



Published in final edited form as:

Magn Reson Med. 2023 August ; 90(2): 417–431. doi:10.1002/mrm.29645.

Stochastic Optimization of 3D Non-Cartesian Sampling Trajectory (SNOPY)*

Guanhua Wang¹, Jon-Fredrik Nielsen¹, Jeffrey A. Fessler^{1,2}, Douglas C. Noll¹

¹Biomedical Engineering, University of Michigan, Michigan, United States

²EECS, University of Michigan, Michigan, United States

Abstract

Purpose: Optimizing 3D k-space sampling trajectories is important for efficient MRI yet presents a challenging computational problem. This work proposes a generalized framework for optimizing 3D non-Cartesian sampling patterns via data-driven optimization.

Methods: We built a differentiable simulation model to enable gradient-based methods for sampling trajectory optimization. The algorithm can simultaneously optimize multiple properties of sampling patterns, including image quality, hardware constraints (maximum slew rate and gradient strength), reduced peripheral nerve stimulation (PNS), and parameter-weighted contrast. The proposed method can either optimize the gradient waveform (spline-based freeform optimization) or optimize properties of given sampling trajectories (such as the rotation angle of radial trajectories). Notably, the method can optimize sampling trajectories synergistically with either model-based or learning-based reconstruction methods. We proposed several strategies to alleviate the severe non-convexity and huge computation demand posed by the large scale. The corresponding code is available as an open-source toolbox.

Results: We applied the optimized trajectory to multiple applications including structural and functional imaging. In the simulation studies, the image quality of a 3D kooshball trajectory was improved from 0.29 to 0.22 (NRMSE) with SNOPY optimization. In the prospective studies, by optimizing the rotation angles of a stack-of-stars (SOS) trajectory, SNOPY reduced the NRMSE of reconstructed images from 1.19 to 0.97 compared to the best empirical method (RSOS-GR). Optimizing the gradient waveform of a rotational EPI trajectory improved participants' rating of the PNS from 'strong' to 'mild.'

Conclusion: SNOPY provides an efficient data-driven and optimization-based method to tailor non-Cartesian sampling trajectories.

Keywords

magnetic resonance imaging; non-Cartesian sampling; data-driven optimization; image acquisition; deep learning

*This manuscript is submitted to *Magnetic Resonance in Medicine* on 09/15/22.

Correspondence: Guanhua Wang. guanhuaw@umich.edu.

1 | INTRODUCTION

Most magnetic resonance imaging systems sample data in the frequency domain (k-space) following prescribed sampling trajectories. Efficient sampling strategies can accelerate acquisition and improve image quality. Many well-designed sampling strategies and their variants, such as spiral, radial, CAIPIRINHA, and PROPELLER^{1,2,3,4}, have enabled MRI's application to many areas^{5,6,7,8}. Sampling patterns in k-space are either located on the Cartesian raster or arbitrary locations (non-Cartesian sampling). This paper focuses on optimizing 3D non-Cartesian trajectories and introduces a generalized gradient-based optimization method for automatic trajectory design or tailoring.

The design of sampling patterns usually considers certain properties of k-space signals. For instance, the variable density (VD) spiral trajectory⁹ samples more densely in the central k-space where more energy is located. For higher spatial frequency regions, the VD spiral trajectory uses larger gradient strengths and slew rates to cover k-space as quickly as possible. Compared to 2D sampling, designing 3D sampling analytically is more challenging for several reasons. The number of parameters increases in 3D, and the parameter selection is more difficult due to the larger search space. For example, a 3D radial trajectory with 10000 spokes has 20000 degrees of freedom, while its 2D multi-slice counterpart with 200 spokes per slice has only 200 degrees of freedom. Additionally, analytical designs usually are based on the Shannon-Nyquist relationship^{10,11,12} that might not fully consider properties of sensitivity maps and non-linear reconstruction methods. For 3D sampling patterns with high undersampling (acceleration) ratios, there are limited analytical tools for designing sampling patterns with an anisotropic FOV and resolution. The peripheral nerve stimulation (PNS) effect¹³ is also more severe in 3D imaging because of the additional spatial encoding gradient, further complicating manual designs. For these reasons, automatic designs of 3D sampling trajectories are crucial for efficient acquisition.

Many 3D sampling approaches exist. The 'stack-of-2D' strategy stacks 2D sampling patterns in the slice direction^{6,12}. This approach is easier to implement and enables slice-by-slice 2D reconstruction. Another design applies Cartesian sampling in the frequency-encoding direction and non-Cartesian sampling in the phase-encoding direction^{14,15}. However, these approaches do not fully explore the design space in three dimensions and may not perform as well as true 3D sampling trajectories¹⁶.

Recently, 3D SPARKLING¹⁶ proposes to optimize 3D sampling trajectories based on the goal of conforming to a given density while distributing samples as uniformly as possible¹⁷. That method demonstrated improved image quality compared to the 'stack-of-2D-SPARKLING' approach. In both 2D and 3D, the SPARKLING approach uses a pre-specified sampling density in k-space that is typically an isotropic radial function. This density function cannot readily capture distinct energy distributions of different imaging protocols, therefore adaptive density functions were recently proposed¹⁸. In SPARKLING, the PNS effects are not controlled explicitly, and the user may need to lower the slew rate to reduce PNS. SPARKLING optimizes the location of every sampling point, or the gradient waveform (freeform optimization), and cannot optimize parameters of existing sampling patterns.

In addition to analytical methods, learning-based methods have been investigated for designing trajectories. Since different anatomies have distinct energy distributions in the frequency domain, an algorithm may learn to optimize sampling trajectories from training datasets. Several studies have shown that different anatomies produce distinct optimized sampling patterns, and these optimized sampling trajectories can improve image quality^{19,20,21,22,23,24,25,26,27}. Some methods can optimize sampling trajectories with respect to specific reconstruction algorithms to further enhance reconstruction image quality^{14,28}. Several recent studies also applied learning-based approaches to 3D non-Cartesian trajectory design. J-MoDL¹⁴ proposes to learn sampling patterns and model-based deep learning reconstruction algorithms jointly. J-MoDL optimizes the sampling locations in the phase-encoding direction, to avoid the computation cost of non-uniform Fourier transform. PILOT/3D-FLAT^{22,29} jointly optimizes freeform 3D non-Cartesian trajectories and a reconstruction neural network with gradient-based methods. These studies use the standard auto-differentiation approach to calculate the gradient used in optimization, which can be inaccurate and lead to sub-optimal optimization results²⁸.

This work extends our previous methods^{20,28} and introduces a generalized Stochastic optimization framework for 3D **NO**n-Cartesian sam**PL**ing trajector**Y** (**SNOPY**). The proposed method can automatically tailor given trajectories and learn k-space features from training datasets. We present several optimization objectives, including image quality, hardware constraints, PNS effect suppression and image contrast. Users can simultaneously optimize one or multiple characteristics of a given sampling trajectory. Similar to previous learning-based methods^{14,20,21,22}, the sampling trajectory can be jointly optimized with trainable reconstruction algorithms to improve image quality. The joint optimization approach can thus exploit the synergy between acquisition and reconstruction, and learn optimized trajectories specific for different anatomies and reconstruction methods^{14,20,28,30,31}. The algorithm can optimize various properties of a sampling trajectory, such as readout waveforms, or rotation angles of readout shots, making it more practical and applicable. We also introduced several techniques to improve efficiency, enabling large-scale 3D trajectory optimization. We tested the proposed methods with multiple imaging applications, including structural and functional imaging. These applications benefited from the SNOPY-optimized sampling trajectories in both simulation and prospective studies.

2 | THEORY

This section describes the proposed gradient-based methods for trajectory optimization. We use the concept of differentiable programming to compute the descent gradient with respect to sampling trajectories required in the gradient-based methods. The sampling trajectory and reconstruction parameters are differentiable parameters, whose gradients can be computed by auto-differentiation. To learn or update these parameters, one may apply (stochastic) gradient descent algorithms. Fig. 1 illustrates the basic idea. The sampling trajectories can be optimized in conjunction with the parameters of learnable reconstruction algorithms so that the learned sampling trajectories and reconstruction methods are in synergy and produce high-quality images. The SNOPY algorithm combines several optimization objectives to ensure that the optimized sampling trajectories have desired properties. Sec. 2.1 delineates these objective functions. Sec. 2.3 shows that the proposed method is applicable to multiple

scenarios with different parameterization strategies. For non-Cartesian sampling, the system model usually involves non-uniform fast Fourier transforms (NUFFT). Sec. 2.4 briefly describes an efficient and accurate way to calculate the gradient involving NUFFTs. Sec. 2.5 suggests several engineering approaches to make this large-scale optimization problem solvable and efficient.

2.1 | Optimization objectives

This section outlines the optimization objectives in SNOPI. As SNOPI is a stochastic gradient descent-like algorithm, the objective function, or loss function, is by default defined on a mini-batch of data. The final loss function can be a linear combination of following loss terms to ensure the optimized trajectory possesses multiple required properties.

