

Autotuning Plug-and-Play Algorithms for MRI

Saurav K. Shastri¹, Rizwan Ahmad^{1,2}, and Philip Schniter¹

¹ Dept. of Electrical and Computer Engineering, The Ohio State University, Columbus, OH, 43210.

² Dept. of Biomedical Engineering, The Ohio State University, Columbus, OH, 43210.

Email: {shastri.19, schniter.1, ahmad.46}@osu.edu

Abstract—For magnetic resonance imaging (MRI), recently proposed “plug-and-play” (PnP) image recovery algorithms have shown remarkable performance. These PnP algorithms are similar to traditional iterative algorithms like FISTA, ADMM, or primal-dual splitting (PDS), but differ in that the proximal update is replaced by a call to an application-specific image denoiser, such as BM3D or DnCNN. The fixed-points of PnP algorithms depend upon an algorithmic stepsize parameter, however, which must be tuned for optimal performance. In this work, we propose a fast and robust auto-tuning PnP-PDS algorithm that exploits knowledge of the measurement-noise variance that is available from a pre-scan in MRI. Experimental results show that our algorithm converges very close to genie-tuned performance, and does so significantly faster than existing autotuning approaches.

Index Terms—Plug-and-play algorithms, primal-dual splitting, autotuning, magnetic resonance imaging, Morozov’s discrepancy principle

I. INTRODUCTION

We consider parallel magnetic resonance imaging (MRI) with $C \geq 1$ coils, where the k-space (i.e., Fourier) measurements from the i th coil, $\mathbf{y}_i \in \mathbb{C}^M$, can be expressed as

$$\underbrace{\begin{bmatrix} \mathbf{y}_1 \\ \mathbf{y}_2 \\ \vdots \\ \mathbf{y}_C \end{bmatrix}}_{\mathbf{y}} = \underbrace{\begin{bmatrix} \mathbf{PFS}_1 \\ \mathbf{PFS}_2 \\ \vdots \\ \mathbf{PFS}_C \end{bmatrix}}_{\mathbf{A}} \mathbf{x} + \mathbf{w}. \quad (1)$$

In (1), $\mathbf{x} \in \mathbb{C}^N$ is the (vectorized) N -pixel image, $\mathbf{S}_i \in \mathbb{C}^{N \times N}$ is a diagonal matrix containing the sensitivity map for the i th coil, $\mathbf{F} \in \mathbb{C}^{N \times N}$ is the discrete Fourier transform, $\mathbf{P} \in \mathbb{R}^{M \times N}$ is a subsampling matrix with $M < N$, $\mathbf{w} \in \mathbb{C}^{CM}$ contains measurement noise of variance σ^2 , and $\mathbf{A} \in \mathbb{C}^{CM \times N}$ is the forward operator. Our goal is to recover the image $\mathbf{x}$ from measurements $\mathbf{y}$ assuming knowledge of $\mathbf{A}$.

Often, image recovery is accomplished by posing and then solving an optimization problem of the form

$$\hat{\mathbf{x}} = \underset{\mathbf{x} \in \mathbb{C}^N}{\operatorname{argmin}} \{ \ell(\mathbf{A}\mathbf{x}; \mathbf{y}) + \lambda \phi(\mathbf{x}) \}, \quad (2)$$

where $\ell(\cdot; \mathbf{y})$ is a loss term, $\phi(\cdot)$ is a regularization term, and $\lambda > 0$ is a tuning parameter. Many algorithms have been developed to solve (2) in the case that both $\ell(\cdot; \mathbf{y})$ and $\phi(\cdot)$ are convex, including the alternating direction method of multipliers (ADMM) [1], the fast iterative shrinkage-thresholding algorithm (FISTA) [2], and primal-dual splitting (PDS) [3]. All

three of these algorithms alternate a loss-reducing operation with a so-called “proximal” operation of the form

$$\operatorname{prox}_{\nu\phi}(s) \triangleq \underset{\mathbf{x}}{\operatorname{argmin}} \{ \lambda \phi(\mathbf{x}) + \frac{1}{2\nu} \|\mathbf{x} - s\|_2^2 \}, \quad (3)$$

where s and $\nu > 0$ are algorithm- and iteration-dependent.

Because (3) can be interpreted as maximum a posteriori denoising of s under the prior pdf $p(\mathbf{x}) \propto e^{-\lambda\phi(\mathbf{x})}$, it has been proposed [4], [5], [6] to replace (3) with a call to a sophisticated application-specific denoiser like BM3D [7] or DnCNN [8]. In many applications, including MRI [9], these “plug-and-play” (PnP) image recovery methods tend to provide much cleaner recoveries than the traditional optimization approach (2).

