Leveraging joint sparsity in hierarchical Bayesian learning *

Jan Glaubitz† and Anne Gelb‡

Abstract. We present a hierarchical Bayesian learning approach to infer jointly sparse parameter vectors from multiple measurement vectors. Our model uses separate conditionally Gaussian priors for each parameter vector and common gamma-distributed hyper-parameters to enforce joint sparsity. The resulting joint-sparsity-promoting priors are combined with existing Bayesian inference methods to generate a new family of algorithms. Our numerical experiments, which include a multi-coil magnetic resonance imaging application, demonstrate that our new approach consistently outperforms commonly used hierarchical Bayesian methods.

Key words. Multiple measurement vectors, joint sparsity, hierarchical Bayesian learning, conditionally Gaussian priors, (generalized) gamma hyper-priors

AMS subject classifications (2020). 65F22, 62F15, 65K10, 68U10

Code repository. https://github.com/jglaubitz/LeveragingJointSparsity

DOI. Not yet assigned

1. Introduction.

Parameter estimation from observable measurements is of fundamental importance in science and engineering applications. Multiple measurement vectors (MMVs) can often be obtained from various sources, each having distinct underlying parameter vectors due to differences in spatial or temporal conditions [21, 48, 1]. This situation can be modeled as a set of linear inverse problems given by

$$\mathbf{y}_l = F_l \mathbf{x}_l + \mathbf{e}_l, \quad l = 1, \dots, L, \tag{1.1}$$

where $\mathbf{y}_1, \dots, \mathbf{y}_L$ are the available MMVs, $\mathbf{x}_1, \dots, \mathbf{x}_L$ represent the sought-after parameter vectors, $F_1, \dots, F_L$ are explicitly known linear forward operators, and $\mathbf{e}_1, \dots, \mathbf{e}_L$ denote the unknown noise component. The linear forward operators are often poorly conditioned, and the measurements may be limited in number or resolution, and contaminated by noise, causing the set of inverse problems (1.1) to be ill-posed.

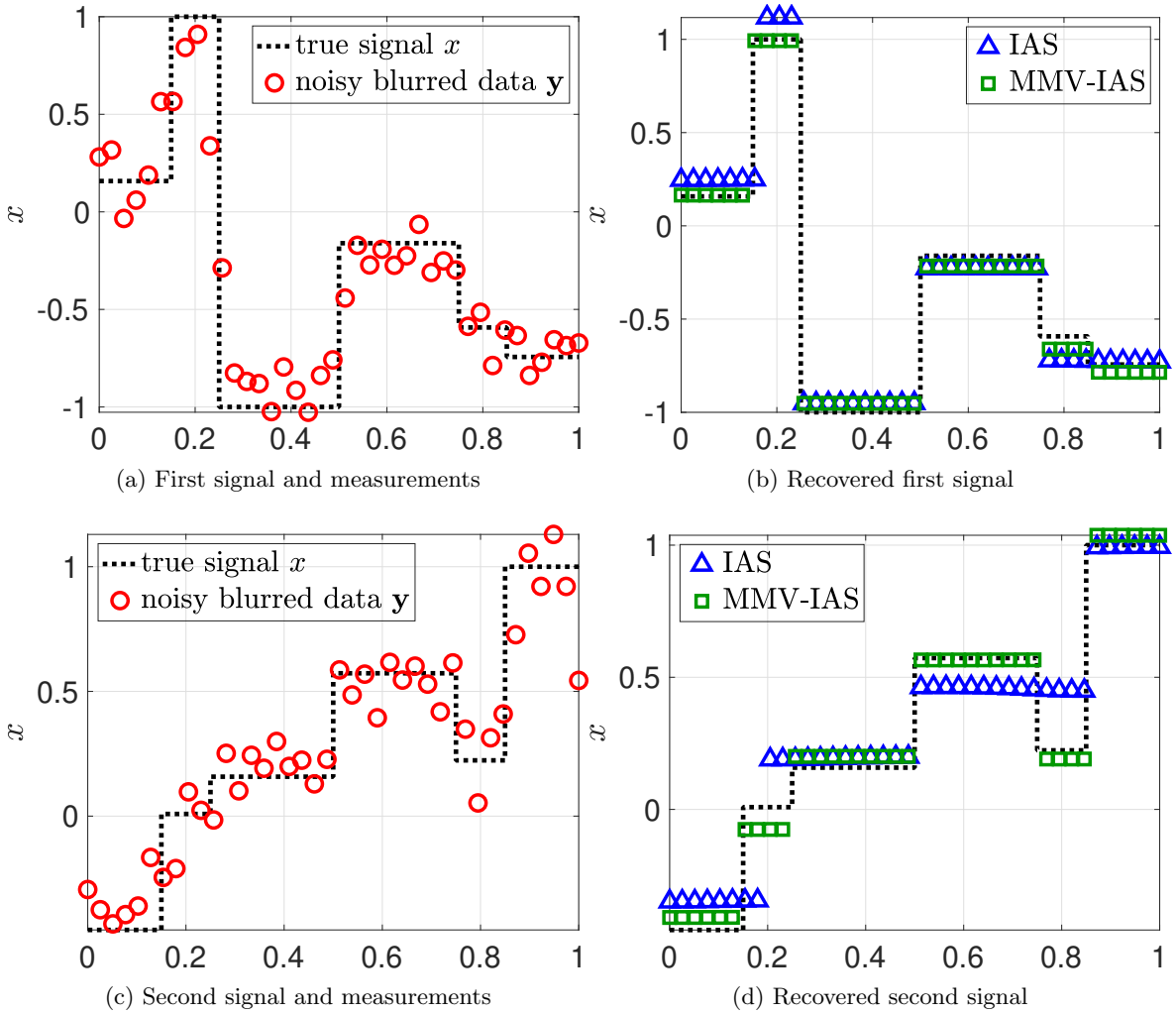
A well-known effective strategy used to mitigate ill-posedness is to incorporate prior information regarding the unknown parameter vectors, and in this study, we assume that these parameter vectors exhibit *joint sparsity*. Specifically, we assume there exists a linear operator R (e.g., a discrete gradient or wavelet transform) such that $R\mathbf{x}_1, \dots, R\mathbf{x}_L$ are sparse and have common support. For example, the parameter vectors could correspond to piecewise constant signals with the same interior edge locations but different values. Figures 1a and 1c depict this scenario for the first two of four jointly sparse piecewise constant signals. Joint sparsity arises in various applications, including signal processing, source location, neuro-electromagnetic imaging, parallel MRI, hyper-spectral imaging, and SAR imaging. For further reading on this topic, see [21, 48, 1, 53] and related references.

*March 6, 2024

Corresponding author: Jan Glaubitz

†Department of Aeronautics and Astronautics & Laboratory for Information and Decision Systems, Massachusetts Institute of Technology, Cambridge, MA, USA (glaubitz@mit.edu, orcid.org/0000-0002-3434-5563)

‡Department of Mathematics, Dartmouth College, Hanover, NH 03755, USA (Anne.E.Gelb@Dartmouth.edu, orcid.org/0000-0002-9219-4572)



(a) First signal and measurements

(b) Recovered first signal

(c) Second signal and measurements

(d) Recovered second signal

Figure 1: First column: The first two of four piecewise constant signals with a common edge profile and noisy blurred measurements. Second column: Reconstructions of the signals using the existing IAS algorithm to separately recover them (blue triangles) and the proposed MMV-IAS algorithm to jointly recover them (green squares). See subsection 6.2 for more details.

Current methodology. Various deterministic methods address the ill-posedness in the set of linear inverse problems (1.1) by transforming it into a set of nearby regularized optimization problems. Under the joint sparsity assumption, established compressive sensing methods [23, 24, 27] can be used to *individually* recover the desired parameter vectors. By leveraging their joint sparsity structure, the compressive sensing methods in [21, 25, 1] *jointly* recover these vectors. The approach in [1] significantly enhanced the recovery process's robustness and accuracy. For more recent works in this area, see [28, 39, 52, 51] and their references.

However, regularized inverse problems often face two significant challenges: (i) determining the appropriate regularization parameters and (ii) quantifying uncertainty in the recovered solution. Because the available measurements in (1.1) are often insufficient and noisy, it is essential to quantify the subsequent uncertainty in the parameters of interest. In particular, uncertainty in the parameters leads to

uncertainty in predictions and decision-making.

In this work we employ a hierarchical Bayesian approach [31, 11, 42] to solve the MMV inverse problem (1.1), with the parameters of interest and the measurements modeled as random variables. The sought-after posterior distribution for the parameters of interest is characterized using Bayes' theorem, which connects the posterior density to the prior and likelihood densities. The prior encodes information available on the parameters of interest before any data are observed, while the likelihood density incorporates the data model and a stochastic description of measurements. A primary benefit of this framework is that it enables uncertainty quantification while avoiding the need for fine-tuning regularization parameters.

One particularly effective class of priors for promoting sparsity is the conditional Gaussian prior. This choice has proven successful in various applications such as sparse basis selection [44, 47], signal and image recovery [16, 3, 29, 52], and edge detection [20, 50]. Additionally, conditional Gaussian priors are computationally convenient and lead to highly efficient inference algorithms, see [13, 9, 8, 46, 29] and references therein. Although existing sparsity-promoting hierarchical Bayesian algorithms can be used to infer the parameter vectors separately, such approaches do not exploit the *joint sparsity* of the parameter vectors.

Our contribution. We present a hierarchical Bayesian learning approach that leverages joint sparsity in multiple parameter vectors described by the MMV data model (1.1). Our approach utilizes separate priors for each parameter vector while sharing common hyper-parameters drawn from (generalized) gamma distributions, which capture the sparsity profile of the parameter vectors. The advantage of using joint-sparsity-promoting priors is demonstrated in comparison to well-established sparsity-promoting algorithms, such as the generalized sparse Bayesian learning (GSBL) [44, 47, 29] and iterative alternating sequential (IAS) [13, 9, 8] algorithms. Our results indicate that the proposed MMV-GSBL and MMV-IAS algorithms, which by design use joint-sparsity-promoting priors, outperform the existing methods. Figure 1 compares the existing IAS algorithm with the proposed MMV-IAS algorithm to recover the first two out of four piecewise-constant signals with a common edge profile from noisy blurred data. Observe that the joint sparsity enhancement in either approach consistently results in superior signal recovery compared to reconstructing signals individually. In particular, while the performance of the IAS and GSBL algorithms may vary depending on the specific problem and model parameters, incorporating joint sparsity consistently offers advantageous outcomes. Further numerical experiments demonstrate that the proposed method increases the robustness and accuracy of the recovered parameter vectors, better catches the sparsity profile encoded in the hyper-parameters, and reduces uncertainty. The findings of this research highlight the significant improvement in performance that can be achieved by exploiting joint sparsity in hierarchical Bayesian models. In particular, we demonstrate its potential application to parallel MRI. Additionally, we note that utilizing separate priors with shared hyper-parameters is not limited to conditionally Gaussian priors and that our approach can be adapted to other hierarchical prior models. Finally, our approach shares similarities with the one proposed in [48] for classical SBL that we discuss in subsection 5.4.

Outline. We present the joint-sparsity-promoting conditionally Gaussian priors and the resulting hierarchical Bayesian model in Section 2, with the Bayesian MAP estimation discussed in Section 3. In Section 4, we analyze the new MMV-IAS algorithm and compare it to the existing IAS algorithm. Section 5 extends the idea of joint-sparsity-promoting priors to the GSBL framework to form the MMV-GSBL algorithm. Numerical experiments are showcased in Section 6, which include applications of the proposed MMV-IAS and -GSBL algorithms to parallel MRI. We summarize the work in Section 7.

Notation. We use normal and boldface capital letters, such as X and $\mathbf{X}$, to denote scalar- and vector-valued random variables, respectively. For a density π, we write $X \sim \pi$ when X is distributed according to π. If $L \in \mathbb{N}$ and $\mathbf{X}_1, \dots, \mathbf{X}_L$ are random variables, then we denote their collection by $\mathbf{X}_{1:L} = (\mathbf{X}_1, \dots, \mathbf{X}_L)$. The same notation applies to dummy variables $\boldsymbol{x} \in \mathbb{R}^n$.

2. The joint hierarchical Bayesian model.

We present the joint-sparsity promoting Bayesian model considered in this investigation. The conditionally Gaussian prior in subsection 2.2 is particularly important for developing our new method.

2.1. The likelihood function.

The likelihood density function models the connection between the parameter and measurement vectors. Consider the linear MMV data model (1.1) with MMVs $\mathbf{y}_l \in \mathbb{R}^{M_l}$, known forward operators $F_l \in \mathbb{R}^{M_l \times N}$, desired parameter vectors $\mathbf{x}_l \in \mathbb{R}^N$, and additive Gaussian noise $\mathbf{e}_l \sim \mathcal{N}(\mathbf{0}, \Sigma_l)$. Since Σ_l is a symmetric positive definite (SPD) covariance matrix, there exists a Cholesky decomposition of the form $\Sigma_l = C_l C_l^T$ with invertible C_l. The noise can then be whitened by multiplying both sides of (1.1) with C_l^{-1} from the left-hand side so that we can assume $\Sigma_l = I$, where I is the $M_l \times M_l$ identity matrix. The *lth likelihood function* is then

$$\pi_{\mathbf{Y}_l|\mathbf{X}_l}(\mathbf{y}_l|\mathbf{x}_l) \propto \exp\left(-\frac{1}{2}\|F_l\mathbf{x}_l - \mathbf{y}_l\|_2^2\right), \quad l = 1, \dots, L. \tag{2.1}$$

Assuming that $\mathbf{Y}_{1:L}$ are jointly independent conditioned on $\mathbf{X}_{1:L}$, the *joint likelihood function* is

$$\pi_{\mathbf{Y}_{1:L}|\mathbf{X}_{1:L}}(\mathbf{y}_{1:L}|\mathbf{x}_{1:L}) = \prod_{l=1}^{L} \pi_{\mathbf{Y}_l|\mathbf{X}_l}(\mathbf{y}_l|\mathbf{x}_l) \propto \exp\left(-\frac{1}{2}\sum_{l=1}^{L}\|F_l\mathbf{x}_l - \mathbf{y}_l\|_2^2\right). \tag{2.2}$$

Note that (2.2) is a conditionally Gaussian density function, which is convenient for Bayesian inference. A couple of remarks are in order.