2.1.1 | Image quality—For many MRI applications, efficient acquisition and reconstruction aim to produce high-quality images. Consequently, the learning objective should encourage images reconstructed from sampled k-space signals to match the reference images. We formulate this similarity objective as the following image quality training loss:

$$\mathcal{L}_{\text{recon}} = \ell(f_{\theta, c}(\mathbf{A}(\boldsymbol{\omega}(c))\mathbf{x} + \boldsymbol{\varepsilon}) - \mathbf{x}). \quad (1)$$

Here, $\boldsymbol{\omega}(c) \in \mathbb{R}^{N_{\text{fe}} \times N_s \times N_d}$ denotes the trajectory to be optimized, with N_s shots, N_{fe} sampling points in each shot, and N_d image dimensions. For 3D MRI, $N_d = 3$. $\boldsymbol{\varepsilon}$ is simulated complex Gaussian noise. $\mathbf{A}(\boldsymbol{\omega})$ is the forward system matrix for sampling trajectory $\boldsymbol{\omega}(c)$ ³². c denotes the parameterization coefficients of sampling trajectories $\boldsymbol{\omega}$, which is introduced in Sec. 2.3. In this study, \mathbf{A} also incorporated multi-coil sensitivity information³³. \mathbf{x} denotes the reference image from the training set \mathcal{X} , which is typically reconstructed from fully-sampled signals. In addition to contrast-weighted imaging, if the training dataset \mathcal{X} includes *quantitative parameter maps*, one may also simulate \mathbf{x} using the Bloch equation, and \mathbf{A} can subsequently consider imaging physics such as relaxation. $f_{\theta, \omega}(\cdot)$ is the reconstruction algorithm to be delineated in Sec. 2.2. θ denotes the reconstruction algorithm's parameters. It can be kernel weights in a convolutional neural network (CNN), or the regularizer coefficient in a model-based reconstruction method. The similarity term $\ell(\cdot)$ can be ℓ_1 norm, ℓ_2 norm, or a combination of both. There are also other ways to measure the distance between \mathbf{x} and $f_{\theta, \omega}(\mathbf{A}(\boldsymbol{\omega})\mathbf{x} + \boldsymbol{\varepsilon})$, such as the structural similarity index (SSIM³⁴). For simplicity, this work used a linear combination of ℓ_1 norm and square-of- ℓ_2 norm, which is a common practice in deep learning-based image reconstruction³⁵.

2.1.2 | Hardware limits—The gradient system of MR scanners has physical constraints, namely maximum gradient strength and slew rate. Ideally, we would like to enforce a set of constraints of the form

$$\|\mathbf{g}_i[j, :]\|_2 \leq g_{\text{max}}, \quad \mathbf{g}_i = \mathbf{D}_1 \boldsymbol{\omega}[:, i, :]/(\gamma \Delta t) \in \mathbb{R}^{(N_{\text{fe}} - 1) \times N_d},$$

for every shot $i = 1, \dots, N_s$ and time sample $j = 1, \dots, N_{\text{fe}}$, where \mathbf{g}_i denotes the gradient strength of the i shot and g_{max} denotes the desired gradient upper bound. One may use a

Euclidean norm along the spatial axis so that any rotation of the sampling trajectory still obeys the constraint. Applying the penalty to each individual gradient axis is also feasible. A similar constraint is enforced on the Euclidean norm of the slew rate $s_i = \mathbf{D}_2 \boldsymbol{\omega}[:, i, :]/(\gamma \Delta t^2)$, where \mathbf{D}_1 and \mathbf{D}_2 denote first-order and second-order finite difference operators applied along the readout dimension. Δt denotes the raster time interval and γ denotes the gyromagnetic ratio.

To make the optimization more practical, we follow previous studies^{20,22}, and formulate the hardware constraint as a soft penalty term:

$$\mathcal{L}_g = \sum_{i=1}^{N_s} \sum_{j=1}^{N_{fe}-1} \phi_{g_{\max}}(\|\mathbf{g}_i[j, :]\|_2) \quad (2)$$

$$\mathcal{L}_s = \sum_{i=1}^{N_s} \sum_{j=1}^{N_{fe}-2} \phi_{s_{\max}}(\|s_i[j, :]\|_2). \quad (3)$$

Here ϕ is a penalty function, and we use a simple soft-thresholding function $\phi_i(\mathbf{x}) = \max(|\mathbf{x}| - \lambda, 0)$, because it is sub-differentiable and easy to implement. It is possible to use more sophisticated functions. Since ϕ here is a soft penalty and the optimization results may exceed the threshold, s_{\max} and g_{\max} can be slightly lower than the scanner's actual physical limits to ensure that the optimization results are feasible on the scanner. Applying a sanity check before sequence programming is also useful. In addition to the soft-penalty approach, recent studies³⁶ also used projection-based methods.

2.1.3 | Suppression of PNS effect—The additional gradient axis in 3D imaging can result in stronger peripheral nerve stimulation (PNS) effects compared to 2D imaging. To quantify possible PNS effects of candidate gradient waveforms, SNOPI uses a convolution model described in Ref. 37:

$$\mathbf{R}_{id}(t) = \frac{1}{s_{\min}} \int_0^t \frac{s_{id}(\theta)c}{(c+t-\theta)^2} d\theta, \quad (4)$$

where \mathbf{R}_{id} denotes the PNS effect of the gradient waveform from the i th shot and the d th dimension. s_{id} is the slew rate of the i th shot in the d th dimension. c (chronaxie) and s_{\min} (minimum stimulation slew rate) are scanner-specific parameters.

Likewise, we discretize the convolution model and use a soft penalty term as the following loss function:

$$\begin{aligned}
 p_{id}[j] &= \sum_{k=1}^j \frac{s_{id}[k]c\Delta t}{s_{\min}(c + j\Delta t - k\Delta t)^2}, \\
 \mathcal{L}_{\text{pns}} &= \sum_{i=1}^{N_s} \sum_{j=1}^{N_{\text{fc}}} \phi_{p_{\max}} \left(\left(\sum_{d=1}^{N_d} p_{id}[j]^2 \right)^{\frac{1}{2}} \right).
 \end{aligned} \tag{5}$$

Again, ϕ denotes the soft-thresholding function, with PNS threshold p_{\max} (usually 80%³⁷). This model combines the 3 spatial axes via the sum-of-squares manner and does not consider anisotropic characteristics of PNS³⁸. The implementation may use an FFT (with zero padding) for efficient convolution.

2.1.4. | Image contrast—In many applications, the optimized sampling trajectory should maintain certain parameter-weighted contrasts. For example, ideally the (gradient) echo time (TE) should be identical for each shot. Again, we replace this hard constraint with an echo time penalty. Other parameters, like repetition time (TR) and inversion time (TI), can be predetermined in the RF pulse design. Specifically, the corresponding loss function encourages the sampling trajectory to cross the k-space center at certain time points:

$$\mathcal{L}_c = \sum_{\{i,j,d\} \in C} \phi_0(|\omega[i,j,d]|), \tag{6}$$

where c is a collection of gradient time points that are constrained to cross the k-space zero point. ϕ is still a soft-thresholding function, with threshold 0.

The total loss function is a linear combination of the above terms

$$\mathcal{L} = \lambda_{\text{recon}} \mathcal{L}_{\text{recon}} + \lambda_g \mathcal{L}_g + \lambda_s \mathcal{L}_s + \lambda_{\text{pns}} \mathcal{L}_{\text{pns}} + \lambda_c \mathcal{L}_c.$$

Note that not every term is required. For example, experiment 3.2.2 only used the $\mathcal{L}_{\text{recon}}$. Sec. 5 further discusses how to choose linear weights λ s.

2.2 | Reconstruction

In (1), the reconstruction algorithm $f_{\theta,\omega}(\cdot)$ can be various algorithms. Consider a typical cost function for regularized MR image reconstruction

$$\hat{\mathbf{x}} = \arg \min_{\mathbf{x}} \| \mathbf{A}(\boldsymbol{\omega})\mathbf{x} - \mathbf{y} \|_2^2 + \mathcal{R}(\mathbf{x}). \tag{7}$$

$\mathcal{R}(\mathbf{x})$ here can be a Tikhonov regularization $v \| \mathbf{x} \|_2^2$ (CG-SENSE³⁹), a sparsity penalty $v \| \mathbf{T}\mathbf{x} \|_1$ (compressed sensing⁴⁰, \mathbf{T} is a finite-difference operator), a roughness penalty $v \| \mathbf{T}\mathbf{x} \|_2^2$ (penalized least squares, PLS), or a neural network (model-based deep learning, MoDL⁴¹). Sec. 4 shows that different reconstruction algorithms lead to distinct optimized

sampling trajectories. In training, y is retrospectively simulated as $\bar{y} = \mathbf{A}(\boldsymbol{\omega})\mathbf{x} + \varepsilon$ (following (1)). In prospective studies, y is the acquired k-space data.

To get a reconstruction estimation $\hat{\mathbf{x}}$, one may use iterative reconstruction algorithms. Specifically, the algorithm should be step-wise differentiable (or sub-differentiable) to enable differentiable programming. The backpropagation uses the chain rule to traverse every step of the iterative algorithm to calculate gradients with respect to variables such as $\boldsymbol{\omega}$.

2.3 | Parameterization

As is shown in Ref. 20, directly optimizing every k-space sampling location (or equivalently every gradient waveform time point) may lead to sub-optimal results. Additionally, in many applications, one may need to optimize certain properties of existing sampling patterns, such as the rotation angles of a multi-shot spiral trajectory, so that the optimized trajectory can be easily integrated into existing workflows. For these needs, we propose two parameterization strategies.

The first approach, spline-based freeform optimization, represents the sampling pattern using a linear basis, i.e., $\boldsymbol{\omega} = \mathbf{B}\mathbf{c}$, where \mathbf{B} is a matrix of samples of a basis such as quadratic B-spline kernels and \mathbf{c} denotes the coefficients to be optimized^{20,22}. This approach fully exploits the generality of a gradient system. Using a linear parameterization like B-splines reduces degrees of freedom and facilitates applying hardware constraints^{20,42}. Additionally, the parameterization can be combined with multi-scale optimization to avoid sub-optimal local minima and further improve optimization results^{17,20,22}. However, freeformly optimized trajectories could introduce implementation challenges. For example, some MRI systems can not store hundreds of different gradient waveforms.

