A Convergent Generalized Krylov Subspace Method for Compressed Sensing MRI Reconstruction With Gradient-Driven Denoisers

Tao Hong, *Member, IEEE*, Umberto Villa, *Member, IEEE*, and Jeffrey A. Fessler, *Fellow, IEEE*

***Abstract*—Model-based reconstruction plays a key role in compressed sensing (CS) MRI, as it incorporates effective image regularizers to improve the quality of reconstruction. The Plug-and-Play and Regularization-by-Denoising frameworks leverage advanced denoisers (e.g., convolutional neural network (CNN)-based denoisers) and have demonstrated strong empirical performance. However, their theoretical guarantees remain limited, as practical CNNs often violate key assumptions. In contrast, gradient-driven denoisers achieve competitive performance, and the required assumptions for theoretical analysis are easily satisfied. However, solving the associated optimization problem remains computationally demanding. To address this challenge, we propose a generalized Krylov subspace method (GKSM) to solve the optimization problem efficiently. Moreover, we also establish rigorous convergence guarantees for GKSM in nonconvex settings. Numerical experiments on CS MRI reconstruction with spiral and radial acquisitions validate both the computational efficiency of GKSM and the accuracy of the theoretical predictions. The proposed optimization method is applicable to any linear inverse problem.**

***Index Terms*—CS MRI, gradient-driven denoiser, Krylov subspace, convergence, spiral and radial acquisitions.**

I. INTRODUCTION

MAGNETIC resonance imaging (MRI) scanners acquire k-space data that represents the Fourier coefficients of the image of interest. However, the acquisition process is inherently slow due to physical, hardware, and sampling constraints [1]. This slow acquisition presents several practical challenges, including patient discomfort, motion artifacts, and reduced throughput. Since the seminal work in [2], compressed sensing (CS) MRI has attracted significant attention in the MRI community [3], [4] for accelerating the acquisition process through structured sampling patterns. Modern CS MRI methods incorporate multiple receiver coils (a.k.a. parallel imaging [5], [6]) to further improve acquisition speed. Image reconstruction in CS MRI requires solving the following composite minimization problem [7]:

$$\mathbf{x}^* = \arg\min_{\mathbf{x}\in\mathbb{C}^N} F(\mathbf{x}) \triangleq \underbrace{\frac{1}{2}\|\mathbf{A}\mathbf{x}-\mathbf{y}\|_2^2}_{h(\mathbf{x})} + \lambda f(\mathbf{x}), \qquad (1)$$

where $\mathbf{A} \in \mathbb{C}^{MC\times N}$ denotes the forward operator that maps the image $\mathbf{x} \in \mathbb{C}^N$ to the measured k-space data $\mathbf{y} \in \mathbb{C}^{MC}$. Here, we consider C receiver coils. The encoding operator $\mathbf{A}$ is a stack of C submatrices $\mathbf{A}_c \in \mathbb{C}^{M\times N}$, each defined as $\mathbf{A}_c = \mathbf{PFS}_c$, where $\mathbf{P}$ is the sampling mask, $\mathbf{F}$ is the (non-uniform) Fourier transform, and $\mathbf{S}_c$ is the coil sensitivity map corresponding to the cth coil, which is patient-dependent. The trade-off parameter $\lambda > 0$ balances $h(\mathbf{x})$ and $f(\mathbf{x})$.

The data-fidelity term $h(\mathbf{x})$ in (1) promotes consistency with the acquired k-space data. In practice, often $M \ll N$ due to under-sampling, making (1) ill-posed. Therefore, incorporating prior knowledge through the regularizer $f(\mathbf{x})$ is essential for stabilizing the reconstruction. The choice of regularization plays a crucial role in reconstruction quality. Traditional hand-crafted regularizers include wavelets [8], total variation (TV) [7], [9], combinations of wavelets and TV [2], [10], dictionary learning [11], [12], and low-rank models [13], to name a few. For reviews of various choices for $f(\mathbf{x})$, see [4], [14], [15].

In the past decade, deep learning (DL) has gained prominence in MRI reconstruction due to its capacity to learn complex image priors directly from large training datasets [16]. Roughly speaking, DL-based approaches can be broadly categorized into end-to-end networks [17] and physics-driven unrolled algorithms [18], [19], [20]. Recently, generative models have emerged as a powerful class for modeling priors in MRI, achieving impressive results across various settings [21].

An alternative to classical DL pipelines is the Plug-and-Play (PnP) [22] and REgularization-by-Denoising (RED) [23] frameworks. PnP and RED integrate powerful denoisers into iterative reconstruction algorithms and have demonstrated competitive performance across various imaging tasks [24], [25], [26], [27], [28], [29]. Early PnP/RED frameworks employed classical denoisers such as the median filter [30], non-local means [31], and BM3D [32]. As convolutional neural network (CNN)–based

Received 17 August 2025; revised 3 December 2025; accepted 9 January 2026. Date of publication 19 January 2026; date of current version 2 February 2026. The work of Tao Hong and Umberto Villa was supported in part by NIH under Grant R01 EB034261 and in part by NIH under Grant R01 EB035618 and Grant R21 EB034344. The associate editor coordinating the review of this article and approving it for publication was Prof. Tolga Çukur. *(Corresponding author: Tao Hong.)*

Tao Hong and Umberto Villa are with the Department of Biomedical Engineering and the Oden Institute for Computational Engineering and Sciences, University of Texas at Austin, Austin, TX 78712 USA (e-mail: tao.hong@austin.utexas.edu; uvilla@austin.utexas.edu).

Jeffrey A. Fessler is with the Department of Electrical and Computer Engineering, University of Michigan, Ann Arbor, MI 48109 USA (e-mail: fessler@umich.edu).

This article has supplementary downloadable material available at https://doi.org/10.1109/TCI.2026.3655489, provided by the authors.

Digital Object Identifier 10.1109/TCI.2026.3655489

denoisers have shown superior performance to classical ones, modern PnP/RED methods typically integrate CNN-based denoisers. Unlike end-to-end or unrolled DL methods that require retraining for each imaging task, PnP and RED leverage learned image priors to flexibly adapt to changes in the forward model without retraining. This adaptability is particularly beneficial in CS MRI reconstruction, where scan-specific variations (e.g., different sampling trajectories and patient-specific sensitivity maps) are common. See [33] for a review of PnP methods that incorporate both classical and CNN-based denoisers in MRI reconstruction.

Despite the empirical success of PnP and RED, their theoretical convergence guarantees remain an active area of research; see [26], [34], [35], [36], [37], [38]. These works typically require that the denoisers either approximate maximum a posteriori or minimum mean squared error estimators, or satisfy the nonexpansiveness condition. However, many successful denoisers—especially those based on CNNs—do not satisfy these assumptions. As a result, PnP and RED with such denoisers cannot be rigorously interpreted as optimization algorithms. Although optimization-free perspectives have been proposed, understanding the behavior of these frameworks remains challenging [26]. One alternative is to train denoisers with additional regularization that enforces a bounded Lipschitz constant [36], [37]. However, guaranteeing strict and tight boundedness in practice remains an open challenge.

Recent efforts have aimed to close the gap between the theoretical foundations and practical effectiveness of PnP and RED by introducing gradient-driven denoisers [39], [40], [41]. In this approach, the unknown image $\mathbf{x}$ is recovered by solving

$$\mathbf{x}^* = \arg\min_{\mathbf{x}\in\mathcal{C}} F(\mathbf{x}) := \underbrace{\frac{1}{2}\|\mathbf{A}\mathbf{x}-\mathbf{y}\|_2^2}_{h(\mathbf{x})} + \lambda f_{\boldsymbol{\theta}}(\mathbf{x}), \tag{2}$$

where $f_{\boldsymbol{\theta}}(\mathbf{x})$ is a scalar-valued energy function parameterized by CNNs that serves as a learned image prior and $\mathcal{C}$ is a closed convex set in $\mathbb{C}^N$. The parameters $\boldsymbol{\theta}$ are learned by enforcing $\mathbf{x} - \nabla_{\mathbf{x}} f_{\boldsymbol{\theta}}(\mathbf{x})$ to act as a denoiser. Thus, the only required assumption is the differentiability of $f_{\boldsymbol{\theta}}$ with respect to $\mathbf{x}$, which allows one to integrate deep learning into inverse problems while maintaining a degree of interpretability—an essential requirement in medical imaging, where reconstructions directly influence diagnostic decisions. For notational simplicity, we omit the subscript $\boldsymbol{\theta}$ and use $\nabla f(\mathbf{x})$ instead of $\nabla_{\mathbf{x}} f(\mathbf{x})$. Moreover, we absorb λ into $f(\mathbf{x})$ in the following discussion since λ is fixed throughout the minimization once it is selected.

Although both $h(\mathbf{x})$ and $f(\mathbf{x})$ in (2) are differentiable, $f(\mathbf{x})$ is generally nonconvex, which poses challenges for designing convergent and efficient algorithms. Cohen et al. [39] applied a projected gradient descent method with a line search to solve (2). Alternatively, Hurault et al. [40] employed the proximal gradient descent method with line search. Both approaches provide convergence guarantees under the assumption that ∇f is Lipschitz continuous. However, these methods typically require hundreds of iterations to converge, which limits their practical applicability. Recently, Hong et al. [42] proposed a convergent complex quasi-Newton proximal method (CQNPM) that significantly reduces the computational time required to solve (2). Their convergence is established under the assumptions that ∇f is Lipschitz continuous and that the proximal Polyak-Łojasiewicz condition holds. Although CQNPM converges faster than existing methods for solving (2), it requires solving a weighted proximal mapping (as defined in [42, equation (3)]) at each iteration. This step requires computing $\mathbf{A}\mathbf{x}$ and $\mathbf{A}^{\mathrm{H}}\mathbf{x}$ multiple times,[1] which can increase the overall computational complexity. Computing $\mathbf{A}\mathbf{x}$ is expensive in MRI reconstruction with many coils, high-resolution images, or many interleaves or spokes in non-Cartesian acquisitions. Drawing inspiration from Krylov subspace methods (KSMs) [43], we propose a generalized Krylov subspace method (GKSM) for efficiently solving (2), which requires computing $\mathbf{A}\mathbf{x}$, $\mathbf{A}^{\mathrm{H}}\mathbf{x}$, and $\nabla f(\mathbf{x})$ only once per iteration. Our main contributions are summarized as follows:

- We propose a generalized Krylov subspace method (GKSM) for efficiently solving (2).
- We present a rigorous convergence analysis of GKSM in nonconvex settings, along with the convergence rate of the cost function values.
- We extensively evaluate the performance of GKSM on brain (respectively, knee) images from the dataset described in the MoDL paper [18] (respectively, the NYU fastMRI dataset [44]). The k-space data are simulated from the reconstructed complex-valued images using spiral and radial sampling trajectories. We also empirically validate the accuracy of the convergence analysis.

The rest of this paper is organized as follows. Section II reviews the preliminaries on KSMs and discusses related work that generalizes KSMs for solving inverse problems. Section III describes GKSM in detail. Section IV provides a rigorous convergence analysis of GKSM. Section V reports experimental results that evaluate the performance of GKSM and empirically validate the theoretical analysis.

II. PRELIMINARIES ON KRYLOV SUBSPACE METHODS

This section first introduces KSMs, which were primarily developed for solving linear equations. We then review related generalized Krylov methods for linear inverse problems, along with existing theoretical results. Our main goal in this section is to provide a sketch of the key developments in KSMs, from their origins in solving linear equations to their generalization for inverse problems.

KSMs are a class of iterative algorithms for solving problems of the form

$$\bar{\mathbf{A}}\mathbf{x} = \mathbf{b}, \tag{3}$$

where $\bar{\mathbf{A}} \in \mathbb{R}^{N\times N}$ is typically sparse, ill-conditioned, and large-scale. At kth iteration, KSMs construct an approximate solution to $\mathbf{x}^*$ within the Krylov subspace:

$$\mathcal{K}_k(\bar{\mathbf{A}}, \bar{\mathbf{r}}_1) = \text{span}\left\{\bar{\mathbf{r}}_1, \bar{\mathbf{A}}\bar{\mathbf{r}}_1, \bar{\mathbf{A}}^2\bar{\mathbf{r}}_1, \ldots, \bar{\mathbf{A}}^{k-1}\bar{\mathbf{r}}_1\right\}, \tag{4}$$

where $\bar{\mathbf{r}}_1 = \mathbf{b} - \bar{\mathbf{A}}\mathbf{x}_1$ is the initial residual. The approximate solution $\mathbf{x}_{k+1}$ is obtained by seeking $\mathbf{x}_{k+1} \in \mathbf{x}_1 + \mathcal{K}_k(\bar{\mathbf{A}}, \bar{\mathbf{r}}_1)$ that minimizes a chosen norm of the residual. The most widely

---

[1] H denotes the Hermitian transpose operator.

used KSMs include, but are not limited to, the conjugate gradient method [45], LSQR [46], BiCGSTAB [47], and the generalized minimal residual method [48]. These methods are designed for different types of matrices $\bar{\mathbf{A}}$, such as symmetric positive definite, non-symmetric, or indefinite, to name a few [43]. Moreover, KSMs can incorporate preconditioners to further accelerate convergence [49].

Many inverse problems with variational regularizers [50], [51] can be modeled as the following ℓ_p-ℓ_q optimization problem:

$$\min_{\mathbf{x}\in\mathbb{R}^N} \frac{1}{p}\|\mathbf{A}\mathbf{x}-\mathbf{y}\|_p^p + \frac{\lambda}{q}\|\mathbf{W}\mathbf{x}\|_q^q, \tag{5}$$

where $0 < p, q \leq 2$, and $\mathbf{W}$ represents a transform such as a wavelet transform. KSMs have been generalized to solve problems such as (5). For $p = q = 2$, Lampe et al. [52] presented a generalized KSM to address (5) in which the parameter λ is adaptively adjusted to keep $\|\mathbf{A}\mathbf{x}-\mathbf{y}\|_2$ sufficiently close to a prescribed tolerance. Lanza et al. [53] proposed to solve (5) using a KSM along with an iteratively reweighted approach. Moreover, Huang et al. combined Krylov subspace-based method with majorization minimization for solving (5) [54]. To avoid the inner-outer iterations when using the iteratively approach in [53], several flexible Krylov subspaces were proposed to improve efficiency [55], [56], [57]. See [50], [58] for a review of using KSMs for inverse problems.

Besides their extension to inverse problems, rigorous convergence analyses of KSMs remain an open research area. Lanza et al. [53] showed that the iterates converged to a minimizer of (5) for $1 \leq p, q \leq 2$ if $\ker(\mathbf{A}^{\mathrm{T}}\mathbf{A}) \cap \ker(\mathbf{W}^{\mathrm{T}}\mathbf{W}) = \{\mathbf{0}\}$, where $\ker(\cdot)$ denotes the null space of the matrix, and the constructed Krylov subspace fully represents the entire image domain. Similar convergence results can also be found in [54], [59]. Other works [55], [56], [57] proved that the cost function values are monotonically decreasing and that the iterates converge to a stationary point. For brevity, we present only the convergence results of KSMs for inverse problems. See [60], [61] and the references therein for discussions on KSMs and their convergence in other contexts.

III. PROPOSED METHOD

This section provides the details of our GKSM for solving (2). We first discuss the case where $\mathcal{C} = \mathbb{C}^N$ and then describe how to incorporate a convex constraint. Lastly, we provide further discussion of GKSM to offer additional insights.