For MRI image recovery, PnP shows similar performance to end-to-end deep neural networks (DNNs) like [10], [11] but offers several advantages [9]. For example, end-to-end DNNs often require a huge corpus of fully sampled k-space data for training, whereas DNN denoisers can be trained using patches from a few images. Also, the training of end-to-end DNNs is typically dependent on some choice of the forward operator $\mathbf{A}$, which can cause performance degradation when the network is applied to recover images measured under a different $\mathbf{A}$. In PnP, the denoiser training does not depend on $\mathbf{A}$ and thus there can be no such mismatch error.

In this work, we focus on the PnP-PDS algorithm [5], which alternates the following two steps:

$$\mathbf{x}_k = \mathbf{f}(\mathbf{x}_{k-1} - \gamma_1 \mathbf{A}^H \mathbf{v}_{k-1}) \quad (4a)$$

$$\mathbf{v}_k = \operatorname{prox}_{\gamma_2 \ell^*}(\mathbf{v}_{k-1} + \gamma_2 \mathbf{A}(2\mathbf{x}_k - \mathbf{x}_{k-1})), \quad (4b)$$

where $\mathbf{f} : \mathbb{C}^N \rightarrow \mathbb{C}^N$ is the denoiser, γ_1 and γ_2 are positive stepsizes, and $\operatorname{prox}_{\gamma_2 \ell^*}(\mathbf{s}) = \mathbf{s} - \gamma_2 \operatorname{prox}_{\gamma_2^{-1} \ell}(\gamma_2^{-1} \mathbf{s})$, where $\ell^*(\cdot; \mathbf{y})$ denotes the convex conjugate of $\ell(\cdot; \mathbf{y})$. The stepsizes are chosen such that they satisfy $\gamma_1 \gamma_2 \leq 1/\|\mathbf{A}\|_2^2$. We focus on PnP-PDS since it has advantages over both PnP-FISTA [6] and PnP-ADMM [4]: it yields a first-order algorithm for typical choices of $\ell(\cdot; \mathbf{y})$ and does not upper-bound the stepsize γ_1 .

In the common case of quadratic loss $\ell(\mathbf{z}; \mathbf{y}) = \frac{1}{2} \|\mathbf{z} - \mathbf{y}\|_2^2$, the PnP-PDS algorithm (4) becomes

$$\mathbf{x}_k = \mathbf{f}(\mathbf{x}_{k-1} - \gamma_1 \mathbf{A}^H \mathbf{v}_{k-1}) \quad (5a)$$

$$\mathbf{v}_k = \frac{1}{1+\gamma_2} \mathbf{v}_{k-1} + \frac{\gamma_2}{1+\gamma_2} (\mathbf{A}(2\mathbf{x}_k - \mathbf{x}_{k-1}) - \mathbf{y}). \quad (5b)$$

Rather than solving an optimization problem of the form (2), the PnP-PDS algorithm (5) seeks the solution $\hat{\mathbf{x}}$ to the fixed-point equation

$$\hat{\mathbf{x}} = \mathbf{f}(\hat{\mathbf{x}} - \gamma_1 \mathbf{A}^H (\mathbf{A}\hat{\mathbf{x}} - \mathbf{y})). \quad (6)$$

This work was supported in part by NIH-R01HL135489.

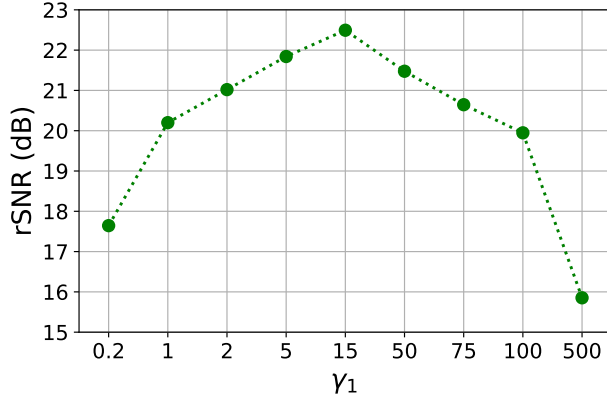


Fig. 1. Asymptotic rSNR of quadratic-loss PnP-PDS (5) versus stepsize γ_1 .

or, equivalently, the solution to a consensus equilibrium problem [12].