Remark 2.1 (Complex-valued forward operators). This framework also allows for complex-valued forward operators and observations. Specifically for $F \in \mathbb{C}^{M\times N}$, we can use the equivalent real-valued forward operator $[\mathrm{Re}(F); \mathrm{Im}(F)] \in \mathbb{R}^{2M\times N}$, where $\mathrm{Re}(F)$ and $\mathrm{Im}(F)$ denote the real and imaginary part of F.

Remark 2.2 (Non-linear data models). For simplicity, we restrict our attention to linear data models. However, the proposed approach can be extended to non-linear models using methods such as Kalman filtering [26, 41, 32].

Remark 2.3 (Dependent measurement vectors). For simplicity, we have assumed that the MMVs $\mathbf{Y}_{1:L}$ are jointly independent conditioned on the parameter vectors $\mathbf{X}_{1:L}$. This assumption facilitated the expression of the joint likelihood function as detailed in (2.2). However, it is important to recognize that in practical scenarios, the assumption of independence among measurement vectors might not hold due to inherent interdependencies. To illustrate, consider the case where $\mathbf{y}_{1:L}|\mathbf{x}_{1:L} \sim \mathcal{N}(\mathbf{0}|\Sigma)$. In this more general scenario, the joint likelihood function is

$$\pi_{\mathbf{Y}_{1:L}|\mathbf{X}_{1:L}}(\mathbf{y}_{1:L}|\mathbf{x}_{1:L}) \propto \exp\left(-\frac{1}{2}\left\|\Sigma^{-1}\left(F\mathbf{x} - \mathbf{y}\right)\right\|_2^2\right), \tag{2.3}$$

where $F = \mathrm{diag}(F_1, \dots, F_L)$, $\mathbf{x} = [\mathbf{x}_1, \dots, \mathbf{x}_L]^T$, and $\mathbf{y} = [\mathbf{y}_1, \dots, \mathbf{y}_L]^T$. We can again whiten the noise by computing the Cholesky decomposition $\Sigma = CC^T$ and multiplying F and $\mathbf{y}$ by C^{-1} from the left. It is noteworthy that the resulting forward operator matrix might not maintain a block-diagonal form.

Consequently, the parameter vector updates, as discussed in subsection 3.2, no longer decouple. In this case, the optimization problems in (3.5) transforms into

$$\mathbf{x}_{1:L} = \underset{\mathbf{x}_1,\dots,\mathbf{x}L}{\arg\min} \left\{ \|F\mathbf{x} - \mathbf{y}\|_2^2 + \sum_{l=1}^{L} \|D\boldsymbol{\theta}^{-1/2} R\mathbf{x}_l\|_2^2 \right\} \tag{2.4}$$

with $D_{\boldsymbol{\theta}} = \operatorname{diag}(\boldsymbol{\theta})$. Notably, (2.4) poses a more computationally demanding problem compared to the original parallelizable optimization problems (3.5).

2.2. The joint-sparsity promoting conditionally Gaussian prior. The prior density models our prior belief about the desired parameter vectors $\mathbf{x}_{1:L}$. Here we assume that they are *jointly* sparse, i.e., there exist a linear transform $R \in \mathbb{R}^{K\times N}$ such that $R\mathbf{x}_1, \dots, R\mathbf{x}_L$ are sparse and have the same support (the indices of their non-zero values are the same). We start by modeling the sparsity of $R\mathbf{x}_l$ in a probabilistic setting by choosing the *lth prior* as the conditionally Gaussian density

$$\pi_{\mathbf{X}_l|\boldsymbol{\Theta}_l}(\mathbf{x}_l|\boldsymbol{\theta}_l) \propto \det(D_{\boldsymbol{\theta}_l})^{-1/2} \exp\left(-\frac{1}{2}\|D_{\boldsymbol{\theta}_l}^{-1/2} R\mathbf{x}_l\|_2^2\right), \quad l = 1, \dots, L, \tag{2.5}$$

with hyper-parameter vector $\boldsymbol{\theta}_l = [(\theta_l)_1, \dots, (\theta_l)_K]$, covariance matrix $D_{\boldsymbol{\theta}_l} = \operatorname{diag}(\boldsymbol{\theta}_l)$, and unknown variance parameters $(\theta_l)_k > 0$ for $k = 1, \dots, K$ and $l = 1, \dots, L$.

Remark 2.4. Following [10, 29], the conditional Gaussian prior (2.5) can be motivated by its asymptotic behavior: Assume that $(\theta_l)_1 = \cdots = (\theta_l)_K$, then (2.5) favors $\mathbf{x}_l$ for which $\|R\mathbf{x}_l\|_2$ is close to zero, since such an $\mathbf{x}_l$ has a higher probability. For instance, when $R\mathbf{x}_l$ corresponds to the increments of $\mathbf{x}_l$, i.e., $[R\mathbf{x}_l]_k = (x_l)_{k+1} - (x_l)_k$, then (2.5) with $(\theta_l)_1 = \cdots = (\theta_l)_K$ favors $\mathbf{x}_l$ to have little variation. However, if one of the hyper-parameters, say $(\theta_l)_k$, is significantly larger than the others, a jump between $(x_l)_{k+1}$ and $(x_l)_k$ becomes more likely. In this way, (2.5) promotes sparsity of $R\mathbf{x}_l$. Furthermore, we can connect the support of $R\mathbf{x}_l$ to the hyper-parameters $(\theta_l)_1, \dots, (\theta_l)_K$. In particular, we expect the support of $R\mathbf{x}_l$ to coincide with the hyper-parameters significantly larger than most others.

We next model $R\mathbf{x}_1, \dots, R\mathbf{x}_L$ having the same support. To this end, motivated by Remark 2.4, we connect the supports of $R\mathbf{x}_1, \dots, R\mathbf{x}_L$ to the hyper-parameter vectors, $\boldsymbol{\theta}_1, \dots, \boldsymbol{\theta}_L$, by assuming that $\boldsymbol{\theta}_1 = \cdots = \boldsymbol{\theta}_L$. Denoting the common hyper-parameter vector as $\boldsymbol{\theta}$, (2.5) then reduces to

$$\pi_{\mathbf{X}_l|\boldsymbol{\Theta}}(\mathbf{x}_l|\boldsymbol{\theta}) \propto \det(D_{\boldsymbol{\theta}})^{-1/2} \exp\left(-\frac{1}{2}\|D_{\boldsymbol{\theta}}^{-1/2} R\mathbf{x}_l\|_2^2\right), \quad l = 1, \dots, L. \tag{2.6}$$

That is, the priors are now all conditioned on the *same* hyper-parameters. Finally, assuming the $\mathbf{X}_{1:L}$ are jointly independent conditioned on $\boldsymbol{\Theta}$, the *joint prior* is

$$\pi_{\mathbf{X}_{1:L}|\boldsymbol{\Theta}}(\mathbf{x}_{1:L}|\boldsymbol{\theta}) = \prod_{l=1}^{L} \pi_{\mathbf{X}_l|\boldsymbol{\Theta}}(\mathbf{x}_l|\boldsymbol{\theta}) \propto \det(D_{\boldsymbol{\theta}})^{-L/2} \exp\left(-\frac{1}{2}\sum_{l=1}^{L} \|D_{\boldsymbol{\theta}}^{-1/2} R\mathbf{x}_l\|_2^2\right). \tag{2.7}$$

Remark 2.5 (Other sparsity-promoting hierarchical priors). The conditionally Gaussian prior in (2.7) not only enforces joint sparsity but also enables convenient Bayesian inference due to its compatibility with the Gaussian likelihood (2.2). However, the joint-sparsity-promoting approach using a common hyper-parameter vector $\boldsymbol{\theta}$ can be extended to other hierarchical sparsity-promoting priors, such as horseshoe [15, 45] and neural network priors [34, 2, 33].

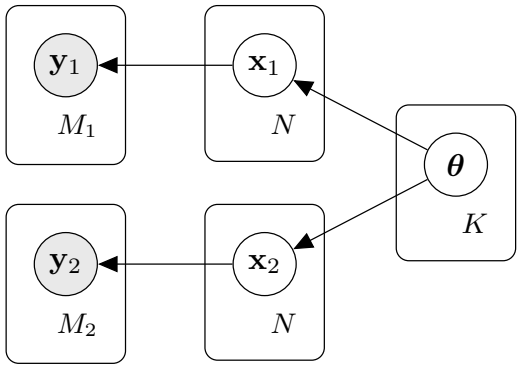


Figure 2: Graphical representation of the hierarchical Bayesian model promoting joint sparsity for two ($L = 2$) measurement and parameters vectors, $\mathbf{y}_1, \mathbf{y}_2$ and $\mathbf{x}_1, \mathbf{x}_2$, respectively. Shaded and plain circles represent observed and unobserved (hidden) random variables, respectively. The arrows indicate how the random variables influence each other: The parameter vectors $\mathbf{x}_1, \mathbf{x}_2$ are connected to the measurement vectors $\mathbf{y}_1, \mathbf{y}_2$, respectively, via the likelihood (2.2); The common hyper-parameters $\boldsymbol{\theta}$ are connected to $\mathbf{x}_1, \mathbf{x}_2$ via the joint-sparsity-promoting prior (2.7). Using common gamma hyper-parameters $\boldsymbol{\theta}$ (instead of separate ones for $\mathbf{x}_1, \mathbf{x}_2$) results in $R\mathbf{x}_1$ and $R\mathbf{x}_2$ having the same support.

2.3. The generalized gamma hyper-prior. The price to pay for the hierarchical joint prior model (2.7) is that we now need to estimate not only the parameter vectors $\mathbf{x}_{1:L}$ but also the common hyper-parameter vector $\boldsymbol{\theta}$. By Bayes' theorem, the *joint posterior density* of $(\mathbf{X}_{1:L}, \boldsymbol{\Theta})$ given $\mathbf{Y}_{1:L}$ is

$$\pi_{\mathbf{X}_{1:L},\boldsymbol{\Theta}|\mathbf{Y}_{1:L}}(\mathbf{x}_{1:L}, \boldsymbol{\theta}|\mathbf{y}_{1:L}) \propto \pi_{\mathbf{Y}_{1:L}|\mathbf{X}_{1:L}}(\mathbf{y}_{1:L}|\mathbf{x}_{1:L})\, \pi_{\mathbf{X}_{1:L}|\boldsymbol{\Theta}}(\mathbf{x}_{1:L}|\boldsymbol{\theta})\, \pi_{\boldsymbol{\Theta}}(\boldsymbol{\theta}). \tag{2.8}$$

From Remark 2.4, it is evident that to promote sparsity of $R\mathbf{x}_1, \dots, R\mathbf{x}_L$ the hyper-prior $\pi_{\boldsymbol{\Theta}}$ should favor small values of $\theta_1, \dots, \theta_K$ while allowing occasional large outliers for the conditionally Gaussian prior (2.6). Following [9, 8], this can be achieved by treating $\theta_1, \dots, \theta_K$ as random variables with an uninformative generalized gamma density function

$$\pi_{\boldsymbol{\Theta}}(\boldsymbol{\theta}) = \prod_{k=1}^{K} \mathcal{GG}(\theta_k|r, \beta, \vartheta_k) \propto \det(D_{\boldsymbol{\theta}})^{r\beta-1} \exp\left(-\sum_{k=1}^{K} (\theta_k/\vartheta_k)^r\right). \tag{2.9}$$

Here, $\mathcal{GG}$ is the generalized gamma distribution

$$\mathcal{GG}(\theta_k|r, \beta, \vartheta_k) \propto \theta_k^{r\beta-1} \exp\left(-(\theta_k/\vartheta_k)^r\right), \tag{2.10}$$

where $r \in \mathbb{R} \setminus \{0\}$, $\beta > 0$, and $\vartheta_k > 0$ for $k = 1, \dots, K$. Figure 2 provides a graphical illustration and summary of our joint-sparsity-promoting hierarchical Bayesian model.

3. Bayesian inference. We now address Bayesian inference for the joint-sparsity-promoting hierarchical Bayesian model proposed in Section 2. To this end, for given MMVs $\mathbf{y}_{1:L}$, we solve for the *maximum a posterior (MAP) estimate* $(\mathbf{x}_{1:L}^{\text{MAP}}, \boldsymbol{\theta}^{\text{MAP}})$, which is the maximizer of the posterior density (2.8). Equivalently, the MAP estimate is the minimizer of the negative logarithm of the posterior, i.e.,

$$(\mathbf{x}_{1:L}^{\text{MAP}}, \boldsymbol{\theta}^{\text{MAP}}) = \operatorname*{arg\,min}_{\mathbf{x}_{1:L},\boldsymbol{\theta}} \left\{\mathcal{G}(\mathbf{x}_{1:L}, \boldsymbol{\theta}\right\}, \tag{3.1}$$

where the objective function $\mathcal{G}$ is

$$\mathcal{G}(\mathbf{x}_{1:L}, \boldsymbol{\theta}) = -\log \pi_{\mathbf{X}_{1:L},\boldsymbol{\Theta}|\mathbf{Y}_{1:L}}(\mathbf{x}_{1:L}, \boldsymbol{\theta}|\mathbf{y}_{1:L}). \tag{3.2}$$

Substituting (2.2), (2.7), and (2.9) into (3.2), we obtain

$$\mathcal{G}(\mathbf{x}_{1:L},\boldsymbol{\theta}) = \frac{1}{2}\left(\sum_{l=1}^{L}\|F_l\mathbf{x}_l-\mathbf{y}_l\|_2^2+\|D_{\boldsymbol{\theta}}^{-1/2}R\mathbf{x}_l\|_2^2\right)+\sum_{k=1}^{K}\left(\frac{\theta_k}{\vartheta_k}\right)^r-\eta\sum_{k=1}^{K}\log(\theta_k) \tag{3.3}$$

up to constants that neither depend on $\mathbf{x}_{1:L}$ nor $\boldsymbol{\theta}$, where $\eta = r\beta-(L/2+1)$. In what follows we discuss how the minimizer of $\mathcal{G}$ — and therefore the MAP estimate $(\mathbf{x}_{1:L}^{\mathrm{MAP}},\boldsymbol{\theta}^{\mathrm{MAP}})$ — can be approximated.

