LNFA2384849, VOL 00, ISS 0

Iteratively Refined Image Reconstruction with Learned Attentive Regularizers

Mehrsa Pourya, Sebastian Neumayer, and Michael Unser

QUERY SHEET

This page lists questions we have about your paper. The numbers displayed at left are hyperlinked to the location of the query in your paper.

The title and author names are listed on this sheet as they will be published, both on your paper and on the Table of Contents. Please review and ensure the information is correct and advise us if any changes need to be made. In addition, please review your paper as a whole for typographical and essential corrections.

Your PDF proof has been enabled so that you can comment on the proof directly using Adobe Acrobat. For further information on marking corrections using Acrobat, please visit http://journalauthors.tandf.co.uk/production/acrobat.asp; https://authorservices.taylorandfrancis.com/how-to-correct-proofs-with-adobe/

The CrossRef database (www.crossref.org/) has been used to validate the references.

AUTHOR QUERIES

Q1 Please confirm that the city details in the affiliations, as set out in the proof, are correct.
Q2 Please check that the heading levels have been correctly formatted throughout.
Q3 Please confirm the email address of the corresponding author as set in the proof is accurate.
Q4 As per the journal style, we have reordered the references in sequential order.
Q5 As per the journal style, “equations” should be section-wise. Please check.
Q6 As per the journal style, “enunciations” should be section-wise. Please check.

NUMERICAL FUNCTIONAL ANALYSIS AND OPTIMIZATION
2024, VOL. 00, NO. 0, 1–30
https://doi.org/10.1080/01630563.2024.2384849

Iteratively Refined Image Reconstruction with Learned Attentive Regularizers

Mehrsa Pourya[a], Sebastian Neumayer[b], and Michael Unser[a]

[a]Biomedical Imaging Group, EPFL, Lausanne, Switzerland; [b]Professorship of Inverse Problems, TU Chemnitz, Chemnitz, Germany

ABSTRACT

We propose a regularization scheme for image reconstruction that leverages the power of deep learning while hinging on classic sparsity-promoting models. Many deep-learning-based models are hard to interpret and cumbersome to analyze theoretically. In contrast, our scheme is interpretable because it corresponds to the minimization of a series of convex problems. For each problem in the series, a mask is generated based on the previous solution to refine the regularization strength spatially. In this way, the model becomes progressively attentive to the image structure. For the underlying update operator, we prove the existence of a fixed point. As a special case, we investigate a mask generator for which the fixed-point iterations converge to a critical point of an explicit energy functional. In our experiments, we match the performance of state-of-the-art learned variational models for the solution of inverse problems. Additionally, we offer a promising balance between interpretability, theoretical guarantees, reliability, and performance.

ARTICLE HISTORY
Received 13 February 2024
Accepted 22 July 2024

KEYWORDS
Convex regularization; data-driven priors; fixed-point equations; inverse problems; majorization minimization; solution-driven models

1. Introduction

In biomedical imaging [1], including magnetic resonance imaging (MRI) and computed tomography, reconstructions are often achieved via the resolution of an inverse problem. Its task is to recover an unknown signal $\mathbf{x} \in \mathbb{R}^N$ from noisy measurements $\mathbf{y} = \mathbf{H}\mathbf{x} + \mathbf{n} \in \mathbb{R}^M$, where $\mathbf{H} \in \mathbb{R}^{M,N}$ encodes the data-acquisition process and the noise $\mathbf{n} \in \mathbb{R}^M$ accounts for imperfections in this description. From a variational perspective [2], one defines the reconstruction as the solution to the minimization problem

$$\arg\min_{\mathbf{x}\in\mathbb{R}^N} \big(\mathrm{E}(\mathbf{H}\mathbf{x}, \mathbf{y}) + \lambda \mathcal{R}(\mathbf{x})\big), \tag{1}$$

which involves a data-fidelity term $\mathrm{E}\colon \mathbb{R}^M \times \mathbb{R}^M \to \mathbb{R}_{\geq 0}$ and a regularizer $\mathcal{R}\colon \mathbb{R}^N \to \mathbb{R}_{\geq 0}$. In (1), the data fidelity term ensures the consistency of

CONTACT Sebastian Neumayer sebastian.neumayer@mathematik.tu-chemnitz.de Professorship of Inverse Problems, TU Chemnitz, Chemnitz, Germany.

the reconstruction with the measurements, while the regularization, whose strength is controlled by $\lambda \in \mathbb{R}_{>0}$, imposes some regularity constraints (prior information) on the solution.

For a large variety of data-acquisition and noise models, a well-studied zoo of data fidelities E can be found in the literature. While an instance-specific E is natural, it is desirable that the regularizer $\mathcal{R}$ is agnostic to $\mathbf{H}$ and $\mathbf{n}$ and solely depends on the properties of the underlying images. Hence, a regularizer that captures these inherent properties would be of great interest. Attempts can be traced as far back as to the Tikhonov regularization [3], where images are modeled as smooth signals. Later, this approach was outperformed by compressed sensing [4]. Such models either assume that the signal is sparse in some latent space (e.g., wavelet decomposition [5]) or involve a filter-based regularizer $\mathcal{R}$ such as the total variation (TV) [4, 6] and its generalizations [7]. These classic signal-processing approaches achieve a baseline performance with the advantage that they provide stability and robustness guarantees [8].

With the emergence of deep-learning techniques for the solution of inverse problems [9], the traditional approaches have been outperformed in many applications. The end-to-end training achieves state-of-the-art performance in terms of quantitative metrics such as the peak signal-to-noise ratio (PSNR). However, such models are often neither interpretable nor trustworthy for sensitive applications such as biomedical imaging [10, 11]. Therefore, a recent line of research [12–15] is focusing on the use of deep learning for the solution of inverse problems within the variational framework (1). There, instead of learning the whole reconstruction pipeline in an end-to-end manner, one only learns the regularizer $\mathcal{R}$. Up to now, these models have relied mostly on deep architectures to parameterize $\mathcal{R}$, which makes an interpretation difficult. To bypass this issue, the authors in [16] have proposed to parameterize the learnable $\mathcal{R}$ as

$$\mathcal{R}\colon \mathbf{x} \mapsto \sum_{c=1}^{N_C} \left\langle \mathbf{1}_N, \boldsymbol{\psi}_c(\mathbf{W}_c\mathbf{x}) \right\rangle, \tag{2}$$

with channel-wise data-driven convolutional matrices $\mathbf{W}_c \in \mathbb{R}^{N,N}$ and $\boldsymbol{\psi}_c(\mathbf{x}) = (\psi_c(x_k))_{k=1}^N$, where the convex and symmetric profiles ψ_c are members of $\mathcal{C}^{1,1}_{\geq 0}(\mathbb{R})$, the space of nonnegative differentiable functions with Lipschitz-continuous derivatives. Based on the architecture (2), the authors of [16] obtain the best performance among known convex regularizers in their experiments. Moreover, (2) has a clear interpretation as a filter-based regularizer. To further improve the reconstruction performance, we need to look beyond convexity. As an extension of the model (2), the authors of [17] have proposed to learn symmetric potentials $\psi_c \in \mathcal{C}^{1,1}_{\geq 0}(\mathbb{R})$ with $\psi_c'' \geq -\rho$ a.e., namely ρ-weakly convex ones. This relaxation significantly improves over the convex setting. In particular, it gets close to the performance of the DRUNet-based model [18], which is among the best-performing methods with a (loose) energy interpretation.

1.1. Outline and contribution

First, we introduce the theoretical concepts in Section 2. Then, we establish in Section 3.1 a link between the use of a ρ-weakly convex ψ_c within (2) and spatially-adaptive regularization [19–22]. To this end, we investigate the regularizer

$$\mathcal{R}_{\mathrm{MMR}} \colon \mathbf{x} \mapsto \sum_{c=1}^{N_C} \big\langle \mathbf{1}_N, \boldsymbol{\psi}_c\big(\mathbf{B}_c\,|\mathbf{W}_c\mathbf{x}|\,\big)\big\rangle, \tag{3}$$

where the convolutional matrices $\mathbf{W}_c \in \mathbb{R}^{N,N}$ and $\mathbf{B}_c \in \mathbb{R}^{N,N}_{\geq 0}$, and the $\boldsymbol{\psi}_c(\mathbf{x}) = (\psi_c(x_k))_{k=1}^N$ with concave potentials $\psi_c \in \mathcal{C}^{1,1}_{\geq 0}(\mathbb{R}_{\geq 0})$ are data-driven. In (3), $|\cdot|$ is applied component-wise to the vector $\mathbf{W}_c\mathbf{x}$ and $\mathbf{B}_c$ is constrained to have normalized rows. As shorthand, we introduce the notations $\mathbf{W} = (\mathbf{W}_c)_{c=1}^{N_C}$, $\mathbf{B} = (\mathbf{B}_c)_{c=1}^{N_C}$, and $\boldsymbol{\psi} = (\boldsymbol{\psi}_c)_{c=1}^{N_C}$. For the regularizer (3), we show in Theorem 3 that the variational problem (1) is guaranteed to have at least one minimizer. To reach the latter, we propose to use the iterative majorization-minimization regularization (MMR) characterized by

$$\mathbf{x}_{k+1} \in \operatorname*{arg\,min}_{\mathbf{x}\in\mathbb{R}^N} \big(\mathrm{E}(\mathbf{H}\mathbf{x}, \mathbf{y}) + \lambda \mathcal{R}_{\mathrm{MMR},k}(\mathbf{x})\big), \tag{4}$$

with initialization $\mathbf{x}_1 \in \mathbb{R}^N$ and

$$\mathcal{R}_{\mathrm{MMR},k} \colon \mathbf{x} \mapsto \sum_{c=1}^{N_C} \big\langle \boldsymbol{\Lambda}_c(\mathbf{x}_k), |\mathbf{W}_c\mathbf{x}| \big\rangle, \tag{5}$$

where $\boldsymbol{\Lambda} = (\Lambda_c)_{c=1}^{N_C} \colon \mathbb{R}^N \to (\mathbb{R}^N_{\geq 0})^{N_C}$ with $\boldsymbol{\Lambda}_c(\mathbf{x}) = \mathbf{B}_c^\top \boldsymbol{\psi}_c'(\mathbf{B}_c\,|\mathbf{W}_c\mathbf{x}|)$. If E is strictly convex and differentiable, then (4), namely the majorized problem at the k-th step, is strictly convex. Its unique minimum can be computed using the forward-backward splitting (FBS) algorithm [23]. Hence, we can rewrite (4) using the associated update operator $T_{\Lambda,\mathbf{W},\mathbf{y}} \colon \mathbb{R}^N \to \mathbb{R}^N$ as

$$\mathbf{x}_{k+1} = T_{\Lambda,\mathbf{W},\mathbf{y}}(\mathbf{x}_k). \tag{6}$$

In Theorem 4, we prove that the iterations (6) converge to a critical point of the underlying problem (1).

In (5), we can interpret $\boldsymbol{\Lambda}_c$ as a channel-wise spatial adaption of the regularization strength, which is attentive to image structures. This viewpoint of solution-driven spatial adaptivity [24, 25] serves as a starting point for the generalization of the MMR model in Section 3.2. More precisely, we propose to replace $\boldsymbol{\Lambda}$ in (5) with a more expressive convolutional neural network $\tilde{\Lambda}$. For this, we relax the constraints on the activation functions and linear operators in the mask generator Λ associated with (3), see Figure 1. This leads to the solution-adaptive fixed-point iterations (SAFI) as reconstruction scheme, which involves

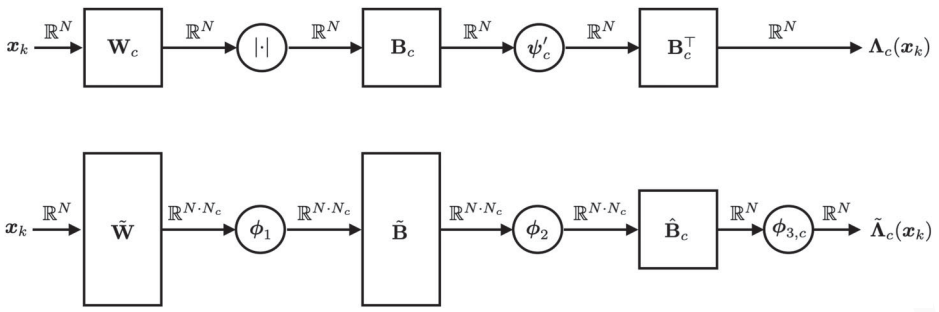

Figure 1. Mask-generation architecture of the majorization-minimization (top) and the solution-driven (below) setting. Above each arrow, we denote the signal dimension at the corresponding stage.

the regularizers

$$\mathcal{R}_{\mathrm{SAFI},k}\colon \mathbf{x} \mapsto \sum_{c=1}^{N_C} \left\langle \tilde{\Lambda}_c(\mathbf{x}_k), |\mathbf{W}_c\mathbf{x}| \right\rangle. \tag{7}$$

In (7), $\tilde{\Lambda}\colon \mathbb{R}^N \to ([0,1]^N)^{N_C}$ is a 3-layer network with $\tilde{\Lambda}_c(\mathbf{x}) = \boldsymbol{\phi}_{3,c}(\hat{\mathbf{B}}_c\boldsymbol{\phi}_2(\tilde{\mathbf{B}}\boldsymbol{\phi}_1(\tilde{\mathbf{W}}\mathbf{x})))$. Its convolutional operators have dimensions $\tilde{\mathbf{W}} \in \mathbb{R}^{(N_C\cdot N),N}$, $\tilde{\mathbf{B}} \in \mathbb{R}^{(N_C\cdot N),(N_C\cdot N)}$, and $\hat{\mathbf{B}}_c \in \mathbb{R}^{N,(N_C\cdot N)}$. The activation functions $\boldsymbol{\phi}_1(\mathbf{x}) = (\phi_{1,\lceil k/N\rceil}(x_k))_{k=1}^{N_C\cdot N}$ and $\boldsymbol{\phi}_2(\mathbf{x}) = (\phi_{2,\lceil k/N\rceil}(x_k))_{k=1}^{N_C\cdot N}$ share linear splines $\phi_{r,c} \in \mathcal{C}(\mathbb{R})$ on input blocks of size N. The final activation functions are $\boldsymbol{\phi}_{3,c}(\mathbf{x}) = (\phi_{3,c}(x_k))_{k=1}^{N}$, where each $\phi_{3,c} \in \mathcal{C}(\mathbb{R})$ is composed of a linear spline and a Sigmoid function. The latter enforces that the entries of each $\tilde{\Lambda}_c$ remain in $[0,1]$. For the regularizers (7), the minimization problem (4) is still strictly convex. Therefore, each update (6) in the pipeline is numerically tractable and gives rise to an update operator $T_{\tilde{\Lambda},\mathbf{W},\mathbf{y}}\colon \mathbb{R}^N \to \mathbb{R}^N$. In Theorem 5, we prove that $T_{\tilde{\Lambda},\mathbf{W},\mathbf{y}}\colon \mathbb{R}^N \to \mathbb{R}^N$ admits at least one fixed point. In this relaxed setting, the convergence of the SAFI scheme to a fixed point is encouraged by the use of regularization techniques during training [26]. The parameterization details for the architectures (5) and (7) are given in Section 3.3. The learning of the associated parameters on denoising problems is discussed in Section 4.