The second approach is to optimize attributes \mathbf{c} of existing trajectories, where $\boldsymbol{\omega}(\mathbf{c})$ is a differentiable function of the attributes \mathbf{c} . For example, many applications use radial trajectories, where the rotation angles can be optimized. Suppose $\mathbf{s} \in \mathbb{R}^{3 \times N}$ is one radial sampling spoke, and consider an M -shot 3D radial trajectory,

$$\boldsymbol{\omega} = [\mathbf{R}_1 \ \dots \ \mathbf{R}_M] \mathbf{I}_M \otimes \mathbf{s}, \quad (8)$$

where $\mathbf{R}_i \in \mathbb{R}^{3 \times 3}$ denotes a rotation matrix, \mathbf{I}_M denotes an identity matrix of size M , and \otimes denotes the Kronecker product. In this case, the list of \mathbf{R}_i is the coefficient to be optimized. This approach is easier to implement on scanners, and can work with existing workflows.

2.4 | Efficient and accurate Jacobian calculation

In the similarity loss (1), the sampling trajectory is embedded in the forward system matrix \mathbf{A} . The system matrix for non-Cartesian sampling usually includes NUFFT operators³². Updating the sampling trajectory in each optimization step requires the Jacobian, or the gradient with respect to the sampling trajectory. The NUFFT operator typically involves interpolation in the frequency domain, which is non-differentiable due to rounding operations. Several previous works used auto-differentiation (with sub-gradients)

to calculate an approximate numerical gradient^{22,29}, but that approach is inaccurate and slow²⁸. We derived an efficient and accurate Jacobian approximation method²⁸. For example, the efficient Jacobian of a forward system model \mathbf{A} is:

$$\frac{\partial \mathbf{A} \mathbf{x}}{\partial \boldsymbol{\omega}^{[d]}} = -\iota \text{diag}\{\mathbf{A}(\mathbf{x} \odot \mathbf{r}^{[d]})\}, \quad (9)$$

where $d \in \{1, 2, 3\}$ denotes the spatial dimensions, $\mathbf{r}^{[d]}$ denotes the Euclidean spatial grid, \odot denotes the Hadamard product, and ι is the imaginary unit. Calculating this Jacobian simply uses another NUFFT, which is more efficient than the auto-differentiation approach. See Ref. 28 for more cases, such as $\frac{\partial \mathbf{A}' \mathbf{A} \mathbf{x}}{\partial \boldsymbol{\omega}^{[d]}}$ and the detailed derivation.

2.5 | Efficient optimization

2.5.1 | Optimizer—Generally, to optimize the sampling trajectory $\boldsymbol{\omega}$ and other parameters (such as reconstruction parameters $\boldsymbol{\theta}$) via stochastic gradient descent-like methods, each update takes a step (in the simplest form)

$$\begin{aligned} \boldsymbol{\theta}^K &= \boldsymbol{\theta}^{K-1} - \eta_{\boldsymbol{\theta}} \frac{\partial \mathcal{L}}{\partial \boldsymbol{\theta}}(\boldsymbol{\omega}^{K-1}, \boldsymbol{\theta}^{K-1}) \\ \boldsymbol{\omega}^K &= \boldsymbol{\omega}^{K-1} - \eta_{\boldsymbol{\omega}} \frac{\partial \mathcal{L}}{\partial \boldsymbol{\omega}}(\boldsymbol{\omega}^{K-1}, \boldsymbol{\theta}^{K-1}), \end{aligned}$$

where \mathcal{L} is the loss function described in Section 2.1 and where $\eta_{\boldsymbol{\theta}}$ and $\eta_{\boldsymbol{\omega}}$ denote step-size parameters.

The optimization is highly non-convex and may suffer from sub-optimal local minima. We used stochastic gradient Langevin dynamics (SGLD)⁴³ as the optimizer to improve results and accelerate training. Each update of SGLD injects Gaussian noise into the gradient to introduce randomness

$$\begin{aligned} \boldsymbol{\theta}^K &= \boldsymbol{\theta}^{K-1} - \eta_{\boldsymbol{\theta}} \frac{\partial \mathcal{L}}{\partial \boldsymbol{\theta}^{K-1}} + \sqrt{2\eta_{\boldsymbol{\theta}}} \mathcal{N}(0, 1) \\ \boldsymbol{\omega}^K &= \boldsymbol{\omega}^{K-1} - \eta_{\boldsymbol{\omega}} \frac{\partial \mathcal{L}}{\partial \boldsymbol{\omega}^{K-1}} + \sqrt{2\eta_{\boldsymbol{\omega}}} \mathcal{N}(0, 1). \end{aligned} \quad (10)$$

Across most experiments, we observed that SGLD led to improved results and faster convergence compared with SGD or Adam⁴⁴. Fig. 2 shows a loss curve of SGLD and Adam of experiment 3.2.3.

2.5.2 | Memory saving techniques—Due to the large dimension, the memory cost for naive 3D trajectory optimization would be prohibitively intensive. We developed several techniques to reduce memory use and accelerate training.

As discussed above, the efficient Jacobian approximation uses only 10% of the memory required by the standard auto-differentiation approach²⁸. We also used in-place operations in certain reconstruction steps, such as the conjugate gradient (CG) method, because with

careful design it will not interrupt auto-differentiation. (See our open-source code* for details.) The primary memory bottleneck relates to 3D NUFFT operators. One can pre-calculate the Toeplitz embedding kernel to save memory and accelerate computation^{45,46}. In the training phase, we used NUFFTs with lower accuracy, for instance, with a smaller oversampling ratio for gridding²⁸. Table 1 shows the incrementally improved efficiency achieved with these techniques. Without the proposed techniques, optimizing 3D trajectories would require hundreds of gigabytes of memory, which would be impractical for a single node. SNOPI enables solving this otherwise prohibitively large problem on a single GPU.

3 | METHODS

3.1 | Datasets

We used two publicly available datasets; both of them contain 3D multi-coil raw k-space data. SKM-TEA⁴⁷ is a 3D quantitative double-echo steady-state (qDESS⁴⁸) knee dataset. It was acquired by 3T GE MR750 scanners and 15/16-channel receiver coils. SKM-TEA includes 155 subjects. We used 132 for training, 10 for validation, and 13 for the test. Calgary brain dataset⁴⁹ is a 3D brain T1w MP-RAGE⁵⁰ k-space dataset. It includes 67 available subjects, acquired by an MR750 scanner and 12-channel head coils. We used 50 volumes for training, 6 for validation, and 7 for testing. All sensitivity maps were calculated by ESPIRiT⁵¹.

3.2 | Simulation experiments

We experimented with multiple scenarios to show the broad applicability of the proposed method. All the experiments used a node equipped with an Nvidia Tesla A40 GPU for training.

3.2.1 | Optimizing 3D gradient waveform—We optimized the sampling trajectory with a 3D radial (‘kooshball’) initialization^{52,53}. As is described in 2.3, the experiment optimized the readout waveform of each shot with B-spline parameterization, to reduce the number of degrees of freedom and enable multi-scale optimization. The initial 3D radial trajectory had a 5.12 ms long readout (raster time = 4 μ s) and 1024 shots (8 \times acceleration), using the rotation angle described in Ref. 16. The training used the SKM-TEA dataset. The retrospectively cropped FOV was 158 \times 158 \times 51 mm³ with 0.76 \times 0.62 \times 1.6 mm³ simulated resolution. The receiver bandwidth was \pm 125 kHz (dwell time = 4 μ s). The training loss was

$$\mathcal{L} = \mathcal{L}_{\text{recon}} + 10^3 \mathcal{L}_g + 10^3 \mathcal{L}_s + \mathcal{L}_{\text{pns}}.$$

The gradient strength (g_{max}) and slew rate (s_{max}) were 50 mT/m and 150 mT/m/ms (for individual axis). The PNS threshold (p_{max}) was 80%. The simulated noise ϵ was 0. The batch size was 3. The learning rate η_w decayed from 10^{-4} to 0 linearly. For multi-level optimization, we used 3 levels (with B-spline kernel widths = 32, 16, and 8), and each level used 200 epochs. The total training time was \sim 240 hrs. The trajectory was optimized with

* <https://github.com/guanhuaw/Bjork>

respect to several image reconstruction algorithms. We used a regularizer weight $\nu = 10^{-3}$ and 30 CG iterations for CG-SENSE and PLS. For learning-based reconstruction, we used the MoDL⁴¹ network that alternates between a neural network-based denoiser and data consistency updates. We used a 3D version of the denoising network⁵⁴, 20 CG iterations for the data consistency update, and 6 outer iterations. Similar to previous investigations^{14,20}, SNOPIY jointly optimized the neural network's parameters and the sampling trajectory using (10).

3.2.2 | Optimizing rotation angles of stack-of-stars trajectory—This experiment optimized the rotation angles of a stack-of-stars trajectory, which is a widely used volumetric imaging sequence. The training used the Calgary brain dataset. We used PLS as the reconstruction method for simplicity, with $\nu = 10^{-3}$ and 30 iterations. The simulated noise ϵ was 0 and the batch size was 12. We used 200 epochs and a learning rate linearly decaying from 10^{-4} to 0. The FOV was retrospectively cropped to $256 \times 218 \times 32 \text{ mm}^3$ with 1 mm^3 resolution. We used 40 spokes per kz location ($6 \times$ acceleration), and 1280 spokes in total. The readout length was 3.5 ms. The receiver bandwidth was $\pm 125 \text{ kHz}$ (dwell time = $4 \mu\text{s}$). The trajectory was a stack of 32 kz planes, hence SNOPIY optimized 1280 rotation angles in this case.