3.1. The iterative alternating sequential algorithm. We use a block-coordinate descent approach [49, 4] to approximate the MAP estimate. In the context of conditionally Gaussian priors for which the covariance is assumed to follow a (generalized) gamma distribution, a prevalent block-coordinate descent method is the so-called iterative alternating sequential (IAS) algorithm [10, 6, 13, 9]. The IAS algorithm computes the minimizer of the objective function $\mathcal{G}$ by alternatingly (i) minimizing $\mathcal{G}$ w.r.t. $\mathbf{x}_{1:L}$ for fixed $\boldsymbol{\theta}$ and (ii) minimizing $\mathcal{G}$ w.r.t. $\boldsymbol{\theta}$ for fixed $\mathbf{x}_{1:L}$. Given an initial guess for the hyper-parameter vector $\boldsymbol{\theta}$, the IAS algorithm proceeds through a sequence of updates of the form

$$\mathbf{x}_{1:L} = \arg\min_{\mathbf{x}_{1:L}}\left\{\mathcal{G}(\mathbf{x}_{1:L},\boldsymbol{\theta})\right\},\quad \boldsymbol{\theta} = \arg\min_{\boldsymbol{\theta}}\left\{\mathcal{G}(\mathbf{x}_{1:L},\boldsymbol{\theta})\right\}, \tag{3.4}$$

until a convergence criterion is met.[1] Efficient implementation of the two update steps in (3.4) is discussed below.

3.2. Updating the parameter vectors. Updating $\mathbf{x}_{1:L}$ given $\boldsymbol{\theta}$ reduces to solving the quadratic optimization problems

$$\mathbf{x}_l = \arg\min_{\mathbf{x}}\left\{\|F_l\mathbf{x}-\mathbf{y}_l\|_2^2+\|D_{\boldsymbol{\theta}}^{-1/2}R\mathbf{x}\|_2^2\right\},\quad l=1,\dots,L, \tag{3.5}$$

with $D_{\boldsymbol{\theta}} = \mathrm{diag}(\boldsymbol{\theta})$. Note that the optimization problems (3.5) are decoupled and can thus be solved efficiently in parallel. Furthermore, assuming the *common kernel condition* (also see [29, 51])

$$\mathrm{kernel}(F_l)\cap\mathrm{kernel}(R) = \{\mathbf{0}\},\quad l=1,\dots,L, \tag{3.6}$$

holds, each optimization problem in (3.5) has a unique solution. Here, $\mathrm{kernel}(G) = \{\,\mathbf{x}\in\mathbb{R}^N \mid G\mathbf{x}=\mathbf{0}\,\}$ is the kernel of an operator $G:\mathbb{R}^N\to\mathbb{R}^M$, i.e., the set of vectors that are mapped to zero by G. The common kernel condition (3.6) guarantees that the combination of prior information and the given measurements will result in a well-posed problem, which is a commonly accepted assumption in regularized inverse problems [31, 43]. Finally, we can efficiently solve the quadratic optimization problems (3.5) using various existing methods, including the fast iterative shrinkage-thresholding (FISTA) algorithm [5], the preconditioned conjugate gradient (PCG) method [36], potentially combined with an early stopping based on Morozov's discrepancy principle [6, 7, 9], and the gradient descent approach [29]. There is no general advantage of one method over another, and the choice should be made based on the specific problem (and the structure of F_l and R) at hand.

3.3. Updating the hyper-parameters. We next address the update for the hyper-parameters $\boldsymbol{\theta}$, for which we must solve

$$\boldsymbol{\theta} = \arg\min_{\boldsymbol{\theta}}\left\{\mathcal{G}(\mathbf{x}_{1:L},\boldsymbol{\theta})\right\} \tag{3.7}$$

[1] In our implementation, we stop if the relative change in the $\mathbf{x}_l$ variables falls below a given threshold. For simplicity, we initialize the hyper-parameter vector as $\boldsymbol{\theta} = [1,\dots,1]$.

for fixed parameter vectors $\mathbf{x}_{1:L}$. Substituting (3.3) into (3.7) and ignoring all terms that do not depend on $\boldsymbol{\theta}$, (3.7) is equivalent to

$$\theta_k = \arg\min_{\theta_k} \left\{ \theta_k^{-1} \left(\sum_{l=1}^{L} [R\mathbf{x}_l]_k^2/2 \right) + \left(\frac{\theta_k}{\vartheta_k} \right)^r - \eta \log(\theta_k) \right\}, \tag{3.8}$$

where $\eta = r\beta - (L/2+1)$ and $[R\mathbf{x}_l]_k$ denotes the k-th entry of the vector $R\mathbf{x}_l \in \mathbb{R}^K$. Differentiating the objective function in (3.8) w.r.t. θ_k and setting this derivative to zero yields

$$0 = -\theta_k^{-2} \left(\sum_{l=1}^{L} [R\mathbf{x}_l]_k^2/2 \right) + \theta_k^{r-1} \left(\frac{r}{\vartheta_k^r} \right) - \theta_k^{-1}\eta. \tag{3.9}$$

For some values of r, (3.9) admits an analytical solution. For instance, if $r = 1$, (3.9) is equivalent to

$$\theta_k = \frac{\vartheta_k}{2} \left(\eta + \sqrt{\eta^2 + 2\vartheta^{-1} \sum_{l=1}^{L} [R\mathbf{x}_l]_k^2} \right), \quad k = 1, \dots, K, \tag{3.10a}$$

where $\eta = r\beta - (L/2+1)$, and respectively for $r = -1$, we have

$$\theta_k = \frac{\sum_{l=1}^{L} [R\mathbf{x}_l]_k^2/2 + \vartheta_k}{-\eta}, \quad k = 1, \dots, K. \tag{3.10b}$$

By assuming $\eta > 0$ in (3.10a) and $\eta < 0$ in (3.10b), the hyper-parameters are ensured to be positive. We refer to [9, 8] for details on how (3.9) can be solved numerically in the general case.

3.4. Proposed algorithm and its relationship to current methodology. Algorithm 3.1 summarizes the above procedure to approximate the MAP estimate $(\mathbf{x}_{1:L}^{\mathrm{MAP}}, \boldsymbol{\theta}^{\mathrm{MAP}})$ of our joint-sparsity-promoting hierarchical Bayesian model proposed in Section 2. We will refer to this method as the *MMV-IAS algorithm*.

Algorithm 3.1 The MMV-IAS algorithm

1: Choose model parameters $(r, \beta, \boldsymbol{\vartheta})$ and initialize $\boldsymbol{\theta}$
2: **repeat**
3: Update the parameter vectors $\mathbf{x}_{1:L}$ (in parallel) according to (3.5)
4: Update the hyper-parameters $\boldsymbol{\theta}$ according to (3.10)
5: **until** convergence or the maximum number of iterations is reached

Relationship to the IAS algorithm. Our MMV-IAS algorithm builds upon the standard IAS algorithm [9, 8] and reduces to it when handling a single measurement and parameter vector ($L = 1$). However, it is important to note that using the standard IAS to recover each $\mathbf{x}_{1:L}$ separately from the MMVs $\mathbf{y}_{1:L}$ is *not* equivalent to the MMV-IAS algorithm as it does not consider joint sparsity. Our numerical examples in Section 6 demonstrate that this can lead to suboptimal results.

Remark 3.1 (Extensions of the IAS algorithm). Several advancements to the IAS algorithm have recently been made. In [8], hybrid versions were proposed to balance convex and non-convex models to enhance sparsity while avoiding stopping at non-global minima. In [40], path-following methods were

used to smoothly transition from convex to non-convex models. Additionally, in [32], the IAS algorithm was generalized for non-linear data models with ensemble Kalman methods. While beyond the scope of this current investigation, it would be beneficial to integrate our joint-sparsity-promoting approach with these advancements as deemed appropriate for a particular application.

Relationship to iteratively re-weighted least squares. The update steps (3.5) and (3.10) of Algorithm 3.1 can be understood as an iteratively re-weighted least squares (IRLS) algorithm [17, 22] with automatic inter-signal coupling. IRLS algorithms aim to recover sparse signals by assigning individual weights to the components of $\mathbf{x}$ and updating these weights iteratively. This concept is also applied in iteratively re-weighted ℓ^1-regularization methods [14]. The MMV-IAS framework provides a Bayesian interpretation for weighting strategies and can be used to tailor these weights based on statistical assumptions of the problem.

Relationship to group sparsity. We next address a possible generalization of our joint-sparsity-promoting hierarchical Bayesian model, and in the process, reveal its connection to group-sparsity-promoting models, as discussed in [6]. More precisely, we show that our joint-sparsity-promoting approach can be re-interpreted as a group-sparsity-promoting model, and generalizes the one in [6] in several ways. Observe that the joint prior (2.7) can be rewritten in product form as

$$\pi_{\mathbf{X}_{1:L}|\boldsymbol{\Theta}}(\mathbf{x}_{1:L}|\boldsymbol{\theta}) \propto \prod_{k=1}^{K} \left\{ \theta_k^{-L/2} \exp\left(-\frac{1}{2\theta_k} \sum_{l=1}^{L} [R\mathbf{x}_l]_k^2 \right) \right\}. \tag{3.11}$$

Furthermore, denoting by $[R\mathbf{x}_\bullet]_k = \big([R\mathbf{x}_1]_k, \dots, [R\mathbf{x}_L]_k\big) \in \mathbb{R}^L$ the vector that contains the kth components of the vectors $R\mathbf{x}_1, \dots, R\mathbf{x}_L$, (3.11) becomes

$$\pi_{\mathbf{X}_{1:L}|\boldsymbol{\Theta}}(\mathbf{x}_{1:L}|\boldsymbol{\theta}) \propto \prod_{k=1}^{K} \left\{ \theta_k^{-L/2} \exp\left(-\frac{1}{2\theta_k} \|[R\mathbf{x}_\bullet]_k\|_2^2 \right) \right\}. \tag{3.12}$$

Note that, in combination with a suitable generalized gamma hyper-prior, (3.12) promotes only a few of the groups $[R\mathbf{x}_\bullet]_1, \dots, [R\mathbf{x}_\bullet]_K$ not to be zero, where the kth group $[R\mathbf{x}_\bullet]_k$ is zero if and only if $\|[R\mathbf{x}_\bullet]_k\|_2 = 0$. This allows us to re-interpret the "joint-sparsity-promoting" prior (2.7) as a "group-sparsity-promoting" prior. Furthermore, observe that that is straightforward to replace $\|\cdot\|_2$ with other weighted norms. The product form (3.12) of the joint prior density implies that

$$[R\mathbf{x}_\bullet]_k \sim \mathcal{N}(\mathbf{0}, \theta_k I), \tag{3.13}$$

where $I \in \mathbb{R}^{L\times L}$ is the usual identity matrix. Or, in other words, the L components of the vector $[R\mathbf{x}_\bullet]_k \in \mathbb{R}^L$ are independent and identically distributed. However, we can relax this assumption to allow non-trivial correlations between the components to be modeled, which is necessary in some applications. For instance, see [6], which considered a hierarchical Bayesian model for the inverse problem of magnetoencephalography (MEG)—aiming at estimating electromagnetic cerebral activity from measurements of the magnetic fields outside the head. To this end, we can replace (3.13) by

$$[R\mathbf{x}_\bullet]_k \sim \mathcal{N}(\mathbf{0}, \theta_k C_K), \tag{3.14}$$

where $C_k \in \mathbb{R}^{L\times L}$ is an arbitrary covariance matrix. In this case, the joint prior (3.12) becomes

$$\pi_{\mathbf{X}_{1:L}|\boldsymbol{\Theta}}(\mathbf{x}_{1:L}|\boldsymbol{\theta}) \propto \prod_{k=1}^{K} \left\{ \theta_k^{-L/2} \exp\left(-\frac{1}{2\theta_k} \|[R\mathbf{x}_\bullet]_k\|_{C_k}^2 \right) \right\} \tag{3.15}$$

with norm $\|\mathbf{b}\|_{C_k}^2 = \mathbf{b}^T C_k^{-1}\mathbf{b}$. The usual Euclidean norm $\|\cdot\|_2$ corresponds to the special case $C_k = I$. Moreover, the objective function $\mathcal{G}$ in (3.3), which is the negative logarithm of the posterior, is then

$$\mathcal{G}(\mathbf{x}_{1:L}, \boldsymbol{\theta}) = \frac{1}{2}\sum_{l=1}^{L}\|F_l\mathbf{x}_l - \mathbf{y}_l\|_2^2 + \frac{1}{2}\sum_{k=1}^{K}\theta_k^{-1}\|[R\mathbf{x}_\bullet]_k\|_{C_k}^2 + \sum_{k=1}^{K}\left(\frac{\theta_k}{\vartheta_k}\right)^r - \eta\sum_{k=1}^{K}\log(\theta_k), \tag{3.16}$$

up to constants that neither depend on $\mathbf{x}_{1:L}$ nor $\boldsymbol{\theta}$, where $\eta = r\beta - (L/2+1)$. While the $\mathbf{x}_l$-updates of the IAS algorithm in subsection 3.1 are still the same as in subsection 3.2, the $\boldsymbol{\theta}$-update (3.8) in subsection 3.3 transforms into

$$\theta_k = \arg\min_{\theta_k}\left\{\frac{1}{2\theta_k}\|[R\mathbf{x}_\bullet]_k\|_{C_k}^2 + \left(\frac{\theta_k}{\vartheta_k}\right)^r - \eta\log(\theta_k)\right\}. \tag{3.17}$$

Hence we recover the hierarchical Bayesian model and the update rules in [6] as the special case of $L = 3$, $R = I$, and a usual gamma hyper-prior ($r = 1$).