Given a subspace basis $\mathbf{V}_k \in \mathbb{C}^{N\times k}$ satisfying $\mathbf{V}_k^{\mathrm{H}}\mathbf{V}_k = \mathbf{I}_k$ where $\mathbf{I}_k$ is the identity matrix with dimension k, a Hermitian positive definite matrix $\mathbf{B}_k \in \mathbb{C}^{N\times N}$, $\mathbf{B}_k \succ 0$, and $\alpha_k \in \mathbb{R}$, $\alpha_k > 0$, GKSM solves the following problem at the kth iteration to obtain the coefficient $\boldsymbol{\beta}_k \in \mathbb{C}^k$:

$$\boldsymbol{\beta}_k = \arg\min_{\boldsymbol{\beta}\in\mathbb{C}^k} \underbrace{\frac{1}{2}\|\mathbf{A}\mathbf{x}-\mathbf{y}\|_2^2 + \bar{f}(\mathbf{x},\mathbf{x}_k,\mathbf{B}_k,\alpha_k)}_{\bar{F}(\mathbf{x},\mathbf{x}_k)}, \tag{6}$$

where $\mathbf{x} = \mathbf{V}_k\boldsymbol{\beta}$, $\bar{f}(\mathbf{x},\mathbf{x}_k,\mathbf{B}_k,\alpha_k) \equiv \langle\nabla f(\mathbf{x}_k),\mathbf{x}\rangle + \frac{1}{2\alpha_k}\|\mathbf{x}-\mathbf{x}_k\|_{\mathbf{B}_k}^2$ is a quadratic proximal term, and $\|\mathbf{x}\|_{\mathbf{B}_k}^2 = \mathbf{x}^{\mathrm{H}}\mathbf{B}_k\mathbf{x}$. Then the next image iterate is $\mathbf{x}_{k+1} = \mathbf{V}_k\boldsymbol{\beta}_k$. Rewriting (6) in terms of $\boldsymbol{\beta}$ and reorganizing yields

$$\boldsymbol{\beta}_k = \arg\min_{\boldsymbol{\beta}\in\mathbb{C}^k} \left\| \begin{bmatrix} \mathbf{A}\mathbf{V}_k\boldsymbol{\beta} \\ \bar{\mathbf{B}}_k^{\frac{1}{2}}\mathbf{V}_k\boldsymbol{\beta} \end{bmatrix} - \begin{bmatrix} \mathbf{y} \\ \bar{\mathbf{B}}_k^{\frac{1}{2}}\mathbf{w}_k \end{bmatrix} \right\|_2^2, \tag{7}$$

where $\mathbf{w}_k = \mathbf{x}_k - \alpha_k\mathbf{B}_k^{-1}\nabla f(\mathbf{x}_k)$ and $\bar{\mathbf{B}}_k^{1/2}$ denotes the principal matrix square root of $\bar{\mathbf{B}}_k = \mathbf{B}_k/\alpha_k$ and it is unique [62]. Note that $\mathbf{A}\mathbf{V}_k$ is built incrementally during the algorithm. Compared with the image size, the dimension of $\boldsymbol{\beta}_k$ is relatively low because the number of iterations will be significantly smaller than the image dimension. Therefore, we solve (7) directly, i.e.,

$$\boldsymbol{\beta}_k = (\mathbf{V}_k^{\mathrm{H}}\mathbf{A}^{\mathrm{H}}\mathbf{A}\mathbf{V}_k + \mathbf{V}_k^{\mathrm{H}}\bar{\mathbf{B}}_k\mathbf{V}_k)^{-1}\mathbf{V}_k^{\mathrm{H}}(\mathbf{A}^{\mathrm{H}}\mathbf{y} + \bar{\mathbf{B}}_k\mathbf{w}_k). \tag{8}$$

Here the matrix being inverted is only $k \times k$ with $k \ll N$.

To enrich the subspace after the kth iteration, we first compute the gradient of the objective function in (6) with respect to $\mathbf{x}$ at $\mathbf{x} = \mathbf{x}_{k+1}$, i.e.,

$$\mathbf{r}_k = \nabla_{\mathbf{x}}\bar{F}(\mathbf{x}_{k+1},\mathbf{x}_k). \tag{9}$$

Then we set $\mathbf{v}_{k+1} = \tilde{\mathbf{r}}_k/\|\tilde{\mathbf{r}}_k\|$ with $\tilde{\mathbf{r}}_k = (\mathbf{I}_N - \mathbf{V}_k\mathbf{V}_k^{\mathrm{H}})\mathbf{r}_k$. The new subspace basis $\mathbf{V}_{k+1}$ is formulated as $[\mathbf{V}_k \quad \mathbf{v}_{k+1}]$. If $\|\tilde{\mathbf{r}}_k\| = 0$, we simply skip the update of $\mathbf{V}_k$. Note that the dimension of $\boldsymbol{\beta}_k$ is equal to the number of columns of $\mathbf{V}_k$. This dimension may be smaller than k if the event $\|\tilde{\mathbf{r}}_k\| = 0$ occurs, in which case the number of columns of $\mathbf{V}_k$ is also smaller than k. Algorithm 1 summarizes the detailed steps of GKSM. To establish the convergence rate of the cost function values, we introduce an additional step 18 in Algorithm 1, as the generated $\mathbf{V}_k$ does not necessarily span the entire image domain. Note that GKSM reduces to CQNPM [42] when $\mathbf{V}_k = \mathbf{I}_N$. For this case, we simply apply the accelerated gradient descent method to solve (6). If the regularizer f in (2) is a quadratic function, then the subspace spanned by $\mathbf{V}_k$ simplifies to the classical Krylov subspace in (4). To obtain $\mathbf{B}_k$, we adopt the algorithm presented in [42, Algorithm 2] such that $\mathbf{B}_k$ is an estimate of the Hessian matrix of $f(\mathbf{x})$ that is guaranteed to be Hermitian positive definite. For completeness, Algorithm 2 provides the detailed steps for computing $\mathbf{B}_k$. The operator $\Re(\cdot)$ in Algorithm 2 extracts the real part.

A. *Incorporating a Convex Constraint*

This part extends GKSM to handle a convex constraint on $\mathbf{x}$ with a slight increase in computational cost. To ensure $\mathbf{x} \in \mathcal{C}$, we solve the following problem for $\boldsymbol{\beta}_k$ instead of (7):

$$\boldsymbol{\beta}_k = \arg\min_{(\mathbf{V}_k\boldsymbol{\beta})\in\mathcal{C}} \left\| \begin{bmatrix} \mathbf{A}\mathbf{V}_k\boldsymbol{\beta} \\ \bar{\mathbf{B}}_k^{\frac{1}{2}}\mathbf{V}_k\boldsymbol{\beta} \end{bmatrix} - \begin{bmatrix} \mathbf{y} \\ \bar{\mathbf{B}}_k^{\frac{1}{2}}\mathbf{w}_k \end{bmatrix} \right\|_2^2. \tag{11}$$

Letting $\mathbf{z} = \mathbf{V}_k\boldsymbol{\beta}$ and using the fact that $\mathbf{V}_k^{\mathrm{H}}\mathbf{V}_k = \mathbf{I}_k$, we have $\boldsymbol{\beta} = \mathbf{V}_k^{\mathrm{H}}\mathbf{z}$. Thus we rewrite (11) as

$$\mathbf{x}_{k+1} = \arg\min_{\mathbf{z}\in\mathcal{C}} \left\| \begin{bmatrix} \mathbf{A}\mathbf{V}_k\mathbf{V}_k^{\mathrm{H}}\mathbf{z} \\ \bar{\mathbf{B}}_k^{\frac{1}{2}}\mathbf{V}_k\mathbf{V}_k^{\mathrm{H}}\mathbf{z} \end{bmatrix} - \begin{bmatrix} \mathbf{y} \\ \bar{\mathbf{B}}_k^{\frac{1}{2}}\mathbf{w}_k \end{bmatrix} \right\|_2^2. \tag{12}$$

Since the objective function of (12) is differentiable, we simply apply the accelerated projection gradient method [63] to

Algorithm 1: Generalized Krylov Subspace Method (GKSM).

```
Initialization: $\mathbf{x}_1$, stepsize $\alpha_k > 0$, $\mathbf{V}_1 = \frac{\mathbf{A}^{\mathrm{H}}\mathbf{y}}{\|\mathbf{A}^{\mathrm{H}}\mathbf{y}\|}$, $\mathbf{AV}_1$,
  maximal number of subspace iterations $K$, and maximal
  number of total iterations Max_Iter
Iteration:
1:  for $k = 1, 2, \ldots,$ Max_Iter do
2:    Compute $\nabla f(\mathbf{x}_k)$
3:    Set $\mathbf{B}_k$ using Algorithm 2
4:    Compute $\boldsymbol{\beta}_k$ using (8) (or solve (11) for $\boldsymbol{\beta}_k$ if a
      convex constraint is enforced)
5:    Compute $\mathbf{x}_{k+1} \leftarrow \mathbf{V}_k\boldsymbol{\beta}_k$
6:    if $k \leq K$ then
7:      Compute $\mathbf{r}_k \leftarrow \nabla_{\mathbf{x}}\bar{F}(\mathbf{x}_{k+1}, \mathbf{x}_k)$
8:      $\tilde{\mathbf{r}}_k \leftarrow (\mathbf{I} - \mathbf{V}_k\mathbf{V}_k^{\mathrm{H}})\mathbf{r}_k$
9:      if $\|\tilde{\mathbf{r}}_k\| \neq 0$ then
10:       $\mathbf{v}_{k+1} \leftarrow \tilde{\mathbf{r}}_k/\|\tilde{\mathbf{r}}_k\|$
11:       $\mathbf{V}_{k+1} \leftarrow \begin{bmatrix}\mathbf{V}_k & \mathbf{v}_{k+1}\end{bmatrix}$
12:       $\mathbf{AV}_{k+1} \leftarrow \begin{bmatrix}\mathbf{AV}_k & \mathbf{Av}_{k+1}\end{bmatrix}$
13:     else
14:       $\mathbf{V}_{k+1} \leftarrow \mathbf{V}_k$
15:       $\mathbf{AV}_{k+1} \leftarrow \mathbf{AV}_k$
16:     end if
17:   else
18:     $\mathbf{V}_{k+1} \leftarrow \mathbf{I}_N$
19:   end if
20: end for
```

solve (12) efficiently. Although $\mathbf{z}$ has the same dimension as $\mathbf{x}$, solving (12) only requires simple matrix-vector multiplications since $\mathbf{AV}_k$ and $\bar{\mathbf{B}}_k^{\frac{1}{2}}\mathbf{V}_k$ are precomputed and saved. In practice, to extend Algorithm 1 to handle the convex constraint, we only need to replace the computation of $\boldsymbol{\beta}_k$ at step 4 with the solution of (11) by solving (12).

B. Discussion

The dominant computations at each iteration in Algorithm 1 involve computing $\nabla f(\mathbf{x})$, $\mathbf{Ax}$ and $\mathbf{A}^{\mathrm{H}}\mathbf{x}$ once, and the overall computational cost per iteration is lower than that of the methods proposed in [40], [42] that require dozens of evaluations of $\mathbf{Ax}$ and $\mathbf{A}^{\mathrm{H}}\mathbf{x}$ per iteration. Apart from computational efficiency, GKSM requires additional memory, because it must store $\mathbf{V}_k$ and $\mathbf{AV}_k$. Thus, GKSM may become memory-prohibitive for very large-scale problems. A practical heuristic to address this challenge is to use a restart strategy, in which we cyclically set $\mathbf{V}_k = \mathbf{x}_{k+1}/\|\mathbf{x}_{k+1}\|$. This not only reduces the memory usage but also lowers the computational cost. Algorithmically, when a restart is triggered, this is equivalent to rerunning Algorithm 1 with $\mathbf{x}_{k+1}$ as the new initial value and $\mathbf{V}_1 = \mathbf{x}_{k+1}/\|\mathbf{x}_{k+1}\|$. Moreover, in practice, there is no guarantee that a column-orthogonal matrix $\mathbf{V}_{k+1}$ can always be constructed from $\mathbf{r}_k$, since $\|\tilde{\mathbf{r}}_k\|$ may be zero. The restart strategy typically helps to escape such a situation. However, in our experimental settings, we never found that $\|\tilde{\mathbf{r}}_k\| = 0$. We leave the study of restart strategies to future work.

Algorithm 2: Modified Memory Efficient Self-Scaling Hermitian Rank-1 Method.

```
Initialization: $\mathbf{x}_{k-1}$, $\mathbf{x}_k$, $\nabla f(\mathbf{x}_{k-1})$, $\nabla f(\mathbf{x}_k)$, $\delta > 0$,
  $\nu_1 \in (0, 1)$, and $\nu_2 \in (1, \infty)$
Set $\mathbf{s}_k \leftarrow \mathbf{x}_k - \mathbf{x}_{k-1}$ and $\mathbf{m}_k \leftarrow (\nabla f(\mathbf{x}_k) - \nabla f(\mathbf{x}_{k-1}))$
Compute $a$ such that
    $\min_a\{a \in [0,1] | \bar{\mathbf{m}}_k = a\mathbf{s}_k + (1-a)\mathbf{m}_k\}$
    satisfies $\nu_1 \leq \frac{\Re(\langle \mathbf{s}_k, \bar{\mathbf{m}}_k\rangle)}{\langle \mathbf{s}_k, \mathbf{s}_k\rangle}$ and $\frac{\langle \bar{\mathbf{m}}_k, \bar{\mathbf{m}}_k\rangle}{\Re(\langle \mathbf{s}_k, \bar{\mathbf{m}}_k\rangle)} \leq \nu_2$        (10)
Compute $\tau_k \leftarrow \frac{\langle \mathbf{s}_k, \mathbf{s}_k\rangle}{\Re(\langle \mathbf{s}_k, \bar{\mathbf{m}}_k\rangle)} - \sqrt{\left(\frac{\langle \mathbf{s}_k, \mathbf{s}_k\rangle}{\Re(\langle \mathbf{s}_k, \bar{\mathbf{m}}_k\rangle)}\right)^2 - \frac{\langle \mathbf{s}_k, \mathbf{s}_k\rangle}{\langle \bar{\mathbf{m}}_k, \bar{\mathbf{m}}_k\rangle}}$
$\rho_k \leftarrow \Re(\langle \mathbf{s}_k - \tau_k\bar{\mathbf{m}}_k, \bar{\mathbf{m}}_k\rangle)$
if $\rho_k \leq \delta\|\mathbf{s}_k - \tau_k\bar{\mathbf{m}}_k\|\|\bar{\mathbf{m}}_k\|$ then
  $\mathbf{u}_k \leftarrow \mathbf{0}$
else
  $\mathbf{u}_k \leftarrow \mathbf{s}_k - \tau_k\bar{\mathbf{m}}_k$
end if
$\rho_k^{\mathbf{B}} \leftarrow \tau_k^2\rho_k + \tau_k\mathbf{u}_k^{\mathrm{H}}\mathbf{u}_k$
Return: $\mathbf{B}_k \leftarrow \tau_k^{-1}\mathbf{I}_N - \frac{\mathbf{u}_k\mathbf{u}_k^{\mathrm{H}}}{\rho_k^{\mathbf{B}}}$
```

The following convergence analysis shows that GKSM is guaranteed to monotonically decrease the cost function value every iteration. Thus, it is safe to set $K =$ `Max_Iter` in practice. However, the convergence rate of the cost function values to a minimum remains unclear since we cannot guarantee that $\mathbf{V}_k$ will span the entire image space after a finite number of iterations. To better characterize the convergence rate of the cost values, we introduce step 18 in Algorithm 1. This addition allows us to explicitly quantify the cost convergence after K iterations.