Our numerical evaluation for both denoising and MRI reconstruction in Section 5 indicates that the learning of the parameters $\mathbf{W}$, $\mathbf{B}$, and $\boldsymbol{\psi}$ in (3) leads to a reconstruction performance similar to that of the weakly convex model from [17] with $\rho = 1$. By setting $\mathbf{B}_c = \mathbf{Id}$ in (5), we obtain a weakly convex regularizer without a bound on ρ as special case. Hence, our theoretical analysis, corroborated by the numerical results, leads to yet another reasonable explanation for the performance gain of the weakly convex model [17] over the convex one [16]. With the more general regularizer (7) associated to SAFI, the performance gets similar to that of [27], despite the much simpler mask generator $\tilde{\Lambda}$. As in previous works, we observe that the learned regularizers generalize well to the previously unseen inverse problems. Finally, conclusions are drawn in Section 6.

1.2. Relation to previous work

Our regularizer $\mathcal{R}_{\mathrm{MMR}}$ relies on the architecture (2) from [17], where we add the inner activation $|\cdot|$ and the nonnegative convolutional matrices $\mathbf{B}_c$. The decomposition $\psi_c = \mu\psi_{c,\mathrm{cvx}} + \psi_{c,\mathrm{ccv}}$ is proposed in [17] with a convex $\psi_{c,\mathrm{cvx}} \in \mathcal{C}^{1,1}_{\geq 0}(\mathbb{R})$, a concave $\psi_{c,\mathrm{ccv}} \in \mathcal{C}^{1,1}_{\geq 0}(\mathbb{R})$ with $(-\rho) \leq \psi''_{c,\mathrm{ccv}} \leq 0$ a.e., and $\mu \in \mathbb{R}_{\geq 0}$. The convex part $\psi_{c,\mathrm{cvx}}$ of the learned ψ_c is necessary to maintain differentiability at 0. Since our ψ_c only takes positive inputs, we do not have this issue and can drop the term $\psi_{c,\mathrm{cvx}}$. Further, we relax ρ from 1 to ∞ to fully explore the role of concavity. Given that the experimental results are similar, we do not expect that the inclusion of a convex part $\psi_{c,\mathrm{cvx}}$ in (3) leads to a significant gain in performance.

For the regularizer $\mathcal{R}_{\mathrm{SAFI},k}$, we use the absolute value $|\cdot|$ instead of non-convex potentials, as proposed in [27, 28]. Hence, the subproblem (4) for each SAFI update is convex and the deployed optimization algorithm converges to a minimizer. This is stronger than the mere convergence to stationary points of [27]. Since we learn $\mathbf{W}$, our $\mathcal{R}_{\mathrm{SAFI},k}$ generalizes the data-adaptive total-variation model in [20]. Moreover, in contrast to these approaches, we iteratively refine the mask in $\mathcal{R}_{\mathrm{SAFI},k}$ based on $\mathbf{x}_{k+1} = T_{\tilde{\Lambda},\mathbf{W},\mathbf{y}}(\mathbf{x}_k)$. This leads to implicit depth, which is a possible explanation for why complex generators $\tilde{\Lambda}$ are not required in our framework.

The majorization-minimization (MM) perspective also shows up in [21], which deploys MM iterations to minimize a spatially adaptive model that is similar to [27, 28]. To ensure closed-form solutions for the minimization problems (4), the authors deploy $|\cdot|^2$ as potentials instead of $|\cdot|$ in (7). In contrast to the SAFI approach, their masks $\tilde{\Lambda}(\mathbf{x}_k)$ for the MM iterations are induced completely by the underlying regularizer.

2. Preliminaries

Throughout this work, $\mathcal{X} \subseteq \mathbb{R}^N$ denotes a closed convex set.

2.1. Concave functions

A function $f\colon \mathcal{X} \to \mathbb{R}$ is said to be *concave* if it satisfies

$$f\big(\alpha\mathbf{x}_1 + (1-\alpha)\mathbf{x}_2\big) \geq \alpha f(\mathbf{x}_1) + (1-\alpha) f(\mathbf{x}_2), \qquad \forall \mathbf{x}_1, \mathbf{x}_2 \in \mathcal{X}, \quad \forall \alpha \in [0,1]. \tag{8}$$

If $\mathcal{X}$ is open and $f \in \mathcal{C}^1(\mathcal{X})$, then f is concave if and only if its gradient ∇f satisfies

$$\big\langle \nabla f(\mathbf{x}_1) - \nabla f(\mathbf{x}_2), \mathbf{x}_1 - \mathbf{x}_2 \big\rangle \leq 0, \qquad \forall \mathbf{x}_1, \mathbf{x}_2 \in \mathcal{X}. \tag{9}$$

In the special case $N = 1$, condition (9) simply states that the derivative f' is non-increasing on $\mathcal{X}$. Another useful property is that any differentiable concave

function f is upper-bounded by its first-order Taylor expansion

$$f(\mathbf{x}_1) \leq f(\mathbf{x}_2) + \langle \nabla f(\mathbf{x}_2), \mathbf{x}_1 - \mathbf{x}_2 \rangle, \qquad \forall \mathbf{x}_1, \mathbf{x}_2 \in \mathcal{X}. \tag{10}$$

A function $f\colon \mathcal{X} \to \mathbb{R}$ is *convex* if and only if $(-f)$ is concave.

2.2. Majorization-minimization algorithm

For a deeper exposition to MM algorithms, we refer to [29–31]. Here, we only collect some basic definitions and the core results. For a continuous $f\colon \mathcal{X} \to \mathbb{R}$, we investigate the problem

$$\arg\min_{\mathbf{x}\in\mathcal{X}} f(\mathbf{x}). \tag{11}$$

The idea behind MM algorithms is to replace f by a sequence of (approximating) majorizations $g(\cdot, \mathbf{x}_k)$, $\mathbf{x}_k \in \mathcal{X}$ for which the computation of a (global) minimizer is tractable. A function $g\colon \mathcal{X} \times \mathcal{X} \to \mathbb{R}$ is said to be a *majorization* of $f\colon \mathcal{X} \to \mathbb{R}$ if it satisfies

i) the upper-bound $f(\mathbf{x}) \leq g(\mathbf{x}, \mathbf{x}_k), \quad \forall \mathbf{x}, \mathbf{x}_k \in \mathcal{X}$;
ii) and the local tight bound $g(\mathbf{x}_k, \mathbf{x}_k) = f(\mathbf{x}_k), \quad \forall \mathbf{x}_k \in \mathcal{X}$.

Next, we introduce the formal MM algorithm together with a convergence result [31, 32].

Theorem 1. *For a continuous $f\colon \mathcal{X} \to \mathbb{R}$ with majorization $g\colon \mathcal{X} \times \mathcal{X} \to \mathbb{R}$ and a starting point $\mathbf{x}_1 \in \mathcal{X}$, the MM sequence is given by*

$$\mathbf{x}_{k+1} \in \arg\min_{\mathbf{x}\in\mathcal{X}} g(\mathbf{x}, \mathbf{x}_k), \tag{12}$$

and the function values $f(\mathbf{x}_k)$ are non-increasing. If g is continuous, f and every $g(\cdot, \mathbf{x}_k)$ is continuously differentiable, and the sub-level set $\{\mathbf{x} \in \mathcal{X} : f(\mathbf{x}) \leq f(\mathbf{x}_1)\}$ is compact, then all accumulation points of $\{\mathbf{x}_k\}_{k\in\mathbb{N}}$ are critical points of f. Moreover, if the set $\mathcal{X}^* = \{\mathbf{x} : \langle \nabla f(\mathbf{x}), \mathbf{z} - \mathbf{x} \rangle \geq 0,\ \forall \mathbf{z} \in \mathcal{X}\}$ is a singleton or if $\mathcal{X}^*$ is discrete and $\lim_{k\to\infty} \|\mathbf{x}_{k+1} - \mathbf{x}_k\|_2 \to 0$, then the MM iterations $\{\mathbf{x}_k\}_{k\in\mathbb{N}}$ converge to a critical point of f.

Remark 1. The condition that $\mathcal{X}^*$ is a singleton is met if f is strongly convex, namely if $(f - \frac{\sigma}{2}\|\cdot\|_2^2)$ is convex for some $\sigma \in \mathbb{R}_+$. Hence, we get in this setting global convergence guarantees that are similar to those of convex-minimization algorithms. Q6

2.3. Γ-Convergence

Here, we recall the basic concepts of Γ-convergence within our Euclidean framework and refer to [33] for a more detailed exposition. A family of functions $\{J_k\}_{k\in\mathbb{N}}$ with $J_k\colon \mathcal{X} \to [0, \infty]$ is said to Γ-converge to $J\colon \mathcal{X} \to [0, \infty]$ if the following two conditions are fulfilled for every $\mathbf{x} \in \mathcal{X}$:

i) for all $\mathbf{x}_k \to \mathbf{x}$, it holds that $J(\mathbf{x}) \leq \liminf_{k\to\infty} J_k(\mathbf{x}_k)$;
ii) for every $\mathbf{x} \in \mathcal{X}$, there is a sequence $\{\mathbf{x}_k\}_{k\in\mathbb{N}}$ with $\mathbf{x}_k \to \mathbf{x}$ and $\limsup_{k\to\infty} J_k(\mathbf{x}_k) \leq J(\mathbf{x})$.

The importance of Γ-convergence is captured by Theorem 2. Recall that a family of functions $J_k\colon \mathcal{X} \to \mathbb{R}$ is equi-coercive if it is bounded from below by a coercive function.

Theorem 2 (Theorem of Γ-convergence [33]). *Let $\{J_k\}_{k\in\mathbb{N}}$ be an equi-coercive family of functions $J_k\colon \mathcal{X} \to \mathbb{R}$. If J_k Γ-converges to J, then it holds that*
i) the optimal function values converge $\lim_{k\to\infty} \inf_{\mathbf{x}\in\mathcal{X}} J_k(\mathbf{x}) = \inf_{\mathbf{x}\in\mathcal{X}} J(\mathbf{x})$;
ii) all accumulation points of the minimizers of J_k are minimizers of J.

In particular, if all the J_k and J have unique minimizers, then Theorem 2 directly implies convergence of the minimizers of the J_k to the one of J.

3. New perspectives on ridge-based regularization

First, we provide a novel perspective on weakly convex ridge regularizers [17] through the MMR model. Based on this perspective, we then derive our more general SAFI reconstruction scheme.

3.1. Majorization-minimization regularization

For the MMR model, we specify $\mathcal{R}$ in the generic problem (1) as (3) and choose E as the squared norm. Moreover, we allow for linear constraints by minimizing over a closed convex polytope $\mathcal{X} \subset \mathbb{R}^N$. This leads to the problem

$$\arg\min_{\mathbf{x}\in\mathcal{X}} f(\mathbf{x}) \coloneqq \left(\frac{1}{2}\left\|\mathbf{H}\mathbf{x}-\mathbf{y}\right\|_2^2 + \lambda \sum_{c=1}^{N_C} \left\langle \mathbf{1}_N, \boldsymbol{\psi}_c\left(\mathbf{B}_c|\mathbf{W}_c\mathbf{x}|\right)\right\rangle\right). \tag{13}$$

First, we establish the existence of minimizers for (13).

Theorem 3. *Let $\psi_c\colon \mathbb{R} \to \mathbb{R}_{\geq 0}$, $c = 1, \ldots, N_C$, be continuous and piecewise-polynomial functions with finitely many pieces, and let $\mathcal{X} \subset \mathbb{R}^N$ be a closed convex polytope. Then, problem (13) admits a minimizer.*

Proof Each ψ_c partitions $\mathbb{R}$ into finitely many closed[1] intervals $(I_c^m)_{m=1}^{L_c}$ on which it is a polynomial. Hence, if we denote the n-th row of $\mathbf{B}_c$ by $\mathbf{B}_{c,n}$, each $\boldsymbol{\psi}_c(\mathbf{B}_{c,n}|\mathbf{W}_c\cdot|)$ partitions $\mathcal{X}$ into L_c closed unions of polytopes $\Omega_{c,n}^m = \{\mathbf{x} \in \mathbb{R}^N : \mathbf{B}_{c,n}|\mathbf{W}_c\mathbf{x}| \in I_c^m\}$. Based on these, we can further partition $\mathcal{X}$ into finitely many closed polytopes, each of which is contained in one of the $\cap_{c,n=1}^{N_C,N} \Omega_{c,n}^{m_{c,n}}$, where $m_{c,n} \in \{1, \ldots, L_c\}$, and on which all the $\mathbf{B}_{c,n}|\mathbf{W}_c\cdot|$ are linear. The infimum in (13) is the infimum of f on (at least) one of these polytopes, say P.

[1] Such a partition with closed interval exists because ψ_c is continuous.

