c}}$ as a selection matrix to select indices in $\mathscr{C}$.

Appendix C. Algorithm to solve the general projection problem

An iterative algorithm based on proximal gradient descent on the dual is used to solve the following general projection problem.

$$\begin{aligned} &\min_{\mathbf{s}\in\mathbb{R}^{2m}} \quad \frac{1}{2}\|\mathbf{s} - \mathbf{A}^{(2)}\mathbf{c}\|_2^2 \\ &\text{subject to } \|\mathbf{D}_1^{(2)}\mathbf{s}\|_\infty \le \gamma G_{\max} \\ &\qquad\qquad \|\mathbf{D}_2^{(2)}\mathbf{s}\|_\infty \le \gamma S_{\max} \end{aligned} \tag{C.1}$$

$$\equiv \min_{\mathbf{s}\in\mathbb{R}^{2m}} \quad \frac{1}{2}\|\mathbf{s} - \mathbf{A}^{(2)}\mathbf{c}\|_2^2 + \mathcal{I}_{\mathscr{A}_{\alpha_1}}(\mathbf{D}_1^{(2)}\mathbf{s}) + \mathcal{I}_{\mathscr{A}_{\alpha_2}}(\mathbf{D}_2^{(2)}\mathbf{s}) \tag{C.2}$$

where $\alpha_1 = \gamma G_{\max}$, $\alpha_2 = \gamma S_{\max}$, $\mathscr{A}_{\alpha_i}(\mathbf{s}) = \{\mathbf{s} \in \mathbb{R}^{2m}\colon \|\mathbf{s}\|_\infty \le \alpha_i\}$, $i \in \{1,2\}$ and

$$\mathcal{I}_X(\mathbf{x}) = \begin{cases} 0 & ; \quad \mathbf{x} \in \mathscr{X} \\ +\infty & ; \quad \mathbf{x} \notin \mathscr{X} \end{cases}$$

Since, $\mathcal{I}_{\mathscr{A}_{\alpha_i}}(\mathbf{x}) = \sup_{\mathbf{y}\in\mathbb{R}^{2m}} \langle \mathbf{x}, \mathbf{y}\rangle - \alpha_i \|\mathbf{y}\|_*$ where $\|\cdot\|_*$ is the dual norm, (C.2) can be written as

$$\begin{aligned} &\min_{\mathbf{s}\in\mathbb{R}^{2m}} \quad \frac{1}{2}\|\mathbf{s} - \mathbf{A}^{(2)}\mathbf{c}\|_2^2 \\ &+ \sup_{\mathbf{q}_1,\mathbf{q}_2\in\mathbb{R}^{2m}} \langle \mathbf{D}_1^{(2)}\mathbf{s}, \mathbf{q}_1\rangle - \alpha\|\mathbf{q}_1\|_1 + \langle \mathbf{D}_2^{(2)}\mathbf{s}, \mathbf{q}_2\rangle - \beta\|\mathbf{q}_2\|_1 \\ = &\sup_{\mathbf{q}_1,\mathbf{q}_2\in\mathbb{R}^{2m}} \min_{\mathbf{s}\in\mathbb{R}^{2m}} \quad \frac{1}{2}\|\mathbf{s} - \mathbf{A}^{(2)}\mathbf{c}\|_2^2 + \langle \mathbf{D}_1^{(2)}\mathbf{s}, \mathbf{q}_1\rangle + \langle \mathbf{D}_2^{(2)}\mathbf{s}, \mathbf{q}_2\rangle \\ &-\alpha\|\mathbf{q}_1\|_1 - \beta\|\mathbf{q}_2\|_1 \\ = &\sup_{\mathbf{q}_1,\mathbf{q}_2\in\mathbb{R}^{2m}} F(\mathbf{q}_1,\mathbf{q}_2) - Q(\mathbf{q}_1,\mathbf{q}_2) \\ = &\inf_{\mathbf{q}_1,\mathbf{q}_2\in\mathbb{R}^{2m}} -F(\mathbf{q}_1,\mathbf{q}_2) + Q(\mathbf{q}_1,\mathbf{q}_2) \end{aligned} \tag{C.3}$$

where $F(\mathbf{q}_1,\mathbf{q}_2) = \min_{\mathbf{s}\in\mathbb{R}^{2m}} \ \frac{1}{2}\|\mathbf{s} - \mathbf{A}^{(2)}\mathbf{c}\|_2^2 + \langle \mathbf{D}_1^{(2)}\mathbf{s}, \mathbf{q}_1\rangle + \langle \mathbf{D}_2^{(2)}\mathbf{s}, \mathbf{q}_2\rangle$ is differentiable and convex and $Q(\mathbf{q}_1,\mathbf{q}_2) = \alpha\|\mathbf{q}_1\|_1 + \beta\|\mathbf{q}_2\|_1$ is non-differentiable and convex. Hence, the dual Eq. (C.3) can be solved using proximal gradient descent [53].