IV. CONVERGENCE ANALYSIS

This section provides a rigorous convergence analysis of using Algorithm 1 to solve (2). Because the unconstrained problem is a special case of the constrained one, we focus our analysis on using GKSM for problems with constraints. We use the notation $F_{\mathcal{C}}(\mathbf{x}) = F(\mathbf{x}) + \iota_{\mathcal{C}}(\mathbf{x})$ in the following analysis, where $\iota_{\mathcal{C}}(\mathbf{x})$ denotes the characteristic function, defined as $\iota_{\mathcal{C}}(\mathbf{x}) = 0$ if $\mathbf{x} \in \mathcal{C}$, and $\iota_{\mathcal{C}}(\mathbf{x}) = +\infty$ otherwise. Here, we assume that $\mathcal{C}$ is convex and that its indicator function $\iota_{\mathcal{C}}$ is lower semicontinuous. Before presenting our main convergence results, we first review the definition of Kurdyka–Łojasiewicz (KL) inequality and make one assumption, followed by four supporting lemmas.

Definition 1 (Kurdyka–Łojasiewicz inequality [64], [65]): Let $\chi(\mathbf{x}) : \mathbb{C}^N \to (-\infty, +\infty]$ be a proper, lower semicontinuous function. We say that χ satisfies the Kurdyka–Łojasiewicz (KL) inequality at a point $\bar{\mathbf{x}} \in \mathrm{dom}(\partial\chi)$ if there exist a $\eta > 0$, a neighborhood $\mathcal{U}$ of $\bar{\mathbf{x}}$, and a continuous concave function

$\varphi : [0, \eta) \to \mathbb{R}_+$ that is continuously differentiable on $(0, \eta)$ and satisfies $\varphi(0) = 0$ and $\varphi'(s) > 0$ for all $s \in (0, \eta)$, such that

$$\varphi' \left(|\chi(\mathbf{x}) - \chi(\bar{\mathbf{x}})|\right) \cdot \operatorname{dist}\left(\mathbf{0}, \partial\chi(\mathbf{x})\right) \geq 1.$$

holds for all $\mathbf{x} \in \mathcal{U} \cap \{\mathbf{x} \in \mathbb{C}^N : |\chi(\mathbf{x}) - \chi(\bar{\mathbf{x}})| < \eta\}$. Here, $\partial\chi(\mathbf{x})$ denotes the subgradient of $\chi(\mathbf{x})$, and $\operatorname{dist}(\cdot, \cdot)$ denotes Euclidean distance.

In Definition 1, $\varphi(s)$ is called the desingularization function. If $\varphi(s)$ holds the form $\varphi(s) = cs^{1-t}$ for $t \in [0, 1)$ and $c > 0$, then we say that $\chi(\mathbf{x})$ has the KL property at $\bar{\mathbf{x}}$ with an exponent of t.

Assumption 1 (L-Smooth f): Assume that $f : \mathbb{C}^n \to (-\infty, +\infty]$ is a proper, lower semicontinuous, and lower bounded function. Further assume that the gradient of f is L-Lipschitz continuous. That is, $\forall \mathbf{x}_1,\ \mathbf{x}_2 \in \mathbb{C}^N$, there exists a $L > 0$ such that the following inequality holds:

$$\|\nabla f(\mathbf{x}_1) - \nabla f(\mathbf{x}_2)\| \leq L \|\mathbf{x}_1 - \mathbf{x}_2\|. \tag{13}$$

Lemma 1 (Majorizer of f [42, Lemma 1]): Let $f : \mathbb{C}^N \to (-\infty, \infty]$ be an L-smooth function. Then for any $\mathbf{x}_1, \mathbf{x}_2 \in \mathbb{C}^N$, we have

$$f(\mathbf{x}_2) \leq f(\mathbf{x}_1) + \Re\left\{\langle \nabla f(\mathbf{x}_1), \mathbf{x}_2 - \mathbf{x}_1\rangle\right\} + \frac{L}{2}\|\mathbf{x}_1 - \mathbf{x}_2\|_2^2. \tag{14}$$

Lemma 2 (Bounded Hessian [42, Lemma 4]): The approximate Hessian matrices $\mathbf{B}_k$ generated by Algorithm 2 satisfy the following inequality

$$\underline{\eta}\,\mathbf{I} \preceq \mathbf{B}_k \preceq \overline{\eta}\,\mathbf{I},$$

where $0 < \underline{\eta} < \overline{\eta} < \infty$.

Lemma 3: By running Algorithm 1 for solving (2), we have the following inequality at kth iteration,

$$\begin{aligned}\Re\left\{\langle \nabla f(\mathbf{x}_k), \mathbf{x}_{k+1} - \mathbf{x}_k\rangle\right\} \leq\ & h(\mathbf{x}_k) - h(\mathbf{x}_{k+1}) \\ & - \frac{1}{2}\|\mathbf{x}_k - \mathbf{x}_{k+1}\|^2_{(\frac{2}{\alpha_k}\mathbf{B}_k - \underline{\eta}\mathbf{I}_N)}.\end{aligned} \tag{15}$$

Lemma 4: Suppose the elements in $\{\phi_k > 0\}_{k\geq 1}$ satisfy

$$\phi_{k+1}^{2t} \leq \gamma(\phi_k - \phi_{k+1}) \text{ and } \phi_{k+1} \leq \phi_k,$$

where $t \in (0, 1)$ and $\gamma > 0$. Then we have the following upper bounds for ϕ_{k+1},

$$\phi_{k+1} \leq \begin{cases} \left(1 - \frac{\phi_1^{2t-1}}{\gamma + \phi_1^{2t-1}}\right)^k \phi_1, & t \in (0, \frac{1}{2}] \\ \left(\phi_1^{1-2t} + \frac{(2t-1)(1-\sigma)^{2t}}{2\gamma} k\right)_+^{\frac{1}{1-2t}}, & t \in (\frac{1}{2}, 1), \end{cases} \tag{16}$$

where $\sigma \in (0, 1)$ and $(\cdot)_+ = \max(\cdot, 0)$.

Lemmas 1 and 2 were already demonstrated in [42], so we omit their proofs here. The proofs of Lemmas 3 and 4 are provided in Appendices A and B. Lemma 4 is used to establish the convergence rates of the cost function sequence for different values of t in the KL inequality when running Algorithm 1. Theorems 1 and 2 summarize our main convergence results.

Theorem 1 (Descent properties of Algorithm 1, $K \leq +\infty$): Let $\alpha_k \in (0, \frac{2\underline{\eta}}{\underline{\eta}+L})$ and $\Delta_k \triangleq \min_{k' \leq k} \|\mathbf{x}_{k'+1} - \mathbf{x}_{k'}\|_2^2$. Under Assumption 1, by running $k < K$ iterations of Algorithm 1 to solve (2), we have

- $\Delta_k \leq \frac{F(\mathbf{x}_1) - F^*}{\upsilon k}$ and $F(\mathbf{x}_{k+1}) \leq F(\mathbf{x}_k)$, where $\upsilon = \min_k\{\underline{\eta}/\alpha_k - (\underline{\eta} + L)/2\}$, F^* denotes the minimum of (2), and $\mathbf{x}_1$ is the initial iterate.
- $\|\mathbf{x}_{k+1} - \mathbf{x}_k\| \to 0$ as $k \to \infty$.

Theorem 2 (Convergence rates, $K < +\infty$): Let $\alpha_k \in (0, \frac{2\underline{\eta}}{\underline{\eta}+L})$ and $\mathcal{B}(\mathbf{x}^*, \Lambda) = \{\mathbf{x} \in \mathbb{C}^N \mid \|\mathbf{x} - \mathbf{x}^*\| \leq \Lambda\}$. Under Assumption 1, by running Algorithm 1 $k > K$ iterations to solve (2), we have

- $\|\mathbf{x}_{k+1} - \mathbf{x}_k\| \to 0$ as $k \to \infty$ and all cluster points of the sequence $\{\mathbf{x}_k\}_{k>K}$ are critical points of (2).
- Assume $\mathbf{x}_k$ converges to $\mathbf{x}^*$ and $F_{\mathcal{C}}$ satisfies the KL inequality. There exists $\Lambda > 0$ such that $F_{\mathcal{C}}$ at $\bar{\mathbf{x}} = \mathbf{x}^*$ in a neighborhood of $\mathcal{U}$ containing $\mathcal{B}(\mathbf{x}^*, \Lambda)$. Then there also exists $K' > K$, such that, $\mathbf{x}_k \in \mathcal{B}(\mathbf{x}^*, \Lambda)$ and $|F_{\mathcal{C}}(\mathbf{x}_k) - F^*| < \eta$ for all $k \geq K'$. For $k \geq K'$, we have the following convergence rates of the cost function values for $t \in [0, 1)$:
  1) $F(\mathbf{x}_{k+1}) - F^* \leq (F_{K'} - \frac{1}{\gamma}(k - K' + 1))_+, t = 0,$
  2) $F(\mathbf{x}_{k+1}) - F^* \leq \left(1 - \frac{F_{K'}^{2t-1}}{F_{K'}^{2t-1} + \gamma}\right)^{k-K'+1} F_{K'}, \quad t \in (0, \frac{1}{2}],$
  3) $F(\mathbf{x}_{k+1}) - F^* \leq (F_{K'}^{1-2t} + q(k - K' + 1))_+^{\frac{1}{1-2t}}, \quad t \in (\frac{1}{2}, 1).$

  where $F_{K'} = F(\mathbf{x}_{K'}) - F^*$, $\sigma \in (0, 1)$, $\gamma = \max_k ([c(L\alpha_k + \overline{\eta})]^2)/(\upsilon(1-t)^2\alpha_k^2))$, and $q = (t - 1/2)(1 - \sigma)^{2t}/\gamma$.

Theorem 1 states that GKSM leads to $\|\mathbf{x}_{k+1} - \mathbf{x}_k\| \to 0$ as $k \to \infty$ for any K, either finite or infinite. Our first result in Theorem 2 establishes all cluster points of the sequence $\{\mathbf{x}_k\}_{k>K}$ are critical points of (2) for finite K. The second result in Theorem 2 provides convergence rates of the cost values for $t \in [0, 1)$ after $K' > K$ iterations. During the algorithm's execution, all iterates remain in $\mathcal{C}$, so that $F_{\mathcal{C}}(\mathbf{x}_k) = F(\mathbf{x}_k)$ for all k. For ease of notation, we use $F(\mathbf{x})$ instead of $F_{\mathcal{C}}(\mathbf{x})$ to express the convergence rates in Theorem 2. Note that if $F(\mathbf{x}_{k+1}) = F^*$, the left hand side is identically zero and therefore the convergence rate bounds in Theorem 2 hold trivially.

Note that Algorithm 1 reduces to CQNPM [42] when K is finite and $k > K$. Theorem 2 then extends the theoretical analysis in [42] to a more general class of functions F_C. Specifically, our analysis relies on the KL inequality, which is weaker than the Polyak-Łojasiewicz (PL) inequality[2] used in [42]. Section V-C empirically studies the convergence behavior of Algorithm 1 to validate our theoretical analysis.

V. NUMERICAL EXPERIMENTS

This section studies the performance of GKSM for CS MRI reconstruction with spiral and radial sampling trajectories. Note that the experimental and algorithmic settings used here are similar to those in our previous work [42]. For convenience, we briefly re-describe them in this paper. Then, we present the reconstruction results and study the convergence behavior of GKSM empirically.

---

[2] The PL inequality corresponds to a special case of the KL inequality with $t = \frac{1}{2}$.

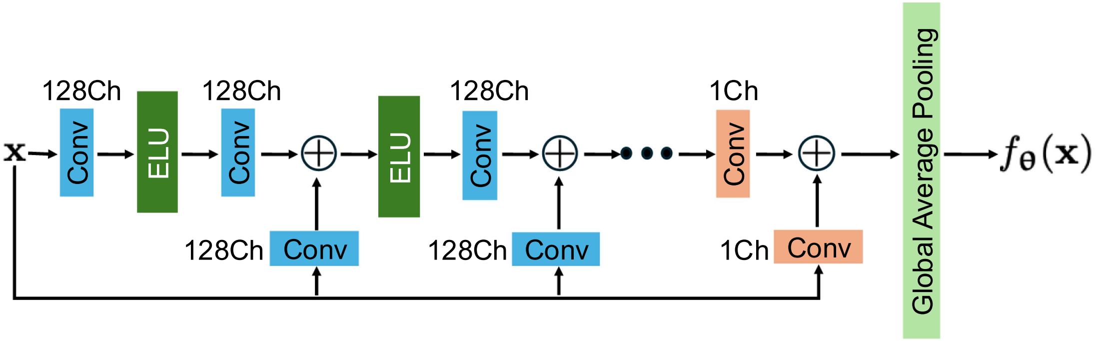
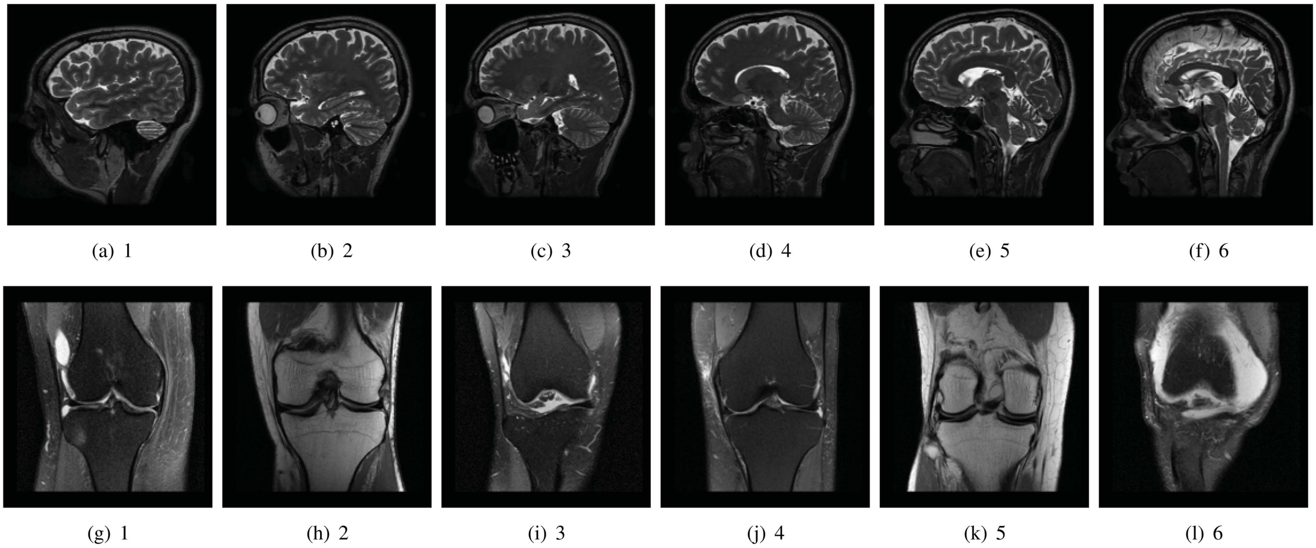


Fig. 1. The neural network architecture used to construct the energy function $f_{\boldsymbol{\theta}}(\mathbf{x})$ is based on [39]. The convolutional kernels have size 3×3 with a stride of one.


(a) 1 (b) 2 (c) 3 (d) 4 (e) 5 (f) 6

(g) 1 (h) 2 (i) 3 (j) 4 (k) 5 (l) 6

Fig. 2. The magnitude of the six brain and knee complex-valued ground truth images.