3.5. Uncertainty quantification. Although we only solve for the MAP estimate of the posterior density $\pi_{\mathbf{X}_{1:L},\boldsymbol{\Theta}|\mathbf{Y}_{1:L}=\mathbf{y}_{1:L}}$ for given MMV data $\mathbf{y}_{1:L}$, we can still partially quantify uncertainty in the recovered parameter vectors. Specifically, for fixed hyper-parameters $\boldsymbol{\theta}$, Bayes' theorem yields

$$\pi_{\mathbf{X}_{1:L}|\boldsymbol{\Theta}=\boldsymbol{\theta},\mathbf{Y}_{1:L}=\mathbf{y}_{1:L}}(\mathbf{x}_{1:L}) \propto \pi_{\mathbf{Y}_{1:L}|\mathbf{X}_{1:L}}(\mathbf{y}_{1:L}|\mathbf{x}_{1:L})\,\pi_{\mathbf{X}_{1:L}|\boldsymbol{\Theta}}(\mathbf{x}_{1:L}|\boldsymbol{\theta}) \tag{3.18}$$

for the fully conditional posterior for the parameter vectors $\mathbf{X}_{1:L}$. Here, $\pi_{\mathbf{Y}_{1:L}|\mathbf{X}_{1:L}}$ is the likelihood density (2.2) and $\pi_{\mathbf{X}_{1:L}|\boldsymbol{\Theta}}$ is the prior (2.7). Substituting (2.2) and (2.7) into (3.18) yields

$$\pi_{\mathbf{X}_{1:L}|\boldsymbol{\Theta}=\boldsymbol{\theta},\mathbf{Y}_{1:L}=\mathbf{y}_{1:L}}(\mathbf{x}_{1:L}) \propto \exp\left(-\frac{1}{2}\sum_{l=1}^{L}\|F_l\mathbf{x}_l - \mathbf{y}_l\|_2^2 + \|D_{\boldsymbol{\theta}}^{-1/2}R\mathbf{x}_l\|_2^2\right). \tag{3.19}$$

For covariance matrix $\Gamma_l = (F_l^TF_l + R^TD_{\boldsymbol{\theta}}^{-1}R)^{-1}$ and mean $\boldsymbol{\mu}_l = \Gamma_l F_l^T\mathbf{y}_l$, we therefore have

$$\pi_{\mathbf{X}_l|\boldsymbol{\Theta}=\boldsymbol{\theta},\mathbf{Y}_l=\mathbf{y}_l}(\mathbf{x}_l) \propto \mathcal{N}(\mathbf{x}_l|\boldsymbol{\mu}_l,\Gamma_l). \tag{3.20}$$

The common kernel condition (3.6) ensures that Γ_l is an SPD covariance matrix. We can now quantify uncertainty in $\mathbf{X}_l$ by sampling from the normal distribution and subsequently determining, for instance, the sample mean and credible intervals. We refer to [46] for more details on sampling from high-dimensional Gaussian distributions.

Remark 3.2 (Full posterior sampling). Although our approach quantifies uncertainty in the parameter vectors, it does not account for the uncertainty in the hyper-parameters. To fully address uncertainty, Bayesian MAP estimation should be replaced with, for instance, full posterior sampling using a Markov chain Monte Carlo (MCMC) method. The goal of MCMC is to compute realizations of a Markov chain that is stationary w.r.t. the posterior distribution [35]. However, sampling from sparsity-promoting hierarchical models is challenging since (1) they are high-dimensional, which leads to high 'per sample' costs; (2) they can have multiple modes separated by regions of low density, which are challenging to traverse; and (3) there is a strong correlation between the parameters of primary interest and the hyper-parameters, resulting in poor mixing and slow convergence. Recent research, specifically in [12], has made strides in addressing the issue of high 'per sample' costs. This was achieved through a re-parameterization that converts the posterior into a form dominated by white Gaussian noise. Following this transformation, the preconditioned Crank–Nicholson (pCN) scheme was employed to efficiently sample from the transformed posterior. Nonetheless, challenges (2) and (3), concerning mode traversal and parameter correlation, respectively, remain unresolved. Extensive research is needed to address such challenges and is beyond the scope of this investigation.

4. Analysis: Complexity, convexity, and convergence. We briefly analyze the computational complexity, convexity, and convergence of the MMV-IAS algorithm (Algorithm 3.1) and the underlying objective function.

4.1. Computational complexity. Consider L parameter vectors $\mathbf{x}_1, \dots, \mathbf{x}_L \in \mathbb{R}^N$. Different methods can be used to solve the $\mathbf{x}_l$-updates in Algorithm 3.1. Assuming that we use the PCG method, each $\mathbf{x}_l$-update has computational complexity $\mathcal{O}(\tilde{N}_l)$, where $\tilde{N}_l$ is the number of non-zero elements of the matrix $F_l^T F_l + R^T D_{\boldsymbol{\theta}}^{-1/2} R$. In the worst case ($\tilde{N}_l = N^2$ for all $l = 1, \dots, L$), the computational cost for updating all parameter vectors is $\mathcal{O}(LN^2)$. As already noted, however, these updates can be performed in parallel. Furthermore, we perform the $\boldsymbol{\theta}$-update in Algorithm 3.1 using one of the explicit formulas (3.10). If $R \in \mathbb{R}^{K \times N}$ for $l = 1, \dots, L$, then the $\boldsymbol{\theta}$-update has computational complexity $\mathcal{O}(KN)$. See [29, Section 4.1] for more details. In sum, if we run Algorithm 3.1 for I iterations, then its overall order of operations is (at most) $\mathcal{O}(I(LN^2 + KN))$.

4.2. Convexity of the objective function. We next investigate the convexity of the objective function $\mathcal{G}$ in (3.3). Theorem 4.1 provides the choices of hyper-parameters $(r, \beta, \vartheta_{1:K})$ for which the objective function $\mathcal{G}$ is globally or locally convex, that is, when the convexity is restricted to specific values for $\boldsymbol{\theta}$. It also describes how the number of MMVs influences convexity.

Theorem 4.1 (Convexity of the objective function). *Let $\mathcal{G}$ be the objective function in (3.3) and $\eta = r\beta - (L/2 + 1)$.*

(a) If $r \geq 1$ and $\eta > 0$, then $\mathcal{G}$ is globally convex.

(b) If $0 < r < 1$ and $\eta > 0$, or if $r < 0$, then $\mathcal{G}$ is convex provided that

$$\theta_k < \vartheta_k \left(\frac{\eta}{r|r-1|} \right)^{1/r}, \quad k = 1, \dots, K. \tag{4.1}$$

Theorem 4.1 highlights the impact of MMV data on the convexity of the objective function $\mathcal{G}$ in our joint-sparsity-promoting hierarchical Bayesian model. In particular, as L increases, the linear decrease of η results in the condition $\eta > 0$ becoming more restrictive. Furthermore, since η decreases as L increases, we see in part (b) that the right-hand side of (4.1) is also smaller. That is, the convex set in which G is convex shrinks as L increases, revealing a trade-off between promoting joint sparsity and decreasing convexity as the number of coupled MMV data and parameter vectors increases. Theorem 4.1 is a natural extension of [9, Theorem 4.1] (also see [8, Theorem 3.1]), which we recover as the special case of $L = 1$ (and $R = I$). The proof is provided in Appendix A.

Remark 4.2 (Convergence of MMV-IAS). The findings in Theorem 4.1 impact the IAS algorithm's performance. When the objective function $\mathcal{G}$ exhibits global convexity, the MMV-IAS algorithm is guaranteed to converge to the unique minimum of $\mathcal{G}$. This follows from the standard theory for coordinate descent approaches [49, 4]. Although global convexity streamlines the computation of the MAP estimate, there is a strong justification for investigating alternative choices of r that yield hierarchical priors with enhanced sparsity-promoting properties. Empirical studies suggest that when the objective function is not globally convexity, the resulting sparsity of the minimizer is often increased. Nevertheless, a non-convex $\mathcal{G}$ can lead to the emergence of misleading local minima, potentially causing the MMV-IAS algorithm to become entrapped in one. To overcome the risk of the algorithm prematurely converging to an incorrect local minimum, [8] recommends the adoption of hybrid versions of the IAS algorithm in the single-measurement-vector case. In such constructions, the global convergence traits of gamma hyper-priors ($r = 1$) are leveraged initially to close in on the vicinity of the unique global minimum, after which there is a shift to a generalized gamma hyper-prior with $r < 1$ to provide a stronger sparsity stimulus.

5. Extension to generalized sparse Bayesian learning. We now briefly demonstrate how our method for fostering joint sparsity can be incorporated into the GSBL framework [44, 47, 29]. Sparse Bayesian learning (SBL), first introduced in [44], is a statistical approach that employs Bayesian inference to recover sparse solutions from indirect, incomplete, and noisy data. This technique is characterized by combining a conditional zero-mean Gaussian prior with a gamma hyper-prior for the precision of the Gaussian prior. Traditional SBL methods have predominantly operated under the sparsity assumption in the parameter vector $\mathbf{x}$. However, the recent work [29] broadened this framework by proposing that sparsity can also apply to some linear transformation of the parameter vector, denoted as $R\mathbf{x}$. Here, R is permitted to possess a non-trivial kernel, provided the common kernel condition $\ker(F) \cap \ker(R) = \mathbf{0}$ holds. This extension led to the development of the GSBL approach. Within this context, we now further evolve the GSBL method to encourage *joint* sparsity in the case of MMVs corresponding to jointly sparse parameter vectors.

As highlighted in the introduction, both IAS and GSBL are established cases of sparsity-promoting algorithms that can benefit from our joint sparsity-promoting priors in the presence of MMV data. Importantly, these priors are not restricted to the discussed algorithms, as they can also be employed to enhance the performance of other sparsity-promoting MAP estimators, demonstrating their versatile applicability.

5.1. The hierarchical Bayesian model. The main difference between the hierarchical model discussed in Section 2 and the one underlying GSBL is that the latter treats the diagonal entries of the precision (inverse covariance) matrix $D_{\boldsymbol{\theta}}$ as gamma distributed random variables. In this case, the joint prior is

$$\pi_{\mathbf{X}_{1:L}|\boldsymbol{\Theta}}(\mathbf{x}_{1:L}|\boldsymbol{\theta}) \propto \det(D_{\boldsymbol{\theta}})^{L/2} \exp\left(-\frac{1}{2}\sum_{l=1}^{L} \|D_{\boldsymbol{\theta}}^{1/2} R\mathbf{x}_l\|_2^2\right) \tag{5.1}$$

rather than (2.7) and the gamma hyper-prior is

$$\pi_{\boldsymbol{\Theta}}(\boldsymbol{\theta}) = \prod_{k=1}^{K} \mathcal{GG}(\theta_k|1,\beta,\vartheta_k) \propto \det(D_{\boldsymbol{\theta}})^{\beta-1} \exp\left(-\sum_{k=1}^{K} \theta_k/\vartheta_k\right) \tag{5.2}$$

rather than (2.9). We still assume the joint likelihood function (2.2).

5.2. Bayesian inference. We perform Bayesian inference for the GSBL model by again solving for the MAP estimate of its posterior. To this end, a block-coordinate descent approach similar to the IAS algorithm was recently investigated in [29] (also see [51]). For the GSBL model above, the objective function $\mathcal{G}$ that is minimized by the MAP estimate is

$$\mathcal{G}(\mathbf{x}_{1:L},\boldsymbol{\theta}) = \frac{1}{2}\left(\sum_{l=1}^{L} \|F_l\mathbf{x}_l - \mathbf{y}_l\|_2^2 + \|D_{\boldsymbol{\theta}}^{1/2} R\mathbf{x}_l\|_2^2\right) + \sum_{k=1}^{K} \frac{\theta_k}{\vartheta_k} + (-L/2+1-\beta)\sum_{k=1}^{K} \log(\theta_k) \tag{5.3}$$

up to constants that neither depend on $\mathbf{x}_{1:L}$ nor $\boldsymbol{\theta}$. We again minimize $\mathcal{G}$ by alternatingly (i) minimizing $\mathcal{G}$ w.r.t. $\mathbf{x}_{1:L}$ for fixed $\boldsymbol{\theta}$ and (ii) minimizing $\mathcal{G}$ w.r.t. $\boldsymbol{\theta}$ for fixed $\mathbf{x}_{1:L}$. In the case of GSBL, updating $\mathbf{x}_{1:L}$ given $\boldsymbol{\theta}$ reduces to solving the quadratic optimization problems

$$\mathbf{x}_l = \arg\min_{\mathbf{x}} \left\{\|F_l\mathbf{x} - \mathbf{y}_l\|_2^2 + \|D_{\boldsymbol{\theta}}^{1/2} R\mathbf{x}\|_2^2\right\}, \quad l = 1,\dots,L. \tag{5.4}$$

Moreover, using the same arguments as in subsection 3.3, the minimizer of $\mathcal{G}$ w.r.t. $\boldsymbol{\theta}$ for fixed $\mathbf{x}_{1:L}$ is

$$\theta_k = \frac{L/2 - 1 + \beta}{\sum_{l=1}^{L} [R\mathbf{x}_l]_k^2/2 + \vartheta_k^{-1}}, \quad k = 1, \dots, K. \tag{5.5}$$

We refer to [29] for more details. Algorithm 5.1 summarizes the above procedure to approximate the MAP estimate of the joint-sparsity-promoting GSBL model above. Henceforth we refer to this method as the *MMV-GSBL algorithm*.