Equation (6) shows that the PnP-PDS fixed point $\hat{x}$ depends on the choice of the stepsize γ_1 (but not γ_2). Thus, γ_1 must be tuned for best recovery performance. This is illustrated in Fig. 1, which plots asymptotic recovery SNR (rSNR) $\triangleq \frac{\|x\|_2^2}{\|\hat{x} - x\|_2^2}$ versus γ_1 for a typical image recovery experiment. Furthermore, the optimal value of γ_1 is dependent on the forward operator A , the noise variance σ^2 , and the image x . This makes it difficult to tune γ_1 in practice, when x is unknown.

In this paper, we propose an autotuning variant of PnP-PDS that shows fast convergence and robustness to parameter initializations. Our method leverages the fact that, in MRI, it is easy to measure the noise variance σ^2 using a pre-scan. Before describing our method, we briefly review existing approaches to autotuned PnP.

II. EXISTING WORK

In the optimization problem (2), it can be difficult to choose the optimal regularization weight λ . When $\ell(\cdot; y)$ and $\phi(x)$ are both convex, (2) can be equivalently recast [13] as

$$\hat{x} = \operatorname{argmin}_{x \in \mathbb{C}^N} \phi(x) \quad \text{subject to} \quad \ell(Ax; y) \leq \mu, \quad (7)$$

for some λ -dependent parameter $\mu > 0$. The formulation (7) may be preferable to (2) in that μ may be easier to choose than λ . For example, with quadratic loss $\ell(z; y) = \frac{1}{2}\|z - y\|_2^2$ and σ^2 -variance additive noise w , the value of $\ell(z; y)$ at the optimal setting $z = Ax$ concentrates at $CM\sigma^2/2$ (as CM grows large, by the law of large numbers), motivating the choice of

$$\mu \approx CM\sigma^2/2 \quad (8)$$

in (7) when the noise variance σ^2 is known. This idea has been routinely applied for image recovery (see, e.g., [14], [15]) and is sometimes referred to as Morozov's discrepancy principle [16].

Although the above formulation pertains to the optimization approach to image recovery, similar ideas have appeared in the PnP literature, as we now describe.

A. PDS-ATO

Inspired by Morozov's discrepancy principle (7)-(8), Ono [5] proposed PnP-PDS (4) with an indicator loss of the form

$$\ell(z; y) = \begin{cases} 0 & \frac{1}{CM}\|y - z\|^2 \leq \beta\sigma^2 \\ \infty & \text{otherwise,} \end{cases} \quad (9)$$

where $\beta \approx 1$ is a user-adjustable parameter (we fix $\beta = 0.95$ in all of our experiments). This yields the iteration

$$x_k = f(x_{k-1} - \gamma_1 A^H v_{k-1}) \quad (10a)$$

$$v_k = \max \left\{ 0, 1 - \frac{\gamma_2 \sqrt{\beta CM} \sigma}{\|v_{k-1} + \gamma_2 A(2x_k - x_{k-1}) - \gamma_2 y\|_2} \right\} (v_{k-1} + \gamma_2 A(2x_k - x_{k-1}) - \gamma_2 y), \quad (10b)$$

where it is typical to choose $\gamma_2 = 1/(\gamma_1 \|A\|_2^2)$ for stability. We will refer to (10) as the AutoTune-Ono (ATO) approach to PDS, i.e., PDS-ATO. It should be emphasized that, in (10), the stepsize parameters γ_1, γ_2 affect convergence speed but *not* the fixed points. As we will see, (10) has good fixed points but somewhat slow convergence speed, regardless of the choice of γ_1 .

B. PDS-ATM

In the fixed-point expression (6) for quadratic-loss PnP-PDS, it can be seen that γ_1 magnifies the residual measurement-error $y - A\hat{x}$. Thus, a larger value of γ_1 should correspond to a smaller value of $\|y - A\hat{x}\|_2^2$, and vice versa. This observation suggests that Morozov's discrepancy principle can somehow be used to adapt γ_1 at each iteration in quadratic-loss PnP-PDS. For example, if $\|y - A\hat{x}\|_2^2$ is larger than the target value at a given iteration, then γ_1 could be increased. A first incarnation of this approach was proposed by Liu et al. in [17]:

$$x_k = f(x_{k-1} - \gamma_{1,k-1} A^H v_{k-1}) \quad (11a)$$

$$v_k = \frac{1}{1+\gamma_{2,k-1}} v_{k-1} + \frac{\gamma_{2,k-1}}{1+\gamma_{2,k-1}} (A(2x_k - x_{k-1}) - y) \quad (11b)$$

$$\gamma_{1,k} = \gamma_{1,k-1} \frac{\|y - Ax_k\|_2^2}{\beta CM \sigma^2} \quad (11c)$$

$$\gamma_{2,k} = \frac{1}{\gamma_{1,k} \|A\|_2^2}. \quad (11d)$$

We will refer to (11) as PDS-ATM. Note that the multiplicative update step (11c) causes γ_1 to be increased when $\frac{1}{CM}\|y - Ax_k\|^2 > \beta\sigma^2$ and decreased when $\frac{1}{CM}\|y - Ax_k\|^2 < \beta\sigma^2$, with the goal of driving γ_1 to a value for which $\frac{1}{CM}\|y - A\hat{x}\|^2 = \beta\sigma^2$.