Experimental Settings: The performance of GKSM are evaluated on both brain and knee MRI datasets. For the brain images, we used the dataset from [18], which contains 360 training and 164 test images. For the knee images, we adopted the multi-coil knee dataset from the NYU fastMRI [44]. The ESPIRiT algorithm [66] was used to obtain the complex-valued images from the raw k-space data. All images were then resized to a uniform image size of 256×256 and normalized such that the maximum magnitude was one. The network architecture proposed in [39] with the addition of bias terms is used to construct $f(\mathbf{x})$, see Fig. 1. The number of layers is set to six. Note that other neural network architectures can also be used to train the gradient-driven denoiser. The only requirement is that the nonlinear activation functions be differentiable. For example, Hurault et al. [40] used DRUNet by replacing the ReLU activations with ELU to construct $f(\mathbf{x})$. The noisy images were generated by adding i.i.d. Gaussian noise with variance $1/255$ to the clean images. The network was trained using the mean squared error loss. We employed the ADAM optimizer [67] with an initial learning rate of 10^{-3}, which was halved every 4,000 iterations. Training was performed for a total of 18,000 iterations with a batch size of 64. Although we trained separate denoisers for the brain and knee datasets, different sampling trajectories used the same denoiser.

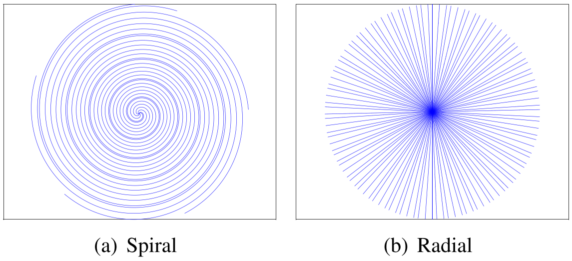

(a) Spiral (b) Radial

Fig. 3. The spiral (a) and radial (b) sampling trajectories.

To assess reconstruction quality, we selected six brain and knee test images as ground truth. Fig. 2 displays the magnitudes of these images. For spiral acquisition, we used six interleaves with 1688 readout points and 32 coils. Radial acquisition employed 55 spokes with golden-angle rotation, 1024 readout points, and 32 coils. Fig. 3 describes these sampling trajectories. To simulate k-space data, we applied the forward model to the ground-truth images and then added the complex

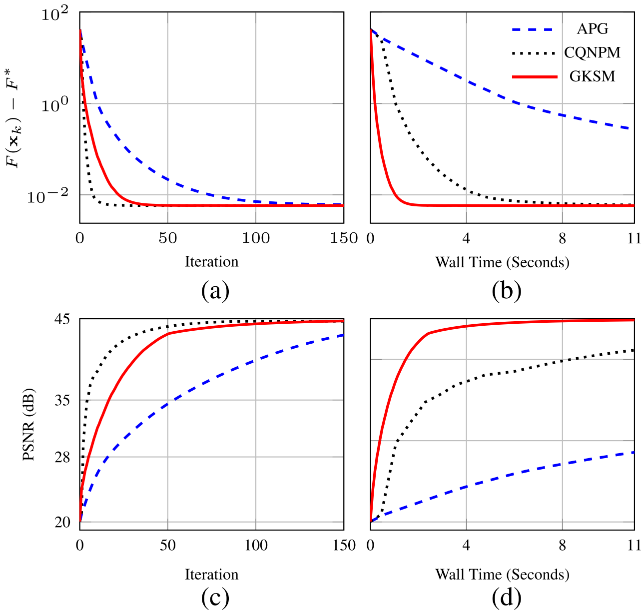


Fig. 4. Comparison of different methods with spiral acquisition on the brain 1 image for $\varepsilon = 5 \times 10^{-3}$. (a), (b): cost values versus iteration and wall time; (c), (d): PSNR values versus iteration and wall time.

i.i.d. Gaussian noise (zero mean, variance 10^{-4}), resulting in an input SNR of approximately 21 dB. In the reconstruction, we employed coil compression [68] to reduce the number of coils from 32 to 20 virtual coils, thereby lowering the computational cost. All experiments used simulated k-space data and were implemented in PyTorch and ran on an NVIDIA A100 GPU.

Algorithmic Settings: The previous work [10] already showed that the accelerated proximal gradient method (dubbed APG) [69] is faster than the projected gradient descent method [39] and the proximal gradient method [40] for addressing (2). Thus, we mainly compared GKSM with CQNPM [10] and APG in this paper. To test the applicability of GKSM for a constrained problem, we used the constraint set $\mathcal{C} = \{\mathbf{x} \mid \|\mathbf{x}\|_\infty \leq 1\}$ in all competing methods. However, in practical CS MRI reconstruction, we generally do not impose such a constraint. For plots involving F^*, we ran APG for 500 iterations and defined $F^* = F(\mathbf{x}_{500}) - \varepsilon$ for a small constant $\varepsilon > 0$. Unless otherwise specified, we set $K = \texttt{Max\_Iter}$ in the following experiments. Algorithm 2 used $\delta = 10^{-8}$, $\nu_1 = 2 \times 10^{-6}$, and $\nu_2 = 200$.

A. Spiral Acquisition Reconstruction

Fig. 4 summarizes the cost and PSNR values versus the number of iterations and wall time for each method on the brain 1 image. Fig. 4.s (a) and (c) show CQNPM was the fastest algorithm in terms of the number of iterations. GKSM achieved similar results to CQNPM after enough iterations. Fig. 4 s (b) and (d) report the cost and PSNR values versus wall time, where GKSM was the fastest algorithm in terms of wall time. In multi-coil CS MRI reconstruction with non-Cartesian sampling, computing $\mathbf{A}\mathbf{x}$ and $\mathbf{A}^{\mathrm{H}}\mathbf{x}$ typically dominate the computational cost. CQNPM (respectively, APG) requires solving a weighted proximal mapping (respectively, a proximal mapping) at each iteration, which involves multiple evaluations of $\mathbf{A}\mathbf{x}$ and $\mathbf{A}^{\mathrm{H}}\mathbf{x}$. In contrast, GKSM requires only a single evaluation of $\mathbf{A}\mathbf{x}$ and $\mathbf{A}^{\mathrm{H}}\mathbf{x}$ per iteration, which significantly reduces computational cost while maintaining relatively fast convergence in terms of the number of iterations.

Fig. 5 presents the reconstructed images and the corresponding error maps at the 50th and 100th iterations of each method. GKSM eventually achieved a reconstruction quality similar to that of CQNPM with same number of iterations, but with significantly less wall time. Table I summarizes the PSNR performance on the rest of five brain images within 150 iterations. Clearly, we observed that GKSM was approximately $7\times$ faster than CQNPM in terms of wall time required to exceed the performance of APG within 150 iterations. Moreover, GKSM achieves nearly the same performance as CQNPM at the 150th iteration, while requiring approximately $9\times$ less time, illustrating the superior performance of our method. The supplementary material includes additional results on the knee images with spiral acquisition, as well as the structural similarity index measure (SSIM) metrics, which exhibit similar behavior.

B. Radial Acquisition Reconstruction

Fig. 6 presents the cost and PSNR values versus the number of iterations and wall time of each method on the knee 1 image with radial acquisition. Fig. 6 s (a) and (c) show CQNPM converged faster than APG in terms of the number of iterations. In this experimental setting, we found that CQNPM made faster progress than GKSM in the early iterations. Then GKSM exceeded CQNPM in the later iterations. This observation is slightly different from Fig. 4. Although all methods have convergence guarantees under the same assumptions, we cannot guarantee that their iterates follow the same trajectory. One possible explanation is that using a subspace in GKSM may sometimes act as an additional constraint, guiding the iterates along a more favorable path toward a minimizer. The detailed study of this direction is beyond the scope of this paper and is left for future work. Fig. 6 s (b) and (d) display the cost and PSNR values of each method versus wall time. Evidently, GKSM converged faster than others in terms of wall time, which is consistent with the previous observation.

Fig. 7 reports the reconstructed images of each method at 50th and 100th iterations. From this experiment, we observed that GKSM demonstrated the best visual quality among all methods. Table II reports the PSNR and wall time performance of each method on the remaining five knee test images. We ran APG for 100 iterations and then compared how many iterations were required by the other methods to exceed the PSNR value achieved by APG. Table II also reports the PSNR values and wall time of CQNPM and GKSM at 100 iterations. Consistently, we observed similar beharior as in Table I. The supplementary

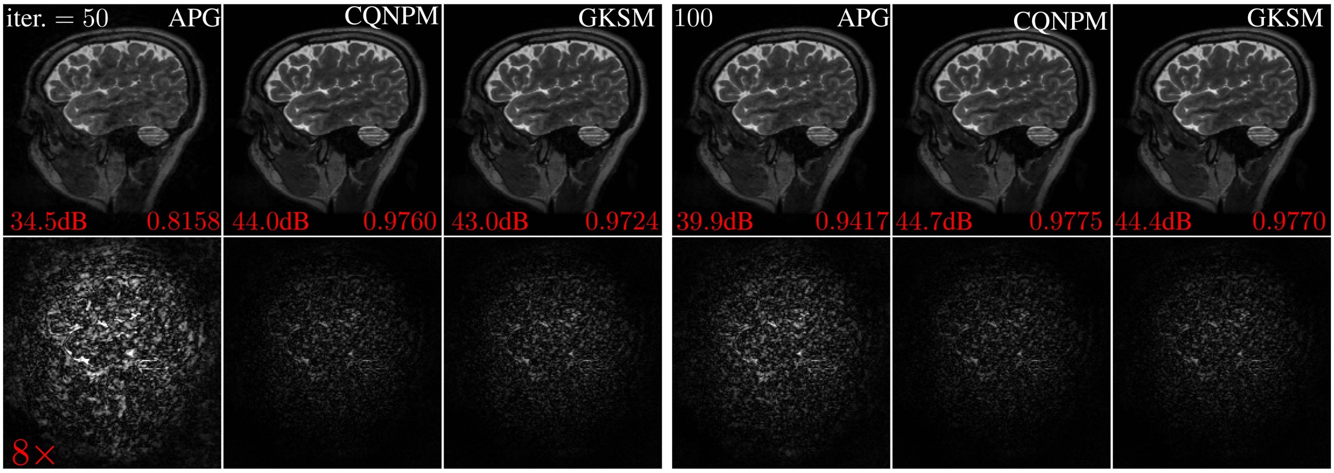


Fig. 5. First row: the reconstructed brain 1 images of each method at 50th and 100th iterations with spiral acquisition. The PSNR (respectively, SSIM) values are labeled at the left (respectively, right) bottom corner of each image. Second row: the associated error maps (8×) of the reconstructed images.

TABLE I

PSNR PERFORMANCE OF EACH METHOD FOR RECONSTRUCTING FIVE ADDITIONAL BRAIN TEST IMAGES WITH SPIRAL ACQUISITION. FOR APG, WE REPORT THE MAXIMUM PSNR (WITHIN 150 ITERATIONS), THE CORRESPONDING NUMBER OF ITERATIONS, AND WALL TIME. FOR OTHER METHODS, WE REPORT THE EARLIEST ITERATION COUNT THAT EXCEEDS THE APG PSNR (ALONG WITH ITS PSNR AND WALL TIME) IN THE FIRST ROW. THE PSNR AND WALL TIME AT THE 150TH ITERATION WERE SUMMARIZED IN THE SECOND ROW. BOLD INDICATES THE SHORTEST WALL TIME AT WHICH THE PSNR OF APG WAS EXCEEDED. THE PSNR VALUES AND WALL TIME AT THE 150TH ITERATION OF CQNPM AND GKSM ARE MARKED WITH AN UNDERLINE. THE BLUE DIGITS DENOTE THE SHORTEST WALL TIME AT THE 150TH ITERATION.

Index	2			3			4			5			6		
Methods	PSNR↑	iter.↓	sec.↓	PSNR↑	iter.↓	sec.↓	PSNR↑	iter.↓	sec.↓	PSNR↑	iter.↓	sec.↓	PSNR↑	iter.↓	sec.↓
APG	42.1	150	86.3	42.8	150	86.4	43.3	150	87.1	42.0	150	86.4	40.5	150	88.7
CQNPM	42.1	29	16.9	42.8	30	19.7	43.3	31	19.1	42.1	30	17.6	40.5	29	17.6
	44.2	150	82.0	44.7	150	70.1	45.0	150	65.5	44.0	150	68.9	42.7	150	77.0
GKSM	42.2	49	**2.4**	42.9	50	**2.4**	43.3	50	**2.4**	42.1	51	**2.5**	40.5	49	**2.3**
	44.2	150	8.4	44.7	150	8.2	45.1	150	8.2	43.9	150	8.3	42.5	150	8.2

material includes additional results on the brain images with radial acquisition, as well as the SSIM metrics that show similar trends.

C. Effect of K and Convergence Validation

This part empirically studies the effect of K and the convergence behavior of GKSM using the brain 1 image and spiral acquisition settings as in Fig. 4. Fig. 8 reports the PSNR values versus the number of iterations and wall time for GKSM with varying K and CQNPM. We observed that different values of K converged to similar PSNR values as CQNPM. Fig. 8(b) presents the PSNR values versus wall time, where we observed that larger values of K led to faster convergence compared with smaller ones. This observation is consistent with our earlier results, as GKSM avoids solving a weighted proximal mapping for iterations $k \leq K$.

We have now empirically validated our theoretical analysis. Fig. 9 presents the cost values and the values of Δ_k/Δ_1 for GKSM with spiral acquisition on six brain test images. As expected, the cost values converged to a constant across all test images, and $\Delta_k/\Delta_1 \to 0$, consistent with our theoretical analysis.

VI. CONCLUSION

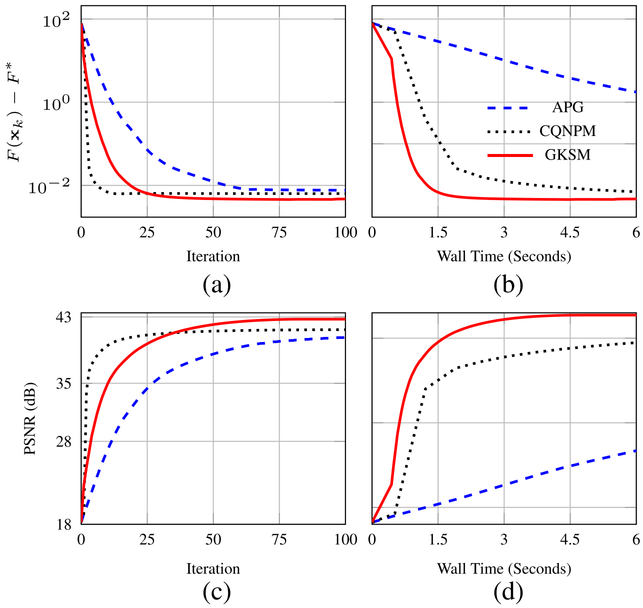
A well-established theoretical foundation is especially important for ensuring reliability in medical imaging applications. Compared with the PnP and RED frameworks, gradient-driven denoisers offer a significantly stronger theoretical foundation. In particular, the only required assumptions are the differentiability of f and the Lipschitz continuity of ∇f, which are easier to satisfy in practice. To efficiently solve the associate nonconvex minimization problem, we developed a generalized Krylov subspace method with convergence guarantees in nonconvex settings. Numerical experiments on multi-coil CS-MRI reconstruction with non-Cartesian sampling trajectories demonstrate that the proposed method can recover images within seconds on a GPU platform. This significantly improves the efficiency of solving the associated optimization problem and enhances the practical applicability of gradient-driven denoisers.