Now, we pick a minimizing sequence $(\mathbf{x}_k)_{k\in\mathbb{N}} \subset P$. Due to the coercivity of $\|\cdot\|_2^2$, we get that the sequence $(\mathbf{H}\mathbf{x}_k)_{k\in\mathbb{N}}$ remains bounded. By construction, there exist diagonal matrices $\mathbf{D}_c \in \mathbb{R}^{N,N}$ such that $\mathbf{B}_{c,n}|\mathbf{W}_c\mathbf{x}| = \mathbf{B}_{c,n}\mathbf{D}_c\mathbf{W}_c\mathbf{x}$ for every $n = 1,\dots,N$ and $\mathbf{x} \in P$. Let $\mathbf{M}$ be the matrix which is the vertical concatenation of $\mathbf{H}$ and all the $\mathbf{B}_{c,n}\mathbf{D}_c\mathbf{W}_c$ with c and n such that $(\mathbf{B}_{c,n}\mathbf{D}_c\mathbf{W}_c\mathbf{x}_k)_{k\in\mathbb{N}}$ remains bounded. Since the sequence $(\mathbf{M}\mathbf{x}_k)_{k\in\mathbb{N}}$ is bounded, we can extract a convergent subsequence with limit $\mathbf{u} \in \mathrm{ran}(\mathbf{M})$. The associated set

$$Q = \{\mathbf{x} \in \mathbb{R}^N \colon \mathbf{M}\mathbf{x} = \mathbf{u}\} = \{\mathbf{M}^\dagger\mathbf{u}\} + \ker(\mathbf{M}) \tag{14}$$

is a closed polytope. It holds that

$$\begin{aligned}\mathrm{dist}(\mathbf{x}_k, Q) &= \mathrm{dist}\big(\mathbf{M}^\dagger\mathbf{M}\mathbf{x}_k + \mathrm{P}_{\ker(\mathbf{M})}(\mathbf{x}_k), Q\big)\\ &\le \mathrm{dist}(\mathbf{M}^\dagger\mathbf{M}\mathbf{x}_k, \mathbf{M}^\dagger\mathbf{u}) \to 0\end{aligned} \tag{15}$$

as $k \to +\infty$ and, thus, that $\mathrm{dist}(P, Q) = 0$. The distance of P and Q is 0 if and only if $P \cap Q \neq \emptyset$ [34, Theorem 1]. For the $\mathbf{B}_{c,n}\mathbf{D}_c\mathbf{W}_c$ that were not added to $\mathbf{M}$, it holds that $\mathbf{B}_{c,n}|\mathbf{W}_c\mathbf{x}_k| \to \infty$. Hence, the interval $I_c^{m_{c,n}}$ has to be unbounded. Since ψ_c is a nonnegative polynomial on it, $\boldsymbol{\psi}_c(\mathbf{B}_{c,n}|\mathbf{W}_c\cdot|)$ has to be constant[2] on P and $\boldsymbol{\psi}_c(\mathbf{B}_{c,n}|\mathbf{W}_c\mathbf{x}_k|) = \boldsymbol{\psi}_c(\mathbf{B}_{c,n}|\mathbf{W}_c\mathbf{x}|)$ for every $\mathbf{x} \in P \cap Q$. Hence, any $\mathbf{x} \in P \cap Q$ is a minimizer for (13). □

Remark 2. A crucial ingredient for our proof is the architecture (3) with $|\cdot|$ as the inner nonlinearity. In general, it is much harder to guarantee the existence of minimizers for piecewise-polynomial functions [35].

The f in (13) is not necessarily convex. Hence, one should not attempt to solve (13) using conventional convex-optimization algorithms. Instead, one can use the majorization-minimization (MM) algorithm defined in (12) to search for stationary points. When f is convex, this algorithm converges to a minimizer. To apply the MM algorithm, we first show that the concavity of the $\psi_c \in \mathcal{C}^{1,1}_{\ge 0}(\mathbb{R})$ implies the concavity of $g_c(\mathbf{x}) = \langle \mathbf{1}_N, \boldsymbol{\psi}_c(\mathbf{B}_c\mathbf{x})\rangle$. Based on this property, we then construct a majorization of $\mathcal{R}_{\mathrm{MMR}}$.

Lemma 1. *If $\psi_c \in C^{1,1}_{\ge 0}(\mathbb{R})$, $c = 1,\dots,N_C$, then g_c is differentiable with $\nabla g_c(\mathbf{x}) = \mathbf{B}_c^\top \boldsymbol{\psi}_c'(\mathbf{B}\mathbf{x})$. Moreover, if ψ_c is also concave, then g_c is concave as well.*

Proof We have that $g_c(\mathbf{x}) = h(\boldsymbol{\psi}_c(\mathbf{B}_c\mathbf{x}))$ with $h(\mathbf{x}) = \langle \mathbf{1}_N, \mathbf{x}\rangle$. Hence, the Jacobian $\mathbf{J}_g$ is given through the chain rule as

$$\begin{aligned}\mathbf{J}_{g_c}(\mathbf{x}) &= \mathbf{J}_{h\circ\boldsymbol{\psi}_c\circ\mathbf{B}_c}(\mathbf{x}) = \mathbf{J}_h(\boldsymbol{\psi}_c(\mathbf{B}\mathbf{x}))\mathbf{J}_{\psi_c}(\mathbf{B}\mathbf{x})\mathbf{B} = \mathbf{1}_N^\top\mathbf{diag}(\boldsymbol{\psi}_c'(\mathbf{B}_c\mathbf{x}))\mathbf{B}_c\\ &= \boldsymbol{\psi}_c'(\mathbf{B}_c\mathbf{x})^\top\mathbf{B}_c,\end{aligned} \tag{16}$$

[2] A non-constant polynomial cannot have a finite limit at $\pm\infty$.

where **diag**: $\mathbb{R}^N \to \mathbb{R}^{N,N}$ returns a diagonal matrix whose diagonal elements are the input vector. As $\nabla g(\mathbf{x}) = \mathbf{J}_g(\mathbf{x})^\top$, the first claim readily follows. Further, it holds for $\mathbf{x}_1, \mathbf{x}_2 \in \mathbb{R}^N$ that

$$\begin{aligned}&\left\langle \nabla g_c(\mathbf{x}_1) - \nabla g_c(\mathbf{x}_2), \mathbf{x}_1 - \mathbf{x}_2 \right\rangle = \left\langle \mathbf{B}_c^\top \boldsymbol{\psi}_c'(\mathbf{B}_c\mathbf{x}_1) - \mathbf{B}_c^\top \boldsymbol{\psi}_c'(\mathbf{B}_c\mathbf{x}_2), \mathbf{x}_1 - \mathbf{x}_2 \right\rangle \\ =&\left\langle \mathbf{B}_c^\top (\boldsymbol{\psi}_c'(\mathbf{B}_c\mathbf{x}_1) - \boldsymbol{\psi}_c'(\mathbf{B}_c\mathbf{x}_2)), \mathbf{x}_1 - \mathbf{x}_2 \right\rangle = \left\langle \boldsymbol{\psi}_c'(\mathbf{B}_c\mathbf{x}_1) - \boldsymbol{\psi}_c'(\mathbf{B}_c\mathbf{x}_2), \mathbf{B}_c(\mathbf{x}_1 - \mathbf{x}_2) \right\rangle \\ =&\langle \boldsymbol{\psi}_c'(\mathbf{B}_c\mathbf{x}_1) - \boldsymbol{\psi}_c'(\mathbf{B}_c\mathbf{x}_2), \mathbf{B}_c\mathbf{x}_1 - \mathbf{B}_c\mathbf{x}_2 \rangle \leq 0, \end{aligned} \tag{17}$$

where the inequality stems from the concavity of ψ_c. By (9), the g_c are concave and the proof is complete. □

Now, we majorize g_c using its first-order Taylor expansion, see (10), and get that

$$\left\langle \mathbf{1}_N, \boldsymbol{\psi}_c(\mathbf{B}_c\mathbf{x}) \right\rangle \leq \left\langle \mathbf{1}_N, \boldsymbol{\psi}_c(\mathbf{B}_c\mathbf{x}_k) \right\rangle + \left\langle \mathbf{B}_c^\top \boldsymbol{\psi}_c'(\mathbf{B}\mathbf{x}_k), \mathbf{x} - \mathbf{x}_k \right\rangle, \qquad \forall \mathbf{x}_k \in \mathbb{R}^N. \tag{18}$$

With the change of variables $\mathbf{x} \mapsto |\mathbf{W}_c\mathbf{x}|$ and by summing over all c, we then get for any $\mathbf{x}_k \in \mathbb{R}^N$ that

$$\begin{aligned}\mathcal{R}_{\mathrm{MMR}}(\mathbf{x}) =& \sum_{c=1}^{N_\mathrm{C}} \left\langle \mathbf{1}_N, \boldsymbol{\psi}_c(\mathbf{B}_c|\mathbf{W}_c\mathbf{x}|) \right\rangle \leq \sum_{c=1}^{N_\mathrm{C}} \left\langle \mathbf{1}_N, \boldsymbol{\psi}_c(\mathbf{B}_c|\mathbf{W}_c\mathbf{x}_k|) \right\rangle \\ &+ \sum_{c=1}^{N_\mathrm{C}} \left\langle \boldsymbol{\Lambda}_c(\mathbf{x}_k), |\mathbf{W}_c\mathbf{x}| - |\mathbf{W}_c\mathbf{x}_k| \right\rangle, \end{aligned} \tag{19}$$

where $\boldsymbol{\Lambda}_c(\mathbf{x}) = \mathbf{B}_c^\top \boldsymbol{\psi}_c'(\mathbf{B}_c|\mathbf{W}_c\mathbf{x}|)$. Regarding the notation from Section 2, we choose

$$\begin{aligned}g(\mathbf{x}, \mathbf{x}_k) =& \frac{1}{2} \left\| \mathbf{H}\mathbf{x} - \mathbf{y} \right\|_2^2 + \lambda \sum_{c=1}^{N_\mathrm{C}} \left\langle \mathbf{1}_N, \boldsymbol{\psi}_c(\mathbf{B}_c|\mathbf{W}_c\mathbf{x}_k|) \right\rangle \\ &+ \lambda \sum_{c=1}^{N_\mathrm{C}} \left\langle \boldsymbol{\Lambda}_c(\mathbf{x}_k), |\mathbf{W}_c\mathbf{x}| - |\mathbf{W}_c\mathbf{x}_k| \right\rangle. \end{aligned} \tag{20}$$

It is easy to verify that the chosen $g(\mathbf{x}, \mathbf{x}_k)$ is a valid majorization of f in (13). Therefore, to compute a stationary point of f based on (12), we have to compute the estimates

$$\mathbf{x}_{k+1} \in \operatorname*{arg\,min}_{\mathbf{x} \in \mathcal{X}} \left(\frac{1}{2} \left\| \mathbf{H}\mathbf{x} - \mathbf{y} \right\|_2^2 + \lambda \sum_{c=1}^{N_\mathrm{C}} \left\langle \boldsymbol{\Lambda}_c(\mathbf{x}_k), |\mathbf{W}_c\mathbf{x}| \right\rangle \right). \tag{21}$$

As $\psi_c' \geq 0$, these majorizations of the original problem can be interpreted as spatially reweighted ℓ_1-analysis regularization, where the strength of the convex summands $\|\mathbf{W}_c\mathbf{x}_k\|_1$ is reweighted by $\boldsymbol{\Lambda}_c(\mathbf{x}_k)$. Accordingly, we rewrite

the convex problem of (21) in a more compact form as

$$\mathbf{x}_{k+1} \in \arg\min_{\mathbf{x}\in\mathcal{X}} \left(\frac{1}{2} \left\| \mathbf{H}\mathbf{x} - \mathbf{y} \right\|_2^2 + \lambda \left\| \mathbf{L}_k \mathbf{x} \right\|_1 \right) \qquad \text{with}$$

$$\mathbf{L}_k = \left[\mathbf{diag}(\mathbf{\Lambda}_c(\mathbf{x}_k))\mathbf{W}_c \right]_{c=1}^{N_C}. \tag{22}$$

In Algorithm 1, we provide an iterative procedure based on FBS [23, 36] to compute (22). To this end, we choose $\frac{1}{2}\left\| \mathbf{H}\cdot - \mathbf{y} \right\|_2^2$ as the differentiable part of the objective and $\lambda \left\| \mathbf{L}_k \cdot \right\|_1$ for the non-differentiable one. The most time-consuming part in Algorithm 1 for generic $\mathbf{L}$ is the evaluation of the proximal operator $\text{prox}_{\alpha\lambda\|\mathbf{L}\cdot\|_1}$ defined as

$$\text{prox}_{\alpha\lambda\|\mathbf{L}\cdot\|_1}(\mathbf{z}) = \arg\min_{\mathbf{w}\in\mathcal{X}} \left(\frac{1}{2} \left\| \mathbf{w} - \mathbf{z} \right\|_2^2 + \alpha\lambda \left\| \mathbf{L}\mathbf{w} \right\|_1 \right). \tag{23}$$

For computational purposes, it is better to consider the dual problem of (23), which we derive as in [37].

Proposition 1. *Let* $\text{Proj}_{\mathcal{X}} : \mathbb{R}^N \to \mathbb{R}^N$ *denote the orthogonal projection onto* $\mathcal{X}$. *If* $\hat{\mathbf{u}}$ *solves the problem*

$$\arg\min_{\mathbf{u}\in\mathbb{R}^{N_C\cdot N}} \left(\frac{1}{2} \left\| \mathbf{L}^\top \mathbf{u} - \mathbf{z} \right\|_2^2 - \frac{1}{2} \left\| \text{Proj}_{\mathcal{X}}\{\mathbf{L}^\top \mathbf{u} - \mathbf{z}\} - (\mathbf{L}^\top \mathbf{u} - \mathbf{z}) \right\|_2^2 \right) \quad \textit{subject to}$$

$$\left\| \hat{\mathbf{u}} \right\|_\infty \leq \alpha\lambda, \tag{24}$$

then $\text{Proj}_{\mathcal{X}}\{\mathbf{z} - \mathbf{L}^\top \hat{\mathbf{u}}\}$ *equals* (23).