To evaluate $F(\mathbf{q}_1,\mathbf{q}_2)$, we need to find

$$\mathbf{s}^*(\mathbf{q}_1,\mathbf{q}_2) = \arg\min_{\mathbf{s}\in\mathbb{R}^{2m}} \frac{1}{2}\|\mathbf{s} - \mathbf{A}_\mathbf{c}\mathbf{c}\|_2^2 + \langle \mathbf{D}_1^{(2)}\mathbf{s}, \mathbf{q}_1\rangle + \langle \mathbf{D}_2^{(2)}\mathbf{s}, \mathbf{q}_2\rangle \tag{C.4}$$

which has the closed form solution

$$\mathbf{s}^*(\mathbf{q}_1,\mathbf{q}_2) = \mathbf{A}_\mathbf{c}\mathbf{c} - \mathbf{D}_1^{(2)^T}\mathbf{q}_1 - \mathbf{D}_2^{(2)^T}\mathbf{q}_2 \tag{C.5}$$

With this, the proximal gradient descent algorithm to solve (C.3) is summarized below in Algorithm 3. Here, $L = \|\mathbf{D}_1{}^{(2)^T}\mathbf{D}_1{}^{(2)} + \mathbf{D}_2{}^{(2)^T}\mathbf{D}_2{}^{(2)}\|$ is the Lipschitz constant of $\nabla F(\mathbf{q}_1,\mathbf{q}_2)$. The algorithm converges at a rate $O(1/t^2)$ for a fixed step size $t \in (0, 1/L]$ [53,54].

Algorithm 3:

Input: $\mathbf{c} \in \mathbb{R}^{2n}$, $\lambda > 0$, $\varepsilon > 0$, L, α, β

1. Initialization: $\mathbf{q}^{(0)} = [\mathbf{q}_1{}^{(0)^T}\mathbf{q}_2{}^{(0)^T}]^T = \mathbf{0}$, $\mathbf{y}^{(0)} = \mathbf{q}^{(0)}$, $t = 1/L$, $\mathbf{e} = \mathbf{c}$, $k = 0$
2. while $\|\mathbf{e}\|_2{}^2 < = \varepsilon$ do

3. $k = k + 1$
4. $\mathbf{z}^{(k)} = \mathbf{y}^{(k-1)} + t \nabla F(\mathbf{y}^{(k-1)})$
5. $\mathbf{q}^{(k)} = \arg\min_{\mathbf{q}} \frac{1}{2t}\|\mathbf{z}^{(k)} - \mathbf{q}\|_2^2 + Q(\mathbf{q})$
6. $\mathbf{y}^{(k)} = \mathbf{q}^{(k)} + \frac{k-1}{k+2}(\mathbf{q}^{(k)} - \mathbf{q}^{(k-1)})$
7. $\mathbf{e} = \mathbf{q}^{(k)} - \mathbf{q}^{(k-1)}$
8. end