Algorithm 5.1 The MMV-GSBL algorithm

1: Choose model parameters $(\beta, \boldsymbol{\vartheta})$ and initialize $\boldsymbol{\theta}$
2: **repeat**
3: Update the parameter vectors $\mathbf{x}_{1:L}$ (in parallel) according to (5.4)
4: Update the hyper-parameters $\boldsymbol{\theta}$ according to (5.5)
5: **until** convergence or the maximum number of iterations is reached

Remark 5.1 (Uncertainty quantification). We can partially quantify uncertainty in the parameter vectors recovered by the MMV-GSBL method described in Algorithm 5.1 by following the discussion in subsection 3.5. The only difference to the MMV-IAS algorithm is that the covariance matrices Γ_l in (3.20) become $\Gamma_l = (F_l^T F_l + R^T D_{\boldsymbol{\theta}} R)$ in the MMV-GSBL framework.

5.3. Analysis. The analysis carried out for the MMV-IAS model and algorithm in Section 4 can be extended to the MMV-GSBL model and algorithm. Both algorithms share a similar computational complexity of $\mathcal{O}(I(LN^2 + KN))$. However, as mentioned in previous studies [47, 29], the GSBL cost function can exhibit non-convexity with multiple local minima. This non-convexity is expected to persist in the MMV-GSBL cost function as well.

5.4. Connection to existing methods. Recovering jointly sparse signals from MMV data using SBL was considered in [48]. The MMV-SBL method proposed [48] has certain limitations, however, including restrictions on the forward operators, the noise distribution, and the requirement of sparse parameter vectors. Furthermore, the evidence approach used in [48] can slow performance for large problems. In contrast, the MMV-GSBL algorithm (Algorithm 5.1) is more efficient and flexible. It allows for varying forward operators, different noise distributions, and more general regularization operators promoting sparsity in an arbitrary linear transformation of the parameter vectors, This makes the proposed MMV-GSBL algorithm suitable for various MMV problems and large-scale parameter vectors.

6. Numerical results. We conduct numerical experiments to showcase the effectiveness of our joint-sparsity-promoting MMV-IAS and MMV-GSBL algorithms, detailed in Algorithms 3.1 and 5.1. For a fair comparison, we also evaluate the individual signal recovery performance using the traditional IAS and GSBL algorithms with the same model parameters. The MATLAB code used to generate the numerical tests can be found in the code repository https://github.com/jglaubitz/LeveragingJointSparsity.

6.1. Hyper-prior parameter selection. For all signal recovery problems, we either chose $(\beta, \vartheta) = (1, 1.501, 10^{-2})$ for the IAS algorithm and $(\beta, \vartheta) = (1, L/2 + 1.501, 10^{-2})$ for the MMV-IAS algorithm, resulting in globally convex objective functions, or $(r, \beta, \vartheta) = (-1, 1, 10^{-4})$ for the IAS and MMV-IAS algorithm, resulting in non-convex objective functions. Moreover, we use $(\beta, \vartheta) = (1, 10^3)$ for the GSBL and MMV-GSBL algorithm, resulting in a non-convex objective function. Similar parameters were used in [29, 52] and [9, 8] for the GSBL and IAS algorithm, respectively. We did not attempt to optimize any of these parameters.

6.2. Signal deblurring. We first consider (jointly) deblurring four piecewise-constant signals with a shared edge profile. The signals are generated by fixing five transition points in the interval $[0,1]$, dividing $[0,1]$ into six constant subintervals on which the signals are constant, and then randomly assigning signal values drawn from a uniform distribution. The values are then normalized such that the maximum value of each signal is set to 1. Figures 1a and 1c in Section 1 show the first two signals and the given noisy blurred measurements. We aim to recover the nodal values $\mathbf{x}_{1:4}$ of all four signals at $N = 40$ equidistant grid points. The corresponding data model is

$$\mathbf{y}_l = F\mathbf{x}_l + \mathbf{e}_l, \quad l = 1, \dots, 4. \tag{6.1}$$

The discrete forward operator, F, represents the application of the midpoint quadrature to the convolution equation

$$y(s) = \int_0^1 k(s - s')x(s)\,\mathrm{d}s', \tag{6.2}$$

where we assume a Gaussian convolution kernel of the form

$$k(s) = \frac{1}{2\pi\gamma^2}\exp\left(-\frac{s^2}{2\gamma^2}\right) \tag{6.3}$$

with blurring parameter $\gamma = 3 \cdot 10^{-2}$. The forward operator is then given by

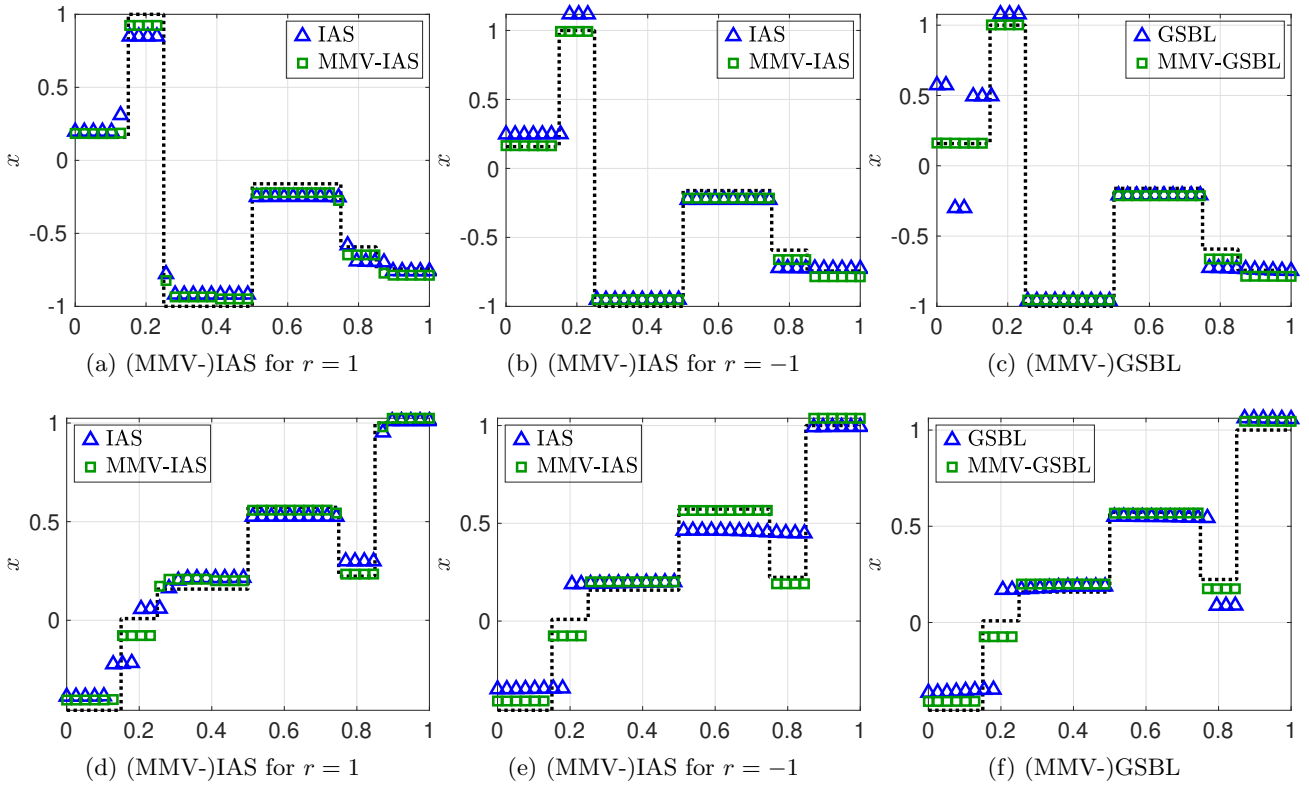
$$[F]_{m,n} = hk(h[i-j]), \quad i,j = 1,\dots,n, \tag{6.4}$$

where $h = 1/n$ is the distance between consecutive grid points. Note that F has full rank but quickly becomes ill-conditioned. The noise vectors $\mathbf{e}_{1:4}$ in (6.1) are i.i.d., with zero mean and a common variance $\sigma^2 = 10^{-2}$. To reflect our prior knowledge that the signals are piecewise constant, we use

$$R = \begin{bmatrix} -1 & 1 & & \\ & \ddots & \ddots & \\ & & -1 & 1 \end{bmatrix} \in \mathbb{R}^{(n-1)\times n} \tag{6.5}$$

for the sparsifying operator. Figures 3a and 3d show the recovered first two signals using the IAS algorithm (to promote sparsity separately) and the proposed MMV-IAS algorithm (to promote sparsity jointly) for $r = 1$, resulting in a globally convex objective function. Figures 3b and 3e show the same results for $r = -1$, resulting in a non-convex objective function. Figures 3c and 3f report on the same test using the GSBL and MMV-GSBL algorithms. The results demonstrate that incorporating joint sparsity into the IAS and GSBL algorithms improves signal recovery accuracy.

The improved accuracy of the MMV-IAS and MMV-GSBL algorithms can be attributed to the use of a common hyper-parameter vector $\boldsymbol{\theta}$ that more accurately detects edge locations compared to separate hyper-parameter vectors $\boldsymbol{\theta}_{1:L}$ used in the IAS and GSBL algorithms. This is evident in Figure 4, which shows the normalized estimated hyper-parameters produced by the IAS/GSBL and MMV-IAS/GSBL algorithms for the first two signals. While the MMV-IAS and MMV-GSBL algorithms accurately capture all edge locations, the IAS and GSBL algorithms produce visibly erroneous hyper-parameter profiles. The impact of missed true edge locations and false detection of others are clearly visible in Figure 3c, which shows that the MMV-GSBL approach eliminates the artificial edges around 0.1 on the horizontal axis. Similarly, Figure 3f demonstrates that the MMV-GSBL approach retains the existing edges around 0.2 on the horizontal axis, which are missed by the GSBL approach.



(a) (MMV-)IAS for $r = 1$ (b) (MMV-)IAS for $r = -1$ (c) (MMV-)GSBL

(d) (MMV-)IAS for $r = 1$ (e) (MMV-)IAS for $r = -1$ (f) (MMV-)GSBL

Figure 3: Different reconstructions of the first (top row) and second (bottom row) of four piecewise constant signals with a common edge profile and noisy blurred measurements using the existing IAS/GSBL algorithm to separately recover them (blue triangles) and the proposed MMV-IAS/-GSBL algorithm to jointly recover them (green squares).

The proposed MMV-IAS and MMV-GSBL algorithms have the additional advantage of quantifying uncertainty in the recovered signals, as described in subsection 3.5 and Remark 5.1. This is demonstrated in Figure 5, which shows the 99.9% credible intervals of the fully conditional posterior densities $\pi_{\mathbf{X}_1|\boldsymbol{\Theta}=\boldsymbol{\theta},\mathbf{Y}_{1:L}=\mathbf{y}_{1:L}}$ of the first recovered signal for the IAS, MMV-IAS, GSBL, and MMV-GSBL model. Here, $\mathbf{y}_{1:L}$ are the given noisy blurred MMVs and $\boldsymbol{\theta}$ is the estimated hyper-parameter vector.