Proof By duality, we have that

$$\alpha\lambda \left\| \mathbf{L}\mathbf{w} \right\|_1 = \max_{\mathbf{u}} \left\{ \mathbf{u}^\top (\mathbf{L}\mathbf{w}) : \left\| \mathbf{u} \right\|_\infty \leq \alpha\lambda \right\}. \tag{25}$$

Plugging this into (23) leads to

$$\min_{\mathbf{w}\in\mathcal{X}} \max_{u} \left(\frac{1}{2} \left\| \mathbf{w} \right\|_2^2 + \frac{1}{2} \left\| \mathbf{z} \right\|_2^2 + \mathbf{w}^\top (\mathbf{L}^\top \mathbf{u} - \mathbf{z}) \right) \quad \text{subject to } \left\| \mathbf{u} \right\|_\infty \leq \alpha\lambda$$

$$= \min_{\mathbf{w}\in\mathcal{X}} \max_{u} \left(\frac{1}{2} \left\| \mathbf{w} - (\mathbf{z} - \mathbf{L}^\top \mathbf{u}) \right\|_2^2 - \frac{1}{2} \left\| \mathbf{z} - \mathbf{L}^\top \mathbf{u} \right\|_2^2 + \frac{1}{2} \left\| \mathbf{z} \right\|_2^2 \right) \quad \text{subject to}$$

$$\left\| \mathbf{u} \right\|_\infty \leq \alpha\lambda. \tag{26}$$

Now, we can swap the min and max because the objective is convex in $\mathbf{w}$ and concave in $\mathbf{u}$ [38, Cor. 37.3.2]. Then, we directly get that $\mathbf{w} = \text{Proj}_{\mathcal{X}}\{\mathbf{z} - \mathbf{L}^\top \mathbf{u}\}$ is optimal for the inner minimization. By removing the constant term $\frac{1}{2}\left\| \mathbf{z} \right\|_2^2$ and a change of sign, we obtain (24). □

To solve (24), we apply once again FBS with the objective as the differentiable part and the constraints for the non-differentiable one, see Algorithm 2. Note that the subtrahend in (24) is the concatenation of a Moreau envelope with the affine map $\mathbf{u} \mapsto (\mathbf{L}^\top \mathbf{u} - \mathbf{z})$. Hence, its gradient reads $(\mathbf{L}(\mathbf{L}^\top \mathbf{u} - \mathbf{z}) -$

$\mathbf{L}\mathrm{Proj}_{\mathcal{X}}\{\mathbf{L}^\top\mathbf{u}-\mathbf{z}\}$), and the overall gradient of the objective in (24) with respect to $\mathbf{u}$ is

$$\mathbf{L}(\mathbf{z}-\mathbf{L}^\top\mathbf{u})-\mathbf{L}\big((\mathbf{z}-\mathbf{L}^\top\mathbf{u})-\mathrm{Proj}_{\mathcal{X}}\{\mathbf{z}-\mathbf{L}^\top\mathbf{u}\}\big)=\mathbf{L}\mathrm{Proj}_{\mathcal{X}}\{\mathbf{z}-\mathbf{L}^\top\mathbf{u}\}. \quad (27)$$

Our last ingredient is the saturating function $\mathbf{clip}_{[\kappa_1,\kappa_2]}\colon \mathbb{R}^N \to \mathbb{R}^N$, which is defined component-wise as

$$[\mathbf{clip}_{[\kappa_1,\kappa_2]}(\mathbf{a})]_k=\mathrm{clip}_{[\kappa_1,\kappa_2]}(a_k)=\begin{cases}\kappa_1, & a_k<\kappa_1\\ a_k, & \kappa_1\le a_k\le\kappa_2\\ \kappa_2, & a_k>\kappa_2.\end{cases} \quad (28)$$

Our proposed MMR scheme is summarized in Algorithm 3. It deploys the FBS (Algorithm 1) to solve the majorization-minimization problems (22). If $\mathbf{H}=\mathrm{Id}$, Algorithm 1 can be terminated after one step. The involved operator $\mathbf{Prox}_{\alpha\lambda\|\mathbf{L}_k\cdot\|_1}$ is computed using again the FBS (Algorithm 2). For both algorithms, our choice of $\{t_k\}_{k\in\mathbb{N}}$ ensures the convergence of the iterates [36]. Under the assumption of infinite precision in the sub-routines, Algorithm 3 finds indeed a critical point of f.

Theorem 4. *Assume that the estimates* (22) *are obtained exactly within Algorithm 3. Then,* $\{f(\mathbf{x}_k)\}_{k\in\mathbb{N}}$ *is non-increasing. If* $\mathbf{H}$ *is invertible, then* f *is coercive and all accumulation points of* $\{\mathbf{x}_k\}_{k\in\mathbb{N}}$ *are in the set of critical points*

$$\mathcal{X}^*=\Big\{\mathbf{x}_1\in\mathcal{X}:\langle\mathbf{H}^\top(\mathbf{H}\mathbf{x}_1-\mathbf{y}),\mathbf{x}_2-\mathbf{x}_1\rangle +\lambda\sum_{c=1}^{N_C}\big\langle\mathbf{\Lambda}_c(\mathbf{x}_1),|\mathbf{W}_c\mathbf{x}_2|-|\mathbf{W}_c\mathbf{x}_1|\big\rangle\ge 0,\ \forall\mathbf{x}_2\in\mathcal{X}\Big\}. \quad (29)$$

Algorithm 1 FBS for solving (22)

Input: filter matrix $\mathbf{L}$, previous minimizer $\mathbf{x}_1$, current iteration k_{out}
Parameters: maximal iteration number K_{FBS}, dynamic tolerance $\epsilon_{\mathrm{FBS}}=f_{\epsilon,\mathrm{FBS}}(k_{\mathrm{out}})>0$
Initialize: $t_1=1,\alpha=1/\|\mathbf{H}\|_2^2,\tilde{\mathbf{x}}_1=\mathbf{x}_1$
for $k=1$ **to** K_{FBS} **do**
  $\mathbf{x}_{k+1}=\mathbf{Prox}_{\alpha\lambda\|\mathbf{L}\cdot\|_1}(\tilde{\mathbf{x}}_k-\alpha\mathbf{H}^\top(\mathbf{H}\tilde{\mathbf{x}}_k-\mathbf{y}),k_{\mathrm{out}},k)$
  $t_{k+1}=(k+5)/3$
  $\tilde{\mathbf{x}}_{k+1}=\mathbf{x}_{k+1}+\frac{t_k-1}{t_{k+1}}(\mathbf{x}_{k+1}-\mathbf{x}_k)$;
  if $\|\mathbf{x}_{k+1}-\mathbf{x}_k\|_2<\epsilon_{\mathrm{FBS}}\|\mathbf{x}_k\|_2$ **then**
    break
  end if
end for
return $\mathbf{x}_{k+1}$

Algorithm 2 Computation of $\operatorname{prox}_{\gamma\|\mathbf{L}\cdot\|_1}$ based on the dual (24) using FBS

Input: vector $\mathbf{z} \in \mathbb{R}^N$, current iteration k_{out}, current iteration k_{FBS}
Parameters: maximal iteration number K_{prox}, dynamic tolerance $\epsilon_{\text{prox}} = f_{\epsilon,\text{prox}}(k_{\text{out}}, k_{\text{FBS}}) > 0$
Initialize: $\mathbf{u}_1 = \mathbf{L}\mathbf{z}$, $\mathbf{v}_1 = \mathbf{L}\mathbf{z}$, $\mathbf{x}_1 = \operatorname{Proj}_{\mathcal{X}}\{\mathbf{z} - \mathbf{L}^\top \mathbf{u}_1\}$, $t_1 = 1$, $\alpha = 1/\|\mathbf{L}\|_2^2$
for $k = 1$ **to** K_{prox} **do**
    $\mathbf{u}_{k+1} = \mathbf{clip}_{[-\gamma,\gamma]}(\mathbf{v}_k - \alpha\mathbf{L}\operatorname{Proj}_{\mathcal{X}}\{\mathbf{L}^\top\mathbf{v}_k - \mathbf{z}\})$
    $t_{k+1} = (k+5)/3$
    $\mathbf{v}_{k+1} = \mathbf{u}_{k+1} + \frac{t_k - 1}{t_{k+1}}(\mathbf{u}_{k+1} - \mathbf{u}_k)$
    $\mathbf{x}_{k+1} = \operatorname{Proj}_{\mathcal{X}}\{\mathbf{z} - \mathbf{L}^\top\mathbf{u}_{k+1}\}$
    if $\|\mathbf{x}_{k+1} - \mathbf{x}_k\|_2 < \epsilon_{\text{prox}}\|\mathbf{x}_k\|_2$ **then**
        break
    end if
end for
return $\mathbf{x}_{k+1}$

Algorithm 3 MMR scheme for (13)

Parameters: maximal iteration number K_{out}, tolerance $\epsilon_{\text{out}} > 0$
Initialize: $\mathbf{x}_1 = \mathbf{0}, \mathbf{L}_1 = [\mathbf{W}_c]_{c=1}^{N_C}$
for $k = 1$ **to** K_{out} **do**
    $\mathbf{x}_{k+1} = \mathbf{FBS}(\mathbf{L}_k, \mathbf{x}_k, k)$
    Compute $\mathbf{L}_{k+1} = [\mathbf{diag}(\Lambda_c(\mathbf{x}_{k+1}))\mathbf{W}_c]_{c=1}^{N_C}$
    if $\|\mathbf{x}_{k+1} - \mathbf{x}_k\|_2 < \epsilon_{\text{out}}\|\mathbf{x}_k\|_2$ **then**
        break
    end if
end for
return $\mathbf{x}_{k+1}$

Moreover, if $\mathcal{X}^*$ is a singleton or if $\mathcal{X}^*$ is discrete and $\lim_{k\to\infty}\|\mathbf{x}_{k+1} - \mathbf{x}_k\|_2 \to 0$, *then the MM iterates $\{\mathbf{x}_k\}_{k\in\mathbb{N}}$ converge to a critical point of f.*

Proof First, we introduce the auxiliary variable $\mathbf{z} \in \mathbb{R}_{\geq 0}^{N_C\cdot\mathbb{N}}$ with grouped components $\mathbf{z}_c = |\mathbf{W}_c\mathbf{x}| \in \mathbb{R}^N$ in (13) and investigate the equivalent problem

$$\arg\min_{\mathbf{x}\in\mathcal{X}} \min_{\mathbf{z}\in\mathbb{R}_{\geq 0}^{N_C\cdot\mathbb{N}}} \tilde{f}(\mathbf{x},\mathbf{z}) \coloneqq \frac{1}{2}\left\|\mathbf{H}\mathbf{x} - \mathbf{y}\right\|_2^2 + \lambda\sum_{c=1}^{N_C}\left\langle \mathbf{1}_N, \boldsymbol{\psi}_c\left(\mathbf{B}_c\mathbf{z}_c\right)\right\rangle \quad \text{subject to}$$
$$\mathbf{z}_c = |\mathbf{W}_c\mathbf{x}|. \tag{30}$$

Then, the majorizations will take the form

$$\tilde{g}\big((\mathbf{x},\mathbf{z}),(\mathbf{x}_k,\mathbf{z}_k)\big) = \frac{1}{2}\left\|\mathbf{H}\mathbf{x}-\mathbf{y}\right\|_2^2 + \lambda\sum_{c=1}^{N_C}\left\langle \mathbf{1}_N, \boldsymbol{\psi}_c(\mathbf{B}_c\mathbf{z}_{k,c})\right\rangle + \lambda\sum_{c=1}^{N_C}\left\langle \mathbf{B}_c^\top \boldsymbol{\psi}_c'(\mathbf{B}_c\mathbf{z}_{k,c}), \mathbf{z}-\mathbf{z}_{k,c}\right\rangle, \tag{31}$$

and their minimization subject to $\mathbf{z}_c = |\mathbf{W}_c\mathbf{x}|$ leads indeed to (4). Observe that $\tilde{g}$ are continuous. Further, both $\tilde{f}$ and the $\tilde{g}(\cdot,(\mathbf{x}_k,\mathbf{z}_k))$ are differentiable. Hence, we can apply Theorem 1 and the claim follows. □

3.2. Solution-adaptive fixed-point iterations

For the MMR model with (4), the mask generator $\Lambda\colon \mathbb{R}^N \to (\mathbb{R}^N_{\geq 0})^{N_C}$ allows for a successive spatial adaption of the regularization strength. So far, the architecture of each $\Lambda_c\colon \mathbb{R}^N \to \mathbb{R}^N_{\geq 0}$ is motivated by the MMR perspective. One might wonder if a more generic $\tilde{\Lambda}\colon \mathbb{R}^N \to ([0,1]^N)^{N_C}$ leads to improvements. This leads to the SAFI scheme based on (7). As the masks generated by $\tilde{\Lambda}_c$ are nonnegative, the resulting SAFI updates

$$\mathbf{x}_{k+1} \in \operatorname*{arg\,min}_{\mathbf{x}\in\mathcal{X}} J(\mathbf{x},\mathbf{x}_k) \coloneqq \frac{1}{2}\left\|\mathbf{H}\mathbf{x}-\mathbf{y}\right\|_2^2 + \lambda\sum_{c=1}^{N_C}\left\langle \tilde{\Lambda}_c(\mathbf{x}_k), |\mathbf{W}_c\mathbf{x}|\right\rangle \tag{32}$$

are minimizers of convex problems. Moreover, if $\mathbf{H}$ is invertible, then (32) is a singleton. Hence, this case gives rise to an update operator $T_{\Lambda,\mathbf{W},\mathbf{y}}\colon \mathcal{X} \to \mathcal{X}$. For the MMR framework in Section 3.1 with $\Lambda_c(\mathbf{x}) = \mathbf{B}_c^\top \boldsymbol{\psi}_c'(\mathbf{B}_c|\mathbf{W}_c\mathbf{x}|)$, we are guaranteed that the fixed-point iterates

$$\mathbf{x}_{k+1} = T_{\Lambda,\mathbf{W},\mathbf{y}}(\mathbf{x}_k) \tag{33}$$

are convergent. Moreover, the resulting fixed point is a critical point of problem (13).

If one is only interested in obtaining convergence of the iterates (33), this choice is overly constraining. For example, convergence can be guaranteed whenever $T_{\Lambda,\mathbf{W},\mathbf{y}}$ is nonexpansive. Any fixed point $\mathbf{x}^*$ of (33) is a critical point of (32) with $\mathbf{x}_k = \mathbf{x}^*$, and we cannot *improve* $\mathbf{x}^*$ by updating the $\tilde{\Lambda}_c$ anymore. In contrast to [27], the spatial adaptivity of the SAFI is driven by every estimate (32), and not only by an initial reconstruction based on the data $\mathbf{y}$. For the special case of total-variation regularization, this was therefore coined as solution-driven adaptivity instead of data-driven adaptivity [24, 25]. In general, the updates (33) are unrelated to the critical points of the non-convex minimization problem

$$\operatorname*{arg\,min}_{\mathbf{x}\in\mathcal{X}} \left(\frac{1}{2}\left\|\mathbf{H}\mathbf{x}-\mathbf{y}\right\|_2^2 + \lambda\sum_{c=1}^{N_C}\left\langle \tilde{\Lambda}_c(\mathbf{x}), |\mathbf{W}_c\mathbf{x}|\right\rangle\right), \tag{34}$$

where one also minimizes over the input of $\tilde{\Lambda}_c$.