C. PDS-ATMI

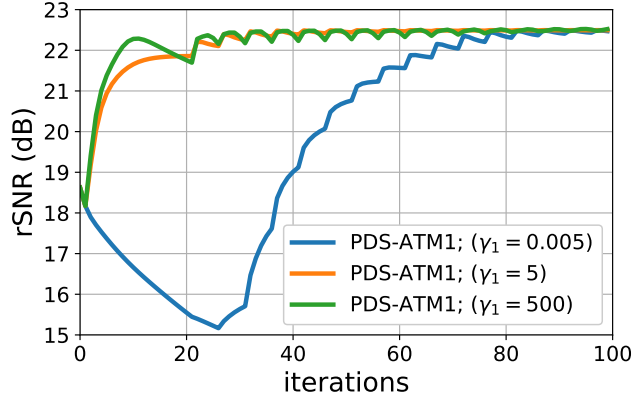
To help promote stability, the authors of [17] made some modifications to (11) for practical implementation. They are summarized in Alg. 1, which we shall refer to as PDS-ATMI. In Alg. 1, the stepsizes $\gamma_{1,k}$ and $\gamma_{2,k}$ are updated only once every 5 iterations, starting from iteration $k = 20$. Although these modifications enhance the stability of PDS-ATMI, they slow down its convergence, as seen in Fig. 2. And even with these modifications, we find that Alg. 1 fails to converge when

Algorithm 1 The PDS-ATM1 algorithm [17]**Require:** $\gamma_1 > 0, \mathbf{x}_0 \in \mathbb{C}^N, \mathbf{v}_0 \in \mathbb{C}^N, \beta \in (0, 1]$

```

1:  $\gamma_{1,0} = \gamma_1$ 
2:  $\gamma_{2,0} = \frac{1}{\gamma_{1,0} \|\mathbf{A}\|_2^2}$ 
3: for  $k = 1, 2, 3, \dots$  do
4:    $\mathbf{x}_k = \mathbf{f}(\mathbf{x}_{k-1} - \gamma_{1,k-1} \mathbf{A}^H \mathbf{v}_{k-1})$ 
5:    $\mathbf{v}_k = \frac{1}{1+\gamma_{2,k-1}} \mathbf{v}_{k-1} + \frac{\gamma_{2,k-1}}{1+\gamma_{2,k-1}} (\mathbf{A}(2\mathbf{x}_k - \mathbf{x}_{k-1}) - \mathbf{y})$ 
6:   if  $k > 19$  and  $\text{mod}(k, 5) == 0$  then
7:      $\gamma_{1,k} = \gamma_{1,k-1} \frac{\|\mathbf{y} - \mathbf{A}\mathbf{x}_k\|_2^2}{\beta CM \sigma^2}$ 
8:      $\gamma_{2,k} = \frac{1}{\gamma_{1,k} \|\mathbf{A}\|_2^2}$ 
9:   else
10:     $\gamma_{1,k} = \gamma_{1,k-1}$ 
11:     $\gamma_{2,k} = \gamma_{2,k-1}$ 
12:   end if
13: end for

```

Fig. 2. rSNR versus iteration for PDS-ATM1 under several initial γ_1 .

the initial γ_1 is too large. These shortcomings motivate our approach to autotuned PnP-PDS, which is detailed in the next section.

III. PROPOSED ALGORITHM: PDS-ATM2

The proposed method, summarized in Alg. 2, is called PDS-ATM2. It builds on the multiplicative-update idea from PDS-ATM (11) but adds two key refinements: damping and restarting. These two refinements help PDS-ATM2 to converge quickly and maintain robustness to the choice of initial γ_1 . A detailed explanation of PDS-ATM2 follows.

Because PDS-ATM2 builds on PDS-ATM, lines 5–6 of Alg. 2 are identical to (11a) and (11b), respectively. These two lines mirror the main update steps of quadratic-loss PnP-PDS, (5a) and (5b).

