





# Provable Preconditioned Plug-and-Play Approach for Compressed Sensing MRI Reconstruction

Tao Hong , *Member, IEEE*, Xiaojian Xu , *Member, IEEE*, Jason Hu , *Graduate Student Member, IEEE*, and Jeffrey A. Fessler , *Fellow, IEEE*

**Abstract**—Model-based methods play a key role in the reconstruction of compressed sensing (CS) MRI. Finding an effective prior to describe the statistical distribution of the image family of interest is crucial for model-based methods. Plug-and-play (PnP) is a general framework that uses denoising algorithms as the prior or regularizer. Recent work showed that PnP methods with denoisers based on pretrained convolutional neural networks outperform other classical regularizers in CS MRI reconstruction. However, the numerical solvers for PnP can be slow for CS MRI reconstruction. This paper proposes a preconditioned PnP (P<sup>2</sup>nP) method to accelerate the convergence speed. Moreover, we provide proofs of the fixed-point convergence of the P<sup>2</sup>nP iterates. Numerical experiments on CS MRI reconstruction with non-Cartesian sampling trajectories illustrate the effectiveness and efficiency of the P<sup>2</sup>nP approach.

**Index Terms**—Magnetic resonance imaging (MRI), non-cartesian sampling, reconstruction, preconditioner, plug-and-play (PnP).

## I. INTRODUCTION

MAGNETIC resonance imaging (MRI) is a noninvasive medical imaging technique that uses magnetic fields to obtain images of organs, tissues, and other structures. MRI scanners acquire the Fourier components of the image of interest, called the k-space. However, the acquisition procedure is slow. To accelerate the acquisition, one strategy is to under-sample the Fourier components, but this violates the condition of the Nyquist sampling theorem, causing aliasing in conventionally reconstructed images. To solve this problem, modern MRI scanners use multiple coils (parallel imaging) to acquire the Fourier components, providing additional spatial information [1], [2], [3]. Moreover, compressed sensing (CS) MRI [4], [5] improves the quality of the reconstructed images by using suitable sampling patterns. In practice, CS MRI is combined with parallel

imaging and the MRI image is reconstructed by solving a composite minimization problem like the following:

$$\hat{x} = \arg \min_{x \in \mathbb{C}^N} \underbrace{\frac{1}{2} \|Ax - y\|_2^2}_{f(x)} + \phi(x), \quad (1)$$

where  $A \in \mathbb{C}^{ML \times N}$  refers to the forward model defining the mapping from the image  $x$  to the acquired k-space data  $y$  and  $L$  is the number of coils.  $A$  consists of a stack of matrices  $A_l$  such that  $A = [A_1; A_2; \dots; A_L]$  where  $A_l \in \mathbb{C}^{M \times N} \triangleq MFC_l$ .  $M \in \mathbb{R}^{M \times N}$  defines the sampling pattern and  $F \in \mathbb{C}^{N \times N}$  denotes the (non-uniform) Fourier transform operator.  $C_l \in \mathbb{C}^{N \times N}$  represents the sensitivity map associated with the  $l$ th coil and is patient specific.

The data-fit term  $f(\cdot)$  encourages consistency of the image  $x$  with the measurements  $y$  and  $\phi(\cdot)$  is a regularizer that describes the statistical distribution of the unknown image  $x$ , often called a prior. Classical choices for  $\phi(\cdot)$  that have shown to be useful for MRI reconstruction include total variation (TV) [4], [6], [7], wavelets [8], [9], dictionary learning [10], [11], and low-rank [12], to name a few. See [13], [14] for reviews of different choices of  $\phi(\cdot)$ . In the past decade, deep learning (DL) has gained a lot of attention in reconstructing MRI images due to its excellent performance. Instead of hand-crafting explicit priors, DL provides a data-driven tool for implicitly encoding image priors. Popular DL-based approaches for MRI reconstruction include end-to-end mapping [15] and model-based (or called physics-informed) deep unrolling [16], [17], [18], [19], [20]. Recently, using generative models to learn a prior for solving MRI reconstruction has received extensive interest [21], [22].

*Plug-and-Play (PnP)* [23] is an alternative to DL that leverages the most effective image denoisers, such as BM3D [24] or DnCNN [25], leading to state-of-the-art performance in various imaging tasks [26], [27], [28], [29], [30], [31]. Differing from DL approaches that usually rely on training with massive data for a predefined imaging task, PnP can be easily customized to a specific application without retraining. This feature is particularly beneficial for solving CS MRI problems, where the sampling patterns, coil sensitivity maps, and image resolution can vary significantly from scan to scan. Moreover, PnP methods can still achieve reasonable reconstruction results even when the testing data is out-of-distribution from the training data used for the denoiser [32]. Detailed discussions about using PnP for MRI reconstruction are found in [33].

Received 6 May 2024; revised 27 August 2024; accepted 1 October 2024. Date of publication 9 October 2024; date of current version 18 October 2024. The work of Tao Hong was supported by the National Institutes of Health under Grant R01NS112233. The associate editor coordinating the review of this article and approving it for publication was Prof. Bo Zhao. (*Corresponding author: Tao Hong.*)

Tao Hong is with the Department of Radiology, University of Michigan, Ann Arbor, MI 48109 USA (e-mail: tahong@umich.edu).

Xiaojian Xu, Jason Hu, and Jeffrey A. Fessler are with the Department of Electrical and Computer Engineering, University of Michigan, Ann Arbor, MI 48109 USA (e-mail: xjxu@umich.edu; jashu@umich.edu; fessler@umich.edu).

This article has supplementary downloadable material available at <https://doi.org/10.1109/TCI.2024.3477329>, provided by the authors.

Digital Object Identifier 10.1109/TCI.2024.3477329

PnP originates from the proximal algorithms [34], which is a class of iterative algorithms for solving (1). At the  $k$ th iteration, the (accelerated) proximal gradient method, which is also called (Fast) Iterative Shrinkage-Thresholding Algorithm ((F)ISTA) [35], updates the next iterate as

$$\begin{aligned} \mathbf{x}_{k+1} &\leftarrow \text{prox}_{\alpha\phi}(\mathbf{z}_k - \alpha\nabla f(\mathbf{z}_k)) \\ \mathbf{z}_{k+1} &\leftarrow \mathbf{x}_{k+1} + c_k(\mathbf{x}_{k+1} - \mathbf{x}_k), \end{aligned} \quad (2)$$

where  $\alpha > 0$  is the step size,  $c_k > 0$  encodes the acceleration mechanism, and  $\nabla f(\cdot)$  denotes the gradient of  $f(\cdot)$ . Here,  $\text{prox}_{\alpha\phi}(\cdot)$  represents the proximal operator defined as

$$\text{prox}_{\alpha\phi}(\cdot) \triangleq \underset{\mathbf{x} \in \mathbb{C}^N}{\text{argmin}} \frac{1}{2} \|\mathbf{x} - \cdot\|_2^2 + \alpha\phi(\mathbf{x}). \quad (3)$$

The observation that (3) can be interpreted as a denoiser inspired the development of PnP-(F)ISTA algorithm, where the proximal operator is replaced with an arbitrary denoiser  $D_\sigma(\cdot)$ . Here,  $\sigma^2$  represents the variance of the additive white Gaussian noise and is related to the strength of the regularizer, e.g., a large  $\sigma$  means more regularization. In PnP practice, the value of  $\sigma$  is often selected via empirical parameter tuning, after choosing the step size  $\alpha$  based on a Lipschitz constant for  $\nabla f$ . PnP-ADMM [36] is another popular PnP approach that comes from the Alternating Direction Method of Multipliers (ADMM) [37]. Section II discusses PnP approaches and their convergence properties.

Despite the rich literature that elaborates on the benefits of using PnP for image reconstruction, efficiently solving PnP optimization problems remains computationally challenging, particularly for multi-coil non-Cartesian sampling CS MRI reconstruction. Frequent use of the forward model  $\mathbf{A}$  results in significant computational expense in CS MRI reconstruction. This work focuses on this challenge and proposes a *Preconditioned* PnP ( $P^2$ nP) method to accelerate the convergence speed of PnP-(F)ISTA algorithm. The main *contributions* of our work are summarized as follows:

- 1) We propose a new preconditioned solver for PnP called  $P^2$ nP that improves the convergence speed of PnP-(F)ISTA algorithm. In particular, we present two different strategies for designing the preconditioners, i.e., *fixed* and *dynamic*, where the fixed one is pre-determined using the forward model  $\mathbf{A}$ , and the dynamic one is estimated at each iteration with negligible cost. Since the estimation of the dynamical preconditioner does not rely on the evaluation of  $\mathbf{A}$ , it is especially useful when computing  $\mathbf{A}\mathbf{x}$  is computationally expensive.
- 2) We establish the theoretical convergence and stability analysis of  $P^2$ nP. Our proofs show that, under mild conditions,  $P^2$ nP achieves fixed-point convergence for both *fixed* and the *dynamic* preconditioners.
- 3) We applied *normalization-equivariant* denoisers in the PnP framework. Our results demonstrated that, compared with ordinary CNN denoisers, noise-adapted and robust normalization-equivariant denoisers can effectively boost the performance of PnP, compared to existing PnP methods for which selecting a denoising strength is tricky.
- 4) We extensively tested  $P^2$ nP for multi-coil non-Cartesian sampling CS MRI reconstruction in a variety of settings,

including with fixed and dynamic preconditioners, spiral and radial sampling trajectories, and a normalization-equivariant denoiser. Our numerical results show that  $P^2$ nP consistently outperformed the baseline methods in terms of both the convergence speed and also the reconstruction performance, both qualitatively and quantitatively.

The rest of the paper is organized as follows. Section II presents some popular existing variants and convergence properties of PnP. Section III proposes our preconditioned PnP ( $P^2$ nP) method and discusses the convergence properties of  $P^2$ nP. Section IV summarizes the experimental validation of  $P^2$ nP for CS MRI reconstruction and the comparison with other known baseline methods. Furthermore, we examine the convergence of  $P^2$ nP to verify our theoretical analyses. Section V concludes the paper.

## II. PRELIMINARIES ON PNP APPROACHES

This section reviews different variants of PnP and their convergence properties. PnP is a family of imaging algorithms that interpret the prior with a black-box denoiser. For example, beyond PnP-ADMM and PnP-(F)ISTA, PnP can be developed from other algorithms, such as PnP-PDS [27], PnP-HQS [38], and PnP-CE [29], etc. An alternative to PnP is regularization by denoising (RED) [39], [40], [41] that forms an explicit denoiser-embedded regularization function. To accelerate the convergence of PnP, Tan et al. [42] proposed a PnP-quasi-Newton approach incorporating quasi-Newton steps into a provable PnP framework. However, their method is quite complicated and the computation at each step is very expensive. Pendu et al. [43] analyzed a preconditioned PnP-ADMM algorithm using a diagonal matrix as the preconditioner. In general, diagonal matrices are ineffective for many applications because the associated Hessian matrices are not diagonally dominant. The method in [43] requires training a locally adjustable denoiser for different preconditioners  $\{\Sigma\}$ , where the denoiser solves

$$\min_{\mathbf{x} \in \mathbb{R}^N} \frac{1}{2} \|\Sigma(\mathbf{x} - \cdot)\|_2^2 + \alpha\phi(\mathbf{x}), \quad (4)$$

where  $\Sigma$  is a diagonal matrix.  $\Sigma$  will be used as an additional channel for training a denoiser. In practical applications, finding a set of  $\{\Sigma\}$  for training is not an easy task. Furthermore, it is also nontrivial to find such a diagonal preconditioner when the explicit formulation of  $\mathbf{A}$  is unknown.

Besides the empirical success of PnP, the convergence analyses of PnP also has made much progress. Sreehari et al. [26] provided sufficient conditions for the convergence of PnP with respect to some implicit objective function. Chan et al. [36] established the fixed-point convergence of PnP-ADMM for bounded denoisers. Buzzard et al. [29] proposed a fixed-point interpretation of PnP from the consensus equilibrium view. Teodoro et al. [44] established the convergence for PnP-ADMM with linearized Gaussian mixture model denoisers. Gavaskar et al. [45] showed the convergence of PnP with an explicit cost function for linear denoisers. Recent analyses showed that the convergence of PnP iterates can also be ensured under other

assumptions about the denoisers [46], [47], [48], [49], [50], [51]. Related work on RED convergence analysis includes [39], [41], [52]. Recent work in [53] demonstrated that RED is guaranteed to converge to a stationary point in a non-convex setting. See [54] and the references therein for detailed discussions of PnP convergence properties.

### III. PROPOSED METHOD

This section first presents our P<sup>2</sup>nP method and then describes two different strategies for choosing the preconditioner: fixed and dynamic. We also provide the convergence and stability analyses for P<sup>2</sup>nP.

Let  $\mathbf{P} \succ \mathbf{0}$  denote a Hermitian positive matrix in  $\mathbb{C}^{N \times N}$ . At the  $k$ th iteration, P<sup>2</sup>nP solves the following problem:

$$\begin{aligned} \mathbf{x}_{k+1} = \operatorname{argmin}_{\mathbf{u} \in \mathbb{C}^N} & \left\{ \langle \nabla f(\mathbf{x}_k), \mathbf{u} - \mathbf{x}_k \rangle \right. \\ & \left. + \frac{1}{2a} (\mathbf{u} - \mathbf{x}_k)^H \mathbf{P}^{-1} (\mathbf{u} - \mathbf{x}_k) + \phi(\mathbf{u}) \right\} \\ = \operatorname{argmin}_{\mathbf{u} \in \mathbb{C}^N} & \left\{ \frac{1}{2} \|\mathbf{u} - (\mathbf{x}_k - \alpha \mathbf{P} \nabla f(\mathbf{x}_k))\|_{\mathbf{P}^{-1}}^2 \right. \\ & \left. + \alpha \phi(\mathbf{u}) \right\}, \end{aligned} \quad (5)$$

where  $\alpha$  is the step-size,  $^H$  denotes the Hermitian transpose, and we define the weighted Euclidean norm  $\|\mathbf{v}\|_{\mathbf{P}^{-1}}^2 \triangleq \mathbf{v}^H \mathbf{P}^{-1} \mathbf{v}$ . Clearly, if  $\mathbf{P} = \mathbf{I}$ , then P<sup>2</sup>nP reverts to PnP-ISTA. Using the fact that  $\|\mathbf{v}\|_{\mathbf{P}^{-1}}^2 \leq \|\mathbf{P}^{-1}\|_2 \|\mathbf{v}\|_2^2$ , we remove the weighting  $\mathbf{P}^{-1}$  in (5) by using the update

$$\mathbf{x}_{k+1} = \operatorname{argmin}_{\mathbf{u} \in \mathbb{C}^N} \frac{1}{2} \|\mathbf{u} - (\mathbf{x}_k - \alpha \mathbf{P} \nabla f(\mathbf{x}_k))\|_2^2 + \frac{\alpha}{\eta} \phi(\mathbf{u}), \quad (6)$$

where  $\eta \triangleq \|\mathbf{P}^{-1}\|_2 > 0$ . Aminifard et al. [55] proved that convergence of the function value sequence  $\{f(\mathbf{x}_k)\}$  in (1) is still guaranteed when solving (6) instead of (5) with an extra line search step. The following section shows a stronger result, namely that convergence of the iterates sequence  $\{\mathbf{x}_k\}$  of P<sup>2</sup>nP still holds under mild assumptions even without an extra line search step.