Since optimizing rotation angles does not impact the gradient strength, slew rate, PNS, and image contrast, we only used the reconstruction loss $\mathcal{L} = \mathcal{L}_{\text{recon}}$. We regarded the method (RSOS-GR) proposed in previous works¹² as the best empirical scheme. We applied 200 epochs with a linearly decaying learning rate from 10^{-3} to 0. The training cost ~ 20 hrs.

3.2.3 | PNS suppression of 3D rotational EPI trajectory for functional imaging—The third application optimizes the rotation EPI (REPI) trajectory⁵⁵, which provides an efficient sampling strategy for fMRI. For high resolution (i.e., 1 mm), we found that subjects may experience strong PNS effects introduced by REPI. This experiment aimed to reduce the PNS effect of REPI while preserving the original image contrast. We optimized one shot of REPI, being parameterized by B-spline kernels (width=16). The optimized readout shot was rotated using the angle scheme similar to Ref. 55 for multi-shot acquisition.

We designed the REPI readout for an oscillating steady imaging (OSSI) sequence, a novel fMRI signal model that can improve the SNR^{56,57}. The FOV was $200 \times 200 \times 12 \text{ mm}^3$, with 1 mm^3 isotropic resolution, $\text{TR} = 16 \text{ ms}$, and $\text{TE} = 7.4 \text{ ms}$. The readout length was 10.6 ms. The receiver bandwidth was $\pm 250 \text{ kHz}$ (dwell time = $2 \mu\text{s}$). The gradient strength (g_{max}), and slew rate (s_{max}) constraints were 58 mT/m and 200 mT/m/ms (3 axes combined).

To accelerate training, the loss term here excluded the reconstruction loss $\mathcal{L}_{\text{recon}}$:

$$\mathcal{L} = 10^{-2} \mathcal{L}_g + 10^{-2} \mathcal{L}_s + \mathcal{L}_{\text{pns}} + 10^2 \mathcal{L}_c.$$

The training used 40,000 steps, with the learning rate decaying linearly from 10^{-4} to 0. The training cost ~ 1 hrs.

3.3 | In-vivo experiments

We implemented the optimized trajectory prospectively on a GE UHP 3.0T scanner equipped with a Nova Medical 32-channel head coil. Participants gave informed consent under local IRB approval. Since the cache space in this MR system cannot load hundreds of distinct gradient waveforms, the experiment 3.2.1 was not implemented prospectively. Readers may refer to the corresponding 2D prospective studies²⁰ for image quality improvement and correction of eddy current effects. For experiment 3.2.2, we programmed the sampling trajectory with a 3D T1w fat-saturated GRE sequence⁵⁸, with TR/TE = 14/3.2 ms and FA = 20°. The experiment included 4 healthy subjects. For experiment 3.2.3, to rate the PNS effect, we asked 3 participants to score the nerve stimulation with a 5-point Likert scale from ‘mild tingling’ to ‘strong muscular twitch.’

3.4 | Reproducible research

The code is publicly available[†]. As an accompanying project, MIRTorch[‡] facilitates applying differentiable programming to MRI sampling and reconstruction.

4 | RESULTS

For the spline-based freeform optimization experiment delineated in 3.2.1, Fig. 3 presents an example of the optimized trajectory, along with zoomed-in regions and plots of a single shot. Similar to the 2D case²⁰ and SPARKLING^{16,17}, the multi-level B-spline optimization generates a swirling trajectory that can cover more k-space in the fixed readout time, to reduce large gaps between sampling locations and, consequently, aliasing artifacts. Notably, the zoomed-in region highlights that different shots were automatically learned not to overlap with each other, which implicitly improved the sampling efficiency¹⁷. Fig. 4 displays point spread functions (PSFs) of trajectories jointly optimized with different reconstruction algorithms. To visualize the sampling density in different regions of k-space, we convolved the trajectory with a Gaussian kernel, and Fig. 4 shows the density of central profiles from different views. Compared with 3D kooshball, the SNOPI optimization led to fewer radial patterns in PSFs, corresponding to fewer streak artifacts in Fig. 5. Trajectories optimized with different reconstruction algorithms generated different PSFs and densities, which agrees with previous studies^{28,30,31}. Table 2 lists the quantitative reconstruction quality of different trajectories. The image quality metric is the average peak signal-to-noise ratio (PSNR) of the test set. SNOPI led to ~4 dB higher PSNR than the kooshball initialization. Fig. 5 includes examples of reconstructed images. Compared to kooshball, SNOPI’s reconstructed images have reduced artifacts and blurring. Though MoDL (and its variants) are well-performing NN-based reconstruction algorithms according to the open fastMRI reconstruction challenge⁵⁹, many important structures are distorted using the kooshball trajectory. Using the SNOPI-optimized trajectory, a simple model-based reconstruction (CG-SENSE) can reconstruct such structures. The gradient strength and the slew rate of optimized sampling trajectories are exhibited in the supplementary materials.

[†] <https://github.com/guanhuaw/SNOPI>

[‡] <https://github.com/guanhuaw/MIRTorch>

SNOPY solves a non-convex problem; therefore, its results depend on the initialization. The supplementary materials compare optimization results with different initializations.

For experiment 3.2.2, Fig. 6 shows the PSF of the optimized and RSOS-GR schemes¹². For the in-plane ($x - y$) PSF, the SNOPY rotation shows noticeably reduced streak-like patterns. In the $y - z$ direction, SNOPY optimization leads to a narrower central lobe and suppressed aliasing artifacts. The prospective in-vivo experiments also support this theoretical finding. In Fig. 6, the example slices (reconstructed by PLS) from prospective studies show that SNOPY reduced streaking artifacts. The average PSNR of SNOPY and RSOS-GR for the 4 participants were 39.23 dB and 37.84 dB, respectively. Supplementary materials show the rotation angles before and after SNOPY optimization.

In experiment 3.2.3, we tested three settings: unoptimized REPI, optimized with PNS threshold (p_{\max} in (5)) = 80%, and optimized with $p_{\max} = 70\%$. Fig. 7 shows one slice of reconstructed images by the CS-SENSE algorithm, as well as the subjective ratings of PNS. Though SNOPY suppressed the PNS effect, the image contrast was well preserved by the image contrast regularizer (6). Fig. 8 presents one shot before and after the optimization, and one plot of simulated PNS effects. The SNOPY optimization effectively reduced subjective PNS effects of given REPI readout gradients in both simulation and in-vivo experiments. Intuitively, SNOPY smoothed the trajectory to avoid a constantly high slew rate, preventing a strong PNS effect.

5 | DISCUSSION

SNOPY presents a novel and intuitive approach to optimizing non-Cartesian sampling trajectories. Via differentiable programming, SNOPY enables the application of gradient-based and data-driven methods to trajectory design. Various applications and in-vivo experiments demonstrated the applicability and robustness of SNOPY and its 2D predecessor²⁰.

Experiments 3.2.1 and 3.2.2 used SNOPY to tailor sampling trajectories according to specific training datasets and reconstruction algorithms, by formulating reconstruction image quality as a training loss. One concern was whether the learned trajectories would overfit the training dataset. In experiment 3.2.2, the training set used an MP-RAGE sequence, while the prospective sequence was an RF-spoiled GRE. Similarly, 2D prospective and retrospective experiments²⁰ showed that trajectories learned with particular pulse sequences and hardware still improved the image quality of other sequences and hardware, and the NN-based reconstruction did not require fine-tuning with respect to prospective experiments. These empirical studies suggest that trajectory optimization is robust to moderate distribution shifts between training and inference. An intuitive explanation is that SNOPY can improve the PSF by reducing aliasing, and such improvements are universally beneficial. Future investigations will explore the robustness of SNOPY in more diverse settings, such as optimizing trajectories with healthy controls and prospectively testing them with pathological participants to examine image quality for pathologies. It will also be desirable to test SNOPY with different FOVs, resolutions, and B_0 strengths.

Our experiments demonstrated that iterative reconstruction with simple analytical regularizers, such as CG-SENSE, can benefit from the SNOPI-optimized sampling trajectories. As depicted in Fig. 3, CG-SENSE with SNOPI optimization can successfully reconstruct many anatomical structures that were blurred in the MoDL reconstruction without SNOPI trajectory. This result is consistent with previous studies²⁸, where compressed sensing algorithms with trajectory optimization also outperformed NN-based reconstruction. These findings indicate untapped potentials of model-based reconstruction by optimizing sampling trajectories.

MRI systems are prone to imperfections such as field inhomogeneity⁶⁰ and eddy currents⁶¹. Many correction approaches exist, such as B_0 -informed reconstruction⁴⁵ and trajectory mapping^{62,63}. SNOPI-optimized trajectories are compatible with existing correction methods. For instance, we demonstrated the feasibility of implementing eddy currents correction for a 2D freeform optimized trajectory in Ref. 20. Additionally, incorporating system imperfections into the forward learning/optimization phase, such as off-resonance maps in the system model \mathbf{A} (as defined in (1)), may enhance the intrinsic robustness of the optimized trajectory. However, this approach requires the distribution of system imperfections, which is typically scanner-specific. To address this limitation, we plan to investigate prospective simulation approaches in future studies. The model mismatch may also happen at the digitization level: the training set typically consists of concrete discrete-space images, whereas real objects are continuous. This inverse crime is common in learning-based methods and may lead to suboptimal results. Future research should investigate strategies for mitigating this issue.

SNOPI uses a relatively simplified model of PNS. More precise models, such as Ref. 38, may lead to improved PNS suppression results. SNOPI can also incorporate other optimization objectives to encourage properties such as robustness to field inhomogeneity and reduction of acoustic noise.