Skipping ahead, line 20 of Alg. 2 performs the (damped) multiplicative update of γ_1 . If the damping parameter α was $= 1$, there would be no damping, and the PDS-ATM2 update of γ_1 in line 20 would be identical to the PDS-ATM update of γ_1 in (11c). But when $\alpha \in (0, 1)$, line 20 slows the update of γ_1 . Importantly, the form of line 20 ensures that the fixed-point value of γ_1 is unaffected by the choice of $\alpha \in (0, 1]$. PDS-ATM2 then updates the value of γ_2 in line 22 of Alg. 2 in the same way that PDS-ATM updates it in (11d).

Algorithm 2 The PDS-ATM2 algorithm (proposed)**Require:** $\gamma_1 > 0, \mathbf{x}_0 \in \mathbb{C}^N, \mathbf{v}_0 \in \mathbb{C}^N, \beta \in (0, 1], \alpha \in (0, 1]$

```

1: restart-now = FALSE, restart-allowed = TRUE
2:  $\gamma_{1,0} = \gamma_1$ 
3:  $\gamma_{2,0} = \frac{1}{\gamma_{1,0} \|\mathbf{A}\|_2^2}$ 
4: for  $k = 1, 2, 3, \dots$  do
5:    $\mathbf{x}_k = \mathbf{f}(\mathbf{x}_{k-1} - \gamma_{1,k-1} \mathbf{A}^H \mathbf{v}_{k-1})$ 
6:    $\mathbf{v}_k = \frac{1}{1+\gamma_{2,k-1}} \mathbf{v}_{k-1} + \frac{\gamma_{2,k-1}}{1+\gamma_{2,k-1}} (\mathbf{A}(2\mathbf{x}_k - \mathbf{x}_{k-1}) - \mathbf{y})$ 
7:   if  $\|\mathbf{y} - \mathbf{A}\mathbf{x}_k\|_2^2 > \|\mathbf{y} - \mathbf{A}\mathbf{x}_{k-1}\|_2^2$  then
8:     restart-now = TRUE
9:   else
10:    restart-now = FALSE
11:   end if
12:   if  $\|\mathbf{y} - \mathbf{A}\mathbf{x}_k\|_2^2 < \beta CM \sigma^2$  then
13:     restart-allowed = FALSE
14:   else if  $\|\mathbf{y} - \mathbf{A}\mathbf{x}_k\|_2^2 > 1.1 \beta CM \sigma^2$  then
15:     restart-allowed = TRUE
16:   end if
17:   if restart-allowed and restart-now then
18:      $\gamma_{1,k} = \gamma_{1,0}$ 
19:   else
20:      $\gamma_{1,k} = \alpha \gamma_{1,k-1} \frac{\|\mathbf{y} - \mathbf{A}\mathbf{x}_k\|_2^2}{\beta CM \sigma^2} + (1 - \alpha) \gamma_{1,k-1}$ 
21:   end if
22:    $\gamma_{2,k} = \frac{1}{\gamma_{1,k} \|\mathbf{A}\|_2^2}$ 
23:   if  $k > 2$  and  $\gamma_{1,k} == \gamma_{1,k-1} == \gamma_{1,k-2}$  then
24:      $\gamma_{1,0} = 10 \gamma_{1,0}$ 
25:   end if
26: end for

```

As visible from lines 7–11 of Alg. 2, the `restart-now` flag is raised when the residual measurement error $\|\mathbf{y} - \mathbf{A}\mathbf{x}_k\|_2^2$ increases from one iteration to the next. This is because residual error is expected to start large and decrease over the iterations; an increasing residual error is a sign that the value of γ_1 has become misadjusted and should be reset, as in line 18.

But there is an exception: we want the residual measurement error $\|\mathbf{y} - \mathbf{A}\mathbf{x}_k\|_2^2$ to increase when it lies below the target value of $\beta CM \sigma^2$. When this situation is detected, in line 12, the `restart-allowed` flag is set to FALSE. If the residual measurement error grows sufficiently above the target value, as in line 14, the `restart-allowed` flag is restored to TRUE.

Finally, a very small γ_1 (re)initialization could cause the algorithm to continually restart. This issue is circumvented by lines 23–25 of Alg. 2, which increase the value of $\gamma_{1,0}$.

IV. NUMERICAL EXPERIMENTS

We evaluated our method using 10 mid-slice, non-fat-suppressed, 128×128 , test images from the single-coil fastMRI knee image database [11]. To construct the multi-coil measurements $\mathbf{y}$, we simulated 4 receive coils using the Biot-Savart law and performed k-space subsampling using a fixed Cartesian mask with acceleration 4 (i.e. $\frac{N}{M} = 4$). White Gaussian noise was then added to obtain images with average signal-to-noise ratio (SNR) in $\{15, 17, 20, 23\}$ dB.