Clearly, (6) is very similar to the definition of the proximal operator in (3). So we replace this proximal operator with a denoiser  $\mathbf{D}_\sigma$ , leading to our proposed P<sup>2</sup>nP algorithm summarized in Algorithm 1. One can treat  $\frac{1}{\eta}$  in (6) as a trade-off parameter. When  $\frac{1}{\eta} > 1$ , we should emphasize  $\phi(\mathbf{u})$  more, so the denoiser should use a larger  $\sigma$  value than the one used for (3). For a given choice of  $\mathbf{P}$  and step size  $\alpha$ , one can fine-tune an appropriate  $\sigma$  value empirically in light of (6).

Similar to PnP-FISTA, one could try to accelerate P<sup>2</sup>nP using momentum. However, we experimentally found that the performance of both PnP-FISTA and the accelerated P<sup>2</sup>nP degraded significantly after running “too many” iterations, a problem needing further investigation in the future. So this paper focuses on Algorithm 1. Next we show two different ways to choose the preconditioner  $\mathbf{P}$ .

---

#### Algorithm 1: Preconditioned Plug-and-Play (P<sup>2</sup>nP).

---

**Require:**  $\mathbf{x}_1$  and step-size  $\alpha > 0$ .  
**1: for**  $k = 1, 2, \dots$  **do**  
**2:**  $\mathbf{x}_{k+1} \leftarrow \mathbf{D}_\sigma(\mathbf{x}_k - \alpha \mathbf{P} \nabla f(\mathbf{x}_k))$ .  
**3: end for**

---

#### A. Fixed Preconditioners

We first discuss the convergence of P<sup>2</sup>nP to gain insights into the kinds of preconditioners  $\mathbf{P}$  that can improve the convergence rate and then show how to choose a fixed  $\mathbf{P}$  efficiently in practice.

1) *Convergence Analysis:* Here we discuss the convergence condition of P<sup>2</sup>nP and its convergence rate bound. First, we assume the denoiser is Lipschitz continuous, a standard assumption in the analysis of fixed-point PnP [47], i.e., Assumption 1.

*Assumption 1:* The denoiser  $\mathbf{D}_\sigma(\cdot) : \mathbb{C}^N \rightarrow \mathbb{C}^N$  is Lipschitz continuous with Lipschitz constant  $1 + \epsilon$  for  $\epsilon > 0$  so that the following inequality is satisfied for all  $\mathbf{x}, \mathbf{y} \in \mathbb{C}^N$ ,

$$\|\mathbf{D}_\sigma(\mathbf{x}) - \mathbf{D}_\sigma(\mathbf{y})\| \leq (1 + \epsilon) \|\mathbf{x} - \mathbf{y}\|.$$

Then Theorem 1 provides a sufficient condition for convergence of the P<sup>2</sup>nP iterates to a fixed-point.

*Theorem 1 (Convergence of P<sup>2</sup>nP with fixed preconditioner):* Assume  $\mathbf{D}_\sigma(\cdot)$  satisfies Assumption 1. Then the iterates sequence generated by Algorithm 1 converges to a fixed-point if

$$r_0 \triangleq (1 + \epsilon) \rho(\mathbf{I} - \alpha \mathbf{P} \mathbf{A}^H \mathbf{A}) < 1, \quad (7)$$

where  $\rho(\cdot)$  denotes the spectral radius, and the convergence rate of the iterates is upper bounded by that  $r_0$ .

See Appendix A for the proof.

From Theorem 1, an ideal  $\alpha \mathbf{P}$  should be chosen to minimize  $\rho(\mathbf{I} - \alpha \mathbf{P} \mathbf{A}^H \mathbf{A})$ . If  $\mathbf{A}^H \mathbf{A} \succ \mathbf{0}$ , the ideal choice would be  $\alpha \mathbf{P} = (\mathbf{A}^H \mathbf{A})^{-1}$  so that Algorithm 1 would converge to a fixed-point in one iteration. However, the computation of such an  $\alpha \mathbf{P}$  is expensive, and, in CS MRI,  $\mathbf{A}^H \mathbf{A}$  is a Hermitian positive semi-definite matrix, so choosing such an  $\alpha \mathbf{P}$  is impractical.

2) *The Choice of Preconditioners  $\alpha \mathbf{P}$ TEXT:* Finding a  $\alpha \mathbf{P}$  to minimize  $\rho(\mathbf{I} - \alpha \mathbf{P} \mathbf{A}^H \mathbf{A})$  is a non-trivial task. One approach that has proved to be useful in the scientific computing community is to minimize an upper bound of  $\rho(\mathbf{I} - \alpha \mathbf{P} \mathbf{A}^H \mathbf{A})$  [56], [57], i.e.,

$$\min_{\alpha \mathbf{P}} \|\mathbf{I} - \alpha \mathbf{P} \mathbf{A}^H \mathbf{A}\|_F^2. \quad (8)$$

In CS MRI, we do not store dense matrix  $\mathbf{A}$  explicitly, so using (8) would be challenging. Moreover,  $\mathbf{A}$  is different for each scan because of the patient-dependent coil sensitivity maps, making any expensive pre-computing strategies impractical.

This paper proposes using polynomial preconditioners [58]. Represent  $\alpha \mathbf{A}^H \mathbf{A}$  as  $\alpha \mathbf{A}^H \mathbf{A} = \mathbf{I} - \bar{\mathbf{A}}$ . Then the ideal preconditioner would be

$$(\mathbf{I} - \bar{\mathbf{A}})^{-1} = \mathbf{I} + \bar{\mathbf{A}} + \bar{\mathbf{A}}^2 + \dots \quad (9)$$

An incomplete inverse of  $\alpha \mathbf{A}^H \mathbf{A}$  is the truncated form of (9), i.e.,  $\mathbf{P} = \sum_{\gamma=1}^{\Gamma} a_\gamma (\bar{\mathbf{A}})^{\gamma-1}$  with  $\Gamma \in \mathbb{Z}_+ : \geq 2$ . Since



$\alpha \mathbf{P} \mathbf{A}^H \mathbf{A} = \sum_{\gamma=1}^{\Gamma} \bar{p}_{\gamma} (\alpha \mathbf{A}^H \mathbf{A})^{\gamma}$ , we know  $\mathbf{P}$  is also a polynomial in  $\mathbf{A}^H \mathbf{A}$ . So we consider the following polynomial form of  $\mathbf{P}$ :

$$\mathbf{P} = \sum_{\gamma=1}^{\Gamma} p_{\gamma} (\alpha \mathbf{A}^H \mathbf{A})^{\gamma-1}, \quad (10)$$

where  $\{p_{\gamma}\}_{\gamma}$  is a set of scalars. We would like to choose this set to minimize  $\rho(\mathbf{I} - \alpha \mathbf{P} \mathbf{A}^H \mathbf{A})$ . Note that the cost of finding  $\{p_{\gamma}\}_{\gamma}$  for an effective  $\mathbf{P}$  can be negligible. For instance, Zulfiqar et al. in [59] set

$$p_{\gamma} = \binom{\Gamma}{\gamma} (-1)^{\gamma-1}, \quad (11)$$

and adapted the polynomial preconditioner to FISTA for wavelet-based image reconstruction, i.e., solving

$$\mathbf{x}_* = \arg \min_{\mathbf{x} \in \mathbb{R}^N} \frac{1}{2} \|\mathbf{A} \mathbf{W}^{-1} \mathbf{x} - \mathbf{y}\|_2^2 + \nu \|\mathbf{x}\|_1 \quad (12)$$

where  $\mathbf{W}$  represents an orthogonal wavelet transform and  $\nu > 0$  is the trade-off parameter. Then the recovered image is  $\mathbf{W}^{-1} \mathbf{x}_*$ . Zulfiqar et al. showed that the preconditioned FISTA converged significantly faster than regular FISTA for addressing (12).

An alternative approach is used in [60], where Iyer et al. showed the effectiveness of Chebyshev polynomial preconditioners for CS MRI reconstruction with low-rank regularization. Because  $\mathbf{A}^H \mathbf{A}$  is Hermitian, all its eigenvalues are real, allowing the use of Chebyshev polynomial preconditioners that are optimal for minimizing  $\rho(\mathbf{I} - \alpha \mathbf{P} \mathbf{A}^H \mathbf{A})$ . By finding the smallest and largest eigenvalue of  $\mathbf{A}^H \mathbf{A}$ , we can get the values of  $\{p_{\gamma}\}_{\gamma}$  analytically. In CS MRI,  $\mathbf{A}$  is usually under-determined, so the smallest eigenvalue of  $\mathbf{A}^H \mathbf{A}$  is zero and we obtain the largest eigenvalue by the power method.

By using a recursive implementation, applying  $\mathbf{P} \mathbf{x}$  needs to compute  $\mathbf{A}^H \mathbf{A} \mathbf{x}$  a total of  $(\Gamma - 1)$  times. In modern CS MRI, performing  $\mathbf{A}^H \mathbf{A} \mathbf{x}$  is also expensive due to the use of multi-coils and non-Cartesian acquisition, so a large  $\Gamma$  can dramatically increase computational costs. Note that a large  $\Gamma$  can yield a more effective preconditioner but leads to higher computational cost. In practice, its efficiency depends on the sampling trajectories, sensitivity maps, number of coils, and matrix size. So, in practice, the best  $\Gamma$  should strike a balance between its effectiveness and the extra computation. In this paper, we simply used  $\Gamma = 2$  for all experiments. To further reduce the computation in the polynomial preconditioners, we study dynamic preconditioners in the next part, where the additional computation at each iteration is negligible.

## B. Dynamic Preconditioners

Theorem 1 indicates that a well-designed preconditioner should approximate the inverse of  $\mathbf{A}^H \mathbf{A}$ . So an alternative approach is to use quasi-Newton methods [61, Ch. 6] to estimate a  $\mathbf{P}$  that approximates  $(\mathbf{A}^H \mathbf{A})^{-1}$ . Following [62], [63], [64], we suggest using the zero-memory self-scaling Hermitian rank-1 (ZMSHR1) for complex data to define such a preconditioner. Algorithm 2 summarizes the ZMSHR1 approach. Since (14)

is a one-dimensional minimization problem, we simply search for the minimal  $a$ . Lemma 1 specifies some properties of the variables generated by Algorithm 2.

**Lemma 1:** Suppose  $f(\mathbf{x}) = \frac{1}{2} \|\mathbf{A} \mathbf{x} - \mathbf{y}\|_2^2$  with  $\mathbf{x} \in \mathbb{C}^N$ . Then  $\tau_k$  and  $\langle \mathbf{s}_k, \mathbf{v}_k \rangle$  generated in Algorithm 2 are real and  $\langle \mathbf{s}_k - \tau_k \mathbf{v}_k, \mathbf{v}_k \rangle \geq 0$ . Moreover, the generated  $\tau_k$  and  $\mathbf{P}_k \in \mathbb{C}^{N \times N}$  for  $\forall k$  are bounded by

$$\frac{1}{2\theta_2} < \tau_k \leq \frac{1}{\theta_1},$$

$$\frac{1}{2\theta_2} \preceq \mathbf{P}_k \preceq \frac{\delta + 1}{\delta \theta_1},$$

where  $\delta > 0$ ,  $\theta_1 \in (0, 1)$ , and  $\theta_2 \in (1, \infty)$ .

See Appendix B for the proof.

Lemma 1 ensures that  $\mathbf{P}_k$  is a positive-definite matrix. Let  $\eta_k \triangleq \|\mathbf{P}_k^{-1}\|_2$ . Then, at the  $k$ th iteration, the dynamic preconditioner version of P<sup>2</sup>nP solves the following problem instead of (6):

$$\mathbf{x}_{k+1} = \underset{\mathbf{u} \in \mathbb{C}^N}{\operatorname{argmin}} \frac{1}{2} \|\mathbf{u} - (\mathbf{x}_k - \alpha \mathbf{P}_k \nabla f(\mathbf{x}_k))\|_2^2 + \frac{\alpha}{\eta_k} \phi(\mathbf{u}). \quad (13)$$

Clearly,  $\eta_k$  differs at each iteration, so solving (13) is not equivalent to executing a denoiser with a fixed  $\sigma$  for all iterations. In [65], Xu et al. proved that if  $\mathbf{D}_{\sigma}(\cdot)$  denotes a minimum mean-squared error (MMSE) denoiser for image data with noise standard deviation  $\sigma$ , then  $\frac{1}{\mu} \mathbf{D}_{\sigma}(\mu \mathbf{x})$ ,  $\mu > 0$  is the MMSE denoiser for an image with noise level  $\sigma^2/\mu^2$ . Moreover, Xu et al. [50] showed many modern denoisers like BM3D [24] and trained CNN denoisers can be treated experimentally as MMSE denoisers. So, for a given  $\eta_k$ , one could, in principle, solve (13) by (somehow) finding a suitable  $\eta_k^*$  and running  $\frac{1}{\eta_k^*} \mathbf{D}_{\sigma}(\eta_k^* \mathbf{x})$  if  $\mathbf{D}_{\sigma}(\cdot)$  is a MMSE denoiser. However, finding  $\eta_k^*$  in practice is a nontrivial task.