The training process incorporates several loss terms, including image quality, PNS suppression, hardware limits, and image contrast. By combining these terms, the optimization can lead to trajectories that have multiple desired characteristics. One may alter the optimization results by controlling the coefficients. For example, with a larger coefficient of the hardware constraint loss, the trajectory will better conform to s_{\max} and g_{\max} . The supplementary materials contain an example of optimization results using different combinations of weights. Setting the weights of several terms can be complicated. Empirically, the weight of soft constraints, including hardware (\mathcal{L}_g and \mathcal{L}_s), PNS suppression (\mathcal{L}_{pns}), and contrast (\mathcal{L}_c) can be tuned to a higher value if the optimized trajectory significantly violates these constraints. Additionally, the training losses may sometimes contradict each other, and the optimization process would get stuck in a local minimum. To address this, several empirical tricks have been employed. Similar to SPARKLING¹⁷, the constraint on maximum gradient strength can be relaxed using a higher receiver bandwidth. Bayesian optimization is another option for finding optimal loss weights, but may increase training time. Using SGLD can introduce randomness that helps

escape local minima. In spline-based optimization, one can use a larger B-spline kernel width in the early stages of a coarse-to-fine search.

Trajectory optimization is a non-convex problem. SNOPI uses several methods, including effective Jacobian approximation, parameterization, multi-level optimization, and SGLD, to alleviate the non-convexity and achieve better optimization results. These methods were also found to be effective in previous studies^{20,28}. Initialization is also important for non-convex problems, as demonstrated in the supplementary materials. SNOPI can leverage existing knowledge of MR sampling as a benign initialization. For instance, our experiments used the widely accepted golden-angle stack-of-stars as optimization bases. The SNOPI algorithm can sequentially improve these skillfully designed trajectories to combine the best of both stochastic optimization and researchers' insights.

SNOPI has a wide range of potential applications, including dynamic and quantitative imaging, particularly if large-scale quantitative datasets are available. These new applications may require task-specific optimization objectives in addition to the ones described in Sec. 2.1. In particular, if the reconstruction method is not easily differentiable, such as the MR fingerprinting reconstruction based on dictionary matching⁶⁴, one needs to design a surrogate objective for image quality.

Supplementary Material

Refer to Web version on PubMed Central for supplementary material.

ACKNOWLEDGMENTS

The authors thank Dr. Melissa Haskell, Dr. Shouchang Guo, Dinank Gupta, and Yuran Zhu for the helpful discussion.

Funding Information

This work was partially supported by National Science Foundation; grant IIS 1838179 and National Institutes of Health; grant R01 EB023618, U01 EB026977, S10 OD026738-01.

Abbreviations:

MRI	magnetic resonance imaging
PSNR	peak signal-to-noise ratio

REFERENCES

1. Ahn CB, Kim JH, Cho ZH. High-speed Spiral-scan Echo Planar NMR Imaging-I. *IEEE Trans Med Imaging* 1986 Mar;5(1):2–7. [PubMed: 18243976]
2. Lauterbur PC. Image Formation by Induced Local Interactions: Examples Employing Nuclear Magnetic Resonance. *Nature* 1973;242(5394):190–191.
3. Breuer FA, Blaimer M, Mueller MF, Seiberlich N, Heidemann RM, Griswold MA, et al. Controlled Aliasing in Volumetric Parallel Imaging (2D CAIPIRINHA). *Magn Reson Med* 2006;55(3):549–556. [PubMed: 16408271]
4. Pipe JG. Motion Correction with PROPELLER MRI: Application to Head Motion and Free-Breathing Cardiac Imaging. *Magn Reson Med* 1999;42(5):963–969. [PubMed: 10542356]

5. Glover GH, Law CS. Spiral-in/out BOLD fMRI for Increased SNR and Reduced Susceptibility Artifacts. *Magn Reson Med* 2001;46(3):515–522. [PubMed: 11550244]
6. Johansson A, Balter J, Cao Y. Rigid-Body Motion Correction of the Liver in Image Reconstruction for Golden-Angle Stack-of-Stars DCE MRI. *Magn Reson Med* 2018;79(3):1345–1353. [PubMed: 28617993]
7. Yu MH, Lee JM, Yoon JH, Kiefer B, Han JK, Choi BI. Clinical Application of Controlled Aliasing in Parallel Imaging Results in a Higher Acceleration (CAIPIRINHA)-Volumetric Interpolated Breathhold (VIBE) Sequence for Gadoteric Acid-Enhanced Liver MR Imaging. *J Magn Reson Imaging* 2013;38(5):1020–1026. [PubMed: 23559147]
8. Forbes KPN, Pipe JG, Bird CR, Heiserman JE. PROPELLER MRI: Clinical Testing of a Novel Technique for Quantification and Compensation of Head Motion. *J Magn Reson Imaging* 2001;14(3):215–222. [PubMed: 11536397]
9. Lee JH, Hargreaves BA, Hu BS, Nishimura DG. Fast 3D Imaging Using Variable-density Spiral Trajectories with Applications to Limb Perfusion. *Magn Reson Med* 2003;50(6):1276–1285. [PubMed: 14648576]
10. Gurney PT, Hargreaves BA, Nishimura DG. Design and Analysis of a Practical 3D Cones Trajectory. *Magn Reson Med* 2006;55(3):575–582. [PubMed: 16450366]
11. Johnson KM. Hybrid Radial-Cones Trajectory for Accelerated MRI. *Magn Reson Med* 2017;77(3):1068–1081. [PubMed: 27017991]
12. Zhou Z, Han F, Yan L, Wang DJJ, Hu P. Golden-Ratio Rotated Stack-of-Stars Acquisition for Improved Volumetric MRI. *Magn Reson Med* 2017;78(6):2290–2298. [PubMed: 28168738]
13. Ham CLG, Engels JML, van de Wiel GT, Machielsen A. Peripheral Nerve Stimulation during MRI: Effects of High Gradient Amplitudes and Switching Rates. *J Magn Reson Imaging* 1997;7(5):933–937. [PubMed: 9307922]
14. Aggarwal HK, Jacob M. Joint Optimization of Sampling Patterns and Deep Priors for Improved Parallel MRI. In: 2020 IEEE Intl. Conf. on Acous., Speech and Sig. Process. (ICASSP); 2020. p. 8901–8905.
15. Bilgic B, et al. Wave-CAIPI for Highly Accelerated 3D Imaging. *Magn Reson Med* 2015;73(6):2152–2162. [PubMed: 24986223]
16. Chaithya GR, Weiss P, Daval-Fr erot G, Massire A, Vignaud A, Ciuciu P. Optimizing Full 3D SPARKLING Trajectories for High-Resolution Magnetic Resonance Imaging. *IEEE Trans Med Imaging* 2022 Aug;41(8):2105–2117. [PubMed: 35254981]
17. Lazarus C, et al. SPARKLING: Variable-density K-space Filling Curves for Accelerated T2*-weighted MRI. *Mag Res Med* 2019 Jun;81(6):3643–61.
18. Chaithya G, Ramzi Z, Ciuciu P. Learning the sampling density in 2D SPARKLING MRI acquisition for optimized image reconstruction. In: 2021 29th Euro. Sig. Process. Conf. (EUSIPCO) IEEE; 2021. p. 960–964.
19. Huijben IAM, Veeling BS, van Sloun RJG. Learning Sampling and Model-based Signal Recovery for Compressed Sensing MRI. In: 2020 IEEE Intl. Conf. on Acous., Speech and Sig. Process. (ICASSP); 2020. p. 8906–8910.
20. Wang G, Luo T, Nielsen JF, Noll DC, Fessler JA. B-Spline Parameterized Joint Optimization of Reconstruction and K-Space Trajectories (BJORK) for Accelerated 2D MRI. *IEEE Trans Med Imaging* 2022 Sep;41(9):2318–2330. [PubMed: 35320096]
21. Bahadir CD, Wang AQ, Dalca AV, Sabuncu MR. Deep-learning-based Optimization of the Under-sampling Pattern in MRI. *IEEE Trans Comput Imag* 2020;6:1139–1152.
22. Weiss T, Senouf O, Vedula S, Michailovich O, Zibulevsky M, Bronstein A. PILOT: Physics-Informed Learned Optimized Trajectories for Accelerated MRI. *MELBA* 2021;p. 1–23.
23. Sanchez T, et al. Scalable Learning-based Sampling Optimization for Compressive Dynamic MRI. In: 2020 IEEE Intl. Conf. on Acous., Speech and Sig. Process. (ICASSP); 2020. p. 8584–8588.
24. Jin KH, Unser M, Yi KM. Self-supervised Deep Active Accelerated MRI; 2020. <http://arxiv.org/abs/1901.04547>.
25. Zeng D, Sandino C, Nishimura D, Vasanaawala S, Cheng J. Reinforcement Learning for Online Undersampling Pattern Optimization. In: *Proc. Intl. Soc. Magn. Reson. Med. (ISMRM)*; 2019. p. 1092.