Following the discussed ideas, we proceed as outlined in Figure 1 and Section 1, and replace $\Lambda_c(\mathbf{x})$ by the richer architecture $\tilde{\Lambda}_c(\mathbf{x}) = \boldsymbol{\phi}_{3,c}(\hat{\mathbf{B}}_c\boldsymbol{\phi}_2(\tilde{\mathbf{B}}\boldsymbol{\phi}_1(\tilde{\mathbf{W}}\mathbf{x})))$. As observed in [27], one expects that $\tilde{\Lambda}_c$ dampens the response of the $\mathbf{W}_c$ to structure and leaves it unchanged for noise or artifacts. Under some conditions, we can show that $T_{\tilde{\Lambda},\mathbf{W},\mathbf{y}}$ admits indeed at least one fixed point. Hence, the definition of reconstructions as fixed points of the operator $T_{\tilde{\Lambda},\mathbf{W},\mathbf{y}}$ makes sense.

Theorem 5. *Let* $\mathbf{H}$ *be invertible and let* $\sigma_{\min}$ *denote its smallest singular value. Then,* $T_{\tilde{\Lambda},\mathbf{W},\mathbf{y}}\colon \mathcal{X} \to \mathcal{X}$ *maps* $\mathcal{X}$ *into a ball centered at* $\mathbf{0}$ *with radius* $2\left\|\mathbf{y}\right\|_2/\sigma_{\min}$. *Further,* $T_{\tilde{\Lambda},\mathbf{W},\mathbf{y}}$ *admits a fixed point.*

Proof First, we investigate the range of $T_{\tilde{\Lambda},\mathbf{W},\mathbf{y}}$. By definition of $T_{\tilde{\Lambda},\mathbf{W},\mathbf{y}}$, it holds for any $\mathbf{x} \in \mathcal{X}$ that

$$\frac{1}{2}\left\|\mathbf{H}T_{\tilde{\Lambda},\mathbf{W},\mathbf{y}}(\mathbf{x}) - \mathbf{y}\right\|_2^2 \leq J(T_{\tilde{\Lambda},\mathbf{W},\mathbf{y}}(\mathbf{x}),\mathbf{x}) \leq J(\mathbf{0},\mathbf{x}) = \frac{1}{2}\left\|\mathbf{y}\right\|_2^2. \tag{35}$$

From this, we conclude that

$$\left\|T_{\tilde{\Lambda},\mathbf{W},\mathbf{y}}(\mathbf{x})\right\|_2 \leq \frac{1}{\sigma_{\min}}\left\|\mathbf{H}T_{\tilde{\Lambda},\mathbf{W},\mathbf{y}}(\mathbf{x})\right\|_2 \leq 2\frac{\left\|\mathbf{y}\right\|_2}{\sigma_{\min}}. \tag{36}$$

For the second part, we want to apply Brouwer's fixed-point theorem. To this end, we additionally need to prove that $T_{\tilde{\Lambda},\mathbf{W},\mathbf{y}}$ is continuous. Due to Theorem 2, it suffices to check equi-coercivity and the conditions for Γ-convergence of the family $J(\cdot,\mathbf{x})$ parameterized by $\mathbf{x} \in \mathcal{X}$. First, note that it holds for any $\mathbf{z} \in \mathcal{X}$ that

$$\frac{\sigma_{\min}^2}{4}\|\mathbf{z}\|_2^2 \leq \frac{1}{4}\|\mathbf{H}\mathbf{z}\|_2^2 \leq \frac{1}{2}\|\mathbf{H}\mathbf{z}-\mathbf{y}\|_2^2 + \|\mathbf{y}\|_2^2 \leq J(\mathbf{z},\mathbf{x}) + \|\mathbf{y}\|_2^2, \tag{37}$$

which implies equi-coercivity. Now, let $\mathbf{x}_k \to \mathbf{x}^*$ and $\mathbf{z}_k \to \mathbf{z}^*$. By the triangle inequality, we get that

$$|J(\mathbf{z}^*,\mathbf{x}^*) - J(\mathbf{z}_k,\mathbf{x}_k)| \leq |J(\mathbf{z}^*,\mathbf{x}^*) - J(\mathbf{z}^*,\mathbf{x}_k)| + |J(\mathbf{z}^*,\mathbf{x}_k) - J(\mathbf{z}_k,\mathbf{x}_k)|. \tag{38}$$

To obtain the lim inf inequality, it suffices to prove that the first two terms converge to 0 as $k \to \infty$. For the first one, we have that

$$|J(\mathbf{z}^*,\mathbf{x}^*) - J(\mathbf{z}^*,\mathbf{x}_k)| \leq \lambda \sum_{c=1}^{N_C}\left\langle|\tilde{\mathbf{\Lambda}}_c(\mathbf{x}^*) - \tilde{\mathbf{\Lambda}}_c(\mathbf{x}_k)|, |\mathbf{W}_c\mathbf{z}^*|\right\rangle \to 0 \tag{39}$$

because Λ_c is continuous. For the second one, we have that

$$\begin{aligned}|J(\mathbf{z}^*,\mathbf{x}_k) - J(\mathbf{z}_k,\mathbf{x}_k)| \leq{}& \frac{1}{2}\left|\|\mathbf{H}\mathbf{z}^*-\mathbf{y}\|_2^2 - \|\mathbf{H}\mathbf{z}_k-\mathbf{y}\|_2^2\right| \\ &+ \lambda\sum_{c=1}^{N_C}\left\langle|\tilde{\mathbf{\Lambda}}_c(\mathbf{x}_k)|, \left||\mathbf{W}_c\mathbf{z}^*| - |\mathbf{W}_c\mathbf{z}_k|\right|\right\rangle\end{aligned}$$

$$\leq \frac{1}{2}\left|\|\mathbf{H}\mathbf{z}^* - \mathbf{y}\|_2^2 - \|\mathbf{H}\mathbf{z}_k - \mathbf{y}\|_2^2\right| + \lambda \sum_{c=1}^{N_C} \left\langle \mathbf{1}_N, |\mathbf{W}_c\mathbf{z}^* - \mathbf{W}_c\mathbf{z}_k| \right\rangle. \tag{40}$$

Again, we conclude that this quantity converges to zero. Hence, we have established the lim inf inequality. For the lim sup inequality, we use the constant recovery sequence $\mathbf{x}_k = \mathbf{x}^*$, for which the claim follows as in (40). In summary, this implies that $T_{\tilde{\Lambda},\mathbf{W},\mathbf{y}}$ is continuous and that a fixed point exists. □

Remark 3. Based on a quasi-variational inequality perspective, the authors of [25] prove the uniqueness of fixed points for certain problems of the form (33). Unfortunately, their assumptions are hard to verify in practice for $\tilde{\Lambda}_c$. Hence, we do not pursue this direction further and only provide a proof of existence.

For finding a fixed point of the SAFI operator $T_{\tilde{\Lambda},\mathbf{W},\mathbf{y}}$, we propose to use the fixed-point iterations (33) detailed in Algorithm 4. Unfortunately, a proof of convergence for these iterations is highly nontrivial. In practice, we encourage this property by using a random number of iterations for the training of the model, as detailed in Section 4. Imposing Lipschitz constraints on the masks could potentially be helpful for proving the convergence of the fixed-point iterations. Note that, for our simple generator $\tilde{\mathbf{\Lambda}}$, we can efficiently enforce such constraints, as detailed in [39]. For Theorems 4 and 5, we require the invertibility of the forward operator $\mathbf{H}$ to define a single-valued update operator $T_{\tilde{\Lambda},\mathbf{W},\mathbf{y}}$. For its set-valued generalization, which naturally arises if we drop the invertibility assumption, a stability analysis of the defining problem (32) was recently established in [28]. Note that in the single-valued case, such results are often key to establish the existence of fixed points. Independent of any theoretical considerations, we observed a converging behavior of both MMR

Algorithm 4 SAFI scheme for (33)

Parameters: maximal iteration number K_{out}, tolerance $\epsilon_{\text{out}} > 0$
Initialize: $\mathbf{x}_1 = \mathbf{0}, \mathbf{L}_1 = [\mathbf{W}_c]_{c=1}^{N_C}$
for $k = 1$ **to** K_{out} **do**
  $\mathbf{x}_{k+1} = \mathbf{FBS}(\mathbf{L}_k, \mathbf{x}_k, k)$
  Compute $\mathbf{L}_{k+1} = [\mathbf{diag}(\tilde{\Lambda}_c(\mathbf{x}_{k+1}))\mathbf{W}_c]_{c=1}^{N_C}$
  if $\|\mathbf{x}_{k+1} - \mathbf{x}_k\|_2 < \epsilon_{\text{out}} \|\mathbf{x}_k\|_2$ **then**
    break
  end if
end for
return $\mathbf{x}_{k+1}$

and SAFI for the compressed-sensing MRI experiment in Section 5, where $\mathbf{H}$ is not invertible.

3.3. Parameterization of the learnable parameters

We now provide details of the parameterization for our two solution-adaptive regularizers. The regularization strength λ in (13) and (32) is learnable for the corresponding reconstruction models. For the MMR model (13) from Section 3.1, the remaining parameters are the linear operators $\{\mathbf{W}, \mathbf{B}\}$ and the concave potentials in Ψ. For the SAFI problem (32) from Section 3.2, the remaining parameters are the linear operators $\{\tilde{\mathbf{W}}, \tilde{\mathbf{B}}, \hat{\mathbf{B}}\}$ and the activation functions $\{\boldsymbol{\phi}_1, \boldsymbol{\phi}_2, \boldsymbol{\phi}_3\}$. Taking a closer look at Algorithms 1–3, we observe that we actually only need access to Ψ' and not to Ψ itself. Hence, we directly parameterize the derivatives Ψ' instead.

3.3.1. Parameterization of linear operators

All linear operators are constructed with the `Conv2d` module from PyTorch. Here, we only detail the construction for the output dimension N_C. More specifically, we decompose each operator into S stacked `Conv2d` modules; each with N_C output channels, a kernel size $(k_s \times k_s)$, and a group size G. This was observed to be more effective than the direct use of a single `Conv2d` module with a larger kernel size [16, 17]. Here, the group size G controls the potential transfer of information across the different channels. In particular, if $G = 1$, then each kernel of the kth layer, $k \in 2, \ldots, S$, is convolved with all the ones of the $(k-1)$th layer. If $G = N_C$, then each kernel is only convolved with the one of its channel.

3.3.2. Constrained linear operators

We impose constraints on some convolution kernels. All the kernels of $\mathbf{W}$ and $\tilde{\mathbf{W}}$ should have zero mean. To ensure this, let $\mathbf{w} \in \mathbb{R}^{k_s^2}$ contain the vectorized elements of the respective kernel. Then, we can use the parameterization $\mathbf{w} \mapsto (\mathbf{w} - (\mathbf{1}^\top \mathbf{w})/k_s^2)$, and optimize over unconstrained variables. For $\mathbf{B}$, we impose that the kernel elements are positive and sum to one. Let $\mathbf{b} \in \mathbb{R}^{k_s^2}$ be the vectorized kernel elements. Here, the implementation of the constraint is nonnegative with the parameterization $\mathbf{b} \mapsto (|\mathbf{b}| \, (\mathbf{1}^\top |\mathbf{b}|))$. Note that $|\cdot|$ is applied element-wise to $\mathbf{b}$.

3.3.3. Learnable activation functions

For the $\{\boldsymbol{\phi}_1, \boldsymbol{\phi}_2, \boldsymbol{\phi}_3\}$ in $\tilde{\Lambda}$, we rely on the learnable linear-spline framework introduced in [40]. More precisely, we use a uniform grid centered at 0 with

stepsize Δ and $2M+1$ points, $M \in \mathbb{N}$, and the B-spline of degree one defined as

$$\beta^1(x) = \begin{cases} 1-|x|, & x \in [-1,1] \\ 0, & \text{otherwise.} \end{cases} \tag{41}$$

Then, we parameterize each $\phi_{p,c}$ based on the vector $\mathbf{d}_{p,c} \in \mathbb{R}^{2M+1}$ of function values at the grid points as

$$\phi_{p,c}(x) = \begin{cases} d_{p,c,1} + \frac{d_{p,c,2}-d_{p,c,1}}{\Delta}(x+M\Delta), & x \in (-\infty, -M\Delta) \\ \sum_{k=-M}^{M} d_{p,c,k+M+1}\beta^1(x/\Delta - k), & x \in [-M\Delta, M\Delta] \\ d_{p,c,2M+1} + \frac{d_{p,c,2M+1}-d_{p,c,2M}}{\Delta}(x-M\Delta), & x \in (M\Delta, \infty). \end{cases} \tag{42}$$

In particular, $\phi_{p,c}$ is nonlinear on $[-M\Delta, M\Delta]$ and extrapolated linearly outside of this interval.

3.3.4. Concave potentials

For the MMR model, we parameterize the ψ_c', $c = 1, \ldots, N_C$, as

$$\psi_c'(x) = \mathrm{clip}_{[0,1]}\big(\sigma_c(r_c x)\big), \tag{43}$$

where $\sigma_c \colon \mathbb{R}_{\geq 0} \to \mathbb{R}$ are learnable linear splines and $r_c \in \mathbb{R}_{>0}$ are learnable scaling constants that adapt the range. To parameterize $\{r_c\}_{c=1}^{N_C}$, we use the `nn.Parameter` module of `PyTorch`. To ensure their positivity, we use $|r_c|$ instead of r_c in the implementation. As σ_c is only defined on $\mathbb{R}_{\geq 0}$, we parameterize it with its $M+1$ values on the nonnegative part of the grid from (42) denoted by $\mathbf{d}_c \in \mathbb{R}^{M+1}$. As ψ_c must be concave, its derivative ψ_c' is constrained to be non-increasing on $\mathbb{R}$. This can be achieved by using a non-increasing σ_c with $\sigma_c(0) = 1$. To enforce the condition $\sigma_c(0) = 1$, it suffices to fix $d_{c,0} = 1$. Let $\mathbf{D} \in \mathbb{R}^{M,M+1}$ be defined via $(\mathbf{D}\mathbf{d}_c)_{m-1} = (d_{c,m} - d_{c,m-1})$, $m = 2, \ldots, M+1$. If all elements of $\mathbf{D}\mathbf{d}_c$ are non-positive, then σ_c is non-increasing. To directly embed this constraint into the parameterization, we define

$$\boldsymbol{P}_{\downarrow}(\mathbf{d}_c) = \mathbf{S}\,\mathrm{clip}_{[-\infty,0]}(\mathbf{D}\mathbf{d}_c) + \mathbf{1}_{M+1}, \tag{44}$$

where $\mathbf{S} \in \mathbb{R}^{M+1,M}$ with $(\mathbf{S}\mathbf{d}_c)_m = \sum_{k=1}^{m-1} d_{c,k}$, $m = 1, \ldots, M+1$. By projecting the unconstrained coefficients $\mathbf{d}_c \in \mathbb{R}^{M+1}$ to $\boldsymbol{P}_{\downarrow}(\mathbf{d}_c)$, we ensure that the corresponding σ_c is non-increasing. With the proposed parameterization, the associated concave profiles $\psi_c \colon \mathbb{R}_{\geq 0} \to \mathbb{R}_{\geq 0}$ satisfy the following properties.

i) They are piecewise-quadratic, nonnegative, and increasing.

ii) We have that $0 \leq \psi_c'(x) \leq 1$ for all $x \in \mathbb{R}_{\geq 0}$ and $\psi_c'(0) = 1$.