The proposed PDS-ATM2 algorithm was compared to PDS-ATM1, PDS-ATO, the U-Net from [11], and genie-tuned PnP-PDS. By “genie-tuned PnP-PDS,” we mean PnP-PDS with the

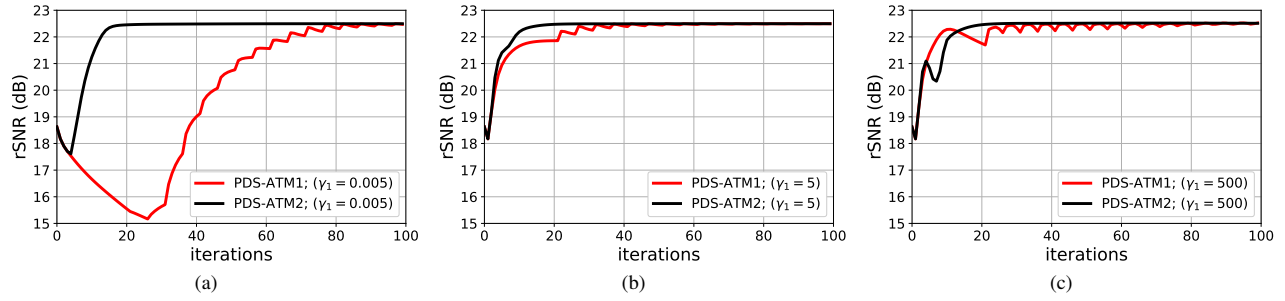


Fig. 3. rSNR versus iteration for PDS-ATM1 and PDS-ATM2 under various choices of initial γ_1 .

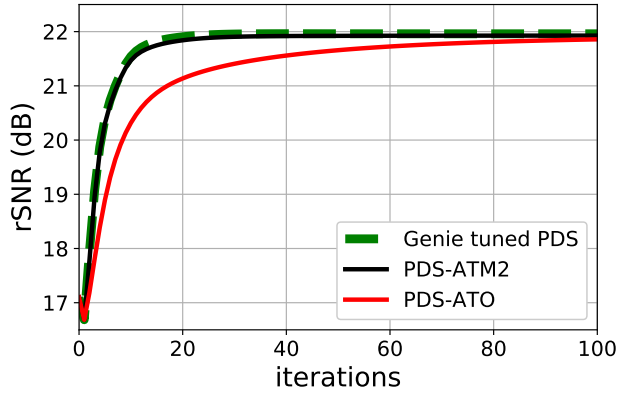


Fig. 4. rSNR versus iteration for several PnP algorithms

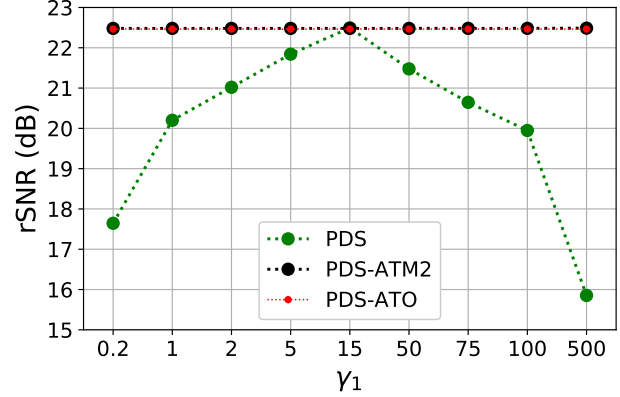


Fig. 5. Asymptotic rSNR versus (initial) γ_1 for several PnP algorithms.

rSNR-maximizing value of γ_1 (found by grid-search). Genie-tuned PnP-PDS is not a practical algorithm, but an upper bound on what one could expect from optimally autotuned version of PnP-PDS.

All image reconstruction methods used an $\mathbf{A}$ constructed with sensitivity maps $\mathbf{S}_i$ estimated from $\mathbf{y}$ using ES-PIRiT [18]. For the denoiser, a modified DnCNN was trained on MRI knee images that were different from the test images; see [9] for details. The U-Net was trained separately at each noise level using the same set of training images used for the denoiser. All autotuned-PnP methods used $\beta = 0.95$, and PDS-ATM2 used the damping value $\alpha = 0.2$.

Figure 3 shows rSNR versus iteration for PDS-ATM1 and PDS-ATM2 for several choices of initial γ_1 . From the figure, we see that PDS-ATM2 converges much faster and smoother than PDS-ATM1 for all choices of initial γ_1 . For this reason, we consider only PDS-ATM2 in the sequel.