APPENDIX A
PROOF OF LEMMA 3

Since $\mathbf{B}_k \succ 0$ (cf. Lemma 2), we know the objective function in (6) is $\underline{\eta}$-strongly convex. By combining with the fact that

TABLE II
PSNR PERFORMANCE OF EACH METHOD FOR RECONSTRUCTING FIVE ADDITIONAL KNEE TEST IMAGES WITH RADIAL ACQUISITION. FOR APG, WE REPORT THE MAXIMUM PSNR (WITHIN 100 ITERATIONS), THE CORRESPONDING NUMBER OF ITERATIONS, AND WALL TIME. FOR OTHER METHODS, WE REPORT THE EARLIEST ITERATION THAT EXCEEDS THE APG PSNR (ALONG WITH ITS PSNR AND WALL TIME) IN THE FIRST ROW. THE PSNR AND WALL TIME AT THE 100TH ITERATION WERE SUMMARIZED IN THE SECOND ROW. BOLD INDICATES THE SHORTEST WALL TIME AT WHICH THE PSNR OF APG WAS EXCEEDED. THE PSNR VALUES AND WALL TIME AT THE 100TH ITERATION OF CQNPM AND GKSM ARE MARKED WITH AN UNDERLINE. THE BLUE DIGITS DENOTE THE SHORTEST WALL TIME AT THE 100TH ITERATION.

Index	2			3			4			5			6		
Methods	PSNR↑	iter.↓	sec.↓	PSNR↑	iter.↓	sec.↓	PSNR↑	iter.↓	sec.↓	PSNR↑	iter.↓	sec.↓	PSNR↑	iter.↓	sec.↓
APG	42.6	100	61.0	41.4	100	61.3	44.6	100	60.7	42.1	100	63.6	44.2	100	64.7
CQNPM	42.7	26	15.8	41.4	30	19.1	44.7	25	16.4	42.2	28	17.0	44.2	25	16.3
	44.1	100	59.9	41.9	100	64.3	45.5	100	65.4	44.1	100	61.9	44.9	100	66.3
GKSM	42.7	41	**2.0**	41.4	33	**1.6**	44.7	38	**1.8**	42.3	36	**1.8**	44.2	39	**1.9**
	44.1	100	5.6	43.3	100	5.5	45.5	100	5.5	44.4	100	5.7	44.9	100	5.5



(a) (b)

(c) (d)

Fig. 6. Comparison of different methods with radial acquisition on the knee 1 image for $\varepsilon = 6 \times 10^{-3}$. (a), (b): cost values versus iteration and wall time; (c), (d): PSNR values versus iteration and wall time.

$\mathbf{V}_k^{\mathrm{H}}\mathbf{V}_k = \mathbf{I}_k$ and the η-strongly convex inequality, we have the following inequality at kth iteration for $\forall\, \mathbf{x} = \mathbf{V}_k\boldsymbol{\beta},\ \mathbf{x} \in \mathcal{C}$:

$$\Re\Bigg\{\Bigg\langle \mathbf{V}_k^{\mathrm{H}}\Bigg[\nabla h(\mathbf{x}_{k+1}) + \nabla \bar{f}(\mathbf{x}_{k+1}, \mathbf{x}_k, \mathbf{B}_k, \alpha_k) + \frac{\eta}{2}(\mathbf{V}_k\boldsymbol{\beta} - \mathbf{x}_{k+1})\Bigg], \boldsymbol{\beta} - \boldsymbol{\beta}_k\Bigg\rangle\Bigg\} \geq 0. \tag{17}$$

Letting $\boldsymbol{\beta} = \begin{bmatrix}\boldsymbol{\beta}_{k-1}\\ 0\end{bmatrix}$, we rewrite (17) as

$$\Re\Bigg\{\Bigg\langle\Bigg[\nabla h(\mathbf{x}_{k+1}) + \frac{1}{\alpha_k}\mathbf{B}_k\,(\mathbf{x}_{k+1} - \mathbf{x}_k) + \nabla f(\mathbf{x}_k) + \frac{\eta}{2}\,(\mathbf{x}_k - \mathbf{x}_{k+1})\Bigg], \mathbf{x}_k - \mathbf{x}_{k+1}\Bigg\rangle\Bigg\} \geq 0. \tag{18}$$

Note that if $\|\tilde{\mathbf{r}}_k\| = 0$, we choose $\boldsymbol{\beta} = \boldsymbol{\beta}_{k-1}$ and (18) is still held. By reorganizing (18) and using the convexity of $h(\mathbf{x})$ ($h(\mathbf{x}_k) \geq h(\mathbf{x}_{k+1}) + \Re\{\langle \nabla h(\mathbf{x}_{k+1}), \mathbf{x}_k - \mathbf{x}_{k+1}\rangle\}$), we get the desired result

$$\Re\{\langle \nabla f(\mathbf{x}_k), \mathbf{x}_{k+1} - \mathbf{x}_k\rangle\} \leq h(\mathbf{x}_k) - h(\mathbf{x}_{k+1}) - \frac{1}{2}\|\mathbf{x}_k - \mathbf{x}_{k+1}\|^2_{(\frac{2}{\alpha_k}\mathbf{B}_k - \underline{\eta}\mathbf{I}_N)}. \tag{19}$$

APPENDIX B
PROOF OF LEMMA 4

Our goal is to derive an upper bound for ϕ_{k+1} by using the facts that $\phi_k - \phi_{k+1} \geq \phi_{k+1}^{2t}/\gamma$ and $0 < \phi_{k+1} \leq \phi_k$. Rewrite $\phi_k - \phi_{k+1} \geq \phi_{k+1}^{2t}/\gamma$ as $\phi_k \geq \phi_{k+1}(1 + \frac{1}{\gamma}\phi_{k+1}^{2t-1})$. Considering $t \in (0, 1/2)$, we know ϕ_{k+1}^{2t-1} is monotonically decreasing since $2t - 1 < 0$. So we have $\phi_{k+1}^{2t-1} \geq \phi_1^{2t-1}$, which implies $\phi_k \geq \phi_{k+1}(1 + \frac{1}{\gamma}\phi_1^{2t-1})$, yielding $\phi_{k+1} \leq (1 - \frac{\phi_1^{2t-1}}{\gamma + \phi_1^{2t-1}})^k\phi_1$. If $t = \frac{1}{2}$, we have $\phi_{k+1} \leq \gamma(\phi_k - \phi_{k+1})$, which yields $\phi_{k+1} \leq \frac{\gamma}{1+\gamma}\phi_k$. Therefore, we can establish the desired result immediately: $\phi_{k+1} \leq (1 - \frac{1}{1+\gamma})^k\phi_1$.

Denote by $\psi(x) = x^{1-2t}$, where $x > 0$. Let $\bar{t} = 2t - 1$. Using the mean value theorem, we have

$$\psi(\phi_{k+1}) - \psi(\phi_k) = -\bar{t}\,\bar{\phi}_k^{-\bar{t}-1}(\phi_{k+1} - \phi_k), \tag{20}$$

with $\phi_{k+1} \leq \bar{\phi}_k \leq \phi_k$. Since $\bar{\phi}_k^{-\bar{t}-1}$ is monotonically decreasing and $\phi_k - \phi_{k+1} \geq \phi_{k+1}^{2t}/\gamma$, we can get the following inequalities from (20) for $t \in (1/2, 1)$

$$\begin{aligned}\psi(\phi_{k+1}) - \psi(\phi_k) &\geq \bar{t}\,\phi_k^{-2t}(\phi_k - \phi_{k+1})\\ &\geq \phi_k^{-2t}\phi_{k+1}^{2t}\frac{\bar{t}}{\gamma}.\end{aligned} \tag{21}$$

Since $0 < \phi_{k+1} \leq \phi_k$, we have $\phi_{k+1}/\phi_k \leq 1$. Suppose we run k iterations. For any $\sigma \in (0, 1)$, we can split the whole iterate indices into two subsets $\mathcal{I}_1$ and $\mathcal{I}_2$ such that $\mathcal{I}_1 = \{k' \mid \phi_{k'+1}/\phi_{k'} \leq 1 - \sigma\}$ and $\mathcal{I}_2 = \{k' \mid \phi_{k'+1}/\phi_{k'} > 1 - \sigma\}$. So, we know that either $|\mathcal{I}_1| \geq k/2$ or $|\mathcal{I}_2| \geq k/2$.

If $|\mathcal{I}_1| \geq k/2$, we get

$$\phi_{k+1} \leq (1 - \sigma)^{k/2}\phi_1. \tag{22}$$

Next, we consider $|\mathcal{I}_2| \geq k/2$. By summing up (21) from 1 to k, we reach

$$\psi(\phi_{k+1}) \geq \psi(\phi_1) + \frac{(2t-1)(1-\sigma)^{2t}}{2\gamma}k.$$

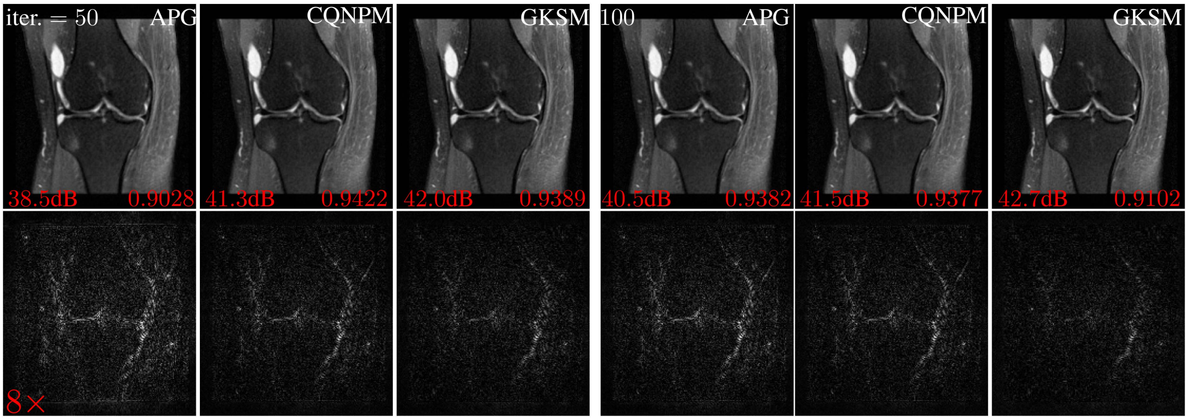
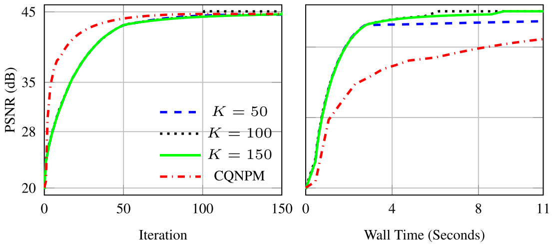
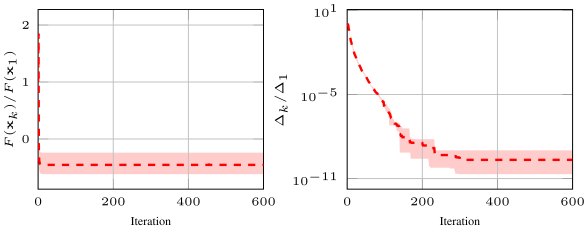


Fig. 7. First row: the reconstructed knee 1 images of each method at 50th and 100th iterations with radial acquisition. The PSNR (respectively, SSIM) values are labeled at the left (respectively, right) bottom corner of each image. Second row: the associated error maps (8×) of the reconstructed images.



(a) (b)

Fig. 8. Comparison of varying K with spiral acquisition on the brain 1 image. (a), (b): PSNR values versus iteration and wall time.



(a) (b)

Fig. 9. (a) Averaged cost values (a) and Δ_k/Δ_1 (b) versus iteration for GKSM. The shaded region of each curve represents the range of the cost values and Δ_k across six brain test images with spiral acquisition.

Using the definition of $\psi(\cdot)$ and the fact that $1-2t<0$, we derive

$$\phi_{k+1} \leq \left(\phi_1^{1-2t} + \frac{(2t-1)(1-\sigma)^{2t}}{2\gamma}k\right)_+^{\frac{1}{1-2t}}. \tag{23}$$

If k is large enough, the bound in (22) is smaller than that in (23), yielding the desired result.

APPENDIX C
PROOF OF THEOREM 1

By using Lemma 1, we have the following inequalities

$$\begin{aligned} f(\mathbf{x}_{k+1}) \leq\ & f(\mathbf{x}_k) + \frac{L}{2}\|\mathbf{x}_{k+1}-\mathbf{x}_k\|_2^2 \\ & + \Re\{\langle \nabla f(\mathbf{x}_k), \mathbf{x}_{k+1}-\mathbf{x}_k\rangle\} \\ \leq\ & f(\mathbf{x}_k) + h(\mathbf{x}_k) - h(\mathbf{x}_{k+1}) \\ & - \frac{1}{2}\|\mathbf{x}_k-\mathbf{x}_{k+1}\|^2_{(2\mathbf{B}_k/\alpha_k-(\underline{\eta}+L)\mathbf{I}_N)}. \end{aligned} \tag{24}$$

The second inequality comes from Lemma 3. Reorganizing (24), we get

$$\frac{1}{2}\|\mathbf{x}_k-\mathbf{x}_{k+1}\|^2_{(2\mathbf{B}_k/\alpha_k-(\underline{\eta}+L)\mathbf{I}_N)} \leq F(\mathbf{x}_k) - F(\mathbf{x}_{k+1}).$$

Letting $\alpha_k < \frac{2\underline{\eta}}{\underline{\eta}+L}$, $\upsilon = \min_k\{\underline{\eta}/\alpha_{k'} - (\underline{\eta}+L)/2\}$, and using Lemma 2, we reach

$$\upsilon\|\mathbf{x}_k-\mathbf{x}_{k+1}\|_2^2 \leq F(\mathbf{x}_k) - F(\mathbf{x}_{k+1}). \tag{25}$$

Since $\upsilon > 0$, we have $F(\mathbf{x}_{k+1}) \leq F(\mathbf{x}_k)$. Summing up (25) from $k' = 1$ to k, we get

$$\begin{aligned} \sum_{k'=1}^{k} \upsilon\|\mathbf{x}_{k'}-\mathbf{x}_{k'+1}\|_2^2 \leq\ & F(\mathbf{x}_1) - F(\mathbf{x}_{K+1}) \\ \leq\ & F(\mathbf{x}_1) - F^*, \end{aligned} \tag{26}$$

where F^* denotes the minimal value of $F(\mathbf{x})$. Letting $\Delta_k = \min_{k'\leq k}\{\|\mathbf{x}_{k'}-\mathbf{x}_{k'+1}\|_2^2\}$, we get the desired result

$$\Delta_k \leq \frac{F(\mathbf{x}_1) - F^*}{\upsilon k}. \tag{27}$$

Let $k\to\infty$, we get $\Delta_k \to 0$. Together with the summation in (26), we obtain $\|\mathbf{x}_{k+1}-\mathbf{x}_k\| \to 0$ as $k\to\infty$.

APPENDIX D
PROOF OF THEOREM 2

For $k > K$, we have $\mathbf{V}_k = \mathbf{I}_N$, so $\mathbf{V}_k^H\mathbf{V}_k = \mathbf{I}_N$ still holds. Therefore, (26) and (27) remain valid for $k > K$. Consequently, we still have $\|\mathbf{x}_{k+1}-\mathbf{x}_k\| \to 0$ as $k\to\infty$. Next, we prove that all cluster points of the sequence $\{\mathbf{x}_k\}_{k>K}$ are critical points of (2).