6.3. Error and success analysis. We now conduct a synthetic sparse signal recovery experiment to further assess the performance of the proposed joint-sparsity-promoting MMV-IAS and MMV-GSBL algorithms. We consider L randomly generated signals, $\mathbf{x}_{1:L}$, each of size N. We fix the number of measurements, M, non-zero components, s, and trials, T. For each trial, $t = 1, \ldots, T$, we proceed as follows:

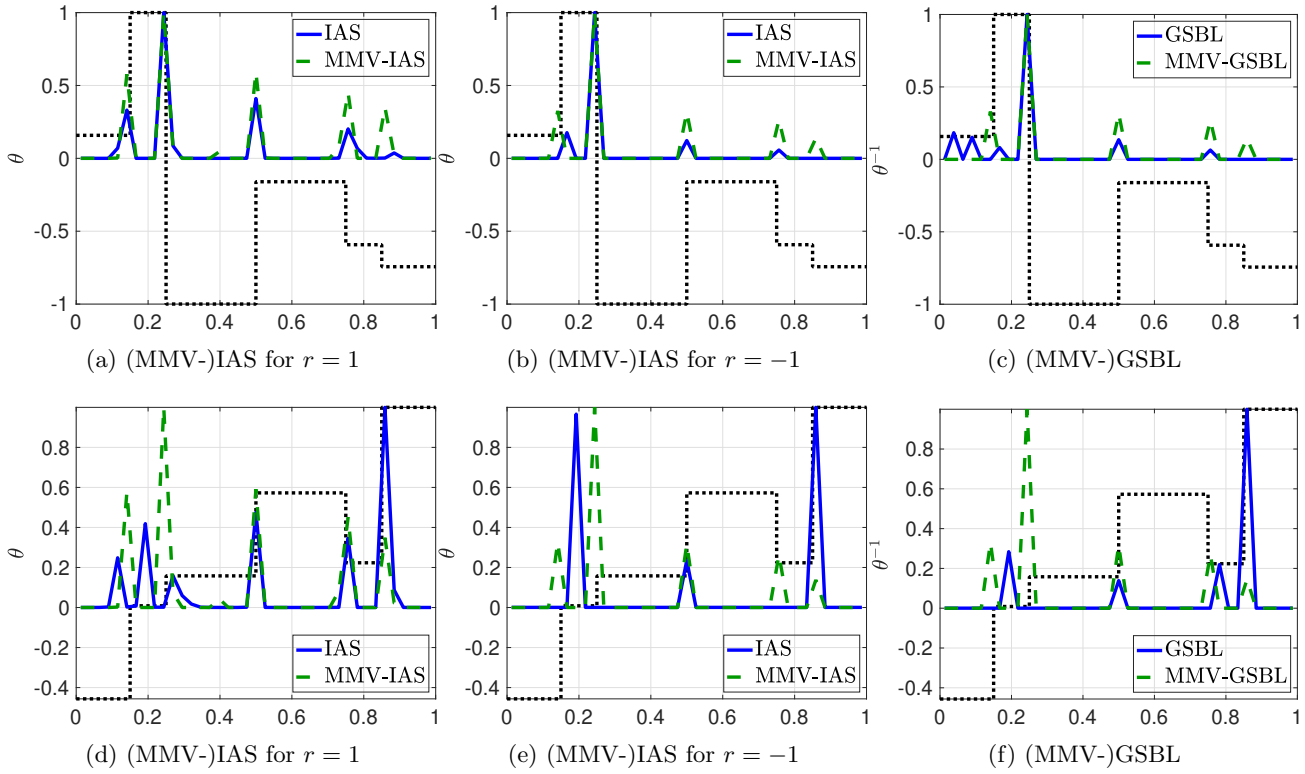
(i) Generate a support set $S \subset \{1, \ldots, N\}$ uniformly at random with size $|S| = s$;

(ii) Define signal vectors $\mathbf{x}_1, \ldots, \mathbf{x}_L$ such that $\text{supp}(\mathbf{x}_1) = \cdots = \text{supp}(\mathbf{x}_L) = S$, where the non-zero entries are drawn from the standard normal distribution;

(iii) Generate a forward operator F as described below, fix a noise variance σ^2, and compute the measurement vectors $\mathbf{y}_l = F\mathbf{x}_l + \mathbf{e}_l$, where $\mathbf{e}_l$ is drawn from $\mathcal{N}(\mathbf{0}, \sigma^2 I)$;

(iv) Compute the reconstructions $\hat{\mathbf{x}}_1, \ldots, \hat{\mathbf{x}}_L$ using the desired algorithm (e.g., IAS or MMV-IAS);

(v) Compute the normalized error $E_t = \sqrt{\sum_{l=1}^{L} \|\mathbf{x}_l - \hat{\mathbf{x}}_l\|_2^2 / \sum_{l=1}^{L} \|\mathbf{x}_l\|_2^2}$ for each algorithm.



(a) (MMV-)IAS for $r = 1$ (b) (MMV-)IAS for $r = -1$ (c) (MMV-)GSBL

(d) (MMV-)IAS for $r = 1$ (e) (MMV-)IAS for $r = -1$ (f) (MMV-)GSBL

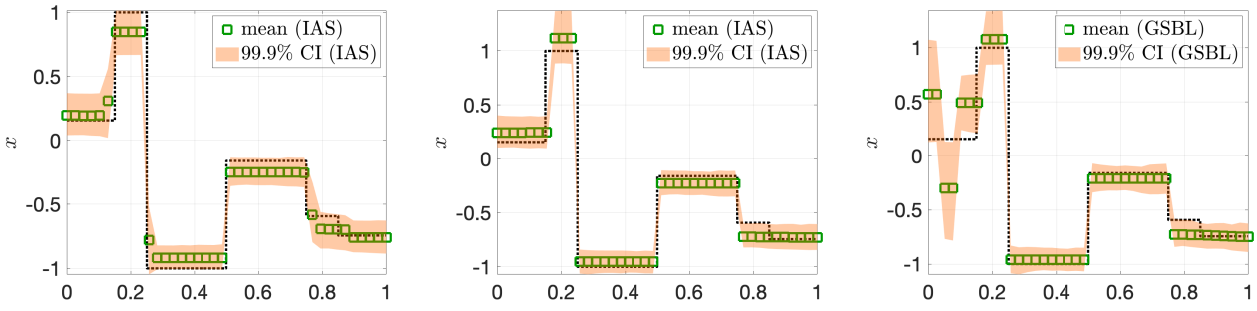
Figure 4: Normalized MAP estimate of the hyper-parameter θ for the first (top row) and second (bottom row) signal using the IAS and MMV-IAS algorithms with $r = \pm 1$ and the GSBL and MMV-GSBL algorithms

Finally, we evaluate the algorithm performance using the average error and empirical success probability (ESP). The *average error* is $E = (E_1 + \cdots + E_T)/T$, i.e., the average of the individual trial errors. The *ESP* is the fraction of trials that successfully recovered the vectors $\mathbf{x}_1, \ldots, \mathbf{x}_L$ up to a given tolerance ε_{tol}, i.e., $E_t < \varepsilon_{\text{tol}}$. We use a subsampled discrete cosine transform (DFT) as the forward operator F. This mimics the situation in which Fourier data are collected (e.g., synthetic aperture radar and magnetic resonance imaging), but some of the data are determined to be unusable due to a system malfunction or obstruction. Specifically, we generate a set $\Omega \subset \{1, \ldots, N\}$ of size M uniformly at random and let

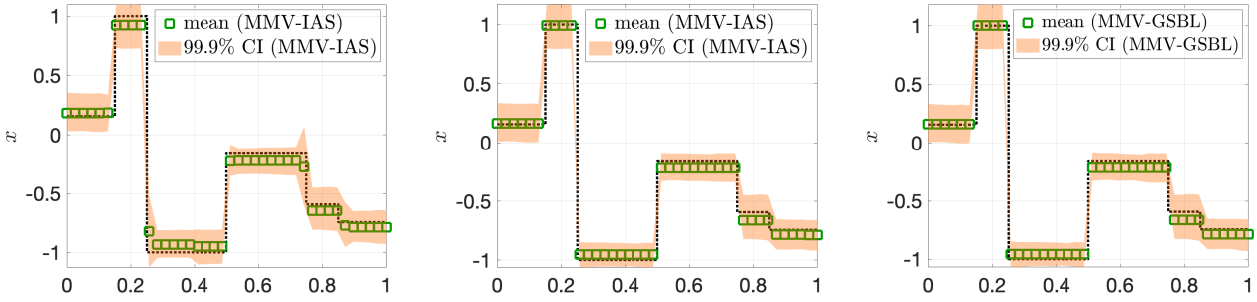
$$F = P_\Omega A, \tag{6.6}$$

where $A \in \mathbb{R}^{N \times N}$ is the DCT matrix and $P_\Omega \in \mathbb{R}^{M \times N}$ is the operator selecting rows of A corresponding to the indices in Ω. The identity matrix is used as the sparsifying operator as the signals are assumed to be sparse. In our experiments we set $N = 100$, $s = 20$, $T = 10$, $\sigma^2 = 10^{-6}$, and $\varepsilon_{\text{tol}} = 10^{-2}$.

The performance of the proposed joint-sparsity-promoting MMV-IAS and MMV-GSBL algorithms is evaluated and compared to the existing IAS and GSBL algorithms in Figures 6 and 7. These figures report the average errors and ESP for different numbers of MMVs ($L = 4, 8, 16$). For brevity, we only report on the IAS and MMV-IAS results for $r = -1$. The results show that the proposed algorithms outperform the existing ones regarding average errors and ESP in most cases. As the number of MMVs increases, the superiority of the proposed algorithms becomes more pronounced. For instance, when $L = 16$, the MMV-IAS algorithm requires only around 40 measurements per signal for successful recovery, while the IAS algorithm requires around 70. The phase transition plots in Figure 8 further



(a) IAS algorithm for $r = 1$ (b) IAS algorithm for $r = -1$ (c) GSBL algorithm



(d) MMV-IAS algorithm for $r = 1$ (e) MMV-IAS algorithm for $r = -1$ (f) MMV-GSBL algorithm

Figure 5: The 99.9% credible intervals (CIs) for the recovered first (top row) and second (bottom row) signal conditioned on the MAP estimate of the hyper-parameter vector $\boldsymbol{\theta}^{\text{MAP}}$ using the IAS and MMV-IAS algorithm with $r = \pm 1$ as well as the GSB and MMV-GSBL algorithm

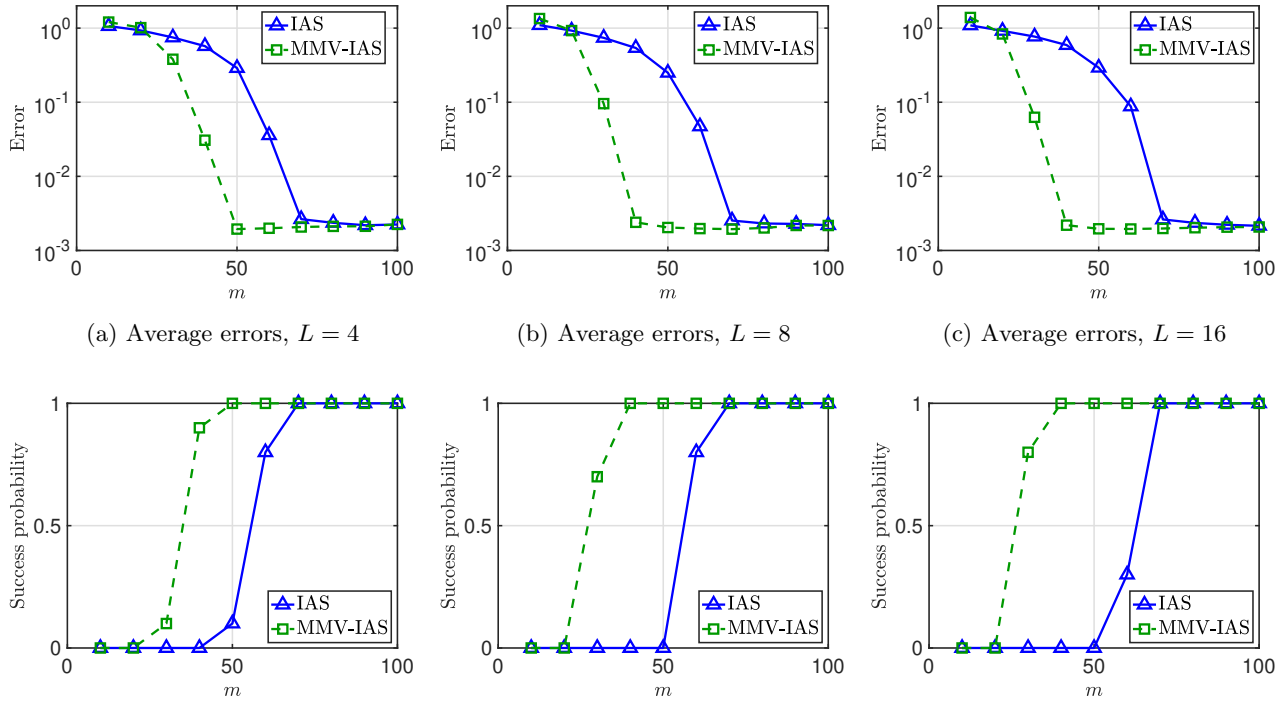
demonstrate the improved performance of the MMV-IAS algorithm, which exhibits a phase transition close to the optimal $m = s$ line. The phase transition profiles for the MMV-GSBL algorithm are similar to the MMV-IAS algorithm but are not included here for brevity.

6.4. Application to parallel magnetic resonance imaging. We next apply the proposed MMV-IAS and MMV-GSBL algorithms to a parallel MRI test problem. Parallel MRI is a multi-sensor acquisition system that uses multiple coils to simultaneously acquire image measurements for recovery. Details on parallel MRI can be found in [30, 19, 18, 1]. A standard discrete data model for parallel MRI is the following: Let $\mathbf{x} \in \mathbb{C}^N$ be the vectorized image to be recovered and L be the number of coils. For the lth coil, the measurements acquired are

$$\mathbf{y}_l = P_{\Omega_l} F \mathbf{x} + \mathbf{e}_l, \tag{6.7}$$

where $F \in \mathbb{C}^{N \times N}$ is the discrete Fourier transform (DFT) matrix, $P_{\Omega_l} \in \mathbb{C}^{M \times N}$ is a sampling operator that selects the rows of F corresponding to the frequencies in Ω_l, and $\mathbf{e}_l \in \mathbb{C}^M$ is noise. The image measurement acquired by each of the L coils is intrinsic to the particular coil. A typical sampling procedure for parallel MRI is to use data taken as radial line sampling in the Fourier space. Figure 9 illustrates the radial sampling maps corresponding to four different coils.