Herbreteau et al. proposed a normalization-equivariant CNN denoiser that performs as well as an ordinary denoiser and can denoise noisy images across various noise levels [66]. Specifically, the normalization-equivariant denoiser has the special property that, for any  $\mu \in \mathbb{R}$  with  $\mu > 0$  and  $\Delta \in \mathbb{C}$ , we have  $\mathbf{D}_{\sigma}(\mu \mathbf{x} + \Delta \mathbf{1}) = \mu \mathbf{D}_{\sigma}(\mathbf{x}) + \Delta \mathbf{1}$ , where  $\mathbf{1}$  denotes the vector of ones. By using a normalization-equivariant denoiser, we can simply apply  $\mathbf{D}_{\sigma}(\mathbf{x}_k - \alpha \mathbf{P}_k \nabla f(\mathbf{x}_k))$  instead of (13) without needing to adjust the associated noise level  $\sigma$  for different  $\eta_k$  values; this property makes using a dynamic preconditioner as practical as the fixed preconditioner case. Algorithm 3 summarizes P<sup>2</sup>nP with dynamic preconditioners.

Section IV-D compares the performance between normalization-equivariant and ordinary denoisers and shows that the normalization-equivariant denoiser outperformed the ordinary denoiser. Next, we discuss the stability property of Algorithm 3, ensuring that the iterates  $\mathbf{x}_k$  generated by Algorithm 3 will always be close to a fixed-point.

**1) Stability Analysis:** In this part, we present the stability analysis of P<sup>2</sup>nP when using dynamic preconditioners. We show that Algorithm 3 is reliable in the sense that the iterates  $\mathbf{x}_k$  remain in a bounded set. We first assume the gradient of  $f$  is upper bounded.



---

**Algorithm 2:** Zero-Memory Self-Scaling Hermitian rank-1 (ZMSHR1) Method.

---

**Require:**  $\mathbf{x}_{k-1}, \mathbf{x}_k, \nabla f(\mathbf{x}_{k-1}), \nabla f(\mathbf{x}_k), \delta > 0$ ,  
 $\theta_1 \in (0, 1)$ , and  $\theta_2 \in (1, \infty)$ .  
1: Set  $\mathbf{s}_k \leftarrow \mathbf{x}_k - \mathbf{x}_{k-1}$  and  $\mathbf{m}_k \leftarrow \nabla f(\mathbf{x}_k) - \nabla f(\mathbf{x}_{k-1})$ .  
2: Compute  $a$  such that

$$\min_a \{a \in [0, 1] \mid \mathbf{v}_k = a\mathbf{s}_k + (1-a)\mathbf{m}_k\} \quad (14)$$

satisfies  $\theta_1 \leq \frac{\langle \mathbf{s}_k, \mathbf{v}_k \rangle}{\langle \mathbf{s}_k, \mathbf{s}_k \rangle}$  and  $\frac{\langle \mathbf{v}_k, \mathbf{v}_k \rangle}{\langle \mathbf{s}_k, \mathbf{v}_k \rangle} \leq \theta_2$ .  
3: Compute  $\tau_k \leftarrow \frac{\langle \mathbf{s}_k, \mathbf{s}_k \rangle}{\langle \mathbf{s}_k, \mathbf{v}_k \rangle} - \sqrt{\left(\frac{\langle \mathbf{s}_k, \mathbf{s}_k \rangle}{\langle \mathbf{s}_k, \mathbf{v}_k \rangle}\right)^2 - \frac{\langle \mathbf{s}_k, \mathbf{s}_k \rangle}{\langle \mathbf{v}_k, \mathbf{v}_k \rangle}}$ .  
4: **if**  $\langle \mathbf{s}_k - \tau_k \mathbf{v}_k, \mathbf{v}_k \rangle \leq \delta \|\mathbf{s}_k - \tau_k \mathbf{v}_k\| \|\mathbf{v}_k\|$  **then**  
5:    $\mathbf{u}_k \leftarrow \mathbf{0}$ .  
6: **else**  
7:    $\mathbf{u}_k \leftarrow \mathbf{s}_k - \tau_k \mathbf{v}_k$ .  
8: **end if**  
9: **Return:**  $\mathbf{P}_k \leftarrow \tau_k \mathbf{I} + \frac{\mathbf{u}_k \mathbf{u}_k^H}{\langle \mathbf{s}_k - \tau_k \mathbf{v}_k, \mathbf{v}_k \rangle}$ .

---



---

**Algorithm 3:** P<sup>2</sup>nP With Dynamic Preconditioners.

---

**Require:**  $\mathbf{x}_1$ , step size  $\alpha \in \mathbb{R}$ ,  
and  $D_\sigma(\cdot)$  is a normalization-equivariant denoiser.  
1: **for**  $k = 1, 2, \dots$  **do**  
2:   **if**  $k=1$  **then**  
3:      $\mathbf{x}_{k+1} \leftarrow D_\sigma(\mathbf{x}_k - \alpha \nabla f(\mathbf{x}_k))$ .  
4:   **else**  
5:     Call Algorithm 2 to get  $\mathbf{P}_k$ .  
6:      $\mathbf{x}_{k+1} \leftarrow D_\sigma(\mathbf{x}_k - \alpha \mathbf{P}_k \nabla f(\mathbf{x}_k))$ .  
7:   **end if**  
8: **end for**

---

*Assumption 2:* For  $\forall \mathbf{x}_k$  generated by Algorithm 3, there exists  $R < \infty$  such that

$$\|\nabla f(\mathbf{x}_k)\| \leq R.$$

The bound  $R$  exists in practice since many denoising algorithms have bounded range spaces [48] and  $\nabla f(\mathbf{x}) = \mathbf{A}^H(\mathbf{A}\mathbf{x} - \mathbf{y})$ . Theorem 2 defines an upper bound on  $\|\mathbf{x}_k - \mathbf{x}_*\|$ .

*Theorem 2 (Stability of Algorithm 3):* Suppose a Hermitian positive matrix  $\mathbf{P}_*$  satisfies  $\mathbf{0} \prec \mathbf{P}_* \preceq \lambda_* \mathbf{I}$  with  $\lambda_* < \infty$ , and  $q \triangleq [(1 + \epsilon) \rho(\mathbf{I} - \alpha \mathbf{P}_* \mathbf{A}^H \mathbf{A})] < 1$ . Let  $\{\mathbf{P}_k\}_k$  denote the dynamic preconditioners generated by Algorithm 2 and assume a normalization-equivariant denoiser is used. Then  $\|\mathbf{x}_k - \mathbf{x}_*\|$  with  $\mathbf{x}_k$  generated by Algorithm 3 and  $\mathbf{x}_* \triangleq D_\sigma(\mathbf{x}_* - \alpha \mathbf{P}_* \nabla f(\mathbf{x}_*))$  is upper bounded by

$$\|\mathbf{x}_{k+1} - \mathbf{x}_*\| \leq q^k \|\mathbf{x}_1 - \mathbf{x}_*\| + \frac{(1 + \epsilon)\alpha(\delta + 1 + \delta\theta_1\lambda_*)}{\delta\theta_1(1 - q)} R. \quad (15)$$

See Appendix C for the proof.

Since  $q < 1$ , when  $k \rightarrow \infty$ , the first term in (15) will vanish and the distance  $\|\mathbf{x}_k - \mathbf{x}_*\|$  is bounded by the second term. In our experiments, we found  $R = \|\nabla f(\mathbf{x}_1)\|$ , so a good initial value with small  $\|\nabla f(\mathbf{x}_1)\|$  can help to control the error. Moreover, one can control the accuracy of the error bound by setting

other parameters to a desired level. However, the algorithm may be very slow if we tune the parameters to enforce a small error bound. In practice, we experimentally identified that the excellent PSNR performance of Algorithm 3 can be achieved without tuning the value of the second term in (15), coinciding with the observation in [48, Thm. 1]. This observation indicates that the parameters which perform well in experiments may result in an extremely high upper bound, suggesting that further analysis could lead to tighter results.

#### IV. NUMERICAL EXPERIMENTS

This section studies the performance of P<sup>2</sup>nP with fixed and dynamic preconditioners for CS MRI reconstruction with spiral and radial sampling trajectories. For the fixed preconditioners, we examined (11) and the Chebyshev polynomials, and investigate the convergence of P<sup>2</sup>nP to verify our analyses. Lastly, we compared the performance of normalization-equivariant and ordinary denoisers. We first present our experimental and algorithmic settings and then show the reconstruction results.

*Experimental Settings:* The brain and knee MRI images were used to study the performance of P<sup>2</sup>nP. For the brain images, we adopted the dataset used in [16] that has 360 images in the training dataset and 164 images in the testing dataset. For the knee images, we used the NYU fastMRI [67] multi-coil knee dataset, where we first applied the ESPIRiT algorithm [68] to recover the complex-valued images and then took 700 and 6 slices from the training and testing datasets, respectively. We then cropped and resized all brain and knee images to  $256 \times 256$ . The noisy images were obtained by adding i.i.d. Gaussian noise with variance  $0.1/255$ . For the denoiser, we used the DRUNet [31] and trained both normalization-equivariant [66] and ordinary denoisers for brain and knee images. In the training procedure, both normalization-equivariant and ordinary denoisers used a batch size of 16 with the mean squared error as the loss function. The ADAM algorithm is used as the optimizer with a learning rate  $10^{-4}$  [69], and a total of  $3 \times 10^3$  iterations are performed to train the denoisers. We trained different denoisers for brain and knee images, but we used the same denoiser for different acquisitions.

We took six images from the brain and knee testing datasets as the ground truth and scaled the maximum magnitude of images to be one. Fig. 1 shows the magnitude of the six complex-valued ground truth images. For the sampling trajectories, we used 6 interleaves, 1688 readout points, and 32 coils (respectively, 21 spokes with golden angle rotation, 1024 readout points, and 32 coils) for the spiral (respectively, radial) trajectory to specify the forward model  $\mathbf{A}$ . Fig. 2 illustrates the trajectories used in this paper. We applied the related forward models to the ground truth images to generate the noiseless multi-coil k-space data and then added complex i.i.d Gaussian noise with mean zero and variance  $10^{-3}$  (respectively,  $3 \times 10^{-4}$ ) to all coils for the brain (respectively, knee) images to form the associated measurements  $\mathbf{y}$ . Section IV-C, IV-D examine the whole six brain test images with the spiral acquisition. The supplementary material provides additional experimental results for a Cartesian sampling trajectory and the way to compute the acceleration

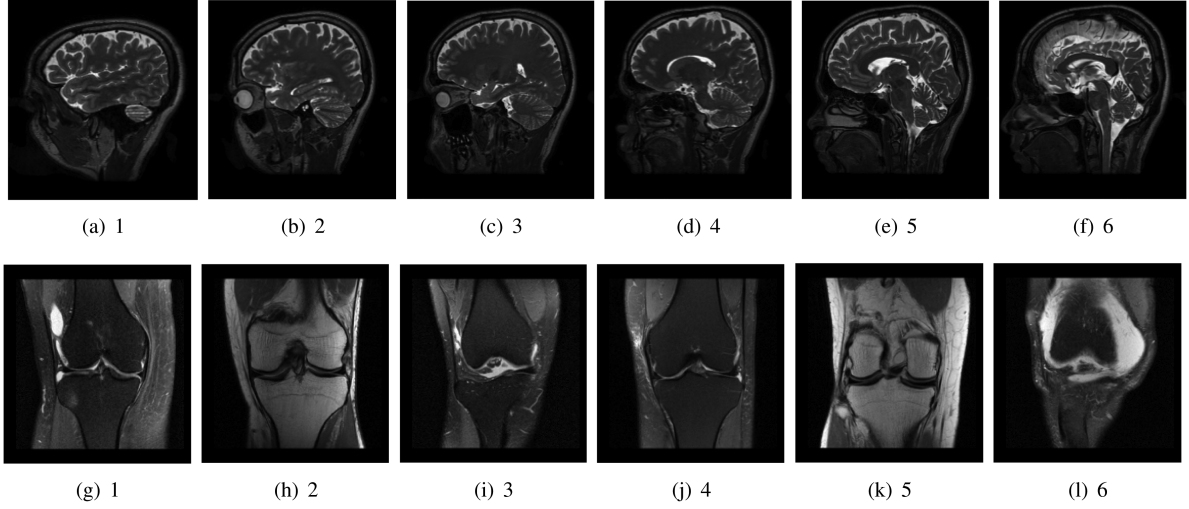


Fig. 1. The magnitude of the six brain and knee complex-valued ground truth images.

TABLE I  
SUMMARY OF THE ALGORITHMIC ROUTINES FOR PnP-ISTA/ADMM AND P<sup>2</sup>nP

PnP-ISTA	$\mathbf{x}_{k+1} = D_{\sigma}(\mathbf{x}_k - \alpha \nabla f(\mathbf{x}_k))$		
PnP-ADMM	$\mathbf{x}_k = \arg \min_{\mathbf{x}} f(\mathbf{x}) + \frac{1}{2} \ \mathbf{x} - \mathbf{z}_{k-1}\ _2^2$		
	$\tilde{\mathbf{z}}_k = D_{\sigma}(\mathbf{x}_k + \mathbf{z}_{k-1})$ $\mathbf{z}_k = \mathbf{z}_{k-1} + \mathbf{x}_k - \tilde{\mathbf{z}}_k$		
P <sup>2</sup> nP	$\mathbf{x}_{k+1} = D_{\sigma}(\mathbf{x}_k - \alpha P \nabla f(\mathbf{x}_k))$	-F-1	$P = 2 - \alpha \mathbf{A}^H \mathbf{A}$
		-F-Cheb	$P = 4 - \frac{10}{3} \alpha \mathbf{A}^H \mathbf{A}$
		-D	$P = \text{Algorithm 2}$

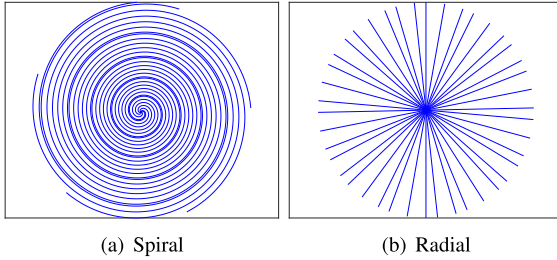


Fig. 2. The non-Cartesian k-space sampling trajectories used in this paper.

factor for spiral and radial acquisitions. All experiments were implemented in PyTorch [70] and run on NVIDIA GeForce RTX 3090.