4. Architecture and training

For our regularizers (5) and (7), we now describe the learning of the parameters detailed in Section 3.3. For both architectures and their respective

reconstruction routines Algorithm 3 and 4, we learn them by solving a denoising problem with additive white Gaussian noise of standard deviation $\sigma \in \{5/255, 15/255, 25/255\}$. Since the training procedure is exactly the same for both architectures, we restrict our discussion to (5).

Let $\{\mathbf{x}^m\}_{m=1}^M$ with $\mathbf{x}^m \in \mathbb{R}^{40\times 40}$ be a set of clean patches from the grayscale BSD500 dataset [41], and let $\{\mathbf{y}^m\}_{m=1}^M = \{\mathbf{x}^m + \mathbf{n}_\sigma^m\}_{m=1}^M$ be some noisy versions, where $\mathbf{n}_\sigma^m$ is a realization of the noise. For all experiments, we train with $M = 238400$ patches. In the following, we collect all the learnable parameters of (5) in the variable θ. Given a noisy patch $\mathbf{y}^m$, we obtain its denoised version $D_{\theta,\sigma}^{n_1,n_2,n_3}(\mathbf{y}^m)$ by applying Algorithm 3 with $K_{\text{out}} = n_1$ steps. As discussed in Section 3.1, fixing $K_{\text{FBS}} = n_2 = 1$ for Algorithm 1 suffices to guarantee convergence in the denoising case. For calculating the involved $\text{prox}_{\gamma\|\mathbf{L}\cdot\|_1}$ based on Algorithm 2, we use $K_{\text{FBS}} = n_3$ steps. During the training phase, all the tolerances are set to (-1) to ensure that the maximum number of steps is used. Now, we propose to learn the optimal parameters $\hat{\theta}$ in $D_{\theta,\sigma}^{n_1,1,n_3}$ based on the empirical risk

$$\hat{\theta} \in \arg\min_{\theta} \sum_{n_1=4}^{6} \sum_{n_3=10}^{12} \sum_{m=1}^{M} \left\| D_{\theta,\sigma}^{n_1,1,n_3}(\mathbf{x}^m + \mathbf{n}_\sigma^m) - \mathbf{x}^m \right\|_2^2. \tag{45}$$

To solve (45), we use the ADAM optimizer [42] with a learning rate of 10^{-3} and a batch size of 128 patches that are reconstructed for a single pair (n_1, n_3). This pair is uniformly drawn at random from one of the possible values for each batch. As documented in [26], using a random numbers of iterations has a regularizing effect when unrolling fixed-point iterations. In particular, this prevents the models from getting overfitted to a specific number of iterations. We perform 40 training epochs and reduce the learning rate by a factor of 0.1 at the 5th and 10th epoch. After each epoch, we evaluate the performance of the model on the Set12 validation data, and choose the output model as the one with the best performance. As a result, we obtain the regularization strength in (13) as well as the linear layers and the potentials that appear in (5).

Remark 4. Instead of pursuing an unrolling approach for training, one can also aim to minimize (45) for $n_1 = n_3 = \infty$ with implicit-differentiation techniques [43, 44]. However, as already observed in [16], it is usually unnecessary to fully compute the involved fixed points in $D_{\theta,\sigma}^{n_1,1,n_3}(\mathbf{y}^m)$ to learn good parameters θ for the regularizer. Moreover, as we have two nested fixed-point problems, namely the problems (22) and (23), this easily gets prohibitively expensive.

4.1. Architecture and initialization

We use the default `nn.Conv2D` initialization for every linear layer, and initialize λ as 10^{-4}. Below, we discuss the remaining hyperparameters and initializations.

MMR model: For the operators $\{\mathbf{W}_c\}_{c=1}^{N_C}$, we proceed as described in Section 3.3 with $N_C = 64$, $S = 2$, $k_s = 7$, and $G = 1$. Further, we force the kernels to be zero-mean. For the modeling of $\{\mathbf{B}_c\}_{c=1}^{N_C}$, we use a linear layer with $N_C = 64$, $S = 2$, $k_s = 7$, and $G = 64$. Here, we enforce that the kernels are positive and normalized. For each concave potential ψ_c, we use 21 gridpoints, which corresponds to $M = 20$ and $\Delta = 0.05$. We initialize the expansion coefficients of the splines with zero, except $d_{c,0} = 1$. Every r_c is initially set to one.

SAFI scheme: To model the linear layers $\{\mathbf{W}_c\}_{c=1}^{N_C}$, we choose $N_C = 64$, $S = 2$, $k_s = 7$, and $G = 1$. We enforce that the kernels are zero-mean. To model $\{\mathbf{W}_{c,1}\}_{c=1}^{N_C}$, $\{\mathbf{W}_{c,2}\}_{c=1}^{N_C}$, and $\{\mathbf{W}_{c,3}\}_{c=1}^{N_C}$, we use linear layers with $N_C = 64$, $S = 1$, $k_s = 7$ and $G = 1$. Further, we use linear splines with no constraints to parameterize $\{\phi_{1,c}\}_{c=1}^{N_C}$, $\{\phi_{2,c}\}_{c=1}^{N_C}$, and $\{\phi_{3,c}\}_{c=1}^{N_C}$, as described in Section 3.3. For this, we use 21 knots, which correspond to $M = 10$ and $\Delta = 0.1$. We initialize all expansion coefficients of the splines with zero.

4.2. Fine tuning

The interpretability of the learned denoiser $D_{\theta,\sigma}^{n_1,1,n_3}$ with the small n_1 and n_2 from the training stage is, however, limited. In particular, we only perform a partial minimization (unrolling) of (13). To remain within our theoretical setup, we need to iterate Algorithms 1–3 until convergence during the evaluation phase. Doing so without modifying the regularization strength λ in (13) has led to over-smoothing in our experiments. Moreover, the training of the model is for denoising only and not necessarily adapted to other inverse problems with $\mathbf{H} \neq \mathbf{Id}$. To deal with these issues, we propose to deploy (5) with the previously learned parameters for (13) and solely fine-tune λ on a small set of task-specific validation data with a coarse-to-fine grid search, as described in [16]. Hence, we get two different denoisers for our numerical evaluation: first, the unrolled version $D_{\theta,\sigma}^{n_1,1,n_3}$, which is exactly what we have trained for, but which is not necessarily a fixed point of (33); second, the *exact* fixed point $D_{\theta,\sigma}^{\infty,1,\infty}$ with an adapted λ, which uses more iterations and for which our theoretical analysis holds. For the inverse problems, we use the parameters of the denoising models that are trained with $\sigma = 15/255$. Here, we only have the fixed-point-based reconstruction operator as we do not train for the task.

4.3. Algorithm hyperparameters for evaluation

We aim to iterate Algorithms 1–3 until convergence, namely, up to machine precision. Still, we enforce an upper bound on the number of iterations for all algorithms. This ensures that we always remain within a reasonable computational budget. Independent of $\mathbf{H}$, we set $K_{\text{prox}} = 500$ and $K_{\text{out}} = 10$. For the denoising case with $\mathbf{H} = \mathbf{Id}$, we know that $K_{\text{FBS}} = 1$ suffices for convergence.

There, we use the iteration-dependent tolerance

$$f_{\epsilon,\text{prox}}(k_{\text{out}}, k_{\text{FBS}}) = \begin{cases} 10^{-3}(0.01)^{\frac{k_{\text{out}}}{5}}, & k_{\text{out}} \leq 5 \\ 10^{-5}, & k_{\text{out}} > 5. \end{cases} \tag{46}$$

For $\mathbf{H} \neq \mathbf{Id}$, we set $K_{\text{FBS}} = 1000$ and use the iteration-dependent tolerances

$$f_{\epsilon,\text{FBS}}(k_{\text{out}}) = \begin{cases} 10^{-3}(0.01)^{\frac{k_{\text{out}}}{5}}, & k_{\text{out}} \leq 5 \\ 10^{-5}, & k_{\text{out}} > 5, \end{cases} \quad \text{and}$$

$$f_{\epsilon,\text{prox}}(k_{\text{out}}, k_{\text{FBS}}) = \begin{cases} 3\epsilon_{\text{FBS}}(\frac{1}{9})^{\frac{k_{\text{FBS}}}{50}}, & k_{\text{FBS}} \leq 50 \\ \frac{\epsilon_{\text{FBS}}}{3}, & k_{\text{FBS}} > 50. \end{cases} \tag{47}$$

To summarize, for efficiency, the inner subproblems are solved with lower precision early on, while the precision for the later stages is higher to ensure convergence. This is a common technique to accelerate majorization minimization models [31] and the FBS algorithm [45, 46].

5. Numerical results

First, we present denoising results as this is our training problem. Then, we deploy the regularizers (5) and (7), which we learned for denoising, to a MRI problem without additional training. For this, we need to adapt the λ in (13) and (32) on some (small) validation set. With this task shift, we want to underline the universality of our approach. The code for our experiments is available on GitHub[3]. In this section, the images of each row in a figure are plotted with the same grayscale.

5.1. Denoising

Before investigating the qualitative behavior of the proposed regularizers (5) and (7), we first compare their quantitative performance with competing learned regularization methods. The achieved PSNR values on the BSD68 test set are given in Table 1. There, we compare our approach with BM3D, which is a popular baseline [47]. We also compare with the WCRR model of [17] and its spatially adaptive extension SARR [27], which both motivated our approach. Finally, we include Prox-DRUNet [18] as a regularizer with a deeper parameterization and some (loose) theoretical guarantees. For the SAFI, we report results for both the training (SAFI_5) and the evaluation configuration (SAFI) with the λ adaption. The performance difference between them is negligible. Hence, from now on, we solely use the evaluation configuration. Additionally, we provide a visual denoising comparison for the *castle* image in Figure 2. Here, we observe that our SAFI scheme recovers the tip of the spire, which is in general hard to achieve for

[3] https://github.com/mehrsapo/MMR_SAFI

Table 1. Denoising performance (in terms of PSNR) on the BSD68 test set.

Method	BM3D [47]	WCRR [17]	MMR	SARR [27]	SAFI	$SAFI_5$	Prox-DRUNet [18]
$\sigma = 5/255$	37.54	37.65	37.67	37.84	37.90	37.91	**37.97**
$\sigma = 15/255$	31.13	31.20	31.05	31.55	31.56	31.60	**31.70**
$\sigma = 25/255$	28.61	28.68	28.62	29.07	29.05	29.10	**29.18**

The average standard deviation of the PSNR for each image (based on 5 reconstructions) is similar for all settings and is roughly 0.02.

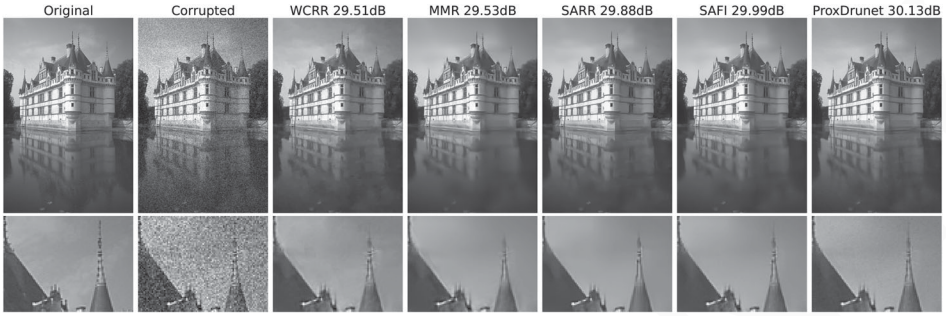

Figure 2. Denoising of the *castle* image corrupted by additive white Gaussian noise with $\sigma = 25/255$.

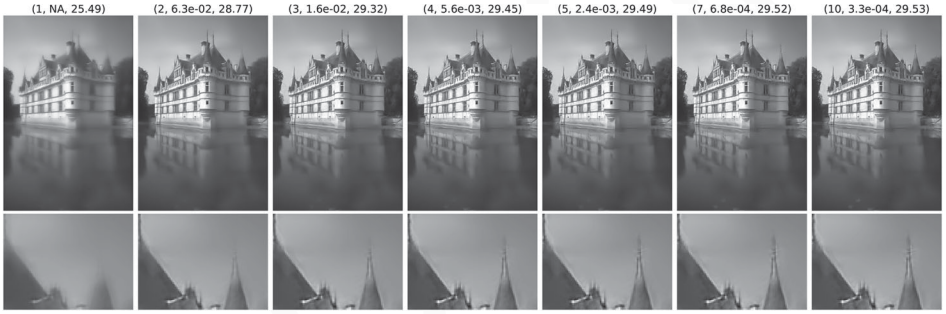

Figure 3. Solution path of the MMR method for denoising with $\sigma = 25/255$. Each image (k, e_k, PSNR_k) represents $\mathbf{x}_{k+1}$ at the kth step of Algorithm 3, with relative error $e_k = \frac{\|\mathbf{x}_{k+1} - \mathbf{x}_k\|_2}{\|\mathbf{x}_k\|_2}$.

$\sigma = 25/255$. The spatial adaptivity helps to preserve sharp edges in the image. Still, all but the Prox-DRUNet method tend to slightly smooth the image.