Figure 4 shows rSNR versus iteration for PDS-ATM2, PDS-ATO, and genie-tuned PDS. There we see that PDS-ATM2 converges as fast as genie-tuned PDS (i.e., < 30 iterations) and significantly faster than PDS-ATO.

Figure 5 shows rSNR versus (initial) γ_1 for PDS-ATM2, PDS-ATO, and PDS. It shows that standard PnP-PDS is very sensitive to the value of γ_1 , while PDS-ATO and PDS-ATM2 are not, as a consequence of autotuning.

Table I shows the asymptotic values of rSNR and SSIM, averaged over the 10 test images, for the methods under con-

sideration. From the table, we see that the average asymptotic-performances of PDS-ATO and PDS-ATM2 are identical and very close to that of genie-aided PnP-PDS, which plays the role of an impractical upper bound. The table also shows that the genie-tuned and autotuned PnP approaches give slightly better average rSNR than the U-Net, but slightly worse average SSIM; overall, their average performance is similar.

Figure 6 shows an example test image, several recoveries, and their corresponding error images. For this example, the three PnP methods performed nearly identically and noticeably better than the U-Net (whose rSNR was 1.18 dB worse).

V. CONCLUSIONS

In this paper, we considered parallel MRI image recovery using the PnP framework. We focused on the PnP-PDS algorithm, since it is both flexible and computationally efficient. We first showed that PnP performance is strongly dependent on the choice of the stepsize, γ_1 , motivating the careful tuning of this parameter. We then reviewed several existing methods that circumvent this tuning issue using Morozov's discrepancy principle, which says that the residual measurement error in each k -space sample should be close to the measurement noise variance, σ^2 , which can be measured using an MRI pre-scan. We showed that, although the existing methods eventually converge to a good recovery, they converge somewhat slowly. Therefore, we proposed a new autotuning PnP-PDS algorithm that converges quickly and maintains robustness to the choice

TABLE I
RSNR AND SSIM AVERAGED OVER THE 10 TEST IMAGES FOR SEVERAL MEASURED SNRS

	Avg meas. SNR: 15 dB		Avg meas. SNR: 17 dB		Avg meas. SNR: 20 dB		Avg meas. SNR: 23 dB	
	rSNR(dB)	SSIM	rSNR(dB)	SSIM	rSNR(dB)	SSIM	rSNR(dB)	SSIM
Genie tuned PDS	21.02	0.804	21.42	0.820	21.96	0.839	22.42	0.854
PDS-ATM2	21.00	0.805	21.44	0.817	21.93	0.833	22.33	0.850
PDS-ATO	21.00	0.805	21.44	0.817	21.93	0.833	22.33	0.850
U-Net	20.77	0.838	21.07	0.845	21.34	0.841	21.50	0.859