**Algorithmic Settings:** The step-size  $\alpha$  was set to be  $1/\|\mathbf{A}^H \mathbf{A}\|_2$ , where the spectral norm was computed by the power method. For all experiments, we used  $\mathbf{A}^H \mathbf{y}$  as the initialization. For P<sup>2</sup>nP with fixed preconditioners, we studied two different strategies: one with  $\Gamma = 2$  such that  $P_1^F = 2 - \alpha \mathbf{A}^H \mathbf{A}$  through (11) dubbed “P<sup>2</sup>nP-F-1”, and the other with  $P_2^F = 4 - \frac{10}{3} \alpha \mathbf{A}^H \mathbf{A}$  through the Chebyshev polynomials dubbed “P<sup>2</sup>nP-F-Cheb”. P<sup>2</sup>nP with dynamic preconditioner is denoted by “P<sup>2</sup>nP-D”. Algorithm 2 used  $\delta = 10^{-8}$ ,  $\theta_1 = 2 \times 10^{-6}$ ,  $\theta_2 = 200$ . We mainly compared P<sup>2</sup>nP with PnP-ISTA and PnP-ADMM methods [36], [47] since those methods also have provable fixed-point convergence under mild assumptions. Table I outlines the algorithmic routines for all comparison methods.

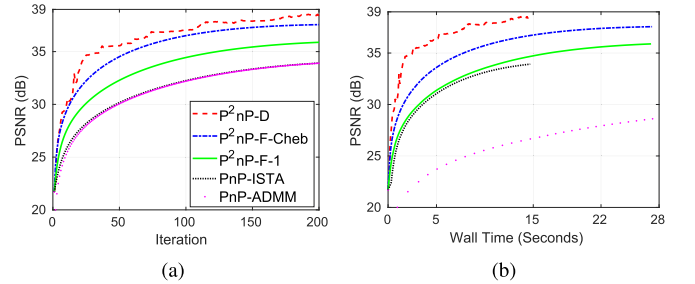


Fig. 3. PSNR values versus iteration and wall (GPU) time on the brain 1 image with spiral acquisition.

We ran all algorithms for 200 iterations except in Section IV-C, where we ran 500 iterations to examine the convergence properties. Moreover, we used the normalization-equivariant denoiser in all the experiments except in Section IV-D, where we compared the difference between normalization-equivariant and ordinary denoisers.

#### A. Spiral Acquisition Reconstruction

Fig. 3 presents the performance of P<sup>2</sup>nP for the brain 1 image. Fig. 3(a) shows that P<sup>2</sup>nP with fixed and dynamic preconditioners converged faster than PnP-ISTA and PnP-ADMM in terms of iteration number, demonstrating the effectiveness of using preconditioners and the fastest convergence speed of Algorithm 3. Moreover, P<sup>2</sup>nP-F-Cheb was faster than P<sup>2</sup>nP-F-1,

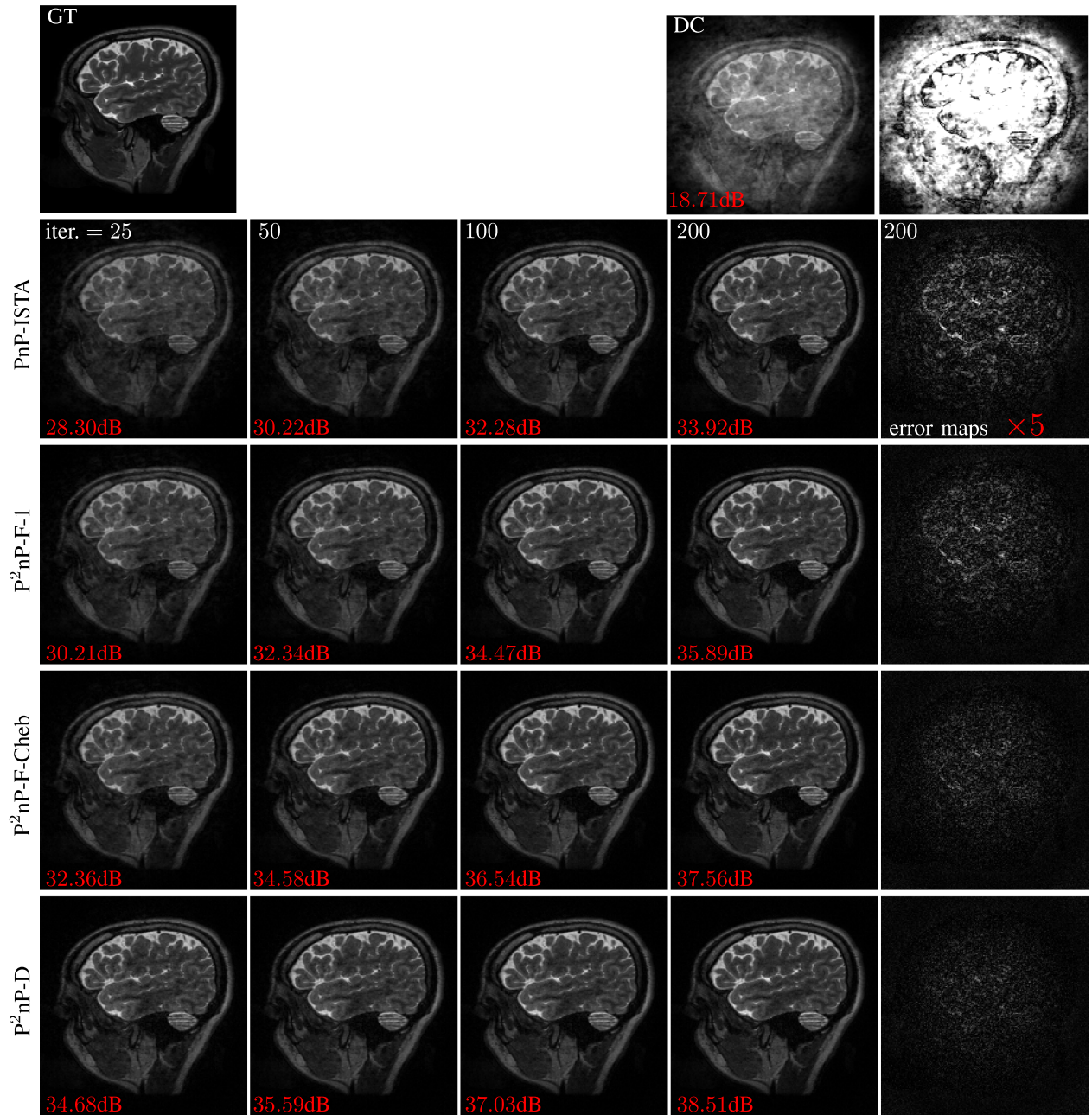


Fig. 4. The reconstructed brain 1 images at 25, 50, 100, 200th iteration with spiral acquisition. The PSNR value is labeled in the left bottom corner of each image. The fifth column shown the error maps ( $\times 5$ ) of the reconstructed images at 200th iteration. We omitted the PnP-ADMM results since it is similar to PnP-ISTA. DC represents the density compensation based reconstruction. Acceleration factor is  $\approx 130$ .

coinciding with our expectation since  $P^2nP$ -F-Cheb is optimal. Fig. 3(b) displays the PSNR values versus wall time.  $P^2nP$ -D was the most appealing algorithm in this experiment because it converged faster (both in terms of iterations run and wall-time) than the other algorithms. Fig. 3(b) shows that  $P^2nP$ -F-1 and  $P^2nP$ -F-Cheb needed almost twice as much wall time as ISTA. This was because  $P^2nP$ -F-1 and  $P^2nP$ -F-Cheb executed  $Ax$  twice as often as PnP-ISTA. Nevertheless,  $P^2nP$ -F-1 and  $P^2nP$ -F-Cheb still converged faster than PnP-ISTA in terms of wall time. PnP-ADMM was the slowest algorithm across

all methods in terms of wall time because ADMM needed to solve a least-squares problem at each iteration, requiring executing  $Ax$  many times. Fig. 4 depicts the reconstructed images at the 25, 50, 100, and 200th iterations and the associated error maps at the 200th iteration and shows that  $P^2nP$  achieved a higher PSNR and clearer image reconstruction than PnP-ISTA with the same number of iterations. Moreover, Fig. 4 also includes the density compensation based reconstruction [71], clearly demonstrating the advantage of the PnP framework.



TABLE II  
PSNR PERFORMANCE OF EACH METHOD FOR RECONSTRUCTING 5 OTHER BRAIN TEST IMAGES WITH SPIRAL ACQUISITION

Index	2			3			4			5			6		
Methods	PSNR↑	iter.↓	sec.↓	PSNR↑	iter.↓	sec.↓	PSNR↑	iter.↓	sec.↓	PSNR↑	iter.↓	sec.↓	PSNR↑	iter.↓	sec.↓
PnP-ADMM	34.29	200	147.6	34.51	200	152.2	34.09	200	149.3	32.78	200	148.7	31.70	200	148.8
PnP-ISTA	34.30	200	14.6	34.52	200	14.4	34.10	200	14.9	32.79	200	14.4	31.71	200	14.5
P <sup>2</sup> nP-F-1	34.32	76	10.4	34.53	76	10.5	34.10	76	10.7	32.82	80	10.9	31.72	79	10.8
	36.00	200	27.4	36.17	200	27.5	35.91	200	28.2	34.85	200	27.1	33.74	200	27.4
P <sup>2</sup> nP-F-Cheb	34.33	37	5.1	34.55	37	5.2	34.13	37	5.2	32.84	39	5.3	31.72	38	5.2
	37.37	200	27.4	37.57	200	28.1	37.38	200	27.1	36.65	200	27.1	35.9	200	27.4
P <sup>2</sup> nP-D	34.41	<b>28</b>	<b>2.0</b>	34.59	<b>24</b>	<b>1.8</b>	34.11	<b>20</b>	<b>1.5</b>	32.87	<b>19</b>	<b>1.4</b>	31.74	<b>33</b>	<b>2.4</b>
	<b>37.40</b>	200	14.5	<b>38.02</b>	200	14.9	<b>38.01</b>	200	14.8	<b>37.43</b>	200	14.4	<b>36.53</b>	200	14.5

For PnP-ADMM, we showed the maximal PSNR and the associated number of iterations, and wall time that is used as the benchmark. For other methods on each test image, the second column of the first row represents the first iteration that exceeded PnP-ADMM PSNR. The related first and third columns are the associated PSNR and wall time, respectively. The bold digits denote the number of iterations and wall time of the fastest algorithm that first exceeded PnP-ADMM. The second row shows the PSNR and wall time at the 200th iteration. The blue digits denote the highest PSNR at the 200th iteration.

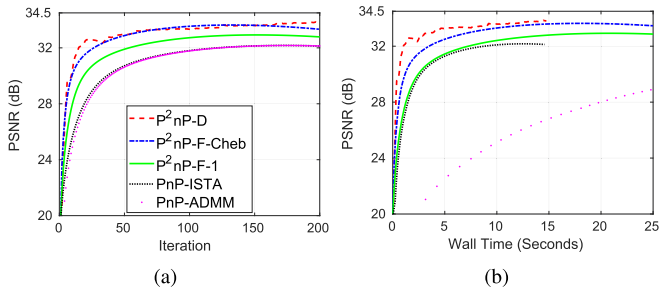


Fig. 5. PSNR values versus iteration and wall (GPU) time on the knee 1 image with radial acquisition.

Table II presents the performance of different methods for the reconstruction of other brain test images, where we used the highest PSNR obtained by PnP-ADMM as a benchmark. Firstly, PnP-ISTA had similar PSNR as PnP-ADMM. Secondly, from the first row of each method in Table II, P<sup>2</sup>nP took fewer iterations and less wall time than PnP-ISTA and PnP-ADMM, to reach a similar PSNR. This consistent advantage of P<sup>2</sup>nP demonstrates the effectiveness of using preconditioners. Moreover, P<sup>2</sup>nP-D was almost 70 times faster than PnP-ADMM and 7~10 times faster than PnP-ISTA to reach a similar PSNR, illustrating the advantages and efficiency of using dynamic preconditioners. The second row of each method in Table II indicated that P<sup>2</sup>nP-D also yielded the highest PSNR after running all scheduled iterations. Overall, in this experiment, P<sup>2</sup>nP-D was the fastest algorithm in terms of iteration and wall time. P<sup>2</sup>nP not only improved the numerical efficiency, but also yielded a higher PSNR. The supplementary material provides the results of knee images with spiral acquisition where we observed similar trends as the brain images.

### B. Radial Acquisition Reconstruction

Fig. 5 describes the performance of different methods for the reconstruction of knee 1 image. Consistent with the observations in the brain-image-based experiments in Section IV-A, P<sup>2</sup>nP converged faster than PnP-ADMM and PnP-ISTA in terms of iteration and wall time. Additionally, in this experiment, P<sup>2</sup>nP-D converged at a rate similar to P<sup>2</sup>nP-F-Cheb in terms of iterations run but P<sup>2</sup>nP-D was faster than P<sup>2</sup>nP-F-Cheb in

terms of wall time. Although the PSNR value of P<sup>2</sup>nP-F-Cheb dropped slightly at the end of the iterations, it was still faster than P<sup>2</sup>nP-F-1. Similar to Section IV-A, P<sup>2</sup>nP-D was the fastest algorithm. Fig. 6 shows the reconstructed images for different methods. Here the reconstructed images had some aliasing artifacts because we only used 21 spokes. The supplementary material shows the reconstruction of knee 1 image with a spiral acquisition that yields much clearer reconstructed images. Moreover, we also tested on the radial acquisition with 55 spokes and the results are summarized in the supplementary material where we saw the artifacts were significantly reduced. P<sup>2</sup>nP yielded only 1 dB higher PSNR than PnP-ISTA instead of  $\approx 4.5$  dB for a spiral acquisition. The main reason is due to the effectiveness of the preconditioner in this acquisition. The supplementary material shows that P<sup>2</sup>nP performs significantly better than PnP-ISTA on the same knee image with spiral and 55-spoke radial acquisitions.