Output: $\mathbf{s} = \mathbf{s}^*(\mathbf{q}_1{}^{(k)}, \mathbf{q}_2{}^{(k)})$

References

[1] Buxton RB. Introduction to functional magnetic resonance imaging: principles and techniques. Cambridge University Press; 2009.
[2] R. S. Likes, Moving gradient zeugmatography, US Patent 4,307,343 (Dec. 22 1981).
[3] Kim D-h, Adalsteinsson E, Spielman DM. Simple analytic variable density spiral design. Magn Reson Med 2003;50(1):214–9.
[4] Lauterbur P. Image Formation by Induced Local Interactions: Examples Employing Nuclear Magnetic Resonance. 2020.
[5] Hennig J, Nauerth A, Friedburg H. RARE imaging: a fast imaging method for clinical MR. Magn Reson Med 1986;3(6):823–33.
[6] Frahm J, Haase A, Matthaei D. Rapid three-dimensional MR imaging using the FLASH technique. J Comput Assist Tomogr 1986;10(2):363–8.
[7] Mansfield P. Multi-planar image formation using NMR spin echoes. J Phys Solid State Phys 1977;10(3):L55.
[8] Griswold MA, Jakob PM, Heidemann RM, Nittka M, Jellus V, Wang J, et al. Generalized autocalibrating partially parallel acquisitions (GRAPPA). Magn Reson Med 2002;47(6):1202–10.
[9] Pruessmann KP, Weiger M, Scheidegger MB, Boesiger P, et al. SENSE: sensitivity encoding for fast MRI. Magn Reson Med 1999;42(5):952–62.
[10] Wright KL, Hamilton JI, Griswold MA, Gulani V, Seiberlich N. Non-cartesian parallel imaging reconstruction. J Magn Reson Imaging 2014;40(5):1022–40.
[11] Wang S, Tan S, Gao Y, Liu Q, Ying L, Xiao T, et al. Learning joint-sparse codes for calibration-free parallel MR imaging. IEEE Trans Med Imaging 2017;37(1):251–61.
[12] Lustig M, Donoho D, Pauly JM. Sparse MRI: The application of compressed sensing for rapid MR imaging. Magn Reson Med 2007;58(6):1182–95.
[13] Lustig M, Donoho DL, Santos JM, Pauly JM. Compressed sensing MRI. IEEE Signal Process Magaz 2008;25(2):72–82.
[14] Wang S, Su Z, Ying L, Peng X, Zhu S, Liang F, et al. Accelerating magnetic resonance imaging via deep learning. 2016 IEEE 13th International Symposium on Biomedical Imaging (ISBI). IEEE; 2016. p. 514–7.
[15] Wang S, Xiao T, Tan S, Liu Y, Ying L, Liang D. Undersampling trajectory design for fast MRI with super-resolution convolutional neural network. Proceedings of the International Society of Magnetic Resonance in Medicine (ISMRM). 2017. p. 3984.
[16] Haldar JP, Kim D. OEDIPUS: An experiment design framework for sparsity-constrained MRI. IEEE Trans Med Imaging 2019;38(7):1545–58.
[17] Zhang Z, Romero A, Muckley MJ, Vincent P, Yang L, Drozdzal M. Reducing uncertainty in undersampled MRI reconstruction with active acquisition. Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 2019. p. 2049–58.
[18] L. Sun, Y. Wu, B. Shu, X. Ding, C. Cai, Y. Huang, J. Paisley, A dual-domain deep lattice network for rapid MRI reconstruction, Neurocomputing.
[19] Wang S, Cheng H, Ying L, Xiao T, Ke Z, Zheng H, et al. DeepcomplexMRI: exploiting deep residual network for fast parallel MR imaging with complex convolution. Magn Reson Imaging 2020;68:136–47.
[20] Liu D-d, Liang D, Liu X, Zhang Y-t. Under-sampling trajectory design for compressed sensing MRI. IEEE International Conference on Engineering in Medicine and Biology Society (EMBC). IEEE; 2012. p. 73–6.
[21] Glover GH. Spiral imaging in fMRI. Neuroimage 2012;62(2):706–12.
[22] Sharma S, Hari K, Leus G. Space filling curves for MRI sampling. IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). IEEE; 2020. p. 1115–9.
[23] Adcock B, Hansen AC, Poon C, Roman B. Breaking the Coherence Barrier: Asymptotic Incoherence and Asymptotic Sparsity in Compressed Sensing. 2020. preprint.
[24] Puy G, Vandergheynst P, Wiaux Y. On variable density compressive sampling. IEEE Signal Process Lett 2011;18(10):595–8.
[25] Chauffert N, Ciuciu P, Weiss P. Variable density compressed sensing in MRI. Theoretical vs heuristic sampling strategies. 2013 IEEE 10th International Symposium on Biomedical Imaging (ISBI). IEEE; 2013. p. 298–301.
[26] Knoll F, Clason C, Diwoky C, Stollberger R. Adapted random sampling patterns for accelerated MRI. Magnetic Resonanc Mater Phys Biol Med 2011;24(1):43–50.