Let $G(\boldsymbol{\beta}) = \|\bar{\mathbf{A}}_k\boldsymbol{\beta} - \bar{\mathbf{y}}_k\|_2^2$ denote the cost function of (11) with $\bar{\mathbf{A}}_k = \begin{bmatrix}\mathbf{A}\\ \mathbf{B}_k^{\frac{1}{2}}\end{bmatrix}$ and $\bar{\mathbf{y}}_k = \begin{bmatrix}\mathbf{y}\\ \mathbf{B}_k^{\frac{1}{2}}\mathbf{w}_k\end{bmatrix}$. Then we rewrite (11) as an unconstrained problem, i.e.,

$$\boldsymbol{\beta}_k = \arg\min_{\boldsymbol{\beta}} G(\boldsymbol{\beta}) + \iota_{\mathcal{C}}(\boldsymbol{\beta}). \tag{28}$$

From the first-order optimality condition of (28) and using the fact that $\mathbf{x}_{k+1} = \boldsymbol{\beta}_k$, we have

$$\mathbf{0} \in \nabla h(\mathbf{x}_{k+1}) + \partial \iota_{\mathcal{C}}(\mathbf{x}_{k+1}) + \nabla f(\mathbf{x}_k) + \frac{1}{\alpha_k}\mathbf{B}_k(\mathbf{x}_{k+1} - \mathbf{x}_k)$$

which implies

$$\begin{aligned} &\nabla f(\mathbf{x}_{k+1}) - \nabla f(\mathbf{x}_k) + \frac{1}{\alpha_k}\mathbf{B}_k(\mathbf{x}_k - \mathbf{x}_{k+1}) \\ &\in \nabla h(\mathbf{x}_{k+1}) + \nabla f(\mathbf{x}_{k+1}) + \partial \iota_{\mathcal{C}}(\mathbf{x}_{k+1}). \end{aligned} \tag{29}$$

Here, we use the definitions of $\bar{f}(\mathbf{x}, \mathbf{x}_k, \mathbf{B}_k, \alpha_k)$ and $h(\mathbf{x})$.

Note that $F_{\mathcal{C}}(\mathbf{x}) = F(\mathbf{x}) + \iota_{\mathcal{C}}(\mathbf{x})$ with $F(\mathbf{x}) = h(\mathbf{x}) + f(\mathbf{x})$. By using (29), we have

$$\begin{aligned} \operatorname{dist}(\mathbf{0}, \partial F_{\mathcal{C}}(\mathbf{x}_{k+1})) &\le \Big\| \nabla f(\mathbf{x}_{k+1}) - \nabla f(\mathbf{x}_k) \\ &\quad + \frac{1}{\alpha_k}\mathbf{B}_k(\mathbf{x}_k - \mathbf{x}_{k+1}) \Big\|, \\ &\le \big\| \nabla f(\mathbf{x}_{k+1}) - \nabla f(\mathbf{x}_k) \big\| \\ &\quad + \frac{\overline{\eta}}{\alpha_k}\|\mathbf{x}_k - \mathbf{x}_{k+1}\| \\ &\le \frac{L\alpha_k + \overline{\eta}}{\alpha_k}\|\mathbf{x}_k - \mathbf{x}_{k+1}\|. \end{aligned} \tag{30}$$

Notice that $\|\mathbf{x}_k - \mathbf{x}_{k+1}\| \to 0$ for $k \to \infty$ and that $\frac{L\alpha_k + \overline{\eta}}{\alpha_k}$ remains finite. So we have $\operatorname{dist}(\mathbf{0}, \partial F_{\mathcal{C}}(\mathbf{x}_{k+1})) \to 0$ for $k \to \infty$, which implies that all cluster points of $\{\mathbf{x}_k\}_{k>K}$ are critical points of (2). This completes the proof of the first term.

Since $F_{\mathcal{C}}$ satisfies the KL inequality and $\mathbf{x}_k$ is converging to $\mathbf{x}^*$, there exist $K' > K$ and $\Lambda > 0$ such that, for all $k \ge K'$, we have $\mathbf{x}_k \in \mathcal{B}(\mathbf{x}^*, \Lambda)$, where $\mathcal{B}(\mathbf{x}^*, \Lambda) = \{\mathbf{x} \in \mathbb{C}^N \mid \|\mathbf{x} - \mathbf{x}^*\| \le \Lambda\}$, and $|F_C(\mathbf{x}_k) - F^*| < \eta$. By letting $\bar{\mathbf{x}} = \mathbf{x}^*$, $\varphi(s) = cs^{1-t}$, and the assumption $\mathcal{B}(\mathbf{x}^*, \Lambda) \subseteq \mathcal{U}$, we get

$$\begin{aligned} (F_{\mathcal{C}}(\mathbf{x}_{k+1}) - F^*)^{2t} &\le c^2(1-t)^2 \operatorname{dist}(\mathbf{0}, \partial F_{\mathcal{C}}(\mathbf{x}_{k+1}))^2 \\ &\overset{(30)}{\le} \frac{[c(L\alpha_k + \overline{\eta})]^2}{(1-t)^2\alpha_k^2}\|\mathbf{x}_k - \mathbf{x}_{k+1}\|_2^2 \\ &\overset{(25)}{\le} \frac{[c(L\alpha_k + \overline{\eta})]^2}{\upsilon(1-t)^2\alpha_k^2}(F(\mathbf{x}_k) - F(\mathbf{x}_{k+1})). \end{aligned} \tag{31}$$

Note that during the algorithm's progress, all iterates remain in $\mathcal{C}$, so that $F_{\mathcal{C}}(\mathbf{x}) = F(\mathbf{x})$. For simplicity, we write $F(\mathbf{x})$ instead of $F_{\mathcal{C}}(\mathbf{x})$ in what follows. Denote by $\gamma = \max_k([c(L\alpha_k + \overline{\eta})]^2)/(\upsilon(1-t)^2\alpha_k^2))$. For $t = 0$, we have

$$F(\mathbf{x}_{k+1}) - F^* \le F(\mathbf{x}_k) - F^* - \frac{1}{\gamma},$$

resulting in

$$F(\mathbf{x}_{k+1}) - F^* \le \left(F(\mathbf{x}_{K'}) - F^* - \frac{1}{\gamma}(k - K' + 1) \right)_+.$$

By using Lemma 4, we get the desired results for $t \in (0, 1)$.

REFERENCES

[1] R. W. Brown, Y.-C. N. Cheng, E. M. Haacke, M. R. Thompson, and R. Venkatesan, *Magnetic Resonance Imaging: Physical Principles and Sequence Design*. Hoboken, NJ, USA: Wiley, 2014.

[2] M. Lustig, D. Donoho, and J. M. Pauly, "Sparse MRI: The application of compressed sensing for rapid MR imaging," *Magn. Reson. Med.*, vol. 58, no. 6, pp. 1182–1195, 2007.

[3] M. Lustig, D. L. Donoho, J. M. Santos, and J. M. Pauly, "Compressed sensing MRI," *IEEE Signal Process. Mag.*, vol. 25, no. 2, pp. 72–82, Mar. 2008.

[4] J. A. Fessler, "Model-based image reconstruction for MRI," *IEEE Signal Process. Mag.*, vol. 27, no. 4, pp. 81–9, Jul. 2010.

[5] K. P. Pruessmann, M. Weiger, M. B. Scheidegger, and P. Boesiger, "SENSE: Sensitivity encoding for fast MRI," *Magn. Reson. Med.*, vol. 42, no. 5, pp. 952–962, 1999.

[6] M. A. Griswold et al., "Generalized autocalibrating partially parallel acquisitions (GRAPPA)," *Magn. Reson. Med.*, vol. 47, no. 6, pp. 1202–1210, 2002.

[7] K. T. Block, M. Uecker, and J. Frahm, "Undersampled radial MRI with multiple coils. iterative image reconstruction using a total variation constraint," *Magn. Reson. Med.*, vol. 57, no. 6, pp. 1086–1098, 2007.

[8] M. Guerquin-Kern, M. Haberlin, K. P. Pruessmann, and M. Unser, "A fast wavelet-based reconstruction method for magnetic resonance imaging," *IEEE Trans. Med. Imag.*, vol. 30, no. 9, pp. 1649–1660, Sep. 2011.

[9] L. I. Rudin, S. Osher, and E. Fatemi, "Nonlinear total variation based noise removal algorithms," *Physica D: Nonlinear Phenomena*, vol. 60, no. 1-4, pp. 259–268, 1992.

[10] T. Hong, L. Hernandez-Garcia, and J. A. Fessler, "A complex Quasi-Newton proximal method for image reconstruction in compressed sensing MRI," *IEEE Trans. Comput. Imag.*, vol. 10, pp. 372–384, 2024.

[11] M. Aharon, M. Elad, and A. Bruckstein, "K-SVD: An algorithm for designing overcomplete dictionaries for sparse representation," *IEEE Trans. Signal Process.*, vol. 54, no. 11, pp. 4311–4322, Nov. 2006.

[12] S. Ravishankar and Y. Bresler, "MR image reconstruction from highly undersampled k-space data by dictionary learning," *IEEE Trans. Med. Imag.*, vol. 30, no. 5, pp. 1028–1041, May 2011.

[13] W. Dong, G. Shi, X. Li, Y. Ma, and F. Huang, "Compressive sensing via nonlocal low-rank regularization," *IEEE Trans. Image Process.*, vol. 23, no. 8, pp. 3618–3632, Aug. 2014.

[14] J. A. Fessler, "Optimization methods for magnetic resonance image reconstruction: Key models and optimization algorithms," *IEEE Signal Process. Mag.*, vol. 37, no. 1, pp. 33–40, Jan. 2020.

[15] S. Ravishankar, J. C. Ye, and J. A. Fessler, "Image reconstruction: From sparsity to data-adaptive methods and machine learning," *Proc. IEEE*, vol. 108, no. 1, pp. 86–109, Jan. 2020.

[16] R. Heckel, M. Jacob, A. Chaudhari, O. Perlman, and E. Shimron, "Deep learning for accelerated and robust MRI reconstruction," *Magn. Reson. Mater. Phys., Biol. Med.*, vol. 37, no. 3, pp. 335–368, 2024.

[17] S. Wang et al., "Accelerating magnetic resonance imaging via deep learning," in *Proc. IEEE 13th Int. Symp. Biomed. Imag.*, 2016, pp. 514–517.

[18] H. K. Aggarwal, M. P. Mani, and M. Jacob, "MoDL: Model-based deep learning architecture for inverse problems," *IEEE Trans. Med. Imag.*, vol. 38, no. 2, pp. 394–405, Feb. 2019.

[19] D. Gilton, G. Ongie, and R. Willett, "Deep equilibrium architectures for inverse problems in imaging," *IEEE Trans. Comput. Imag.*, vol. 7, pp. 1123–1133, 2021.

[20] Z. Ramzi, G. Chaithya, J.-L. Starck, and P. Ciuciu, "NC-PDNet: A density-compensated unrolled network for 2D and 3D non-Cartesian MRI reconstruction," *IEEE Trans. Med. Imag.*, vol. 41, no. 7, pp. 1625–1638, Jul. 2022.

[21] H. Chung and J. C. Ye, "Score-based diffusion models for accelerated MRI," *Med. Image Anal.*, vol. 80, 2022, Art. no. 102479.

[22] S. V. Venkatakrishnan, C. A. Bouman, and B. Wohlberg, "Plug-and-play priors for model based reconstruction," in *Proc. IEEE Glob. Conf. Signal Inf. Process.*, 2013, pp. 945–948.

[23] Y. Romano, M. Elad, and P. Milanfar, "The little engine that could: Regularization by denoising (RED)," *SIAM J. Imag. Sci.*, vol. 10, no. 4, pp. 1804–1844, 2017.

[24] S. Sreehari et al., "Plug-and-play priors for bright field electron tomography and sparse interpolation," *IEEE Trans. Comput. Imag.*, vol. 2, no. 4, pp. 408–423, Dec. 2016.

[25] T. Meinhardt, M. Moeller, C. Hazirbas, and D. Cremers, "Learning proximal operators: Using denoising networks for regularizing inverse imaging problems," in *Proc. IEEE Int. Conf. Comput. Vis.*, Venice, Italy, Oct. 2017, pp. 1799–1808.

[26] G. T. Buzzard, S. H. Chan, S. Sreehari, and C. A. Bouman, "Plug-and-play unplugged: Optimization free reconstruction using consensus equilibrium," *SIAM J. Imag. Sci.*, vol. 11, no. 3, pp. 2001–2020, 2018.

[27] T. Hong, Y. Romano, and M. Elad, "Acceleration of RED via vector extrapolation," *J. Vis. Commun. Image Representation*, vol. 63, 2019, Art. no. 102575.

[28] K. Zhang, Y. Li, W. Zuo, L. Zhang, L. Van Gool, and R. Timofte, "Plug-and-play image restoration with deep denoiser prior," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 44, no. 10, pp. 6360–6376, Oct. 2022.

[29] T. Hong, X. Xu, J. Hu, and J. A. Fessler, "Provable preconditioned plug-and-play approach for compressed sensing MRI reconstruction," *IEEE Trans. Comput. Imag.*, vol. 10, pp. 372–384, 2024.

[30] T. Huang, G. Yang, and G. Tang, "A fast two-dimensional median filtering algorithm," *IEEE Trans. Acoust., Speech, Signal Process.*, vol. 27, no. 1, pp. 13–18, Feb. 1979.

[31] A. Buades, B. Coll, and J.-M. Morel, "A non-local algorithm for image denoising," in *Proc. Comput. Vis. Pattern Recognit., IEEE Comput. Soc. Conf.*, 2005, vol. 2, pp. 60–65.

[32] K. Dabov, A. Foi, V. Katkovnik, and K. Egiazarian, "Image denoising by sparse 3-D transform-domain collaborative filtering," *IEEE Trans. Image Process.*, vol. 16, no. 8, pp. 2080–2095, Aug. 2007.

[33] R. Ahmad et al., "Plug-and-play methods for magnetic resonance imaging: Using denoisers for image recovery," *IEEE Signal Process. Mag.*, vol. 37, no. 1, pp. 105–116, Jan. 2020.

[34] S. H. Chan, X. Wang, and O. A. Elgendy, "Plug-and-play ADMM for image restoration: Fixed-point convergence and applications," *IEEE Trans. Comput. Imag.*, vol. 3, no. 1, pp. 84–98, Mar. 2017.

[35] E. T. Reehorst and P. Schniter, "Regularization by denoising: Clarifications and new interpretations," *IEEE Trans. Comput. Imag.*, vol. 5, no. 1, pp. 52–67, Mar. 2019.

[36] E. Ryu, J. Liu, S. Wang, X. Chen, Z. Wang, and W. Yin, "Plug-and-play methods provably converge with properly trained denoisers," in *Proc. Int. Conf. Mach. Learn.*, 2019, pp. 5546–5557.

[37] M. Terris, A. Repetti, J.-C. Pesquet, and Y. Wiaux, "Building firmly nonexpansive convolutional neural networks," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process.*, 2020, pp. 8658–8662.

[38] U. S. Kamilov, C. A. Bouman, G. T. Buzzard, and B. Wohlberg, "Plug-and-play methods for integrating physical and learned models in computational imaging: Theory, algorithms, and applications," *IEEE Signal Process. Mag.*, vol. 40, no. 1, pp. 85–97, Jan. 2023.