Many techniques have been proposed for parallel MRI, see [19] and references therein. Here we focus on the coil-by-coil approach, first computing the approximate coil images $\hat{\mathbf{x}}_{1:L}$ from (6.7) and then computing an approximation $\hat{\mathbf{x}}$ to the overall image by considering the average of the coil images,



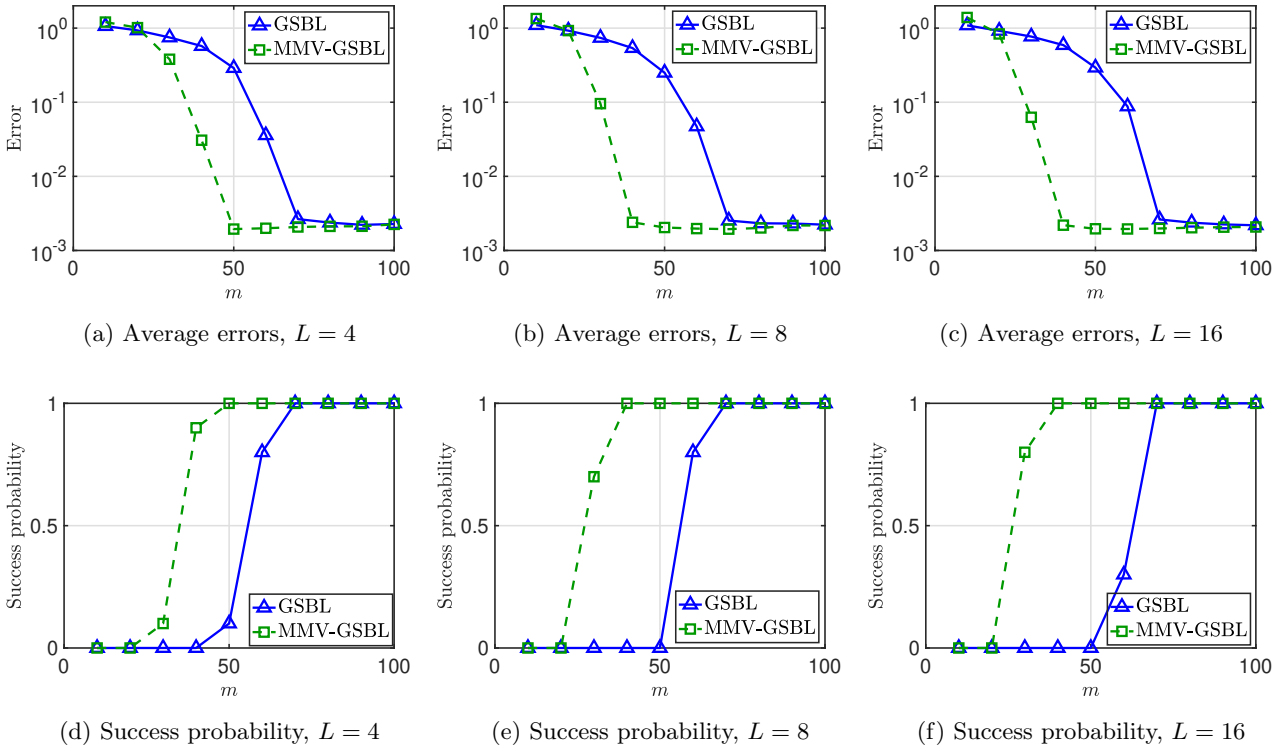
(a) Average errors, $L = 4$ (b) Average errors, $L = 8$ (c) Average errors, $L = 16$

(d) Success probability, $L = 4$ (e) Success probability, $L = 8$ (f) Success probability, $L = 16$

Figure 6: Comparison of the average error and success probability for the sparsity-promoting IAS algorithm (blue triangles) and joint-sparsity-promoting MMV-IAS algorithm (green squares), both for $r = -1$. We recover a signal of size $N = 100$ with $s = 20$ non-zero entries from an increasing number of measurements m.

i.e., $\hat{\mathbf{x}} = (\hat{\mathbf{x}}_1 + \cdots + \hat{\mathbf{x}}_L)/L$. In our experiment, we compare the recovery of the 256×256 Shepp–Logan phantom image in Figure 10a using the least-squares (LS) approach, the existing IAS/GSBL algorithm, and the proposed MMV-GSBL/IAS algorithm to recover the coil images. For brevity, we only report on the IAS and MMV-IAS results for $r = -1$. We used the anisotropic first-order discrete gradient operator as the sparsifying operator. Figure 10 shows the recovered first coil images for 20 lines, noise variance $\sigma^2 = 10^{-3}$, and $L = 4$ coils. The recovered first coil images using the proposed MMV-IAS (Figure 10e) and MMV-GSBL (Figure 10f) algorithms are visibly more accurate than using the corresponding IAS (Figure 10b) and GSBL (Figure 10c) algorithms. Note the sharper transitions between internal structures. Consequently, the proposed MMV-IAS/GSBL algorithm also yields a more accurate approximation to the overall image, compared to the existing IAS/GSBL algorithm, which is demonstrated in Figure 11. To further assess the performance of the proposed joint-sparsity-promoting MMV-IAS and MMV-GSBL algorithms, Figure 12 reports the relative error of the recovered overall image for varying numbers of lines sampled in the Fourier space. The proposed MMV-IAS/GSBL algorithm to jointly recover the coil images consistently yields the smallest error.

6.5. Comparison with a sequential approach. Comparing the proposed MMV-IAS/GSBL algorithm solely with the traditional IAS/GSBL algorithm for separate recovery of parameter vectors may not be entirely equitable. The MMV-IAS/GSBL algorithm leverages information from all parameter vectors to reconstruct each individually. In contrast, the traditional IAS/GSBL does not facilitate information sharing across different parameter vectors. Consequently, we also compare the MMV-IAS/GSBL



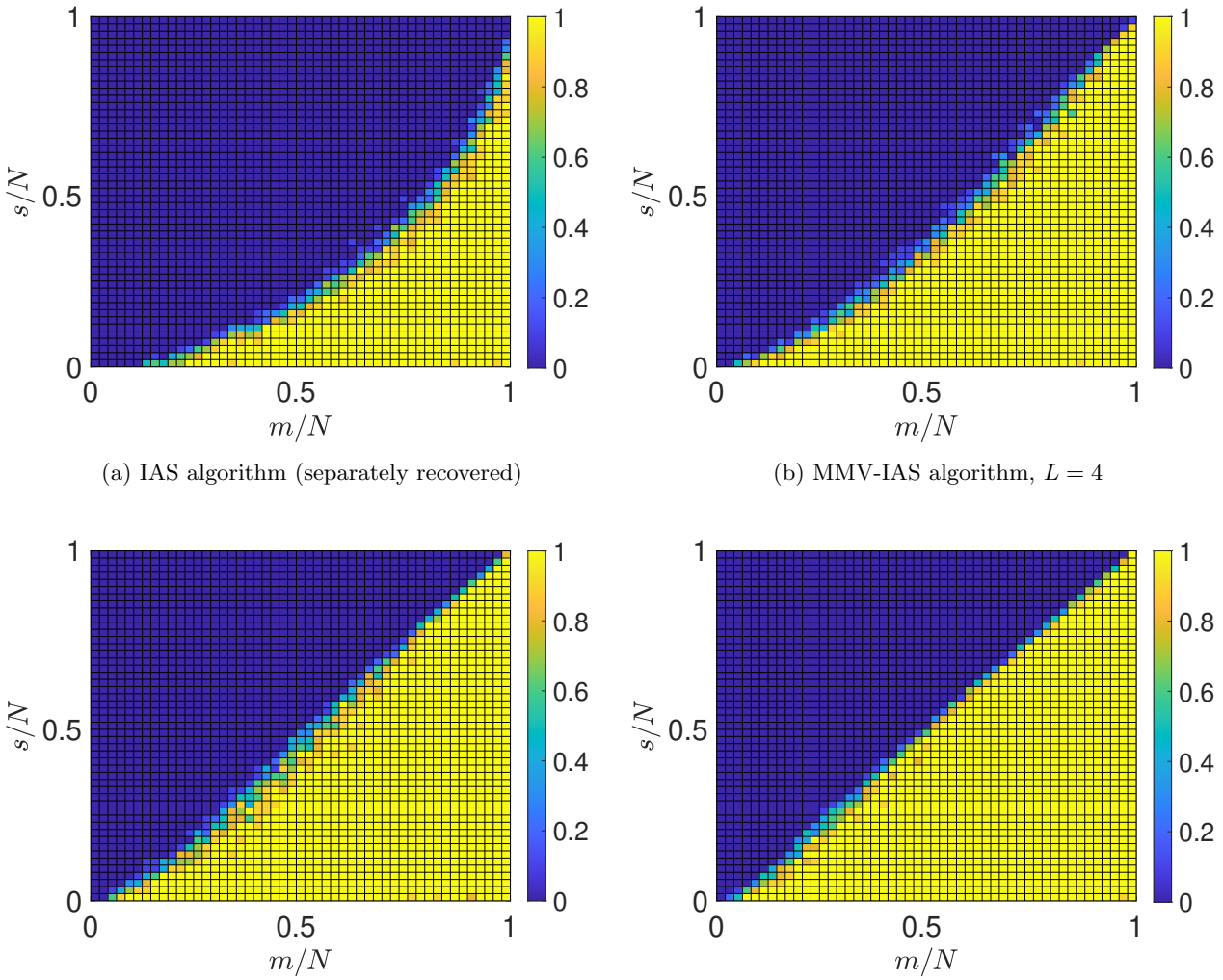
(a) Average errors, $L = 4$ (b) Average errors, $L = 8$ (c) Average errors, $L = 16$

(d) Success probability, $L = 4$ (e) Success probability, $L = 8$ (f) Success probability, $L = 16$

Figure 7: Comparison of the average error and success probability for the sparsity-promoting GSBL algorithm (blue triangles) and joint-sparsity-promoting MMV-GSBL algorithm (green squares). We recover a signal of size $N = 100$ with $s = 20$ non-zero entries from an increasing number of measurements m.

algorithm with a sequential variant of the IAS/GSBL algorithm. In the sequential IAS/GSBL algorithm, we determine the lth parameter vector $\mathbf{x}_l$ and its corresponding hyper-parameter vector $\boldsymbol{\theta}_l$ by approximating the MAP estimate of the lth posterior $\pi_{\mathbf{X}_l,\boldsymbol{\Theta}_l}(\mathbf{x}_l,\boldsymbol{\theta}_l) \propto \pi_{\mathbf{Y}_l|\mathbf{X}_l}(\mathbf{y}_l|\mathbf{x}_l)\,\pi_{\mathbf{X}_l|\boldsymbol{\Theta}_l}(\mathbf{x}_l|\boldsymbol{\theta}_l)\,\pi_{\boldsymbol{\Theta}_l}(\boldsymbol{\theta}_l)$—as it is done in the IAS/GSBL algorithm. However, unlike the traditional IAS/GSBL algorithm, the initial value for $\boldsymbol{\theta}_l$ in the corresponding block-coordinate descent method is chosen as the MAP estimate $\boldsymbol{\theta}_{l-1}^{\mathrm{MAP}}$ derived from the previously learned parameter vector. This approach is reminiscent of strategies employed in time-dependent problems where data are received in sequential batches. For the first parameter vector, the IAS/GSBL and the sequential IAS/GSBL algorithms start with the same initialization for $\boldsymbol{\theta}_1$. For this reason, we do not report on the first parameter vector in the subsequent numerical tests. However, from the second parameter vector onward, their initializations diverge. The sequential approach ensures that the insights obtained from the previous measurement vector $\mathbf{y}_l$ and the learned $\mathbf{x}_l, \boldsymbol{\theta}_l$ are not disregarded, and as such facilitate a reasonable comparison for the proposed MMV-IAS/GSBL algorithm.

Figure 13 compares the reconstructions and normalized hyper-parameter estimates for the last three of four piecewise constant signals with a common edge profile for the IAS, sequential IAS, and MMV-IAS algorithms. All methods use a generalized gamma hyper-prior with $r = 1$, ensuring globally convex objective functions. The results reveal only minor differences between the IAS and sequential IAS algorithms in this particular scenario. The primary reason for this similarity is the global convexity of the objective function common to both algorithms. Despite their differing hyper-parameter initialization,



(a) IAS algorithm (separately recovered)

(b) MMV-IAS algorithm, $L = 4$

(c) MMV-IAS algorithm, $L = 8$

(d) MMV-IAS algorithm, $L = 16$

Figure 8: Phase transition diagrams for the IAS and MMV-IAS algorithm for $r = -1$ and $L = 4, 8, 16$. The diagrams show the success probability for values $1 \leq s \leq N$ and $1 \leq M \leq N$.

they are both theoretically guaranteed to converge to the same unique minimum as the iteration count approaches infinity. The slight discrepancies observed between the IAS and sequential IAS algorithms, such as those noted in Figure 13b, can be attributed to the early termination of the block-coordinate descent method. In comparison, when juxtaposed with the IAS and sequential IAS algorithms, the MMV-IAS algorithm demonstrates improved performance, particularly in terms of more precise edge detection and overall signal recovery.