[27] Chauffert N, Ciuciu P, Kahn J, Weiss P. Variable density sampling with continuous trajectories. Application to MRI. SIAM J Imaging Sci 2014;7(4):1962–92.
[28] Smith DS, Arlinghaus LR, Yankeelov TE, Welch EB. Curvelets as a sparse basis for compressed sensing magnetic resonance imaging. Medical Imaging 2013: Image Processing. vol. 8669. International Society for Optics and Photonics; 2013. p. 866929.
[29] Ravishankar S, Bresler Y. MR image reconstruction from highly undersampled k-space data by dictionary learning. IEEE Trans Med Imaging 2011;30(5):1028.
[30] Afonso MV, Bioucas-Dias JM, Figueiredo MA. An augmented lagrangian approach to the constrained optimization formulation of imaging inverse problems. IEEE Trans Image Process 2011;20(3):681–95.
[31] Needell D, Tropp JA. Cosamp: Iterative signal recovery from incomplete and inaccurate samples. Appl Computat Harmonic Anal 2009;26(3):301–21.
[32] Chauffert N, Ciuciu P, Kahn J, Weiss P. Travelling salesman-based variable density sampling. Proceedings of the 10th SampTA Conference. 2013. p. 509–12.
[33] Hargreaves BA, Nishimura DG, Conolly SM. Time-optimal multidimensional gradient waveform design for rapid imaging. Magn Reson Med 2004;51(1):81–92.
[34] Lustig M, Kim S-J, Pauly JM. A fast method for designing time-optimal gradient waveforms for arbitrary k-space trajectories. IEEE Trans Med Imaging 2008;27(6):866–73.
[35] Davids M, Ruttorf M, Zöllner FG, Schad LR. Fast and robust design of time-optimal k-space trajectories in MRI. IEEE Trans Med Imaging 2015;34(2):564–77.
[36] Chauffert N, Weiss P, Kahn J, Ciuciu P. A projection algorithm for gradient waveforms design in magnetic resonance imaging. IEEE Trans Med Imaging 2016;35(9):2026–39.
[37] Hennig J. K-space sampling strategies. Eur Radiol 1999;9(6):1020–31.
[38] Brodsky EK, Samsonov AA, Block WF. Characterizing and correcting gradient errors in non-cartesian imaging: are gradient errors linear time-invariant (LTI)? Magn Reson Med 2009;62(6):1466–76.
[39] Vannesjo SJ, Graedel NN, Kasper L, Gross S, Busch J, Haeberlin M, et al. Image reconstruction using a gradient impulse response model for trajectory prediction. Magn Reson Med 2016;76(1):45–58.
[40] Yoder DA, Zhao Y, Paschal CB, Fitzpatrick JM. MRI simulator with object-specific field map calculations. Magn Reson Imaging 2004;22(3):315–28.
[41] Simonetti OP, Duerk JL, Chankong V. An optimal design method for magnetic resonance imaging gradient waveforms. IEEE Trans Med Imaging 1993;12(2):350–60.
[42] Chauffert N. Compressed Sampling along Physically Plausible Trajectories MRI. 2020. (Ph.D. thesis, Paris 11 (9 2015)).
[43] Rauhut H. Compressive sensing and structured random matrices. Theoretical Foundations and Numerical Methods for Sparse Recovery. 9. 2010. p. 1–92.
[44] Sharma S, Hari K, Leus G. K-space trajectory design for reduced MRI scan time. ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). IEEE; 2020. p. 1120–4.
[45] De Boor C. A Practical Guide to Splines (Applied Mathematical Sciences). Revised ed. 2001.
[46] Krahmer F, Ward R. Stable and robust sampling strategies for compressive imaging. IEEE Trans Image Process 2014;23(2):612–22.
[47] Kleinberg J, Tardos E. Algorithm Design. Pearson Education India; 2006.
[48] Fessler JA. Michigan Image Reconstruction Toolbox URL http://web.eecs.umich.edu/~fessler/irt/irt/; 2020.
[49] Lustig M. SparseMRI Toolbox Downloaded from http://www.eecs.berkeley.edu/lustig/software.html 2014.
[50] Qin Q. Point spread functions of the t2 decay in k-space trajectories with long echo train. Magn Reson Imaging 2012;30(8):1134–42.
[51] Guerquin-Kern M, Lejeune L, Pruessmann KP, Unser M. Realistic analytical phantoms for parallel magnetic resonance imaging. IEEE Trans Med Imaging 2012;31(3):626–36.
[52] Addy NO, Wu HH, Nishimura DG. Simple method for MR gradient system characterization and k-space trajectory estimation. Magn Reson Med 2012;68(1):120–9.
[53] Parikh N, Boyd S, et al. Proximal algorithms. Found Trends Optimizat 2014;1(3):127–239.
[54] Nesterov Y. A method of solving a convex programming problem with convergence rate O(1/k2). Soviet Mathematics Doklady. 27. 1983. p. 372–6.