[39] R. Cohen, Y. Blau, D. Freedman, and E. Rivlin, "It has potential: Gradient-driven denoisers for convergent solutions to inverse problems," in *Proc. Adv. Neural Inf. Process. Syst.*, 2021, vol. 34, pp. 18152–18164.

[40] S. Hurault, A. Leclaire, and N. Papadakis, "Gradient step denoiser for convergent plug-and-play," in *Proc. Int. Conf. Learn. Representations*, 2022.

[41] S. Chaudhari, S. Pranav, and J. M. Moura, "Gradient networks," *IEEE Trans. Signal Process.*, vol. 73, pp. 324–339, 2025.

[42] T. Hong, Z. Xu, S. Y. Chun, L. Hernandez-Garcia, and J. A. Fessler, "Convergent complex quasi-Newton proximal methods for gradient-driven denoisers in compressed sensing MRI reconstruction," *IEEE Trans. Comput. Imag.*, vol. 11, pp. 1534–1547, 2025.

[43] Y. Saad, *Iterative Methods for Sparse Linear Systems*, Philadelphia, PA, USA: SIAM, 2003.

[44] J. Zbontar et al., "fastMRI: An open dataset and benchmarks for accelerated MRI," 2018, *arXiv:1811.08839*.

[45] M. R. Hestenes et al., "Methods of conjugate gradients for solving linear systems," *J. Res. Nat. Bur. Standards*, vol. 49, no. 6, pp. 409–436, 1952.

[46] C. C. Paige and M. A. Saunders, "LSQR: An algorithm for sparse linear equations and sparse least squares," *ACM Trans. Math. Softw.*, vol. 8, no. 1, pp. 43–71, 1982.

[47] H. A. Van der Vorst, "Bi-CGSTAB: A fast and smoothly converging variant of Bi-CG for the solution of nonsymmetric linear systems," *SIAM J. Sci. Stat. Comput.*, vol. 13, no. 2, pp. 631–644, 1992.

[48] Y. Saad and M. H. Schultz, "GMRES: A generalized minimal residual algorithm for solving nonsymmetric linear systems," *SIAM J. Sci. Stat. Comput.*, vol. 7, no. 3, pp. 856–869, 1986.

[49] Y. Saad, "A flexible inner-outer preconditioned GMRES algorithm," *SIAM J. Sci. Comput.*, vol. 14, no. 2, pp. 461–469, 1993.

[50] J. Chung and S. Gazzola, "Computational methods for large-scale inverse problems: A survey on hybrid projection methods," *SIAM Rev.*, vol. 66, no. 2, pp. 205–284, 2024.

[51] T. Hong, Z. Xu, J. Hu, and J. A. Fessler, "Using randomized nyström preconditioners to accelerate variational image reconstruction," *IEEE Trans. Comput. Imag.*, vol. 11, pp. 1630–1643, 2025.

[52] J. Lampe, L. Reichel, and H. Voss, "Large-scale tikhonov regularization via reduction by orthogonal projection," *Linear Algebra Appl.*, vol. 436, no. 8, pp. 2845–2865, 2012.

[53] A. Lanza, S. Morigi, L. Reichel, and F. Sgallari, "A generalized Krylov subspace method for ℓ_p-ℓ_q minimization," *SIAM J. Sci. Comput.*, vol. 37, no. 5, pp. S30–S50, 2015.

[54] G. Huang, A. Lanza, S. Morigi, L. Reichel, and F. Sgallari, "Majorization–minimization generalized Krylov subspace methods for ℓ_p-ℓ_q optimization applied to image restoration," *BIT Numer. Math.*, vol. 57, no. 2, pp. 351–378, 2017.

[55] S. Gazzola and J. G. Nagy, "Generalized Arnoldi–Tikhonov method for sparse reconstruction," *SIAM J. Sci. Comput.*, vol. 36, no. 2, pp. B225–B247, 2014.

[56] J. Chung and S. Gazzola, "Flexible Krylov methods for ℓ_p regularization," *SIAM J. Sci. Comput.*, vol. 41, no. 5, pp. S149–S171, 2019.

[57] S. Gazzola, J. G. Nagy, and M. S. Landman, "Iteratively reweighted FGMRES and FLSQR for sparse reconstruction," *SIAM J. Sci. Comput.*, vol. 43, no. 5, pp. S47–S69, 2021.

[58] S. Gazzola and M. Sabaté Landman, "Krylov methods for inverse problems: Surveying classical, and introducing new, algorithmic approaches," *GAMM-Mitteilungen*, vol. 43, no. 4, 2020, Art. no. e202000017.

[59] A. Buccini, M. Pasha, and L. Reichel, "Modulus-based iterative methods for constrained ℓ_p-ℓ_q minimization," *Inverse Problems*, vol. 36, no. 8, 2020, Art. no. 084001.

[60] H. D. Sterck, "A nonlinear GMRES optimization algorithm for canonical tensor decomposition," *SIAM J. Sci. Comput.*, vol. 34, no. 3, pp. A1351–A1379, 2012.

[61] H. D. Sterck and Y. He, "On the asymptotic linear convergence speed of anderson acceleration, Nesterov acceleration, and nonlinear GMRES," *SIAM J. Sci. Comput.*, vol. 43, no. 5, pp. S21–S46, 2021.

[62] J. A. Fessler and R. R. Nadakuditi, *Linear Algebra for Data Science, Machine Learning, and Signal Processing*. Cambridge, U.K.: Cambridge Univ. Press, 2024.

[63] A. Beck and M. Teboulle, "A fast iterative shrinkage-thresholding algorithm for linear inverse problems," *SIAM J. Imag. Sci.*, vol. 2, no. 1, pp. 183–202, 2009.

[64] H. Attouch, J. Bolte, P. Redont, and A. Soubeyran, "Proximal alternating minimization and projection methods for nonconvex problems: An approach based on the Kurdyka-łojasiewicz inequality," *Math. Operations Res.*, vol. 35, no. 2, pp. 438–457, 2010.

[65] J. Bolte, S. Sabach, and M. Teboulle, "Proximal alternating linearized minimization for nonconvex and nonsmooth problems," *Math. Program.*, vol. 146, no. 1, pp. 459–494, 2014.

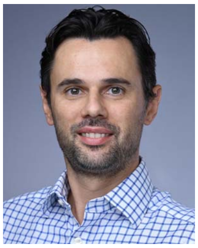
[66] M. Uecker et al., "ESPIRiT—an eigenvalue approach to autocalibrating parallel MRI: Where SENSE meets GRAPPA," *Magn. Reson. Med.*, vol. 71, no. 3, pp. 990–1001, 2014.

[67] D. Kingma and J. Ba, "Adam: A method for stochastic optimization," in *Proc. Int. Conf. Learn. Representations*, 2015.

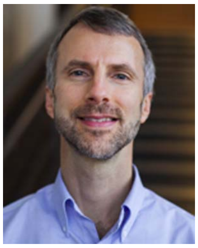
[68] T. Zhang, J. M. Pauly, S. S. Vasanawala, and M. Lustig, "Coil compression for accelerated imaging with Cartesian sampling," *Magn. Reson. Med.*, vol. 69, no. 2, pp. 571–582, 2013.

[69] H. Li and Z. Lin, "Accelerated proximal gradient methods for nonconvex programming," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 28, 2015.

Tao Hong (Member, IEEE) received the Ph.D. degree from the Henry and Marilyn Taub Faculty of Computer Science with the Technion Israel Institute of Technology, Haifa, Israel, in 2021. He is currently a Postdoctoral Fellow with the Oden Institute for Computational Engineering and Sciences, the University of Texas at Austin. Prior to this, he was a Postdoctoral Fellow with the University of Michigan, Ann Arbor, MI, USA. His main research interests include numerical optimization, computational imaging, and reliable AI. His particularly focused on designing efficient and provable computational methods to address the challenges arising in scientific computing, signal processing, machine learning, and computational imaging, as well as on building reliable AI models for imaging science.



Umberto Villa (Member, IEEE) received the B.S. and M.S. degrees in mathematical engineering from the Politecnico di Milano, Milan, Italy, in 2005 and 2007, respectively, and the Ph.D. degree in mathematics from Emory University, Atlanta, GA, USA, in 2012. He is currently an Assistant Professor with the Department of Biomedical Engineering and Core Faculty of the Oden Institute for Computational Engineering and Science, The University of Texas at Austin, Austin, TX, USA. His research interests lie in the computational and mathematical aspects of large-scale inverse problems, imaging science, and uncertainty quantification.



Jeffrey A. Fessler (Fellow, IEEE) received the B.S.E.E. degree from Purdue University, West Lafayette, IN, USA, in 1985, the M.S.E.E. and M.S. degrees from Stanford University, Stanford, CA, USA, in 1986 and 1989, respectively. He is currently the William L. Root Distinguished University Professor of EECS with the University of Michigan. From 1985 to 1988 he was a National Science Foundation Graduate Fellow at Stanford, where he earned a Ph.D. degree in electrical engineering in 1990. Since then, he has been with the University of Michigan since. From 1991 to 1992, he was with the Department of Energy Alexander Hollaender as a Postdoctoral Fellow, the Division of Nuclear Medicine. From 1993 to 1995, he was an Assistant Professor in nuclear medicine and the bioengineering Program. He is currently a Professor with the Departments of Electrical Engineering and Computer Science, Radiology, and Biomedical Engineering. His contributions to the theory and practice of image reconstruction. His research interests are in statistical aspects of imaging problems, and he has supervised doctoral research in PET, SPECT, X-ray CT, MRI, and optical imaging problems. He was the recipient of the Francois Erbsmann award for his IPMI93 presentation, Edward Hoffman Medical Imaging Scientist Award in 2013, IEEE EMBS Technical Achievement Award in 2016, and 2023 Steven S. Attwood Award, highest honor awarded to a Faculty Member by the College of Engineering. He was an Associate Editor for the IEEE TRANSACTIONS ON MEDICAL IMAGING, IEEE SIGNAL PROCESSING LETTERS, IEEE TRANSACTIONS ON IMAGE PROCESSING, IEEE TRANSACTIONS ON COMPUTATIONAL IMAGING (T-CI), *SIAM J. on Imaging Science*, and a Senior AE for T-CI. He has chaired the IEEE T-MI Steering Committee and the ISBI Steering Committee. He was also a co-chair of the 1997 SPIE conference on Image Reconstruction and Restoration, technical program co-chair of the 2002 IEEE International Symposium on Biomedical Imaging (ISBI), and general chair of ISBI 2007.

Supplementary Material:
A Convergent Generalized Krylov Subspace Method for Compressed Sensing MRI Reconstruction with Gradient-Driven Denoisers

Tao Hong *Member, IEEE*, Umberto Villa *Member, IEEE*, and Jeffrey A. Fessler *Fellow, IEEE*

S.I. SPIRAL ACQUISITION RECONSTRUCTION

This section provides additional results on the knee 1 image with spiral acquisition, as well as the SSIM metrics on the brain 1 image with spiral acquisition. Fig. S.1 presents the cost function and PSNR values versus the number of iterations and wall time. Fig. S.2 illustrates the reconstructed images and associated error maps for each method at the 100th and 200th iterations. The observed performance trends are consistent with those reported in the main paper. Table S.2 summarizes the additional results on other five knee images. GKSM converged faster than other competing methods in terms of wall time, consistent with results in the main paper. Tables S.1 and S.3 show the associated SSIM values on the brain and knee 1 images with spiral acquisition. In this setting, we saw that GKSM achieved the highest SSIM values.

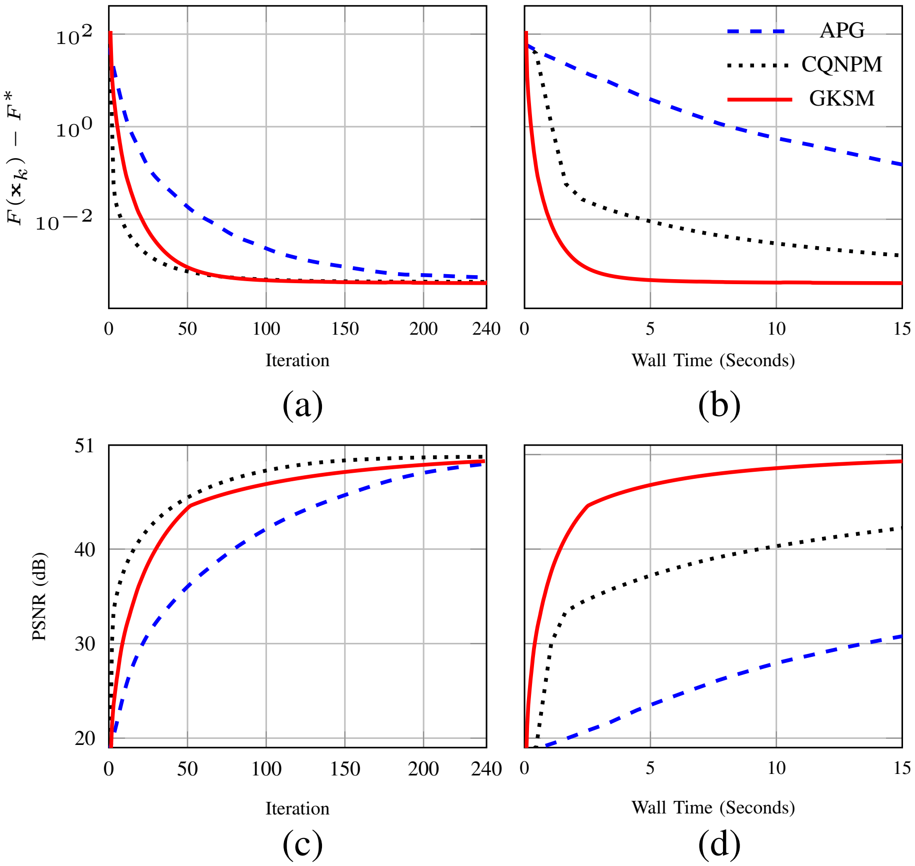


Fig. S.1. Comparison of different methods with spiral acquisition on the knee 1 image with $\varepsilon = 3 \times 10^{-5}$. (a), (b): cost values versus iteration and wall time; (c), (d): PSNR values versus iteration and wall time.

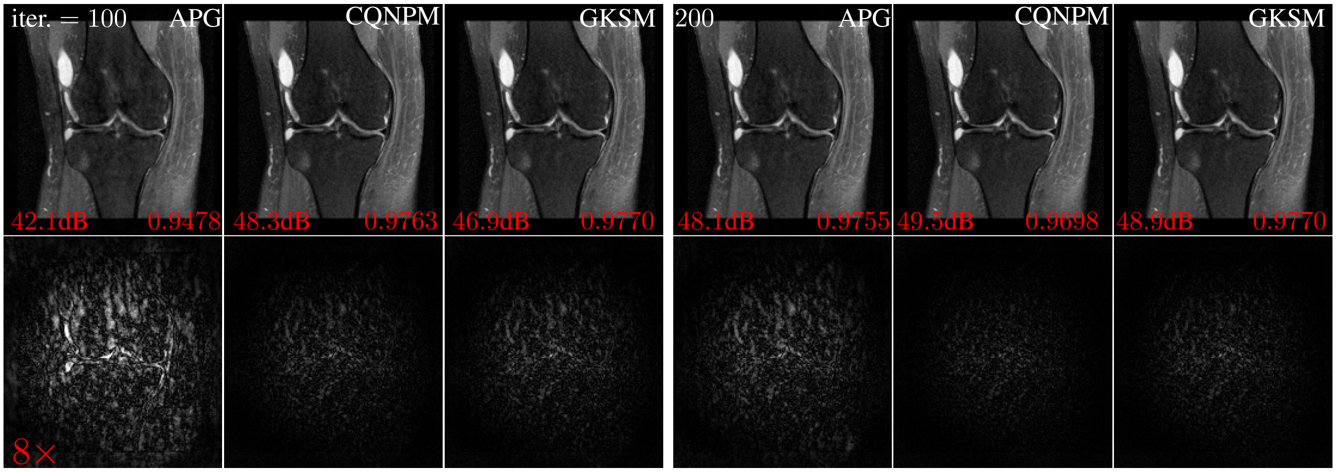


Fig. S.2. First row: the reconstructed knee 1 images of each method at 100th and 200th iterations with spiral acquisition. The PSNR (respectively, SSIM) values are labeled at the left (respectively, right) bottom corner of each image. Second row: the associated error maps (8×) of the reconstructed images.