Similar to Table II, we also tested the other knee images and Table III summarizes the results. Here, PnP-ISTA had very slightly worse PSNR than PnP-ADMM for some test images. Moreover, for some images (i.e., 5 and 6), PnP-ISTA images has slightly lower PSNR towards the end of the iterations. This is expected because fixed-point convergence cannot guarantee the quality of the reconstructed image. Similar to Section IV-A, P<sup>2</sup>nP was faster than PnP-ISTA and PnP-ADMM, and P<sup>2</sup>nP-D was the fastest algorithm to exceed the performance of ADMM. However, for some images (i.e., 2, 3, 4), P<sup>2</sup>nP-F-Cheb achieved the highest PSNR. The supplement summarizes the results of brain images with radial acquisition which showed similar trends.

### C. Convergence Validation

This part studied the convergence of P<sup>2</sup>nP. Denote by  $E(\mathbf{x}_k) = \|\mathbf{x}_k - \mathbf{D}_\sigma(\mathbf{x}_k - \alpha \mathbf{P} \nabla f(\mathbf{x}_k))\|_2^2 / \|\mathbf{x}_1\|_2^2$ , so  $E(\mathbf{x}_k) \rightarrow 0$  if  $\mathbf{x}_k \rightarrow \mathbf{x}_*$  where  $\mathbf{x}_*$  represents the fixed-point. Fig. 7(a) shows the value of  $E(\mathbf{x}_k)$  versus iteration for PnP-ISTA, P<sup>2</sup>nP-F-1, and P<sup>2</sup>nP-F-Cheb. We saw that  $E(\mathbf{x}_k) \rightarrow 0$  for all tested methods and P<sup>2</sup>nP converged faster than PnP-ISTA. Moreover, we noticed that the shaded region was very small, indicating that the convergence properties were similar across different test images. Fig. 7(b) presents the value of  $\|\nabla f(\mathbf{x}_k)\|_2^2 / \|\mathbf{x}_1\|_2^2$

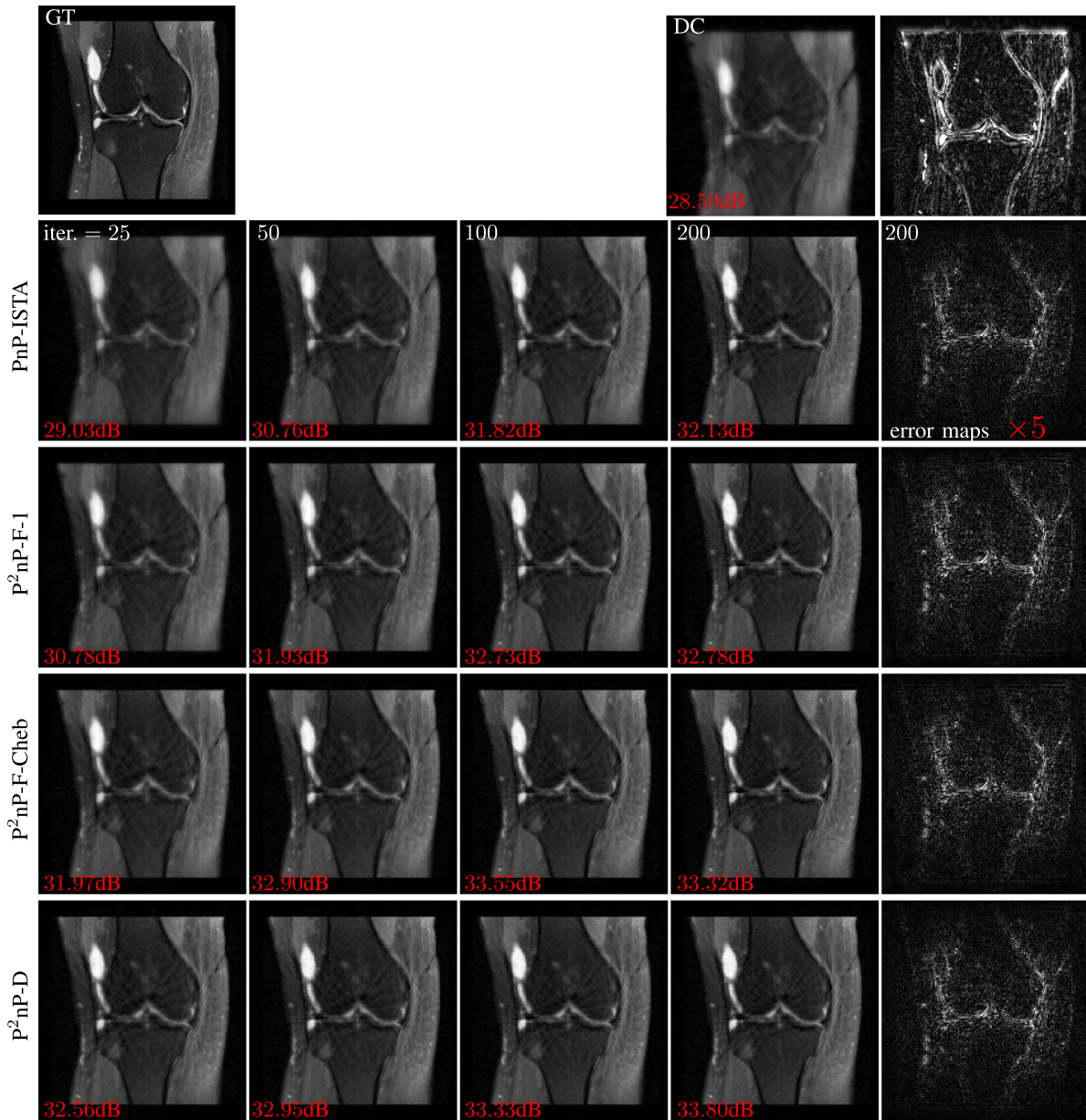


Fig. 6. The reconstructed knee 1 images at 25, 50, 100, 200th iteration with radial acquisition. The PSNR value is labeled in the left bottom corner of each image. The fifth column shown the error maps ( $\times 5$ ) of the reconstructed images at 200th iteration. DC represents the density compensation based reconstruction. Acceleration factor is  $\approx 12$ .

TABLE III  
PERFORMANCE OF EACH METHOD ON THE RECONSTRUCTION OF OTHER KNEE TEST IMAGES WITH THE RADIAL ACQUISITION

Index	2			3			4			5			6		
Methods	PSNR $\uparrow$	iter. $\downarrow$	sec. $\downarrow$	PSNR $\uparrow$	iter. $\downarrow$	sec. $\downarrow$	PSNR $\uparrow$	iter. $\downarrow$	sec. $\downarrow$	PSNR $\uparrow$	iter. $\downarrow$	sec. $\downarrow$	PSNR $\uparrow$	iter. $\downarrow$	sec. $\downarrow$
PnP-ADMM	35.30	153	106.1	31.17	196	145.4	33.64	176	122	31.36	160	120.4	34.77	150	111.5
PnP-ISTA	—	—	—	31.17	200	14.5	—	—	—	31.36	154	11.3	34.77	145	10.4
	35.16	200	14.5	31.17	200	14.5	33.60	200	14.6	31.21	200	14.6	34.55	200	14.3
P²nP-F-1	35.30	54	7.4	31.18	72	9.8	33.64	62	8.5	31.38	57	7.9	34.77	52	7.1
	35.62	200	27.3	31.98	200	27.4	34.16	200	27.5	31.89	200	27.6	35.13	200	27.3
P²nP-F-Cheb	35.30	26	3.5	31.18	35	4.8	33.64	30	4.1	31.42	28	3.8	34.77	25	3.4
	36.01	200	27.3	32.67	200	27.6	34.62	200	27.5	32.45	200	27.7	35.54	200	27.4
P²nP-D	35.38	19	1.4	31.20	20	1.5	33.76	19	1.4	31.76	17	1.3	34.85	20	1.5
	35.95	200	14.6	32.56	200	14.6	34.27	200	14.7	33.11	200	14.6	36.06	200	14.5

The definition of digits here is identical to table ii. “—” means the method cannot reach higher PSNR than ADMM in 200 iterations.

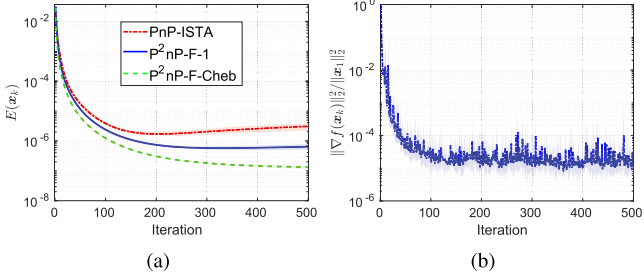


Fig. 7. (a) Numerical test of the fixed-point convergence of PnP-ISTA, P²nP-F-1, and P²nP-F-Cheb. (b)  $\|\nabla f(x_k)\|$  values versus iteration for P²nP-D.

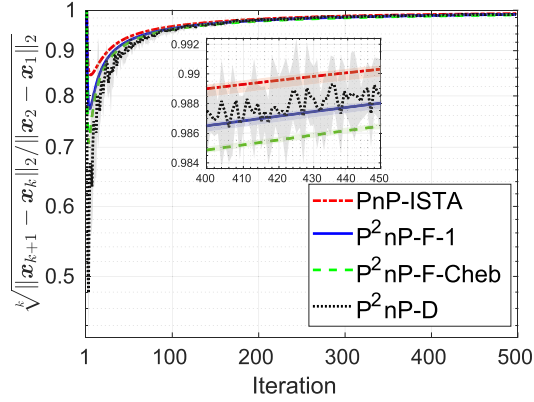


Fig. 8. Numerical test of the convergence rates of PnP-ISTA, P²nP-F-1, P²nP-F-Cheb, and P²nP-D averaged on six brain test images with spiral acquisition. The shaded region of each curve represents the bound of the  $\rho_k$  across all brain test images.

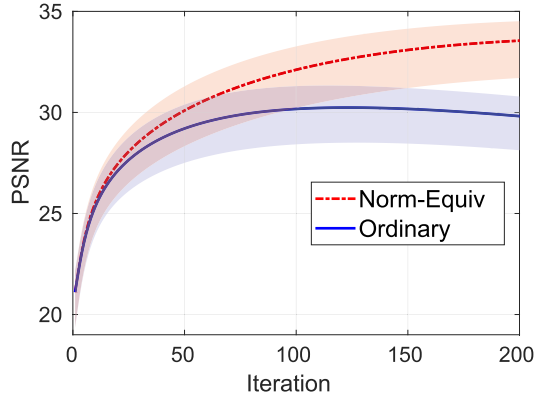


Fig. 9. Averaged PSNR values versus iteration for PnP-ISTA with the normalization-equivariant and ordinary denoisers. The shaded region of each curve represents the bound of the PSNR across all brain test images with spiral acquisition.

for P²nP-D. It can be seen that  $\|\nabla f(x_k)\|_2^2 / \|x_1\|_2^2$  attained its maximal value at the first iteration and tended to zero, indicating  $R = \|\nabla f(x_1)\|$ .

We used  $\rho_k \triangleq (\|x_{k+1} - x_k\|_2 / \|x_2 - x_1\|_2)^{1/k}$  for  $k \geq 1$  to measure the empirical convergence rate, where  $\rho_k < 1$  means  $\|x_{k+1} - x_k\|_2$  tends to zero. Fig. 8 shows  $\rho_k$  versus iteration for PnP-ISTA and P²nP with fixed and dynamic preconditioners.

#### D. Comparing Ordinary and Normalization-Equivariant Denoisers

Fig. 9 compares the PSNR values of PnP-ISTA on the brain test images with ordinary and normalization-equivariant (“Norm-Equiv”) denoisers. PnP-ISTA with the “Norm-Equiv” denoiser had significantly higher PSNR than the ordinary denoiser and was more robust to running more iterations. This is due to the fact that, compared with ordinary denoisers, the “Norm-Equiv” ones are more robust and adaptive to noise level changes [66]. Ref. [65] illustrated the importance of choosing the proper noise level for the denoiser in the PnP framework. Our experiments demonstrated that the “Norm-Equiv” denoiser can be an appealing choice to automatically address this concern.

#### V. CONCLUSION

This paper proposes a preconditioned PnP method (P²nP) with provable convergence. We showed that P²nP can significantly accelerate the convergence speed of its no-preconditioner-based counterparts in the reconstruction of CS MRI. Since the forward model in CS MRI is patient-specific, we proposed two different strategies (fixed and dynamic) for computing preconditioners efficiently. The corresponding numerical experiments demonstrated effectiveness and efficiency of using preconditioners for accelerating MRI reconstruction with spiral and radial trajectories. Moreover, we introduced the use of normalization-equivariant denoisers in the PnP framework. By providing a self-adaptive way to address the noise-level tuning problem in PnP, using normalization-equivariant denoisers not only allows for “solving” (13) but also significantly improves the reconstruction PSNR.

The dynamic preconditioner does not require any explicit knowledge of the forward model, making it a potentially efficient solver for addressing nonlinear inverse problems with PnP, e.g., full waveform inversion [72]. Another promising direction is to unroll P²nP, which may yield a more efficient network than others for image reconstruction. We leave the further investigation of these topics to future work.