Regarding the qualitative behavior, we provide a solution path for MMR and SAFI in Figures 3 and 4, respectively. Somewhat surprisingly, the algorithm outputs a blurred reconstruction after the first step, in which all the noise is removed at the onset. This initial reconstruction is then progressively sharpened throughout the remaining iterations. This behavior is particularly striking as SAFI still recovers the tip of the spire, see Figure 2, which only reemerges in the later iterations. This is only possible since we update the mask iteratively based on the previous reconstruction, which is not the case for the one step method SARR. As guaranteed by Theorem 4, the residuals along the path for MMR in Figure 3 become small. The same is the case for SAFI where the iterates seem to converge to a fixed point, which necessarily exists due to Theorem 5. We

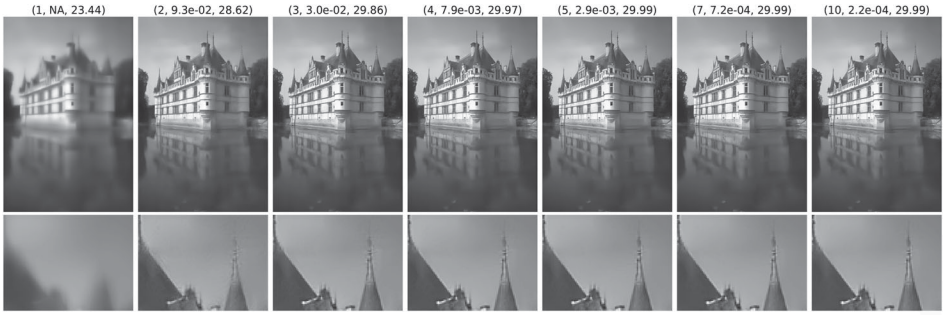

Figure 4. Solution path of the SAFI scheme for denoising with $\sigma = 25/255$. Each image (k, e_k, PSNR_k) represents $\mathbf{x}_{k+1}$ at the kth step of Algorithm 4, with relative error $e_k = \frac{\|\mathbf{x}_{k+1}-\mathbf{x}_k\|_2}{\|\mathbf{x}_k\|_2}$.

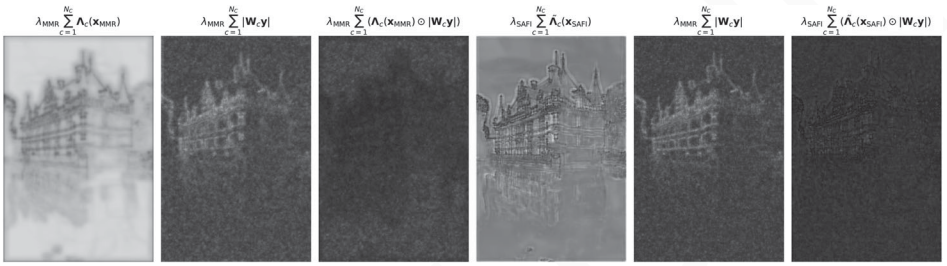

Figure 5. Masks and responses for the learned regularization architectures (5) and (7). Black corresponds to lower values and white to higher ones. Note that $\{\mathbf{W}_c\}_{c=1}^{N_C}$ is learned within the MMR and SAFI frameworks for the first and last three figures (from the left), respectively.

observe the same converging behavior for all images in the BSD68 test set. Also, the visual behavior along the path is very similar in terms of an initial strong smoothing followed by a later recovery of sharp features.

Now, we provide some intuition for the superiority of our approach over its nonadaptive counterpart. If $\|\mathbf{W}\cdot\|_1$ is a well-performing regularizer, the $\mathbf{W}_c$ should not respond to the distinctive properties of an image. To investigate this for both MMR and SAFI, we display the response of the respective $\sum_{c=1}^{N_C} |\mathbf{W}_c\cdot|$ to the noisy image $\mathbf{y}$ in Figure 5. For both cases, the structure of the image is also triggered in addition to the noise. This leaves some room for improvements of the reconstruction results. In particular, we can dampen this undesirable response using the masks. Then, the effect of image structure on the regularization cost becomes less pronounced. In Figure 6, we see how the masks become progressively more attentive to the image structure. Overall, the richer parameterization of the mask generator $\tilde{\mathbf{\Lambda}}$ for SAFI captures the image structure better. In particular, the masks for SAFI can still impose a high penalization in the vicinity of edges, whereas this is impossible for the masks from MMR. Overall, this results in a regularizer for which the image structures are less penalized. To conclude, the SAFI scheme leads to a better reconstruction performance than MMR model.

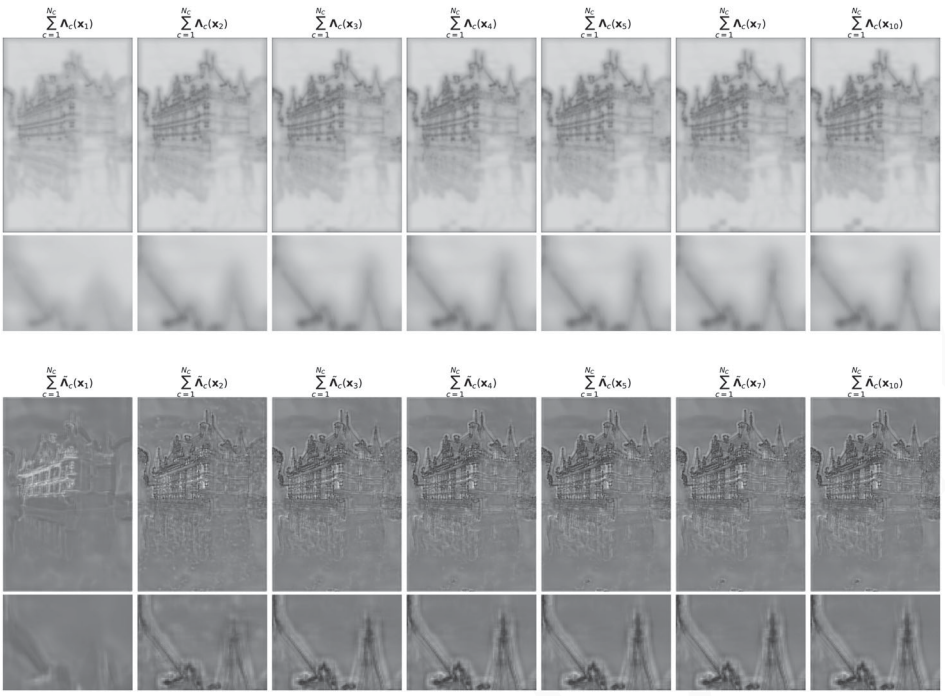

Figure 6. Evaluation of the masks for MMR (top) and SAFI (bottom). Both models become successively attentive to image structure. Still, the extracted structure in the MMR masks is far less pronounced.

5.2. Magnetic resonance imaging

Now, we deploy the proposed regularizers (5) and (7) to solve MRI-reconstruction problems. We use the single- and 15-coil MRI setups detailed in [16]. For each setup, the ground-truth images consist of proton-density-weighted knee images from the fastMRI dataset [11], both with fat suppression (PDFS) and without fat suppression (PD). In total, this leads to four evaluation tasks. For each task, we use a validation set of ten images to fine-tune the regularization strength λ in (13) and (32), respectively. We then report the test performance of the calibrated models on the remaining fifty test images. To generate the ground-truth image, we use the fully sampled k-space measurements. For the single-coil setup, we generate the measurements through a direct masking of the Fourier measures. In the 15-coil setup, we subsample the Fourier transforms of the ground-truth images multiplied by the respective sensitivity maps. For this, we use the BART [48] implementation of the ESPIRiT algorithm [49]. The subsampling rate of each setup is determined by the acceleration factor M_{acc} with the number of columns kept in the k-space being proportional to $1/M_{\text{acc}}$. Our single-coil setup is 4-fold ($M_{\text{acc}} = 4$) and our multi-coil setup is 8-fold ($M_{\text{acc}} = 8$). The measurements are then corrupted with additive white Gaussian noise of standard deviation $\sigma = 2 \cdot 10^{-3}$. In Table 2, we provide both the PSNR and structural-similarity index measure (SSIM) values on centered

Table 2. PSNR (first columns) and SSIM (second columns) values for the MRI experiment.

	4-fold single coil				8-fold multi-coil			
	PD	PDFS	PD	PDFS	PD	PDFS	PD	PDFS
Zero-fill ($\mathbf{H}^\top \mathbf{y}$)	27.40	29.68	0.729	0.745	23.80	27.19	0.648	0.681
TV [23]	32.44	32.67	0.833	0.781	32.77	33.38	0.850	0.824
CRR [16]	33.99	33.75	0.880	0.831	34.29	34.50	0.881	0.852
WCRR [17]	35.78	34.63	0.899	0.838	35.57	35.16	0.894	0.856
SARR [27]	36.25	34.77	0.904	0.839	35.98	35.26	**0.901**	0.858
Prox-DRUNet [18]	36.20	**35.05**	0.901	**0.847**	35.78	35.12	0.894	0.857
MMR	35.63	34.49	0.896	0.833	35.33	34.97	0.891	0.849
SAFI	**36.43**	34.92	**0.908**	0.844	**36.06**	**35.36**	**0.901**	**0.860**

(320×320) patches. Here, we compare against the popular TV regularization, the CRR as a state-of-the-art convex regularizer, its weakly convex extension WCRR, and the Prox-DRUNet as a popular PnP approach. Note that all of these methods are *universal* in the sense that they can be deployed without additional training. The full implementation details for the CRR and WCRR can be found in the respective papers. For Prox-DRUNet, we deploy the DRS-PnP algorithm proposed in [18], which was previously adapted to our experimental setups in [27].

As we observe in Table 2, the MMR model achieves a performance close to that of the weakly convex model introduced in [17]. This underlines again the strong relationship between the two regularization architectures and the associated models. The proposed SAFI regularizer achieves the best performance in three out of the four tasks and is second-best in the other one. Overall, these results indicate that our regularizers (5) and (7) generalize well to inverse problems with the model parameters that were obtained by training on a denoising task. If enough data and compute resources are available, task-specific fine-tuning (second training stage) of the model parameters using the actual data or the forward operator $\mathbf{H}$ can help to further increase the performance.

In Figure 7, we provide multi-coil MRI reconstructions for a PD-type image. There, we observe that MMR results in a reconstruction that is sharper than the one with CRR, while the SAFI scheme yields even better results. Most importantly, these improvements do not come at the price of artifacts in the reconstruction. In terms of quantitative metrics, the Prox-DRUNet solution is comparable to SAFI. However, as we observe in the insets, this solution represents poorly the original texture of the images. In particular, it allows for sharp transitions but smooths out the textured parts of the image in this example. In Figure 8, we investigate the single-coil setup for a PDFS image. This is the only case where the Prox-DRUNet is best on average. Although the Prox-DRUNet solution achieves higher PSNR than the SAFI solution, it is hard to observe pronounced visual differences between them.

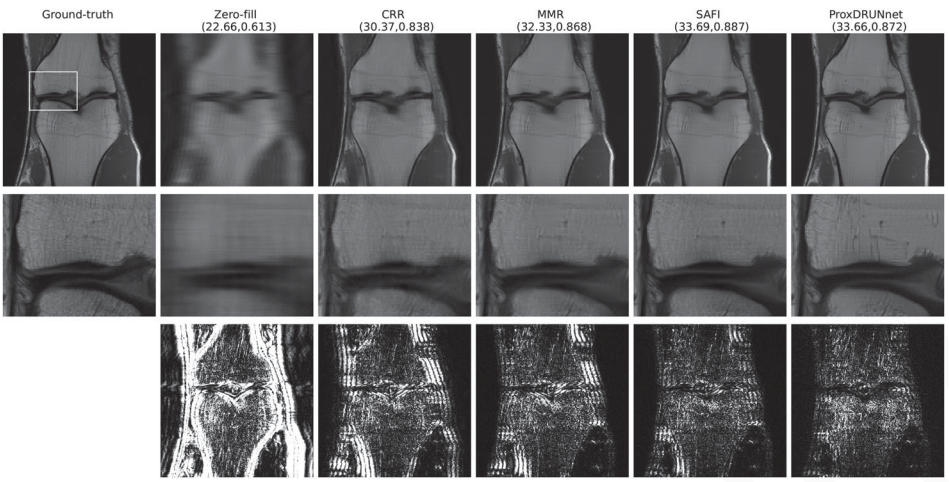

Figure 7. Reconstructions for multi-coil MRI (PD). The reported metric is (PSNR, SSIM). The second row contains the zoomed-in insets. The last row shows the squared value of the residuals, which are cutoff at 0.003.

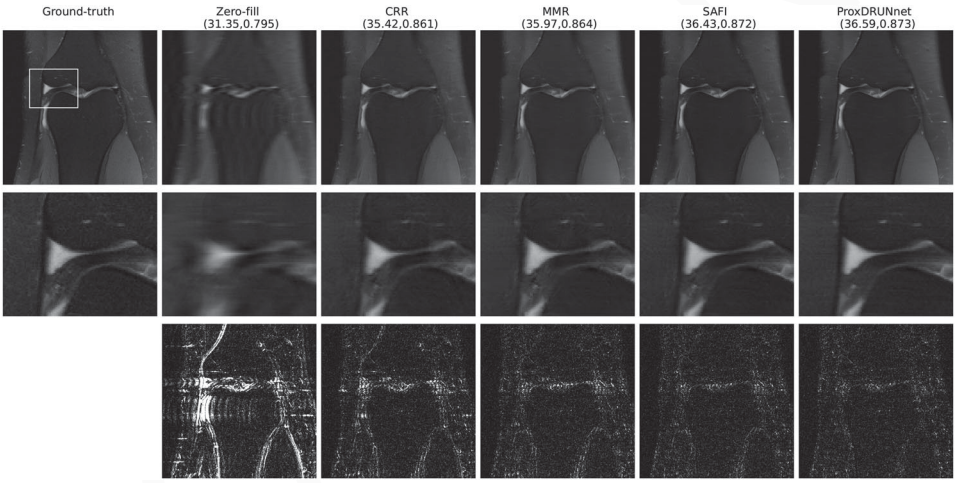

Figure 8. Reconstructions for single-coil MRI (PDFS). The reported metric is (PSNR, SSIM). The second row contains the zoomed-in insets. The last row shows the squared value of the residuals, which are cutoff at 0.003.