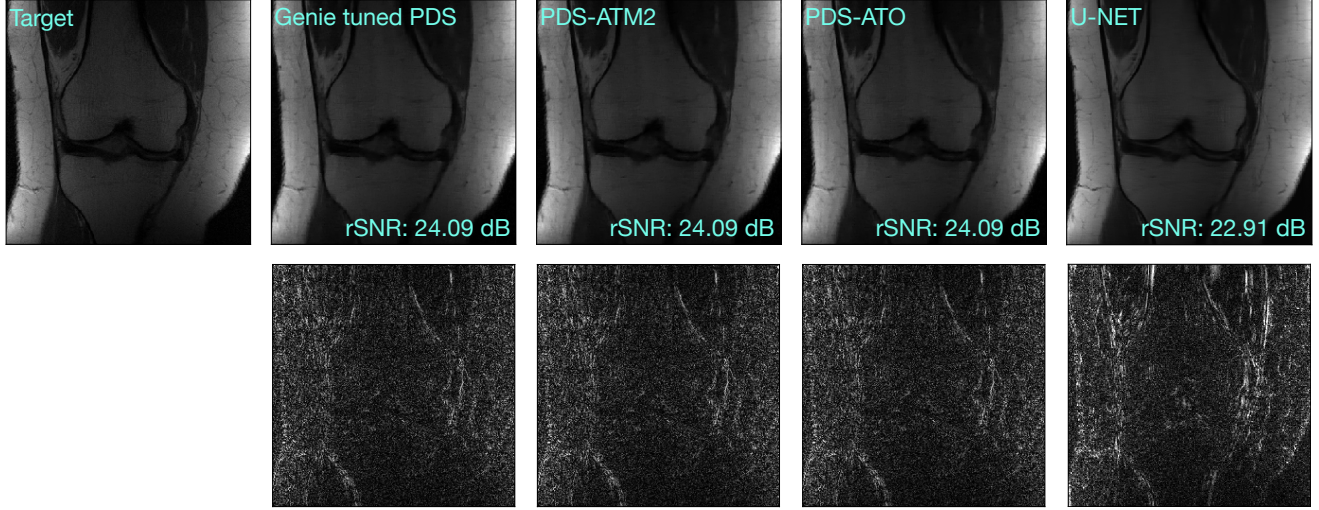


Fig. 6. Example recovery of a knee image from the fastMRI dataset. The top row shows the ground-truth image and several images recoveries with their corresponding rSNR. The bottom row shows the corresponding error images $\times 2.4$.

of initial γ_1 . Through numerical experiments with fastMRI knee images, we showed that the performance of our proposed technique is very close to genie-tuned PnP-PDS, and very close to the U-Net, a recently proposed end-to-end deep-learning method.

ACKNOWLEDGMENT

The authors thank Edward T. Reehorst for sharing the trained Dn-CNN denoiser and the U-Net training code.

REFERENCES

- [1] S. Boyd, N. Parikh, E. Chu, B. Peleato, and J. Eckstein, "Distributed optimization and statistical learning via the alternating direction method of multipliers," *Found. Trends Mach. Learn.*, vol. 3, no. 1, pp. 1–122, 2011.
- [2] 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.
- [3] A. Chambolle and T. Pock, "A first-order primal-dual algorithm for convex problems with applications to imaging," *J. Math. Imaging Vis.*, vol. 40, pp. 120–145, 2011.
- [4] S. V. Venkatakrishnan, C. A. Bouman, and B. Wohlberg, "Plug-and-play priors for model based reconstruction," in *Proc. IEEE Global Conf. Signal Info. Process.*, 2013, pp. 945–948.
- [5] S. Ono, "Primal-dual plug-and-play image restoration," *IEEE Signal Process. Lett.*, vol. 24, no. 8, pp. 1108–1112, 2017.
- [6] Y. Sun, B. Wohlberg, and U. S. Kamilov, "An online plug-and-play algorithm for regularized image reconstruction," *IEEE Trans. Comp. Imag.*, vol. 5, pp. 395–408, 2019.
- [7] 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, 2007.
- [8] 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, 2017.
- [9] R. Ahmad, C. A. Bouman, G. T. Buzzard, S. Chan, S. Liu, E. T. Reehorst, and P. Schniter, "Plug and play methods for magnetic resonance imaging," *IEEE Signal Process. Mag.*, vol. 37, no. 1, pp. 105–116, 2020.
- [10] K. H. Jin, M. T. McCann, E. Froustey, and M. Unser, "Deep convolutional neural network for inverse problems in imaging," *IEEE Trans. Image Process.*, vol. 26, no. 9, pp. 4509–4522, 2017.
- [11] J. Zbontar, F. Knoll, A. Sriram, M. J. Muckley, M. Bruno, A. Defazio, M. Parente *et al.*, "fastMRI: An open dataset and benchmarks for accelerated MRI," *arXiv:1811.08839*, 2018.
- [12] 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.
- [13] D. Lorenz and N. Worliczek, "Necessary conditions for variational regularization schemes," *Inverse Problems*, vol. 29, no. 7, 2013, 075016.
- [14] B. R. Hunt, "The application of constrained least squares estimation to image restoration by digital computer," *IEEE Trans. Computers*, vol. C-22, no. 9, pp. 805–812, 1973.
- [15] N. P. Galatsanos and A. K. Katsaggelos, "Methods for choosing the regularization parameter and estimating the noise variance in image restoration and their relation," *IEEE Trans. Image Process.*, vol. 1, no. 3, pp. 322–336, 1992.
- [16] V. A. Morozov, *Methods for Solving Incorrectly Posed Problems*. Springer Science & Business Media, 2012.
- [17] S. Liu, N. Jin, P. Schniter, and R. Ahmad, "A parameter-free plug-and-play method for accelerated MRI reconstruction," in *Proc. Ann. Mtg. ISMRM*, Aug. 2020, abstract ID: 3607.
- [18] M. Uecker, P. Lai, M. J. Murphy, P. Virtue, M. Elad, J. M. Pauly, S. S. Vasanawala *et al.*, "ESPIRiT—an eigenvalue approach to autocalibrating parallel MRI: Where SENSE meets GRAPPA," *Magnetic Resonance Med.*, vol. 71, no. 3, pp. 990–1001, 2014.