#### APPENDIX A PROOF OF THEOREM 1

The updating scheme in Algorithm 1 can be written as

$$x_{k+1} = T(x_k) \text{ with } T(x) \triangleq D_\sigma(x - \alpha P \nabla f(x)).$$

From the Banach fixed-point theorem, Algorithm 1 is guaranteed to converge to a fixed-point if the mapping  $T(x)$  is contractive. So if  $x_*$  is a fixed-point of  $T(x)$  such that  $x_* = T(x_*)$ , then conditioned on Assumption 1, we have

$$\begin{aligned} \|x_{k+1} - x_*\| &= \|T(x_k) - T(x_*)\| \\ &= \|D_\sigma(x_k - \alpha P \nabla f(x_k)) - D_\sigma(x_* - \alpha P \nabla f(x_*))\| \\ &\leq (1 + \epsilon) \|x_k - x_*\| \\ &\quad - \alpha P (\nabla f(x_k) - \nabla f(x_*)) \| \end{aligned}$$



$$\begin{aligned}
&= (1 + \epsilon) \| \mathbf{x}_k - \mathbf{x}_* - \alpha \mathbf{P} \mathbf{A}^H \mathbf{A} (\mathbf{x}_k - \mathbf{x}_*) \| \\
&\leq (1 + \epsilon) \| \mathbf{I} - \alpha \mathbf{P} \mathbf{A}^H \mathbf{A} \| \cdot \| \mathbf{x}_k - \mathbf{x}_* \|.
\end{aligned}$$

Clearly, if  $(1 + \epsilon)\rho(\mathbf{I} - \alpha \mathbf{P} \mathbf{A}^H \mathbf{A}) < 1$ , the iterates from Algorithm 1 are guaranteed to converge and the convergence rate is at most  $(1 + \epsilon)\rho(\mathbf{I} - \alpha \mathbf{P} \mathbf{A}^H \mathbf{A})$ .

#### APPENDIX B PROOF OF LEMMA 1

The proof that  $\tau_k$  and  $\langle \mathbf{s}_k, \mathbf{v}_k \rangle$  are real comes from [7, Observation 1]. Using the fact  $\nabla f(\mathbf{x}) = \mathbf{A}^H(\mathbf{A}\mathbf{x} - \mathbf{y})$  and following the deduction in [64, Lemma A.3 and Theorem 4.2], we can easily prove that  $\langle \mathbf{s}_k - \tau_k \mathbf{v}_k, \mathbf{v}_k \rangle$  is nonnegative and the bounds of  $\tau_k$  and  $\mathbf{P}_k$  so that we omit the details here. The bound of  $\mathbf{P}_k$  obtained here is much tighter than the one shown in [64, Thm. 4.2] that bound depends on the image size while ours does not. If one considers general nonlinear inverse problems in the complex plane, then Lemma 1 in general is invalid.

#### APPENDIX C PROOF OF THEOREM 2

The updating scheme of Algorithm 3 at the  $k$ th iteration can be represented as

$$\mathbf{x}_{k+1} = T_k(\mathbf{x}_k) \text{ with } T_k(\mathbf{x}) = \mathbf{D}_\sigma(\mathbf{x} - \alpha \mathbf{P}_k \nabla f(\mathbf{x})).$$

Then we have

$$\begin{aligned}
\| \mathbf{x}_{k+1} - \mathbf{x}_* \| &\leq (1 + \epsilon) \| \mathbf{x}_k - \mathbf{x}_* \| \\
&\quad - \alpha (\mathbf{P}_k \nabla f(\mathbf{x}_k) - \mathbf{P}_* \nabla f(\mathbf{x}_*)) \| \\
&\leq (1 + \epsilon) \| \mathbf{I} - \alpha \mathbf{P}_* \mathbf{A}^H \mathbf{A} \| \| \mathbf{x}_k - \mathbf{x}_* \| \\
&\quad + (1 + \epsilon) \alpha \| (\mathbf{P}_k - \mathbf{P}_*) \nabla f(\mathbf{x}_k) \| \\
&\leq q^k \| \mathbf{x}_1 - \mathbf{x}_* \| \\
&\quad + (1 + \epsilon) \alpha \sum_{m=1}^k q^{k-m} \| (\mathbf{P}_m \\
&\quad \quad - \mathbf{P}_*) \nabla f(\mathbf{x}_m) \| \\
&\leq q^k \| \mathbf{x}_1 - \mathbf{x}_* \| \\
&\quad + (1 + \epsilon) \alpha R \beta \sum_{m=1}^k q^{k-m} \\
&\leq q^k \| \mathbf{x}_1 - \mathbf{x}_* \| + (1 + \epsilon) \frac{\alpha R \beta}{1 - q}
\end{aligned}$$

where  $q \triangleq [(1 + \epsilon)\rho(\mathbf{I} - \alpha \mathbf{P}_* \mathbf{A}^H \mathbf{A})]$  and  $\beta \triangleq \max_{m \leq k} \| \mathbf{P}_m - \mathbf{P}_* \| \leq \max_{m \leq k} \| \mathbf{P}_m \| + \| \mathbf{P}_* \| \leq \frac{\delta+1}{\delta\theta_1} + \lambda_*$ . The first inequality is the result of Assumption 1. The second inequality derives from the triangle and Cauchy-Schwarz inequalities. The third inequality is obtained by applying first and second inequalities recursively. The fourth inequality comes from the definitions of  $R$  and  $\beta$ .

#### REFERENCES

- [1] 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.
- [2] M. A. Griswold et al., "Generalized autocalibrating partially parallel acquisitions (GRAPPA)," *Magn. Reson. Med.*, vol. 47, no. 6, pp. 1202–1210, 2002.
- [3] A. Deshmane, V. Gulani, M. A. Griswold, and N. Seiberlich, "Parallel MR imaging," *J. Magn. Reson. Imag.*, vol. 36, no. 1, pp. 55–72, 2012.
- [4] 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.
- [5] 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.
- [6] 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.
- [7] 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.
- [8] M. Guerquin-Kern, M. Haberland, 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] M. V. Zibetti, E. S. Helou, R. R. Regatte, and G. T. Herman, "Monotone FISTA with variable acceleration for compressed sensing magnetic resonance imaging," *IEEE Trans. Comput. Imag.*, vol. 5, no. 1, pp. 109–119, Mar. 2019.
- [10] 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.
- [11] S. Ravishanker 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.
- [12] 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.
- [13] J. A. Fessler, "Model-based image reconstruction for MRI," *IEEE Signal Process. Mag.*, vol. 27, no. 4, pp. 81–89, Jul. 2010.
- [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. Wang et al., "Accelerating magnetic resonance imaging via deep learning," in *Proc. IEEE 13th Int. Symp. Biomed. Imag.*, 2016, pp. 514–517.
- [16] 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.
- [17] 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.
- [18] T. Chen et al., "Learning to optimize: A primer and a benchmark," *J. Mach. Learn. Res.*, vol. 23, no. 189, pp. 1–59, 2022.
- [19] 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.
- [20] Z. Wang et al., "One for multiple: Physics-informed synthetic data boosts generalizable deep learning for fast MRI reconstruction," 2023, *arXiv:2307.13220*.
- [21] Y. Song, L. Shen, L. Xing, and S. Ermon, "Solving inverse problems in medical imaging with score-based generative models," in *Proc. Int. Conf. Learn. Representations*, 2021.
- [22] H. Chung and J. C. Ye, "Score-based diffusion models for accelerated MRI," *Med. Image Anal.*, vol. 80, 2022, Art. no. 102479.
- [23] 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.
- [24] 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.
- [25] K. Zhang, W. Zuo, Y. Chen, D. Meng, and L. Zhang, "Beyond a gaussian denoiser: Residual learning of deep CNN for image denoising," *IEEE Trans. Image Process.*, vol. 26, no. 7, pp. 3142–3155, Jul. 2017.
- [26] 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.

- [27] S. Ono, "Primal-dual plug-and-play image restoration," *IEEE Signal Process. Lett.*, vol. 24, no. 8, pp. 1108–1112, Aug. 2017.
- [28] 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.
- [29] 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.
- [30] W. Dong, P. Wang, W. Yin, G. Shi, F. Wu, and X. Lu, "Denoising prior driven deep neural network for image restoration," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 41, no. 10, pp. 2305–2318, Oct. 2019.
- [31] K. Zhang, Y. Li, W. Zuo, L. Zhang, L. V. 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.
- [32] S. Shoushtari, J. Liu, E. P. Chandler, M. S. Asif, and U. S. Kamilov, "Prior mismatch and adaptation in pnp-admm with a nonconvex convergence analysis," in *Proc. Int. Conf. Mach. Learn.*, 2024.
- [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] N. Parikh et al., "Proximal algorithms," *Foundations Trends Optim.*, vol. 1, no. 3, pp. 127–239, 2014.
- [35] 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.
- [36] 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.
- [37] S. Boyd et al., "Distributed optimization and statistical learning via the alternating direction method of multipliers," *Foundations Trends Mach. Learn.*, vol. 3, no. 1, pp. 1–122, 2011.
- [38] K. Zhang, W. Zuo, S. Gu, and L. Zhang, "Learning deep CNN denoiser prior for image restoration," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2017, pp. 3929–3938.
- [39] 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.
- [40] T. Hong, Y. Romano, and M. Elad, "Acceleration of RED via vector extrapolation," *J. Vis. Commun. Image Representation*, vol. 63, 2019, Art. no. 102575.
- [41] 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.
- [42] H. Y. Tan, S. Mukherjee, J. Tang, and C.-B. Schönlieb, "Provably convergent plug-and-play quasi-newton methods," *SIAM J. Imag. Sci.*, vol. 17, no. 2, pp. 785–819, 2024.
- [43] M. L. Pendu and C. Guillemot, "Preconditioned plug-and-play ADMM with locally adjustable denoiser for image restoration," *SIAM J. Imag. Sci.*, vol. 16, no. 1, pp. 393–422, 2023.
- [44] A. M. Teodoro, J. M. Bioucas-Dias, and M. A. T. Figueiredo, "A convergent image fusion algorithm using scene-adapted Gaussian-mixture-based denoising," *IEEE Trans. Image Process.*, vol. 28, no. 1, pp. 451–463, Jan. 2019.
- [45] R. G. Gavaskar, C. D. Athalye, and K. N. Chaudhury, "On plug-and-play regularization using linear denoisers," *IEEE Trans. Image Process.*, vol. 30, pp. 4802–4813, 2021.
- [46] Y. Sun, B. Wohlberg, and U. S. Kamilov, "An online plug-and-play algorithm for regularized image reconstruction," *IEEE Trans. Comput. Imag.*, vol. 5, no. 3, pp. 395–408, Sep. 2019.
- [47] 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.
- [48] Y. Sun, Z. Wu, X. Xu, B. Wohlberg, and U. S. Kamilov, "Scalable plug-and-play ADMM with convergence guarantees," *IEEE Trans. Comput. Imag.*, vol. 7, pp. 849–863, 2021.
- [49] 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.
- [50] X. Xu, Y. Sun, J. Liu, B. Wohlberg, and U. S. Kamilov, "Provable convergence of plug-and-play priors with MMSE denoisers," *IEEE Signal Process. Lett.*, vol. 27, pp. 1280–1284, 2020.
- [51] J. Liu, S. Asif, B. Wohlberg, and U. Kamilov, "Recovery analysis for plug-and-play priors using the restricted Eigenvalue condition," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 34, 2021, pp. 5921–5933.
- [52] R. Cohen, M. Elad, and P. Milanfar, "Regularization by denoising via fixed-point projection (RED-PRO)," *SIAM J. Imag. Sci.*, vol. 14, no. 3, pp. 1374–1406, 2021.
- [53] B. Iskender, M. L. Klasky, and Y. Bresler, "RED-PSM: Regularization by denoising of factorized low rank models for dynamic imaging," *IEEE Trans. Comput. Imag.*, vol. 10, pp. 832–847, 2024.
- [54] 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.
- [55] Z. Aminifard and S. Babaie-Kafaki, "An approximate Newton-type proximal method using symmetric rank-one updating formula for minimizing the nonsmooth composite functions," *Optim. Methods Softw.*, vol. 38, no. 3, pp. 529–542, 2023.
- [56] M. J. Grote and T. Huckle, "Parallel preconditioning with sparse approximate inverses," *SIAM J. Sci. Comput.*, vol. 18, no. 3, pp. 838–853, 1997.
- [57] N. I. Gould and J. A. Scott, "Sparse approximate-inverse preconditioners using norm-minimization techniques," *SIAM J. Sci. Comput.*, vol. 19, no. 2, pp. 605–625, 1998.
- [58] O. G. Johnson, C. A. Micchelli, and G. Paul, "Polynomial preconditioners for conjugate gradient calculations," *SIAM J. Numer. Anal.*, vol. 20, no. 2, pp. 362–376, 1983.
- [59] M. Zulfikar A. Bhotto, M. O. Ahmad, and M. Swamy, "An improved fast iterative shrinkage thresholding algorithm for image deblurring," *SIAM J. Imag. Sci.*, vol. 8, no. 3, pp. 1640–1657, 2015.
- [60] S. S. Iyer et al., "Polynomial preconditioners for regularized linear inverse problems," *SIAM J. Imag. Sci.*, vol. 17, no. 1, pp. 116–146, 2024.
- [61] J. Nocedal and S. J. Wright, *Numerical Optimization*. Berlin Germany: Springer, 2006.
- [62] M. Osborne and L. Sun, "A new approach to symmetric rank-one updating," *IMA J. Numer. Anal.*, vol. 19, no. 4, pp. 497–507, 1999.
- [63] F. Curtis, "A self-correcting variable-metric algorithm for stochastic optimization," in *Proc. Int. Conf. Mach. Learn.*, 2016, pp. 632–641.
- [64] X. Wang, X. Wang, and Y.-X. Yuan, "Stochastic proximal quasi-newton methods for non-convex composite optimization," *Optim. Methods Softw.*, vol. 34, no. 5, pp. 922–948, 2019.
- [65] X. Xu, J. Liu, Y. Sun, B. Wohlberg, and U. S. Kamilov, "Boosting the performance of plug-and-play priors via denoiser scaling," in *Proc. 54th Asilomar Conf. Signals, Syst., Comput.*, 2020, pp. 1305–1312.
- [66] S. Herbreteau, E. Moebel, and C. Kervrann, "Normalization-equivariant neural networks with application to image denoising," in *Proc. Adv. Neural Inf. Process. Syst.*, 2024, vol. 36.
- [67] J. Zbontar et al., "fastMRI: An open dataset and benchmarks for accelerated MRI," 2018, *arXiv:1811.08839*.
- [68] 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.
- [69] D. Kingma and J. Ba, "Adam: A method for stochastic optimization," 2014, *arXiv:1412.6980*.
- [70] A. Paszke et al., "Pytorch: An imperative style, high-performance deep learning library," in *Proc. Adv. Neural Inf. Process. Syst.*, 2019, vol. 32.
- [71] J. G. Pipe and P. Menon, "Sampling density compensation in MRI: Rationale and an iterative numerical solution," *Magn. Reson. Med.*, vol. 41, no. 1, pp. 179–186, 1999.
- [72] L. Métivier, R. Brossier, J. Virieux, and S. Operto, "Full waveform inversion and the truncated newton method," *SIAM J. Sci. Comput.*, vol. 35, no. 2, pp. B401–B437, 2013.