5.3. Algorithmic aspects for MMR and SAFI

5.3.1. Initialization

In principle, the proposed MMR and SAFI reconstructions depend on the initialization of the schemes. The initializations are required to compute the first masks $\mathbf{\Lambda}$ and $\tilde{\mathbf{\Lambda}}$ for MMR and SAFI, respectively. In Algorithms 3 and 4, we initialize with the solution of the nonadaptive convex problems (21) and (32) with $\mathbf{\Lambda}_c = \mathbf{1}_N$ and $\tilde{\mathbf{\Lambda}}_c = \mathbf{1}_N$, respectively. These plain experiments are denoted by CVX. To evaluate the sensitivity to this choice, we compare it against two alternatives. First, we perturb the proposed initialization by additive white Gaussian noise with $\sigma = 15/255$. These noisy experiments are denoted by Perturbed CVX. As an even stronger deviation, we use a random

Table 3. Robustness study: PSNR value after each MMR/SAFI update for the multi-coil MRI (PD) reconstruction experiment in Figure 7 depending on the initialization.

Update	0	1	2	3	4	5	6	Final
MMR, CVX	34.24	35.93	36.30	36.38	36.41	36.43	36.43	36.44
MMR, Perturbed CVX	24.16	35.10	36.17	36.35	36.41	36.42	36.43	36.43
MMR, Random	0	34.17	36.08	36.33	36.40	36.42	36.43	36.43
SAFI, CVX	33.70	35.55	35.84	35.91	35.94	35.95	35.96	35.97
SAFI, Perturbed CVX	24.10	35.09	35.76	35.90	35.94	35.95	35.96	35.97
SAFI, Random	0	21.66	34.70	35.59	35.82	35.89	35.93	35.96

initialization, where each entry is drawn from the standard normal distribution. These challenging experiments are denoted by Random. The PSNR values of the respective reconstructions for the MRI experiment from Figure 8 are given in Table 3. Most importantly, we observe that all variants eventually lead to the same PSNR. This indicates that the fixed point does not depend on the initialization. Unsurprisingly, the convergence to this fixed point occurs faster with a better initialization. We also observe the same behavior for other images. Finally, as indicated in Algorithms 1 and 2, the involved convex subproblems are always initialized with the minimizer of the previous one to accelerate the convergence.

5.3.2. Computational complexity

For the discussed MRI setups, the iterative SAFI approach is on average five times slower than the Prox-DRUNet approach, which does not incorporate any refinement steps. Reconstruction methods with a similar regularization architecture that do not incorporate a mask refinement (such as WCRR) can be even 50 times faster than SAFI. Memory-wise, SAFI has almost 10 times fewer parameters than ProxDRUNet and about 100 times more than WCRR. Since our approach brings valuable insights regarding weakly convex and spatially adaptive regularization, future work should focus on the improvement of the computational effectiveness of the approach. Since the subproblems (21) and (32) are convex, we can choose from a rich pool of methods for this goal. Moreover, we can draw from the literature on accelerating MM iterations [30, 31].

6. Conclusion

We have proposed to use an iterative majorization-minimization regularization (MMR) along with solution-adaptive-fixed-point iterations (SAFI) as new families of data-driven regularizers. They give rise to a sequence of convex reconstruction problems. Numerically, the minimizers associated with this sequence converged to a fixed point in all of our experiments. Overall, this leads to a robust, universal, and interpretable regularization method for inverse

problems. A benefit of our simple mask generator for SAFI is that it is well-suited to the enforcement of Lipschitz constraints, which are in turn important to obtain stability estimates. Such constraints might be the key to the proof of the convergence of the fixed point iterations. Finally, it could also be interesting to explore other architectural constraints for generating the masks to obtain theoretical guarantees.

Acknowledgments

The authors thank Alexis Goujon for fruitful discussions.

Disclosure statement

None of the authors have competing interests to disclose.

Funding

The research leading to this publication was supported by the European Research Council (ERC) under European Union's Horizon 2020 (H2020), Grant Agreement - Project No 101020573 FunLearn, by the Swiss National Science Foundation, Grant 200020_219356/1, and the German Research Foundation (DFG) within the SPP2298 under the project number 543939932.

References

[1] McCann, M. T., Unser, M. (2019). Biomedical image reconstruction: From the foundations to deep neural networks. *Found. Trends® Signal Process.* 13(3):283–359.

[2] Scherzer, O., Grasmair, M., Grossauer, H., Haltmeier, M., Lenzen, F. (2009). *Variational Methods in Imaging*, volume 167 of Applied Mathematical Sciences. New York: Springer.

[3] Tikhonov, A. N. (1963). Solution of incorrectly formulated problems and the regularization method. *Soviet Math.* 4:1035–1038.

[4] Donoho, D. L. (2006). Compressed sensing. *IEEE Trans. Inf. Theory* 52(4):1289–1306.

[5] Mallat, S. G. (1989). A theory for multiresolution signal decomposition: The wavelet representation. *IEEE Trans. Pattern Anal. Machine Intell.* 11(7):674–693.

[6] Rudin, L. I., Osher, S., Fatemi, E. (1992). Nonlinear total variation based noise removal algorithms. *Phys. D: Nonlinear Phenom.* 60(1–4):259–268.

[7] Lefkimmiatis, S., Bourquard, A., Unser, M. (2011). Hessian-based norm regularization for image restoration with biomedical applications. *IEEE Trans. Image Process.* 21(3):983–995.

[8] del Aguila Pla, P., Neumayer, S., Unser, M. (2023). Stability of image-reconstruction algorithms. *IEEE Trans. Comput. Imag.* 9:1–12.

[9] Arridge, S., Maass, P., Öktem, O., Schönlieb, C.-B. (2019). Solving inverse problems using data-driven models. *Acta Numerica* 28:1–174.

[10] Antun, V., Renna, F., Poon, C., Adcock, B., Hansen, A. C. (2020). On instabilities of deep learning in image reconstruction and the potential costs of AI. *Proc. National Acad. Sci.* 117(48):30088–30095.

[11] Knoll, F., Zbontar, J., Sriram, A., Muckley, M. J., Bruno, M., Defazio, A., Parente, M., Geras, K. J., Katsnelson, J., Chandarana, H., Zhang, Z., Drozdzalv, M., Romero, A., Rabbat, M., Vincent, P., Pinkerton, J., Wang, D., Yakubova, N., Owens, E., Zitnick, C. L., Recht, M. P., Sodickson, D. K., and Lui, Y. W. (2020). fastMRI: A publicly available raw k-space and DICOM dataset of knee images for accelerated MR image reconstruction using machine learning. *Radiol. Artif. Intell.* 2(1):e190007.

[12] Duff, M. A. G., Campbell, N. D. F., Ehrhardt, M. J. (2024). Regularising inverse problems with generative machine learning models. *J. Math. Imag. Vision* 66(1):37–56.

[13] Kobler, E., Effland, A., Kunisch, K., Pock, T. (2020). Total deep variation for linear inverse problems. In: *2020 Conference on Computer Vision and Pattern Recognition*, Virtual, June 14–19, pp. 7549–7558.

[14] Li, H., Schwab, J., Antholzer, S., Haltmeier, M. (2020). NETT: Solving inverse problems with deep neural networks. *Inverse Problems* 36(6):065005.

[15] Lunz, S., Öktem, O., Schönlieb, C.-B. (2018). Adversarial regularizers in inverse problems. In: *Advances in Neural Information Processing Systems*, Vol. 31.

[16] Goujon, A., Neumayer, S., Bohra, P., Ducotterd, S., Unser, M. (2023). A neural-network-based convex regularizer for inverse problems. *IEEE Trans. Comput. Imaging* 9:781–795.

[17] Goujon, A., Neumayer, S., Unser, M. (2024). Learning weakly convex regularizers for convergent image-reconstruction algorithms. *SIAM J. Imaging Sci.* 17(1):91–115.

[18] Hurault, S., Leclaire, A., Papadakis, N. (2022). Proximal denoiser for convergent Plug-and-Play optimization with nonconvex regularization. In: *39th International Conference on Machine Learning*, volume 162 of Proceedings of Machine Learning Research, Hawaii HI, USA, July 23–29, 2022, pp. 9483–9505.

[19] Hintermüller, M., Papafitsoros, K., Rautenberg, C. N. (2017). Analytical aspects of spatially adapted total variation regularisation. *J. Math. Anal. Appl.* 454(2):891–935.

[20] Kofler, A., Altekrüger, F., Antarou Ba, F., Kolbitsch, C., Papoutsellis, E., Schote, D., Sirotenko, C., Zimmermann, F. F., Papafitsoros, K. (2023). Learning regularization parameter-maps for variational image reconstruction using deep neural networks and algorithm unrolling. *SIAM J. Imaging Sci.* 16(4):2202–2246.

[21] Lefkimmiatis, S., Koshelev, I. S. (2023). Learning sparse and low-rank priors for image recovery via iterative reweighted least squares minimization. In: *ICLR*.

[22] Van Chung, C., De los Reyes, J. C., Schönlieb, C. (2017). Learning optimal spatially-dependent regularization parameters in total variation image denoising. *Inverse Probl.* 33(7):074005.

[23] Beck, A., Teboulle, M. (2009). A fast iterative shrinkage-thresholding algorithm for linear inverse problems. *SIAM J. Imaging Sci.* 2(1):183–202.

[24] Lenzen, F., Berger, J. (2015). Solution-driven adaptive total variation regularization. In: *SSVM*. Cham: Springer, pp. 203–215.

[25] Lenzen, F., Lellmann, J., Becker, F., Schnörr, C. (2014). Solving quasi-variational inequalities for image restoration with adaptive constraint sets. *SIAM J. Imaging Sci.* 7(4):2139–2174.

[26] Anil, C., Pokle, A., Liang, K., Treutlein, J., Wu, Y., Bai, S., Kolter, J. Z., Grosse, R. B. (2022). Path independent equilibrium models can better exploit test-time computation. In: *Adv. Neural Inf. Process. Syst.*, Vol. 35, pp. 7796–7809.

[27] Neumayer, S., Pourya, M., Goujon, A., Unser, M. (2023). Boosting weakly convex ridge regularizers with spatial adaptivity. In: *NeurIPS Workshop on Deep Learning and Inverse Problems*.

[28] Neumayer, S., Altekrüger, F. (2024). Stability of data-dependent ridge-regularization for inverse problems. arXiv:2406.12289.

[29] Figueiredo, M. A. T., Bioucas-Dias, J. M., Nowak, R. D. (2007). Majorization–minimization algorithms for wavelet-based image restoration. *IEEE Trans. Image Process.* 16(12):2980–2991.

[30] Hunter, D. R., Lange, K. (2004). A tutorial on MM algorithms. *Amer. Stat.* 58(1):30–37.

[31] Sun, Y., Babu, P., Palomar, D. P. (2017). Majorization-minimization algorithms in signal processing, communications, and machine learning. *IEEE Trans. Signal Process.* 65(3):794–816.

[32] Jacobson, M. W., Fessler, J. A. (2007). An expanded theoretical treatment of iteration-dependent majorize-minimize algorithms. *IEEE Trans. Image Process.* 16(10):2411–2422.

[33] Braides, A. (2002). Γ-*Convergence for Beginners*, volume 22 of Oxford Lecture Series in Mathematics and Its Applications. Oxford: Oxford University Press.

[34] Willner, L. B. (1968). On the distance between polytopes. *Q. Appl. Math.* 26(2):207–212.

[35] Pham, T.-S. (2023). Tangencies and polynomial optimization. *Math. Program.* 199(1–2):1239–1272.

[36] Chambolle, A., Dossal, C. H. (2015). On the convergence of the iterates of "fista". *J. Optim. Theory Appl.* 166(3):25.

[37] Beck, A., Teboulle, M. (2009). Fast gradient-based algorithms for constrained total variation image denoising and deblurring problems. *IEEE Trans. Image Process.* 18(11):2419–2434.

[38] Rockafellar, R. T. (1997). *Convex Analysis*. Princeton Landmarks in Mathematics. Princeton, NJ: Princeton University Press.

[39] Ducotterd, S., Goujon, A., Bohra, P., Perdios, D., Neumayer, S., and Unser, M. (2024). Improving Lipschitz-constrained neural networks by learning activation functions. *J. Mach. Learn. Res.* 25:1–30.

[40] Bohra, P., Perdios, D., Goujon, A., Emery, S., Unser, M. (2021). Learning Lipschitz-controlled activation functions in neural networks for Plug-and-Play image reconstruction methods. In: *NeurIPS 2021 Workshop on Deep Learning and Inverse Problems*, Virtual, December 13, 2021.

[41] Arbeláez, P., Maire, M., Fowlkes, C., Malik, J. (2011). Contour detection and hierarchical image segmentation. *IEEE Trans. Pattern Anal. Mach. Intell.* 33(5):898–916.

[42] Kingma, D. P., Ba, J. (2015). Adam: A method for stochastic optimization. In: *3rd International Conference of Representation Learning (ICLR 2015)*, Poster 9, San Diego CA, USA, May 7–9, 2015.

[43] Bai, S., Kolter, J. Z., Koltun, V. (2019). Deep equilibrium models. In: *Advances in Neural Information Processing Systems*, Vol. 32.

[44] Gilton, D., Ongie, G., Willett, R. (2021). Deep equilibrium architectures for inverse problems in imaging. *IEEE Trans. Comput. Imaging* 7:1123–113.

[45] Schmidt, M., Roux, N., Bach, F. (2011). Convergence rates of inexact proximal-gradient methods for convex optimization. In: *Advances in Neural Information Processing Systems*, Vol. 24.

[46] Villa, S., Salzo, S., Baldassarre, L., Verri, A. (2013). Accelerated and inexact forward-backward algorithms. *SIAM J. Optim.* 23(3):1607–1633.

[47] Dabov, K., Foi, A., Katkovnik, V., Egiazarian, K. (2007). Image denoising by sparse 3-D transform-domain collaborative filtering. *IEEE Trans. Image Process.* 16(8):2080–2095.

[48] Uecker, M., Virtue, P., Ong, F., Murphy, M. J., Alley, M. T., Vasanawala, S. S., Lustig, M. (2013). Software toolbox and programming library for compressed sensing and parallel imaging. In: *ISMRM Workshop on Data Sampling and Image Reconstruction*, pp. 41.

[49] Uecker, M., Lai, P., Murphy, M. J., Virtue, P., Elad, M., Pauly, J. M., Vasanawala, S. S., Lustig, M. (2014). ESPIRiT-An eigenvalue approach to autocalibrating parallel MRI: where SENSE meets GRAPPA. *Magnet. Reson. Med.* 71(3):990–1001.