26. Sherry F, Benning M, Reyes JCD, Graves MJ, Maierhofer G, Williams G, et al. Learning the Sampling Pattern for MRI. *IEEE Trans Med Imaging* 2020 Dec;39(12):4310–21. [PubMed: 32804647]
27. Gözcü B, Mahabadi RK, Li YH, Ilıcak E, Çukur T, Scarlett J, et al. Learning-Based Compressive MRI. *IEEE Trans Med Imaging* 2018 Jun;37(6):1394–1406. [PubMed: 29870368]
28. Wang G, Fessler JA. Efficient Approximation of Jacobian Matrices Involving a Non-Uniform Fast Fourier Transform (NUFFT). *IEEE Trans Comput Imaging* 2023;9:43–54. [PubMed: 37090025]
29. Alush-Aben J, Ackerman-Schraier L, Weiss T, Vedula S, Senouf O, Bronstein A. 3D FLAT: Feasible Learned Acquisition Trajectories for Accelerated MRI. In: Deeba F, Johnson P, Würfl T, Ye JC, editors. *Mach. Learn. Med. Image Reconstr. Lecture Notes in Computer Science*, Cham: Springer International Publishing; 2020. p. 3–16.
30. Gözcü B, Sanchez T, Cevher V. Rethinking Sampling in Parallel MRI: A Data-Driven Approach. In: 2019 27th Euro. Sig. Process. Conf. (EUSIPCO); 2019. p. 1–5.
31. Zibetti MVW, Herman GT, Regatte RR. Fast Data-Driven Learning of Parallel MRI Sampling Patterns for Large Scale Problems. *Sci Rep* 2021 Sep;11(1):19312. [PubMed: 34588478]
32. Fessler JA, Sutton BP. Nonuniform Fast Fourier Transforms Using Min-max Interpolation. *IEEE Trans Sig Process* 2003 Feb;51(2):560–74.
33. Pruessmann KP, Weiger M, Börner P, Boesiger P. Advances in Sensitivity Encoding with Arbitrary k-space Trajectories. *Magn Reson Med* 2001;46(4):638–651. [PubMed: 11590639]
34. Wang Z, Bovik AC, Sheikh HR, Simoncelli EP. Image Quality Assessment: From Error Visibility to Structural Similarity. *IEEE Trans Image Process* 2004 Apr;13(4):600–612. [PubMed: 15376593]
35. Zhao H, Gallo O, Frosio I, Kautz J. Loss Functions for Image Restoration With Neural Networks. *IEEE Trans Comput Imaging* 2017 Mar;3(1):47–57.
36. Radhakrishna CG, Ciuciu P. Jointly Learning Non-Cartesian k-Space Trajectories and Reconstruction Networks for 2D and 3D MR Imaging through Projection. *Bioengineering* 2023 Feb;10(2):158. [PubMed: 36829652]
37. Schulte RF, Noeske R. Peripheral Nerve Stimulation-Optimal Gradient Waveform Design. *Magn Reson Med* 2015;74(2):518–522. [PubMed: 25168778]
38. Davids M, Guérin B, Vom Endt A, Schad LR, Wald LL. Prediction of Peripheral Nerve Stimulation Thresholds of MRI Gradient Coils Using Coupled Electromagnetic and Neurodynamic Simulations. *Magn Reson Med* 2019;81(1):686–701. [PubMed: 30094874]
39. Maier O, et al. CG-SENSE Revisited: Results from the First ISMRM Reproducibility Challenge. *Magn Reson Med* 2021;85(4):1821–1839. [PubMed: 33179826]
40. Lustig M, Donoho DL, Santos JM, Pauly JM. Compressed Sensing MRI. *IEEE Signal Process Mag* 2008 Mar;25(2):72–82.
41. Aggarwal HK, Mani MP, Jacob M. MoDL: Model-based Deep Learning Architecture for Inverse Problems. *IEEE Trans Med Imaging* 2019 Feb;38(2):394–405. [PubMed: 30106719]
42. Hao S, Fessler JA, Noll DC, Nielsen JF. Joint Design of Excitation k-Space Trajectory and RF Pulse for Small-Tip 3D Tailored Excitation in MRI. *IEEE Trans Med Imaging* 2016 Feb;35(2):468–479. [PubMed: 26390450]
43. Welling M, Teh YW. Bayesian Learning via Stochastic Gradient Langevin Dynamics. In: *Proc. 28th Int. Conf. Int. Conf. Mach. Learn (ICML)*. Madison, WI, USA: Omnipress; 2011. p. 681–688.
44. Kingma DP, Ba J, Adam: A Method for Stochastic Optimization; 2017. <http://arxiv.org/abs/1412.6980>.
45. Fessler JA, Lee S, Olafsson VT, Shi HR, Noll DC. Toeplitz-based Iterative Image Reconstruction for MRI with Correction for Magnetic Field Inhomogeneity. *IEEE Trans Sig Process* 2005 Sep;53(9):3393–402.
46. Muckley MJ, Stern R, Murrell T, Knoll F. TorchKbNufft: A High-Level, Hardware-Agnostic Non-Uniform Fast Fourier Transform. In: *ISMRM Workshop on Data Sampling & Image Reconstruction*; 2020..

47. Desai AD, Schmidt AM, Rubin EB, Sandino CM, Black MS, Mazzoli V, et al., SKM-TEA: A Dataset for Accelerated MRI Reconstruction with Dense Image Labels for Quantitative Clinical Evaluation; 2022. <http://arxiv.org/abs/2203.06823>.
48. Welsch GH, Scheffler K, Mamisch TC, Hughes T, Millington S, Deimling M, et al. Rapid Estimation of Cartilage T2 Based on Double Echo at Steady State (DESS) with 3 Tesla. *Magn Reson Med* 2009;62(2):544–549. [PubMed: 19526515]
49. Souza R, Lucena O, Garrafa J, Gobbi D, Saluzzi M, Appenzeller S, et al. An Open, Multi-vendor, Multi-field-strength Brain MR Dataset and Analysis of Publicly Available Skull Stripping Methods Agreement. *NeuroImage* 2018;170:482–494. [PubMed: 28807870]
50. Brant-Zawadzki M, Gillan GD, Nitz WR. MP RAGE: A Three-Dimensional, T1-weighted, Gradient-Echo Sequence—Initial Experience in the Brain. *Radiology* 1992 Mar;182(3):769–775. [PubMed: 1535892]
51. Uecker M, et al. ESPIRiT - An Eigenvalue Approach to Autocalibrating Parallel MRI: Where SENSE Meets GRAPPA. *Mag Reson Med* 2014 Mar;71(3):990–1001.
52. Barger AV, Block WF, Toropov Y, Grist TM, Mistretta CA. Time-Resolved Contrast-Enhanced Imaging with Isotropic Resolution and Broad Coverage Using an Undersampled 3D Projection Trajectory. *Magn Reson Med* 2002;48(2):297–305. [PubMed: 12210938]
53. Herrmann KH, Krämer M, Reichenbach JR. Time Efficient 3D Radial UTE Sampling with Fully Automatic Delay Compensation on a Clinical 3T MR Scanner. *PLOS ONE* 2016 Mar;11(3):e0150371. [PubMed: 26975051]
54. Yu S, Park B, Jeong J. Deep Iterative Down-up CNN for Image Denoising. In: *Proc. of the IEEE Conf. on Comput. Vis. and Patt. Recog. Work. (CVPRW)*; 2019. p. 0–0.
55. Rettenmeier CA, Maziero D, Stenger VA. Three Dimensional Radial Echo Planar Imaging for Functional MRI. *Magn Reson Med* 2022;87(1):193–206. [PubMed: 34411342]
56. Guo S, Noll DC. Oscillating Steady-State Imaging (OSSI): A Novel Method for Functional MRI. *Magn Reson Med* 2020;84(2):698–712. [PubMed: 31912574]
57. Guo S, Fessler JA, Noll DC. High-Resolution Oscillating Steady-State fMRI Using Patch-Tensor Low-Rank Reconstruction. *IEEE Trans Med Imaging* 2020 Dec;39(12):4357–4368. [PubMed: 32809938]
58. Nielsen JF, Noll DC. TOPPE: a framework for rapid prototyping of MR pulse sequences. *Magn Reson Med* 2018;79(6):3128–3134. [PubMed: 29096052]
59. Muckley MJ, Riemenschneider B, Radmanesh A, Kim S, Jeong G, Ko J, et al. Results of the 2020 fastMRI Challenge for Machine Learning MR Image Reconstruction. *IEEE Trans Med Imaging* 2021 Sep;40(9):2306–2317. [PubMed: 33929957]
60. Sutton BP, Noll DC, Fessler JA. Fast, Iterative Image Reconstruction for MRI in the Presence of Field Inhomogeneities. *IEEE Trans Med Imaging* 2003 Feb;22(2):178–188. [PubMed: 12715994]
61. Ahn CB, Cho ZH. Analysis of the Eddy-Current Induced Artifacts and the Temporal Compensation in Nuclear Magnetic Resonance Imaging. *IEEE Trans Med Imaging* 1991 Mar;10(1):47–52. [PubMed: 18222799]
62. Duyn JH, Yang Y, Frank JA, van der Veen JW. Simple correction method for k-space trajectory deviations in MRI. *J Magn Reson* 1998 May;132:150–153. [PubMed: 9615415]
63. Robison RK, Li Z, Wang D, Ooi MB, Pipe JG. Correction of B0 Eddy Current Effects in Spiral MRI. *Magn Reson Med* 2019;81(4):2501–2513. [PubMed: 30444004]
64. Ma D, Gulani V, Seiberlich N, Liu K, Sunshine JL, Duerk JL, et al. Magnetic Resonance Fingerprinting. *Nature* 2013;495(7440):187. [PubMed: 23486058]

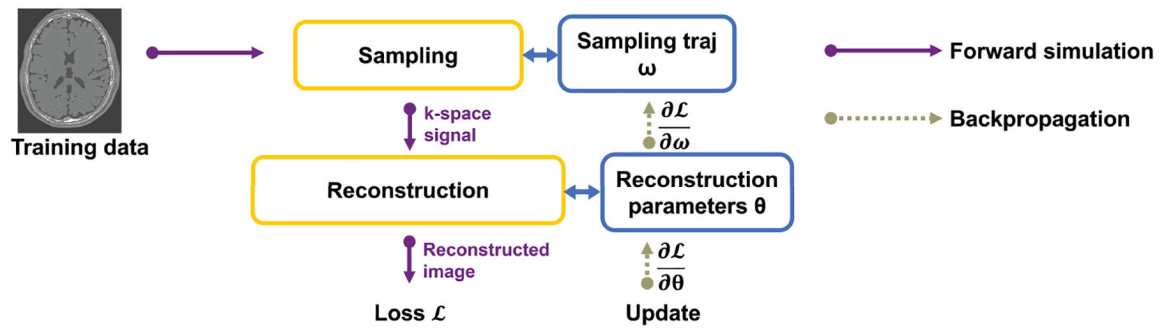


FIGURE 1. Diagram of SNOPI. The sampling trajectory (and possibly reconstruction parameters) are updated using gradient methods. The training/optimization process uses differentiable programming to obtain the gradient necessary for the update.