**Tao Hong** (Member, IEEE) received the Ph.D. degree from the Henry and Marilyn Taub Faculty of Computer Science, the Technion Israel Institute of Technology, Haifa, Israel, in 2021. He is currently a Postdoctoral Fellow with the Functional MRI Lab, University of Michigan, Ann Arbor, MI, USA. His main research interests include numerical optimization and multigrid computational methods. He is particularly focused on designing efficient and provable computational methods to address the computational challenges arising in scientific computing, signal processing, machine learning, and computational imaging.

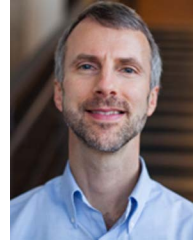


**Xiaojian Xu** (Member, IEEE) received the bachelor degree in communication and information engineering the master's degree in communication and information systems from the University of Electronic Science and Technology of China, in 2014 and 2017, respectively, and the Ph.D. degree in computer science with Washington University, St. Louis, MO, USA, in 2022. She joined the University of Michigan, Ann Arbor, MI, USA, as a Postdoctoral Research Fellow with the Department of Electrical Engineering and Computer Science. Her research interests include

computational imaging, optimization, inverse problems, computer vision, deep learning, machine learning, and signal processing.



**Jason Hu** (Graduate Student Member, IEEE) is currently working toward the Ph.D. degree in electrical and computer engineering with the University of Michigan, Ann Arbor, MI, USA. His research interests include developing generative AI and ML based algorithms for image processing. He primarily focuses on using score-based diffusion models coupled with deep learning and optimization techniques to solve computational imaging problems with applications in medicine, physics, and computer vision, as well as theoretical and foundational analysis of imaging.



**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. degree from Stanford University, Stanford, CA, USA, in 1986, and the M.S. degree in statistics from Stanford University in 1989. From 1985 to 1988 he was a National Science Foundation Graduate Fellow with Stanford, where he earned a Ph.D. in electrical engineering in 1990. He is currently the William L. Root Professor of EECS, University of Michigan, Ann Arbor, MI, USA. He was with the University of

Michigan, since then. From 1991 to 1992 he was a Department of Energy Alexander Hollaender Postdoctoral Fellow with 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 research interests include statistical aspects of imaging problems, and he has supervised doctoral research in PET, SPECT, X-ray CT, MRI, and optical imaging problems. He became a Fellow of the IEEE in 2006, for contributions to the theory and practice of image reconstruction. He was the recipient of the Francois Erbsmann Award for his IPMI93 presentation, the Edward Hoffman Medical Imaging Scientist Award in 2013, and an IEEE EMBS Technical Achievement Award in 2016. He was an Associate Editor for the IEEE TRANSACTIONS ON MEDICAL IMAGING, the IEEE SIGNAL PROCESSING LETTERS, the IEEE TRANSACTIONS ON IMAGE PROCESSING, the IEEE TRANSACTIONS ON COMPUTATIONAL IMAGING (T-CI), and is currently an Associate Editor for *SIAM Journal on Imaging Sciences* and a Senior AE for IEEE TRANSACTIONS ON COMPUTATIONAL IMAGING. He has chaired the IEEE T-MI Steering Committee and the ISBI Steering Committee. He was the 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. He received the 2023 Steven S. Attwood Award, the highest honor awarded to a faculty member by the College of Engineering.



# Supplementary Material: Provable Preconditioned Plug-and-Play Approach for Compressed Sensing MRI Reconstruction

Tao Hong, *Member, IEEE*, Xiaojian Xu, *Member, IEEE*, Jason Hu, *Student Member, IEEE*, and Jeffrey A. Fessler, *Fellow, IEEE*

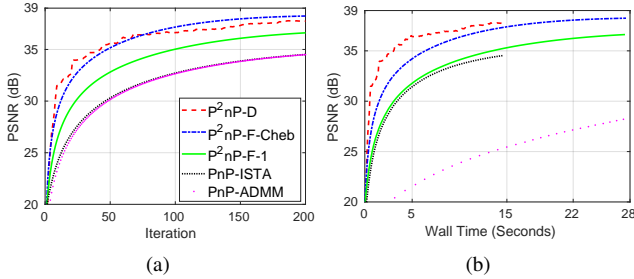


Fig. S.1. PSNR values versus iteration and wall (GPU) time for the knee 1 test image with spiral acquisition.

## S.I. SPIRAL ACQUISITION RECONSTRUCTION

Figure S.1 presents the PSNR values versus iteration and wall time for the knee 1 image with spiral acquisition. Clearly, we saw P²nP-F-1/Cheb and P²nP-D converged faster than PnP-ISTA and PnP-ADMM in terms of iteration number and wall time. Moreover, we observed P²nP-D was faster than P²nP-F-Cheb in terms of wall time, but P²nP-F-Cheb achieved higher PSNR eventually. In this setting, P²nP-F-1 was only slightly faster than PnP-ISTA and PnP-ADMM in terms of wall time. Figure S.2 shows the reconstructed images of each method. Obviously, P²nP required less iterations to achieve higher PSNR than PnP-ISTA and PnP-ADMM, illustrating the effectiveness of using preconditioners. Table S.I describes the results of other knee test images. Evidently, similar trend was observed for the brain test images.

## S.II. RADIAL ACQUISITION RECONSTRUCTION

Figure S.3 describes the PSNR values versus iteration and wall time of the brain 1 test image with radial acquisition. From Figure S.3, we observed P²nP converged faster than PnP-ISTA and PnP-ADMM. Moreover, P²nP-D converged similar to P²nP-F-Cheb in terms of iteration but P²nP-D was faster than P²nP-F-Cheb in terms of wall time. Table S.II shows the results of other test images, illustrating the effectiveness of adding preconditioners.

T. Hong is with the Department of Radiology, University of Michigan, Ann Arbor, MI 48109, USA (Email: tahong@umich.edu). TH was partly supported by National Institutes of Health grant R01NS112233.

J. Hu, X. Xu and J. Fessler are with the Department of Electrical and Computer Engineering, University of Michigan, Ann Arbor, MI 48109, USA (Email: {jashu, xjxu, fessler}@umich.edu).

## S.III. RADIAL ACQUISITION RECONSTRUCTION WITH 55 SPOKES

Figure S.5 describes the reconstruction of knee 1 test image with the 55 spokes radial acquisition. Clearly, the undesired artifacts in the 21 spokes radial acquisition disappeared and P²nP-D yielded the highest PSNR at 200th iteration.

## S.IV. CARTESIAN ACQUISITION RECONSTRUCTION

In this section, we present the results of using Cartesian sampling. Figure S.6 describes the downsampling mask and compares the PSNR values of different methods with respect to iteration and wall time for the knee 1 image. Moreover, we also tested the brain 1 image and the results are presented in Figure S.7. From Figures S.6 and S.7, we saw P²nP still outperforms than PnP-ISTA demonstrating the effectiveness of our approach.

## S.V. THE ACCELERATION FACTOR IN NON-CARTESIAN SAMPLING

In this section, we describe the method we used to compute the acceleration factor in non-Cartesian sampling. Without any confusion, the notation used in this section is self-contained.

For radial acquisition, the following formula is used to compute the acceleration factor (AF):

$$AF = \frac{N_{\text{spokes, fully sampled}}}{N_{\text{spokes, acquired}}}.$$

We approximate  $N_{\text{spokes, fully sampled}}$  through the number of phase encoding lines in Cartesian sampling.

For spiral acquisition, the number of interleaves required for full k-space coverage should ensure the spacing between adjacent spiral arms meeting the Nyquist criterion that

$$N_{\text{interleaves}} = \frac{2\pi k_{\text{max}}}{\Delta k},$$

where  $k_{\text{max}} = \frac{1}{2\Delta x}$  with  $\Delta x$  representing the image resolution and  $\Delta k \leq \frac{1}{\text{FOV}}$ . Then we have

$$N_{\text{interleaves}} = \frac{\pi \times \text{FOV}}{\Delta x} = \pi \times \text{Matrix Size}$$

resulting in

$$AF = \frac{N_{\text{interleaves}}}{N_{\text{acquired, interleaves}}}.$$

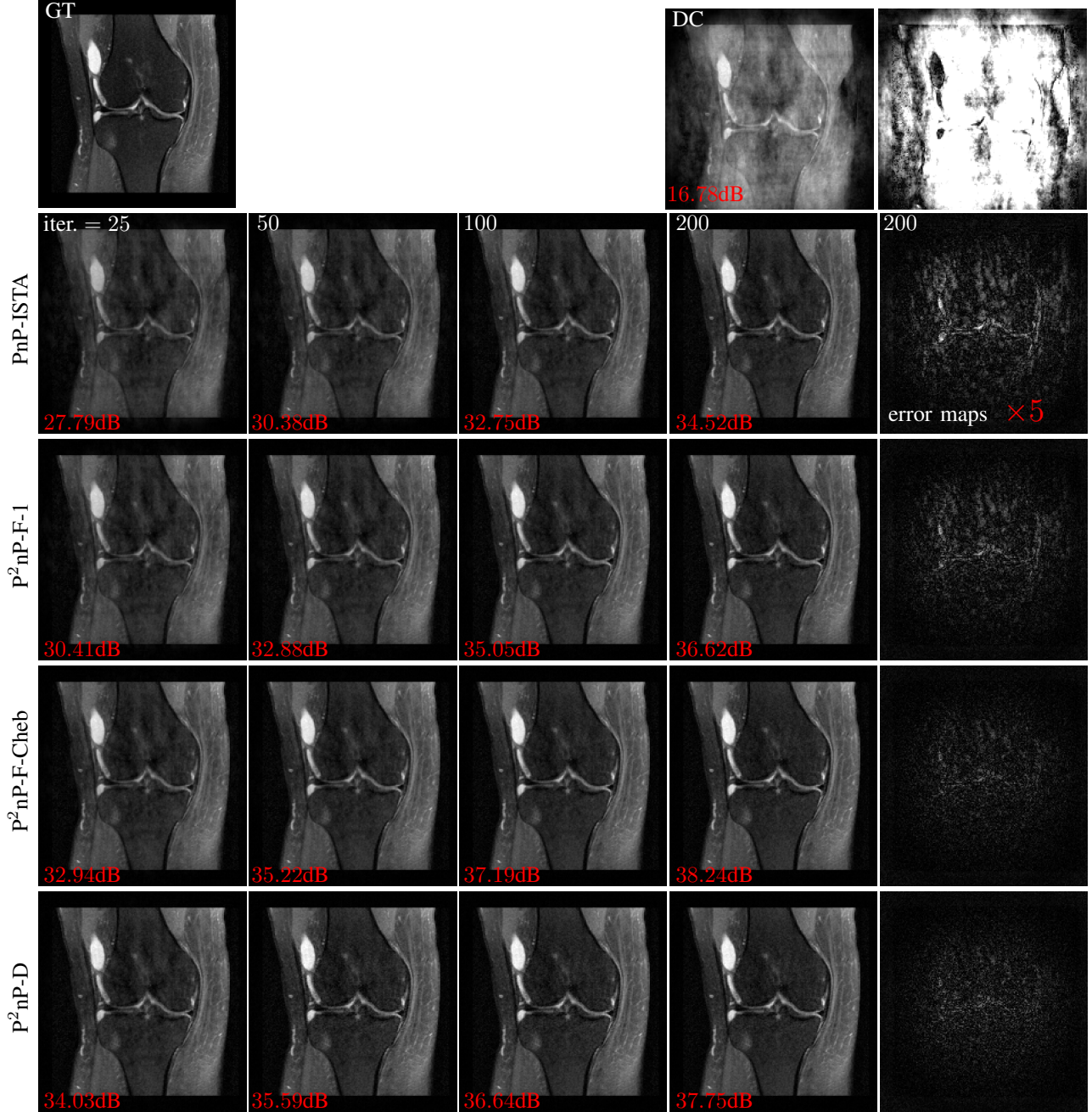


Fig. S.2. The reconstructed knee 1 images at 25, 50, 100, 200th iteration with spiral acquisition. The PSNR value is labeled in the left bottom corner of each image. The fifth column shown the error maps ( $\times 5$ ) of the reconstructed images at 200th iteration. We omitted the PnP-ADMM results since it is similar to PnP-ISTA. DC represents the density compensation based reconstruction. Acceleration factor is  $\approx 130$ .