TABLE S.1

THE SSIM METRICS OF EACH METHOD FOR RECONSTRUCTING FIVE ADDITIONAL BRAIN TEST IMAGES WITH SPIRAL ACQUISITION CORRESPOND TO TABLE I IN THE MAIN PAPER. WE ALSO PROVIDE THE ITERATION COUNTS SO THAT THE SSIM AND PSNR VALUES CAN BE ASSOCIATED WITH THE SAME RECONSTRUCTION ITERATION. THE HIGHEST SSIM VALUES ARE SHOWN IN BOLD.

Index	2		3		4		5		6	
Methods	SSIM↑	iter.↓	SSIM↑	iter.↓	SSIM↑	iter.↓	SSIM↑	iter.↓	SSIM↑	iter.↓
APG	0.9584	150	0.9612	150	0.9677	150	0.9638	150	0.9592	150
CQNPM	0.9628	29	0.9652	30	0.9716	31	0.9689	30	0.9641	29
	0.9704	150	0.9720	150	0.9757	150	0.9744	150	0.9711	150
GKSM	0.9628	49	0.9654	50	0.9713	50	0.9688	51	0.9641	49
	0.9706	150	**0.9722**	150	**0.9760**	150	**0.9746**	150	**0.9713**	150

TABLE S.2

PSNR PERFORMANCE OF EACH METHOD FOR RECONSTRUCTING FIVE ADDITIONAL KNEE TEST IMAGES WITH SPIRAL ACQUISITION. FOR APG, WE REPORT THE MAXIMUM PSNR (WITHIN 200 ITERATIONS), THE CORRESPONDING NUMBER OF ITERATIONS, AND THE WALL TIME. FOR OTHER METHODS, WE REPORT THE EARLIEST ITERATION THAT EXCEEDS THE APG PSNR (ALONG WITH ITS PSNR AND WALL TIME) IN THE FIRST ROW. THE PSNR AND WALL TIME AT THE 200TH ITERATION WERE SUMMARIZED IN THE SECOND ROW. BOLD INDICATES THE SHORTEST WALL TIME AT WHICH THE PSNR OF APG WAS EXCEEDED. THE PSNR VALUES AND WALL TIME AT THE 200TH ITERATION OF CQNPM AND GKSM ARE MARKED WITH AN UNDERLINE. THE BLUE DIGITS DENOTE THE SHORTEST WALL TIME AT THE 200TH ITERATION.

Index	2			3			4			5			6		
Methods	PSNR↑	iter.↓	sec.↓	PSNR↑	iter.↓	sec.↓	PSNR↑	iter.↓	sec.↓	PSNR↑	iter.↓	sec.↓	PSNR↑	iter.↓	sec.↓
APG	45.4	200	113.3	46.2	200	112.5	51.1	200	107.7	47.9	200	127.1	43.1	200	117.0
CQNPM	45.4	96	59.7	46.2	95	58.2	51.1	96	57.6	47.9	128	78.8	42.2	200	125.3
	47.8	200	122.4	47.9	200	123.6	51.2	200	119.4	48.6	200	123.0	42.2	200	125.3
GKSM	45.4	135	**7.1**	46.2	134	**7.2**	51.1	149	**8.4**	47.9	126	**6.6**	43.2	47	**2.2**
	46.6	200	11.7	47.1	200	11.8	51.4	200	12.1	49.2	200	11.7	48.1	200	11.7

TABLE S.3

THE SSIM METRICS OF EACH METHOD FOR RECONSTRUCTING FIVE ADDITIONAL KNEE TEST IMAGES WITH SPIRAL ACQUISITION CORRESPOND TO TABLE S.2. WE ALSO PROVIDE THE ITERATION COUNTS SO THAT THE SSIM AND PSNR VALUES CAN BE ASSOCIATED WITH THE SAME RECONSTRUCTION ITERATION. THE HIGHEST SSIM VALUES ARE SHOWN IN BOLD.

Index	2		3		4		5		6	
Methods	SSIM↑	iter.↓	SSIM↑	iter.↓	SSIM↑	iter.↓	SSIM↑	iter.↓	SSIM↑	iter.↓
APG	0.9674	200	0.9674	200	0.9787	200	0.9786	200	0.9592	200
CQNPM	0.9699	96	0.9683	95	0.9783	96	0.9769	128	0.9600	200
	0.9651	200	0.9593	200	0.9660	200	0.9724	200	0.9600	200
GKSM	**0.9701**	135	**0.9687**	134	**0.9790**	149	**0.9797**	126	0.9635	47
	0.9700	200	0.9669	200	0.9767	200	0.9774	200	**0.9694**	200

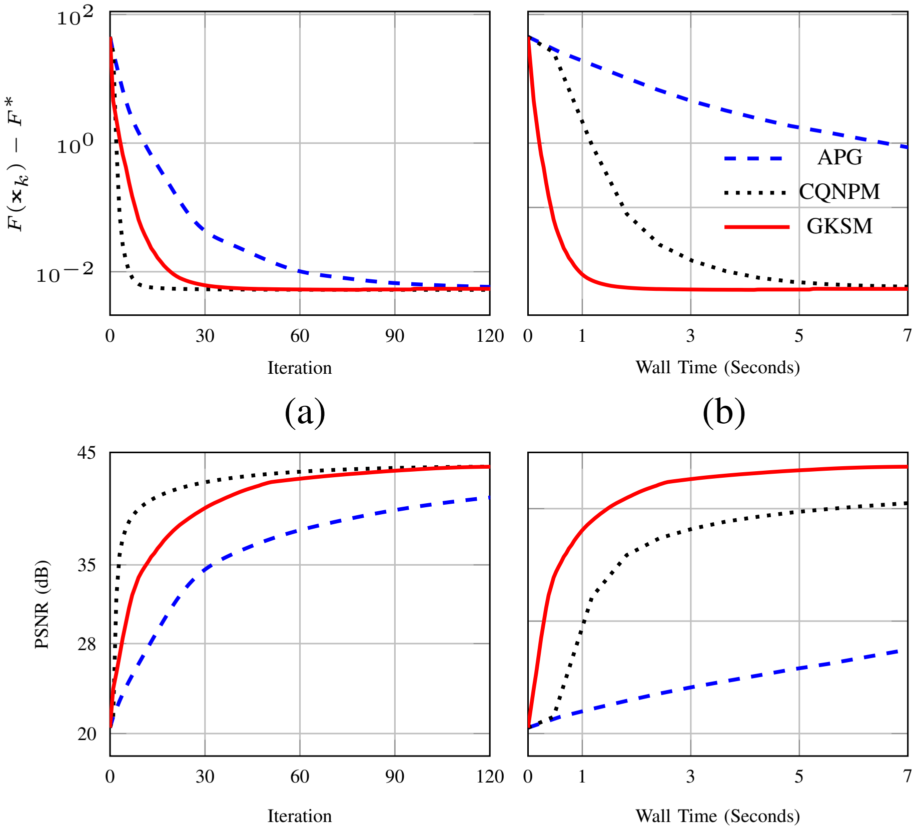
TABLE S.4

THE SSIM METRICS OF EACH METHOD FOR RECONSTRUCTING FIVE ADDITIONAL KNEE TEST IMAGES WITH RADIAL ACQUISITION CORRESPOND TO TABLE II IN THE MAIN PAPER. WE ALSO PROVIDE THE ITERATION COUNTS SO THAT THE SSIM AND PSNR VALUES CAN BE ASSOCIATED WITH THE SAME RECONSTRUCTION ITERATION. THE HIGHEST SSIM VALUES ARE SHOWN IN BOLD.

Index / Methods	2		3		4		5		6	
	SSIM↑	iter.↓	SSIM↑	iter.↓	SSIM↑	iter.↓	SSIM↑	iter.↓	SSIM↑	iter.↓
APG	0.9497	100	**0.9454**	100	0.9536	100	0.9559	100	0.9525	100
CQNPM	**0.9560**	26	0.9450	30	**0.9580**	25	**0.9581**	28	**0.9568**	25
	0.9185	100	0.9409	100	0.9153	100	0.9314	100	0.9155	100
GKSM	0.9556	41	0.9409	33	0.9578	38	0.9562	36	0.9564	39
	0.9376	100	0.9269	100	0.9325	100	0.9352	100	0.9330	100

S.II. RADIAL ACQUISITION RECONSTRUCTION

This section provides additional results on the brain 1 image with radial acquisition. Fig. S.3 shows the cost function and PSNR values versus the number of iterations and wall time. Fig. S.4 illustrates the reconstructed images and associated error maps for each method at the 30th and 100th iterations. The observed performance trends are consistent with those reported in the main paper. Table S.5 summarizes the additional results on five other brain images. GKSM converged faster than other competing methods in terms of wall time. Tables S.4 and S.6 show the associated SSIM values for the brain and knee 1 images with radial acquisition. In this setting, we observed that CQNOM achieved the highest SSIM values in most cases. However, GKSM consistently obtained similar SSIM values.



(a) (b)

(c) (d)

Fig. S.3. Comparison of different methods with radial acquisition on the brain 1 image with $\varepsilon = 7 \times 10^{-4}$. (a), (b): cost values versus iteration and wall time; (c), (d): PSNR values versus iteration and wall time.

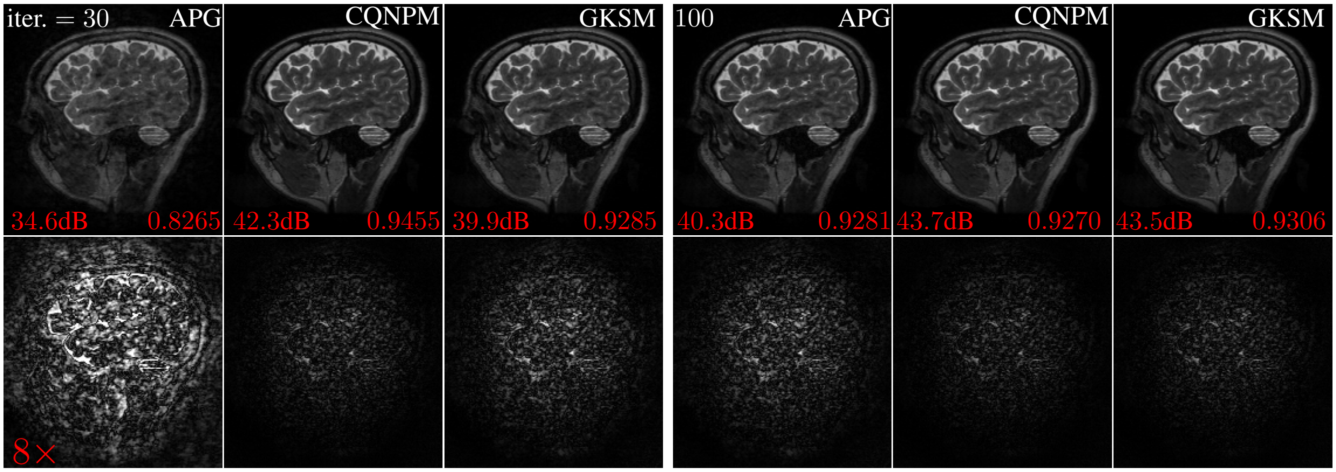


Fig. S.4. First row: the reconstructed knee 1 images of each method at 30th and 100th iterations with spiral acquisition. The PSNR (respectively, SSIM) values are labeled at the left (respectively, right) bottom corner of each image. Second row: the associated error maps (8×) of the reconstructed images.

TABLE S.5

PSNR PERFORMANCE OF EACH METHOD FOR RECONSTRUCTING FIVE ADDITIONAL BRAIN TEST IMAGES WITH RADIAL ACQUISITION. FOR APG, WE REPORT THE MAXIMUM PSNR (WITHIN 100 ITERATIONS), THE CORRESPONDING NUMBER OF ITERATIONS, AND THE WALL TIME. FOR OTHER METHODS, WE REPORT THE EARLIEST ITERATION THAT EXCEEDS THE APG PSNR (ALONG WITH ITS PSNR AND WALL TIME) IN THE FIRST ROW. THE PSNR AND WALL TIME AT THE 100TH ITERATION WERE SUMMARIZED IN THE SECOND ROW. BOLD INDICATES THE SHORTEST WALL TIME AT WHICH THE PSNR OF APG WAS EXCEEDED. THE PSNR VALUES AND WALL TIME AT THE 100TH ITERATION OF CQNPM AND GKSM ARE MARKED WITH AN UNDERLINE. THE BLUE DIGITS DENOTE THE SHORTEST WALL TIME AT THE 100TH ITERATION.

Index	2			3			4			5			6		
Methods	PSNR↑	iter.↓	sec.↓	PSNR↑	iter.↓	sec.↓	PSNR↑	iter.↓	sec.↓	PSNR↑	iter.↓	sec.↓	PSNR↑	iter.↓	sec.↓
APG	40.4	100	62.9	40.4	100	60.4	40.1	100	60.3	39.7	100	62.7	38.2	61.2	100
CQNPM	40.6	11	7.1	40.5	11	7.0	40.1	12	7.1	39.8	11	7.1	38.4	11	6.9
	41.4	100	65.2	43.0	100	63.6	41.0	100	59.7	42.0	100	67.0	42.0	100	64.1
GKSM	40.5	34	**1.7**	40.5	32	**1.5**	40.2	30	**1.5**	39.7	33	**1.6**	38.2	31	**1.5**
	40.6	100	5.5	42.3	100	5.5	43.0	100	5.6	42.7	100	5.5	41.8	100	5.5

TABLE S.6

THE SSIM METRICS OF EACH METHOD FOR RECONSTRUCTING FIVE ADDITIONAL BRAIN TEST IMAGES WITH RADIAL ACQUISITION CORRESPOND TO TABLE S.5. WE ALSO PROVIDE THE ITERATION COUNTS SO THAT THE SSIM AND PSNR VALUES CAN BE ASSOCIATED WITH THE SAME RECONSTRUCTION ITERATION. THE HIGHEST SSIM VALUES ARE SHOWN IN BOLD.

Index	2		3		4		5		6	
Methods	SSIM↑	iter.↓	SSIM↑	iter.↓	SSIM↑	iter.↓	SSIM↑	iter.↓	SSIM↑	iter.↓
APG	0.9382	100	0.9258	100	0.9416	100	0.9335	100	0.9217	100
CQNPM	0.9463	11	**0.9394**	11	0.9478	12	0.9459	11	**0.9381**	11
	0.9516	100	0.9323	100	**0.9523**	100	**0.9472**	100	0.9223	100
GKSM	0.9417	34	0.9332	32	0.9351	30	0.9392	33	0.9285	31
	0.9356	100	0.9250	100	0.9329	100	0.9328	100	0.9252	100