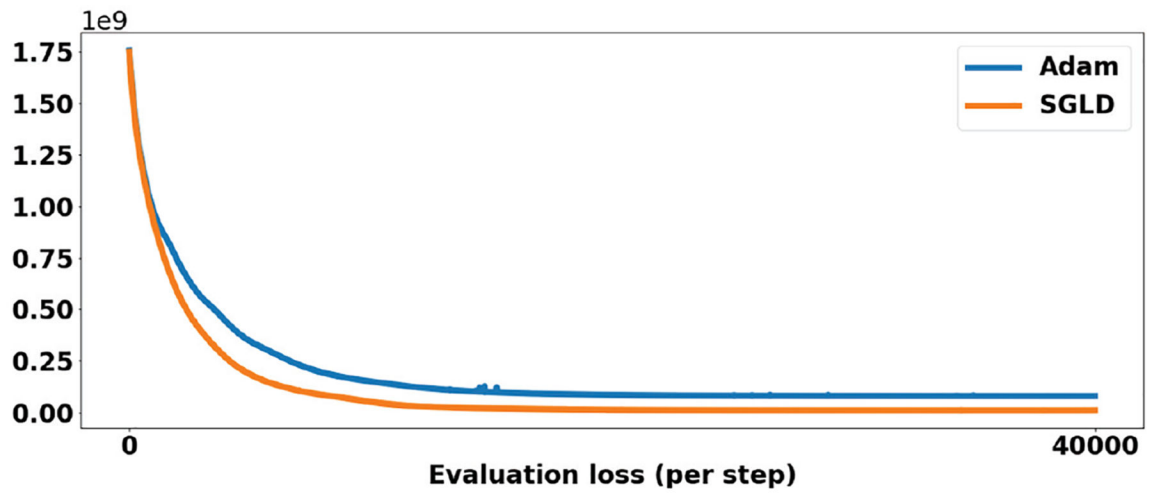


FIGURE 2.

The evaluation loss curve for SGLD and Adam. The training process costs ~1 hrs.

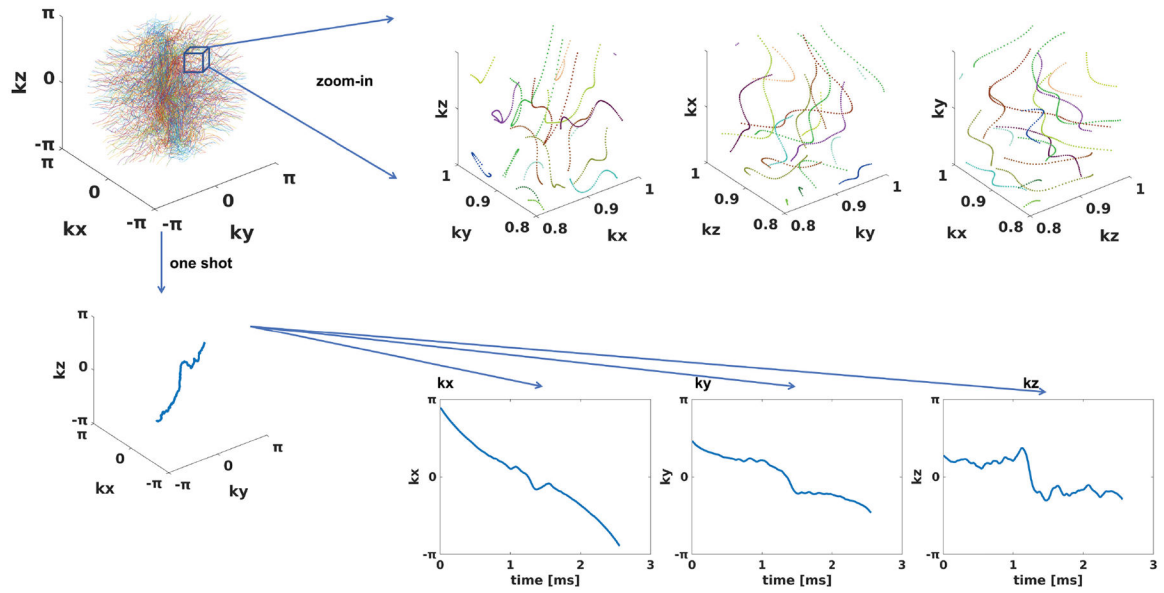


FIGURE 3.

The optimized sampling trajectory of experiment 3.2.1. The training process involves the SKM-TEA dataset and CG-SENSE reconstruction. The upper row shows a zoomed-in region from different viewing perspectives. The lower row displays one shot from different perspectives.

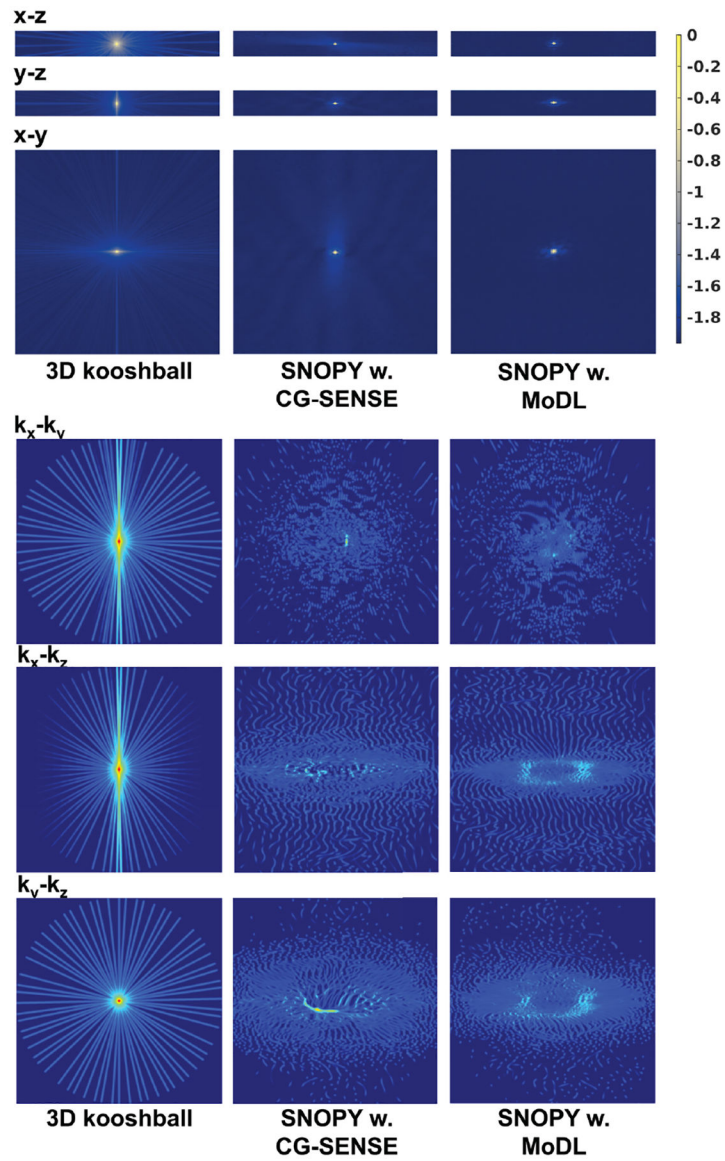


FIGURE 4.

Visualization of the optimized trajectory in experiment 3.2.1. The upper subfigure displays PSFs (log-scaled, single-coil) of trajectories optimized with different reconstruction methods. The lower subfigure shows the density of sampling trajectories, obtained by convolving the sampling points with a Gaussian kernel. Three rows are central profiles from three perspectives.

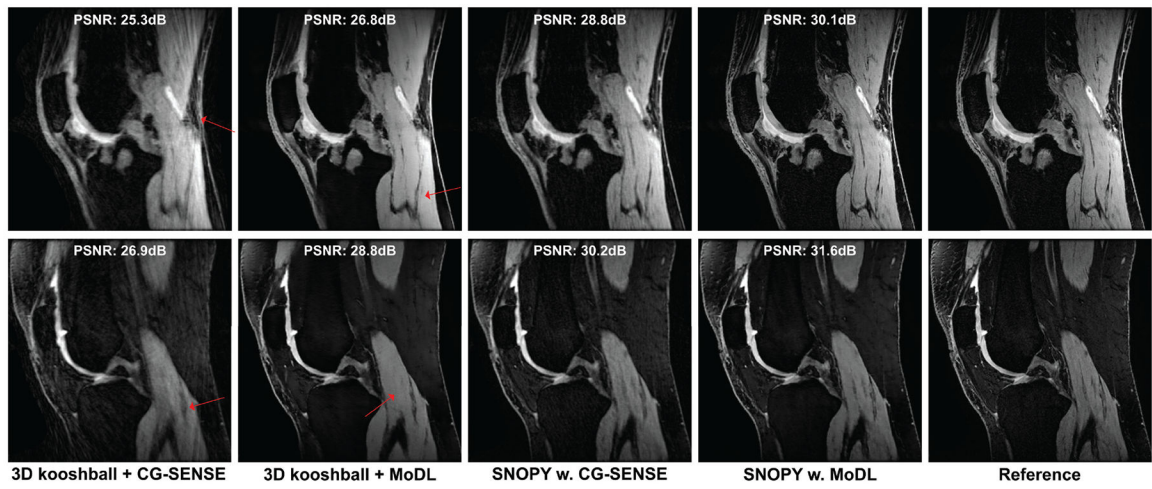


FIGURE 5.