TABLE S.I

PSNR PERFORMANCE OF EACH METHOD FOR RECONSTRUCTING 5 OTHER BRAIN TEST IMAGES WITH SPIRAL ACQUISITION. FOR PnP-ADMM, WE SHOWED THE MAXIMAL PSNR AND THE ASSOCIATED NUMBER OF ITERATIONS, AND WALL TIME THAT IS USED AS THE BENCHMARK. FOR OTHER METHODS ON EACH TEST IMAGE, THE SECOND COLUMN OF THE FIRST ROW REPRESENTS THE FIRST ITERATION THAT EXCEEDED PnP-ADMM PSNR. THE RELATED FIRST AND THIRD COLUMNS ARE THE ASSOCIATED PSNR AND WALL TIME, RESPECTIVELY. THE **BOLD** DIGITS DENOTE THE NUMBER OF ITERATIONS AND WALL TIME OF THE FASTEST ALGORITHM THAT FIRST EXCEEDED PnP-ADMM. THE SECOND ROW SHOWS THE PSNR AND WALL TIME AT THE 200TH ITERATION. THE **BLUE** DIGITS DENOTE THE HIGHEST PSNR 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.↓
PnP-ADMM	37.85	200	147.8	34.33	200	148	34.96	200	147.4	33.53	200	148.9	34.24	200	147.8
PnP-ISTA	37.85	200	14.6	34.33	199	14.6	34.96	199	14.5	33.54	199	14.4	34.25	199	14.5
	37.85	200	14.6	34.34	200	14.7	34.97	200	14.6	33.55	200	14.5	34.26	200	14.6
P <sup>2</sup> nP-F-1	37.85	75	10.3	34.36	85	11.7	34.96	84	11.5	33.57	87	11.8	34.27	90	12.3
	39.03	200	27.4	36.43	200	27.5	36.73	200	27.5	35.92	200	27.2	36.44	200	27.4
P <sup>2</sup> nP-F-Cheb	37.91	37	5	34.36	41	5.7	34.99	41	5.6	33.56	42	5.7	34.28	44	6.1
	39.66	200	27.5	38.03	200	27.5	<b>38.15</b>	200	27.5	37.77	200	27.4	38.14	200	27.5
P <sup>2</sup> nP-D	38.34	<b>36</b>	<b>2.6</b>	34.33	<b>21</b>	<b>1.5</b>	35.06	<b>26</b>	<b>1.9</b>	33.58	<b>35</b>	<b>2.6</b>	34.29	<b>23</b>	<b>1.7</b>
	<b>39.80</b>	200	14.6	<b>38.23</b>	200	14.5	37.90	200	14.8	<b>38.04</b>	200	14.6	<b>38.21</b>	200	14.6

TABLE S.II

PERFORMANCE OF EACH METHOD ON THE RECONSTRUCTION OF OTHER BRAIN TEST IMAGES WITH RADIAL ACQUISITION. THE DEFINITION OF DIGITS IS IDENTICAL TO TABLE S.I. “—” MEANS THE METHOD CANNOT REACH HIGHER PSNR IN 200 ITERATIONS.

Index	2			3			4			5			6		
Methods	PSNR↑	iter.↓	sec.↓	PSNR↑	iter.↓	sec.↓	PSNR↑	iter.↓	sec.↓	PSNR↑	iter.↓	sec.↓	PSNR↑	iter.↓	sec.↓
PnP-ADMM	30.76	181	134.5	31.31	181	132.7	30.79	182	133.3	29.63	175	129.1	29.23	163	121.8
PnP-ISTA	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
	30.73	200	14.5	31.29	200	14.3	30.77	200	14.4	29.59	200	14.4	29.14	200	14.4
P <sup>2</sup> nP-F-1	30.77	66	9.0	31.31	64	8.7	30.81	65	8.8	29.64	61	8.4	29.24	58	8.0
	31.38	200	27.4	31.95	200	27.3	31.46	200	27.2	30.25	200	27.3	29.67	200	27.5
P <sup>2</sup> nP-F-Cheb	30.80	33	4.6	31.35	32	4.4	30.83	32	4.3	29.66	30	4.1	29.23	28	3.8
	31.93	200	27.7	32.50	200	27.3	32.03	200	27.1	30.82	200	27.2	30.13	200	27.3
P <sup>2</sup> nP-D	30.77	<b>18</b>	<b>1.3</b>	31.41	<b>24</b>	<b>1.8</b>	30.83	<b>20</b>	<b>1.4</b>	29.66	<b>25</b>	<b>1.8</b>	<b>29.23</b>	<b>27</b>	<b>2.0</b>
	<b>32.07</b>	200	14.7	<b>32.68</b>	200	14.3	<b>32.04</b>	200	14.4	<b>30.51</b>	200	14.5	28.70	200	14.5

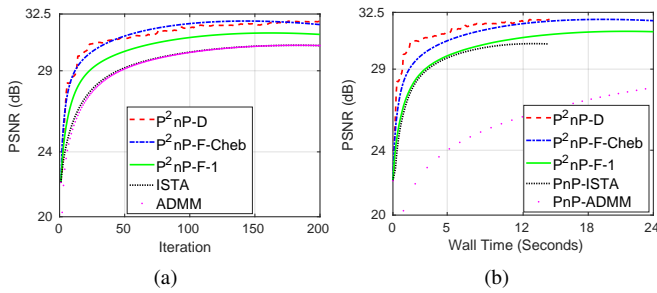


Fig. S.3. PSNR values versus iteration and wall (GPU) time for the brain 1 test image with radial acquisition.



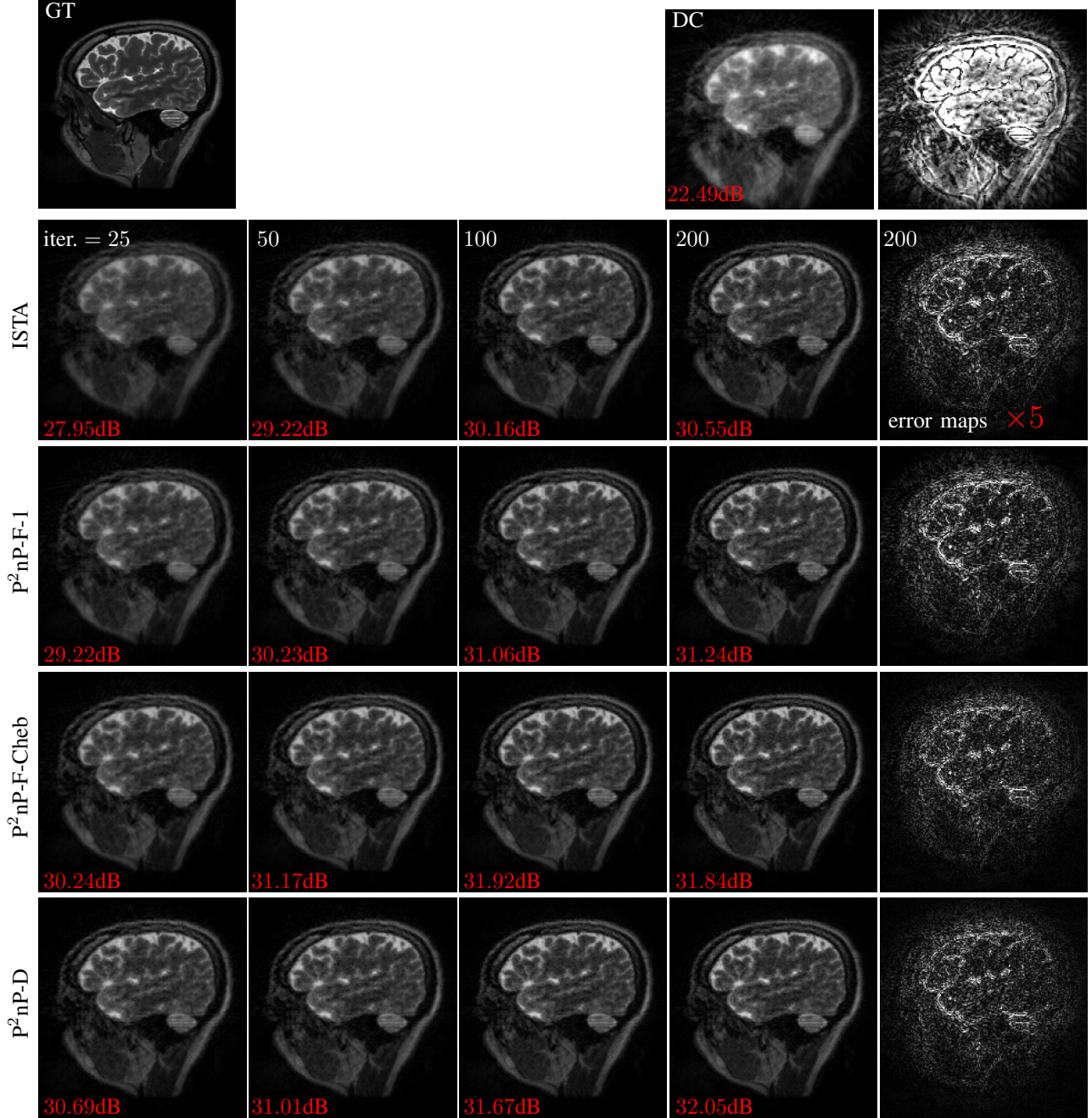


Fig. S.4. The reconstructed brain 1 images at 25, 50, 100, 200th iteration with different algorithms for radial acquisition. The PSNR value is labeled in the left bottom corner of each image. The fifth column shown the error maps ( $\times 5$ ) of the reconstructed images at 200th iteration. DC represents the density compensation based reconstruction. Acceleration factor is  $\approx 12$ .

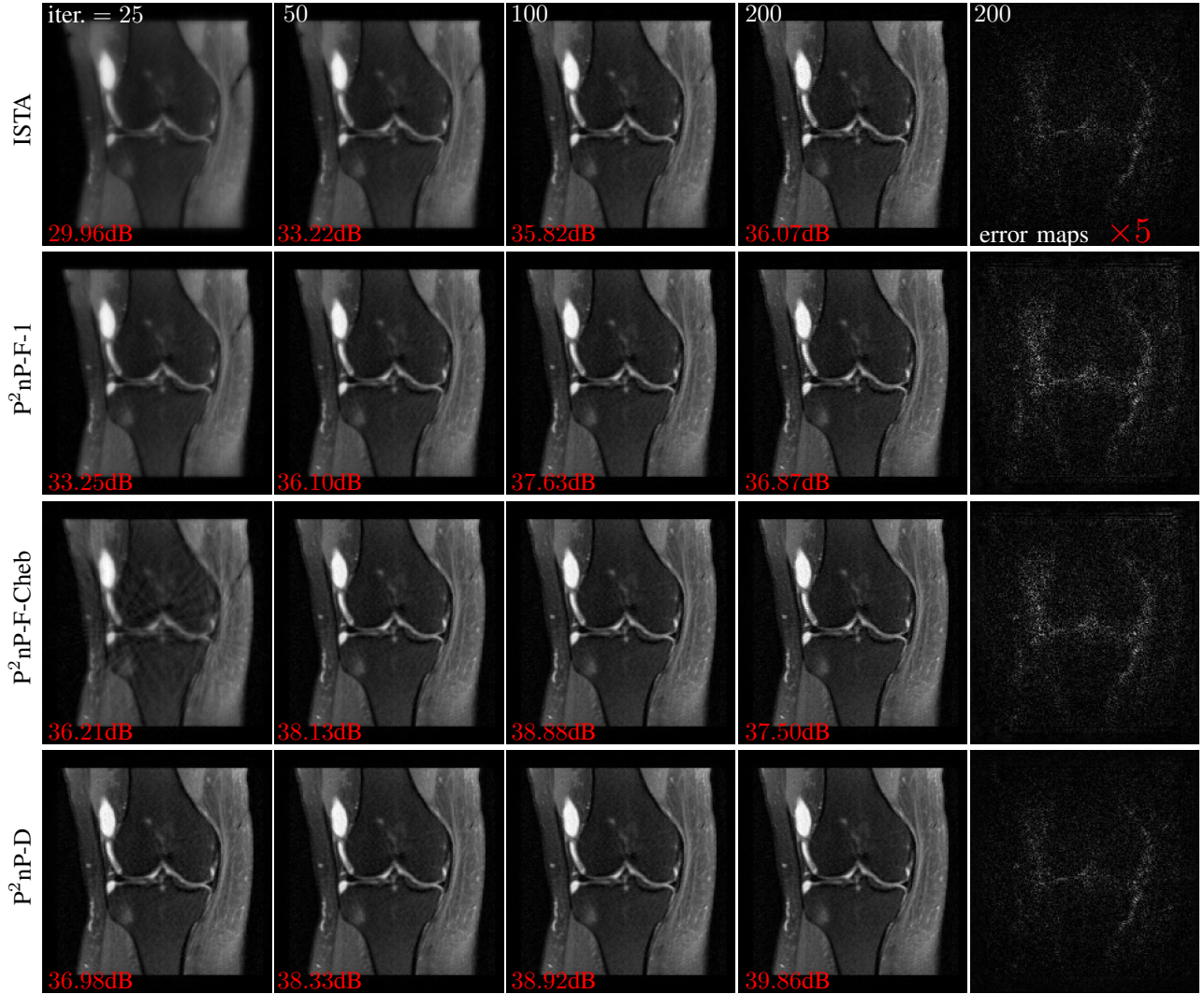


Fig. S.5. The reconstructed knee 1 images at 25, 50, 100, 200th iteration with radial acquisition of 55 spokes. The fifth column shown the error maps ( $\times 5$ ) of the reconstructed images at 200th iteration.

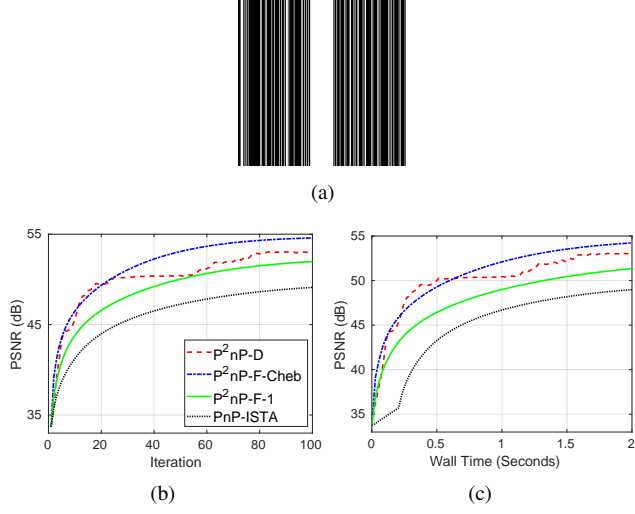


Fig. S.6. Downsampling mask and PSNR values versus iteration and wall time for the knee 1 image with Cartesian sampling. Acceleration factor is 2.

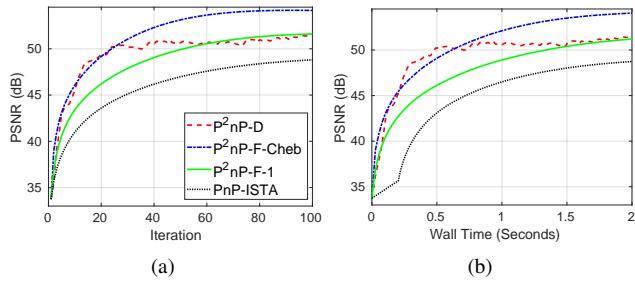


Fig. S.7. PSNR values versus iteration and wall time for the brain 1 image with Cartesian sampling.