Examples of the reconstructed images for two knee slices in experiment 3.2.1.

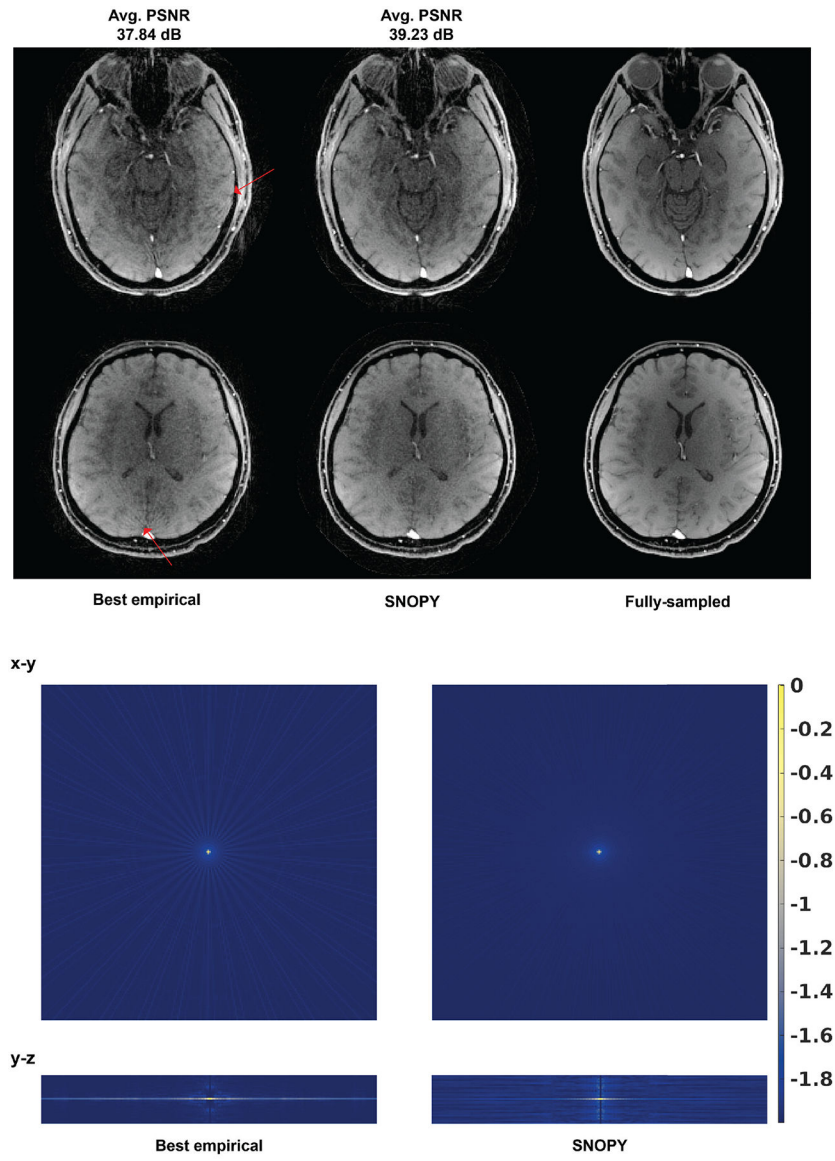


FIGURE 6. Prospective results of 3.2.2, optimizing the rotation angles of the stack-of-stars ($6\times$ acceleration). ‘Best empirical’ uses the design from a previous study¹². The upper subfigure shows two slices from prospective in-vivo experiments. The reconstruction algorithm was PLS. **Avg. PSNR** is the average PSNR of the 4 subjects compared to the fully sampled reference. The lower subfigure shows the log-scaled PSF (single-coil) of two trajectories.

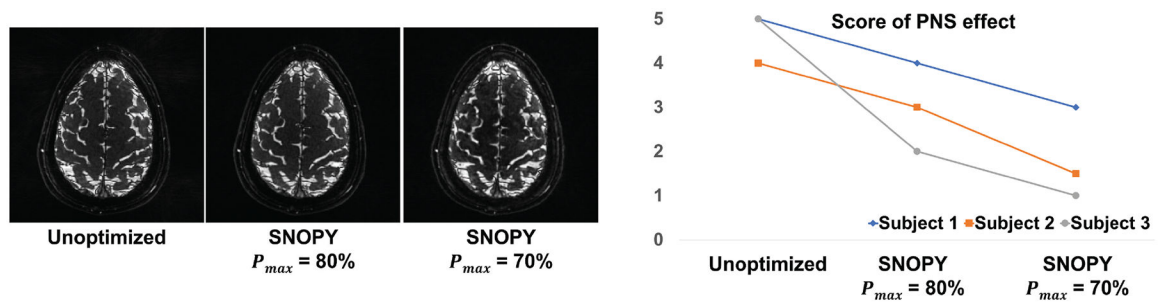
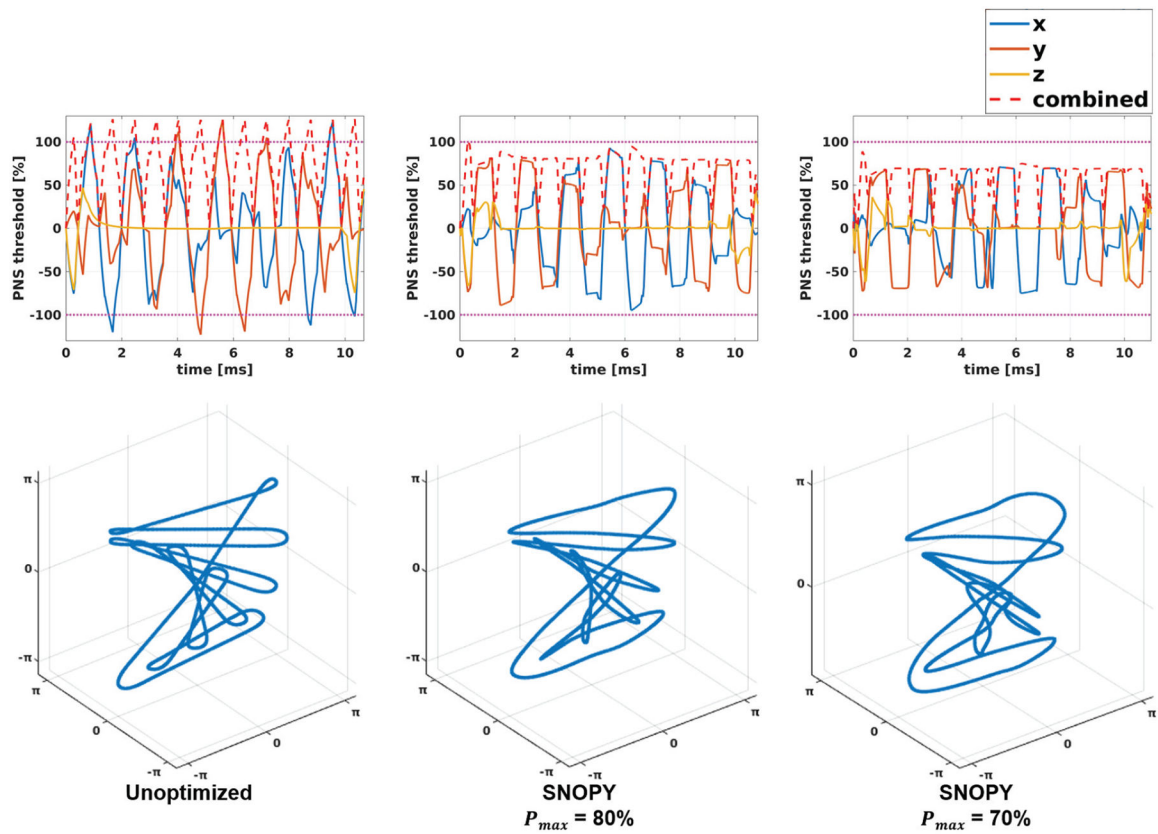


FIGURE 7.

Prospective results of 3.2.3. We showed three different trajectories: the unoptimized REPI, as well as SNOPI-optimized with PNS thresholds of 80% and 70%. The left subfigure shows one slice of reconstructed images. The reconstruction used PLS and 120 shots (volume TR = 2s). The right subfigure shows subjective scores of the PNS effect.

**FIGURE 8.**

The first row of plots displays the PNS effect calculated by the convolution model (5) used in Experiment 3.2.3. The second row shows the corresponding readout trajectories before and after SNOPI optimization.

The memory/time use reduction brought by proposed techniques. Here we used a 2D 400×400 test case, and CG-SENSE reconstruction (20 iterations). '+ ' means adding the technique to previous columns.

TABLE 1

Plain	+Efficient Jacobian	+In-place ops	+Toeplitz embedding	+Low-res NUFFT
5.7GB / 10.4s	272MB / 1.9s	253MB / 1.6s	268MB / 0.4s	136MB / 0.2s

TABLE 2

The quantitative reconstruction quality (NRMSE) of the test set.

	CG-SENSE	PLS	MoDL
3D Kooshball	28.2 dB	28.2 dB	30.1 dB
SNOPI	32.3 dB	32.4 dB	33.6 dB