Figure 14 offers a comparison akin to the previous one, but under a generalized gamma hyper-prior with $r = -1$, leading to non-convex objective functions. In this scenario, there are significant differences between the IAS and sequential IAS algorithms, with the latter appearing to underperform. For example, as shown in Figure 14a, the sequential IAS algorithm fails to detect the final edge at 0.85, whereas the traditional IAS algorithm successfully identifies it. Similar trends are evident in

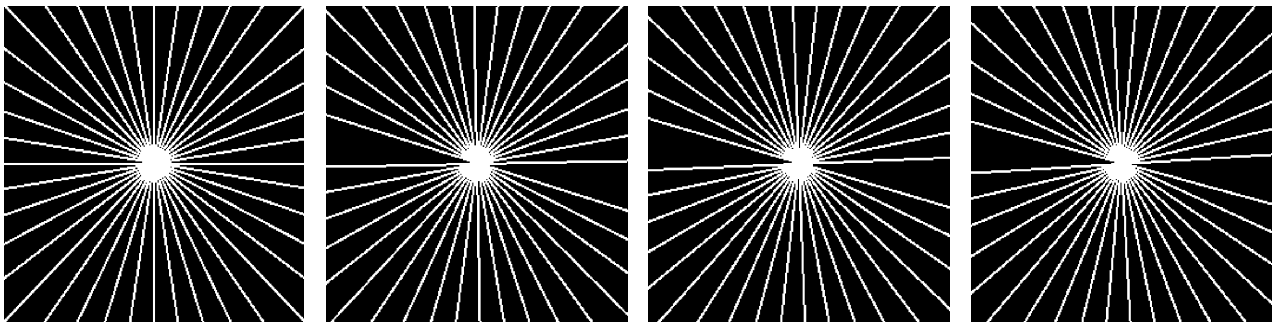


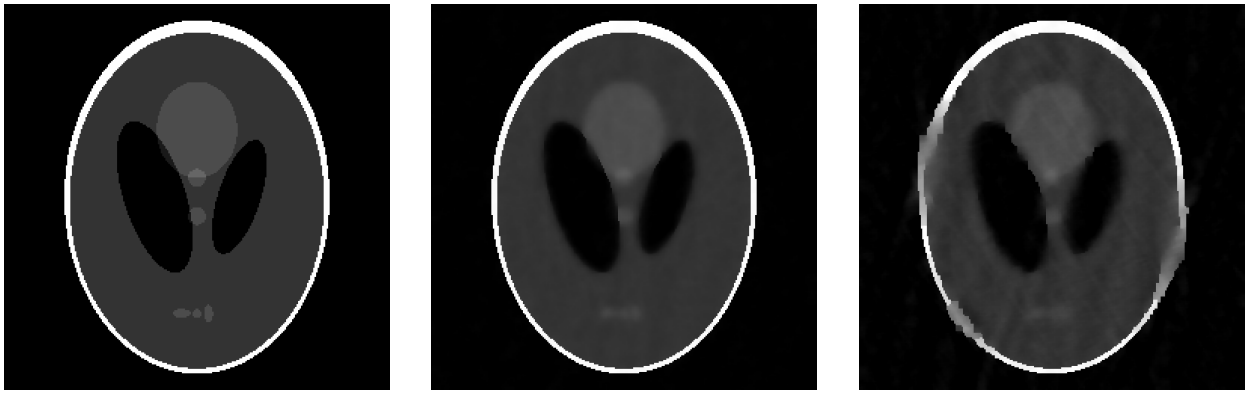
Figure 9: Four different radial sampling maps

Figures 14b and 14c. These results suggest that initializing the hyper-parameter vector $\boldsymbol{\theta}_l$ in the block-coordinate descent method with the previous MAP estimate $\boldsymbol{\theta}_{l-1}^{\text{MAP}}$ might steer the algorithm towards a less favorable local minimum compared to a fresh initialization. Altering the recovery sequence of the parameter vectors could improve the sequential IAS algorithm's performance. Nonetheless, unless guided by the problem's context, identifying an optimal order poses a challenging and non-trivial task. In contrast, when compared with the IAS and sequential IAS algorithms, the MMV-IAS algorithm consistently demonstrates superior performance, especially in terms of more precise edge detection and overall signal recovery.

In the concluding analysis, Figure 15 provides a comparison analogous to the earlier ones, but this time for the GSBL, sequential GSBL, and MMV-GSBL algorithms, all of which lead to non-convex objective functions. In this case, moderate differences are observed between the GSBL and sequential GSBL algorithms. As illustrated in Figures 15a and 15c, the sequential GSBL algorithm produces improved outcomes compared to the GSBL algorithm, while in Figure 15b, the GSBL algorithm demonstrates superior performance over its sequential counterpart. These findings suggest that initializing the hyper-parameter vector $\boldsymbol{\theta}_l$ in the block-coordinate descent method with the prior MAP estimate $\boldsymbol{\theta}_{l-1}^{\text{MAP}}$ can variably influence the algorithm, leading it towards either a more or less favorable local minimum compared to a fresh initialization. In contrast, the MMV-GSBL algorithm exhibits superior overall performance.

7. Concluding remarks. We presented a hierarchical Bayesian approach for inferring parameter vectors from MMVs that promotes joint sparsity. The method involves using separate conditionally Gaussian priors for each parameter vector and common hyper-parameters to enforce a common sparsity profile among the parameter vectors. Based on this joint-sparsity-promoting hierarchy, new algorithms, MMV-IAS and MMV-GSBL, were developed and demonstrated to outperform existing IAS and GSBL algorithms in several test cases. Our findings show that incorporating joint sparsity into the current hierarchical Bayesian methodology can significantly improve its performance. The concept of joint-sparsity-promoting priors is flexible and can be extended beyond the conditionally Gaussian priors and (generalized) gamma hyper-priors used in the present study.

Future work will explore the potential of the proposed joint-sparsity-promoting approach by extending it to other hierarchical prior models, such as horseshoe [15, 45] and neural network priors [34, 2, 33]. The generalization to non-linear data models, hybrid-like MAP estimation [8, 40], and other inference strategies will also be addressed. The open problem of automatic selection of the hyper-prior parameters (also noted in [29, 51]) will be tackled in forthcoming works. The promising results in our parallel MRI example suggest the proposed MMV approach can also be helpful in applications. Future work



(a) Reference image (b) IAS reconstruction (c) GSBL reconstruction

(d) Least squares reconstruction (e) MMV-IAS reconstruction (f) MMV-GSBL reconstruction

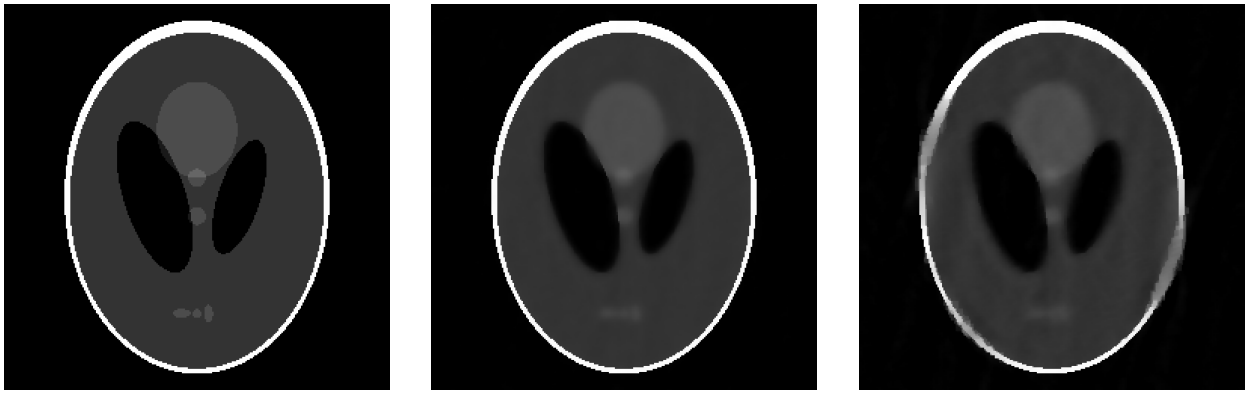
Figure 10: Reference image and the reconstructed first coil images

will consider applications such as synthetic aperture radar (SAR) and electron tomography imaging. In both cases, previous investigations have demonstrated that compressive sensing techniques that exploit joint sparsity of MMVS are effective in recovering point estimates, [37, 38], suggesting that the MMV-IAS or MMV-GSBL might provide an effective Bayesian approach. Employing various sparse transform operators $R_1, \dots, R_L$ may be appropriate in this regard. The case of changing sparsity profiles over time will also be considered, with promising initial results already reported for sequential signaling/imaging [52, 51].

Appendix A. Proof of Theorem 4.1.

In this section, we prove Theorem 4.1. To this end, we first present two auxiliary lemmas.

Lemma A.1. *Let $r \in \mathbb{R} \setminus \{0\}$ and $\beta, \vartheta_k > 0$ for $k = 1, \dots, K$, and denote $\eta = r\beta - (L/2 + 1)$. The*



(a) Reference image (b) IAS reconstruction (c) GSBL reconstruction

(d) Least squares reconstruction (e) MMV-IAS reconstruction (f) MMV-GSBL reconstruction

Figure 11: Reference image and the reconstructed overall images. For all methods, 20 lines/angles (corresponding to around 16% sampling of the k-space), $\sigma^2 = 10^{-3}$, and $L = 4$ coils were used.

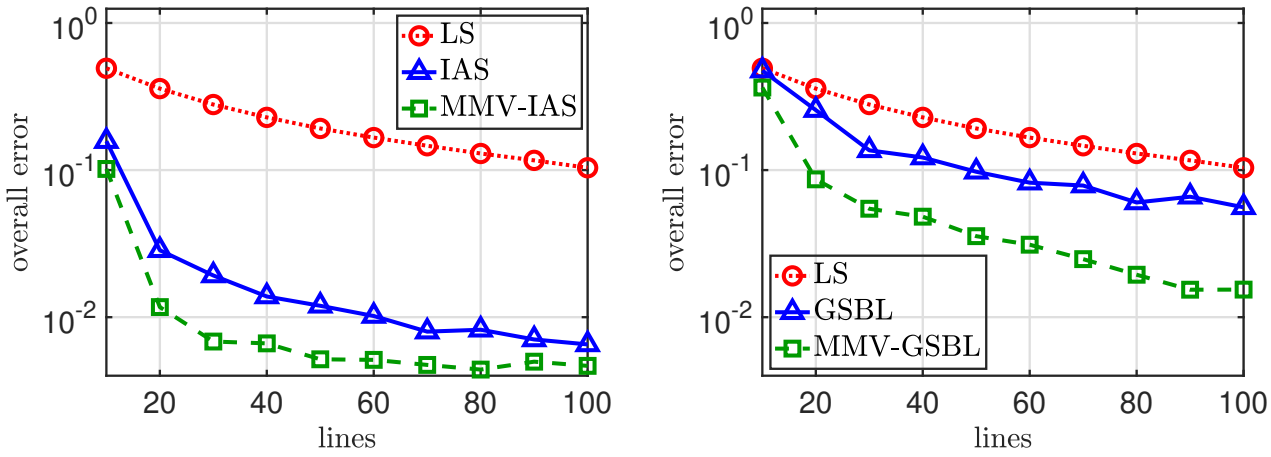
objective function $\mathcal{G}$ *in* (3.3) *has the following second derivatives:*

$$
\begin{aligned}
\nabla_{\mathbf{x}_l} \nabla_{\mathbf{x}_m} \mathcal{G} &= \delta_{l,m} \left(F_l^T F_l + R^T D_{\boldsymbol{\theta}}^{-1} R \right), \\
[\nabla_{\boldsymbol{\theta}} \nabla_{\boldsymbol{\theta}} \mathcal{G}]_{j,k} &= \delta_{j,k} \left(\theta_k^{-3} \left(\sum_{l=1}^{L} [R\mathbf{x}_l]_k^2 \right) + \theta_k^{r-2} \left(\frac{r(r-1)}{\vartheta_k^r} \right) + \theta_k^{-2} \eta \right), \\
\left[\nabla_{\boldsymbol{\theta}} \nabla_{\mathbf{x}_l} \mathcal{G}\right]_{k,n} &= -\theta_k^{-2} [R]_{k,n} [R\mathbf{x}_l]_k,
\end{aligned}
\tag{A.1}
$$

for $j, k = 1, \ldots, K$ *and* $n = 1, \ldots, N$. *Here,* $\delta_{l,m}$ *is the usual Kronecker delta with* $\delta_{l,m} = 1$ *if* $l = m$ *and* $\delta_{l,m} = 0$ *otherwise.*

Proof. Simple calculations show that the first derivatives are

$$
\begin{aligned}
\nabla_{\mathbf{x}_l} \mathcal{G} &= F_l^T \left(F_l \mathbf{x}_l - \mathbf{y}_l \right) + R^T D_{\boldsymbol{\theta}}^{-1} R \mathbf{x}_l, \\
[\nabla_{\boldsymbol{\theta}} \mathcal{G}]_k &= -\theta_k^{-2} \left(\sum_{l=1}^{L} [R\mathbf{x}_l]_k^2 / 2 \right) + \theta_k^{r-1} \left(\frac{r}{\vartheta_k^r} \right) - \theta_k^{-1} \eta,
\end{aligned}
\tag{A.2}
$$



(a) Relative overall error, IAS & MMV-IAS

(b) Relative overall error, GSBL & MMV-GSBL

Figure 12: Relative error of the recovered overall image using the least squares (LS) approach, the existing IAS/GSBL algorithm, and the proposed MMV-IAS/GSBL method. In all cases, we used $L = 4$ coils, noise variance $\sigma^2 = 10^{-3}$, and a varying number of lines.

for $l = 1, \dots, L$ and $k = 1, \dots, K$. Next, we can conclude from (A.2) that

$$\begin{aligned} \nabla_{\mathbf{x}_l} \nabla_{\mathbf{x}_m} \mathcal{G} &= \delta_{l,m} \left(F_l^T F_l + R^T D_{\boldsymbol{\theta}}^{-1} R \right), \\ [\nabla_{\boldsymbol{\theta}} \nabla_{\boldsymbol{\theta}} \mathcal{G}]_{j,k} &= \delta_{j,k} \left(\theta_k^{-3} \left(\sum_{l=1}^{L} [R\mathbf{x}_l]_k^2 \right) + \theta_k^{r-2} \left(\frac{r(r-1)}{\vartheta_k^r} \right) + \theta_k^{-2} \eta \right), \end{aligned} \tag{A.3}$$

for $l, m = 1, \dots, L$ and $j, k = 1, \dots, K$. To determine the mixed derivatives $\nabla_{\boldsymbol{\theta}} \nabla_{\mathbf{x}_l} \mathcal{G}$, note that

$$\left[R^T D_{\boldsymbol{\theta}}^{-1} R\mathbf{x}_l \right]_n = \sum_{j=1}^{K} \left[R^T \right]_{n,j} \left[D_{\boldsymbol{\theta}}^{-1} R\mathbf{x}_l \right]_j = \sum_{j=1}^{K} [R]_{j,n} \, \theta_k^{-1} \, [R\mathbf{x}_l]_j \tag{A.4}$$

for $n = 1, \dots, N$ and $l = 1, \dots, L$. Hence we obtain

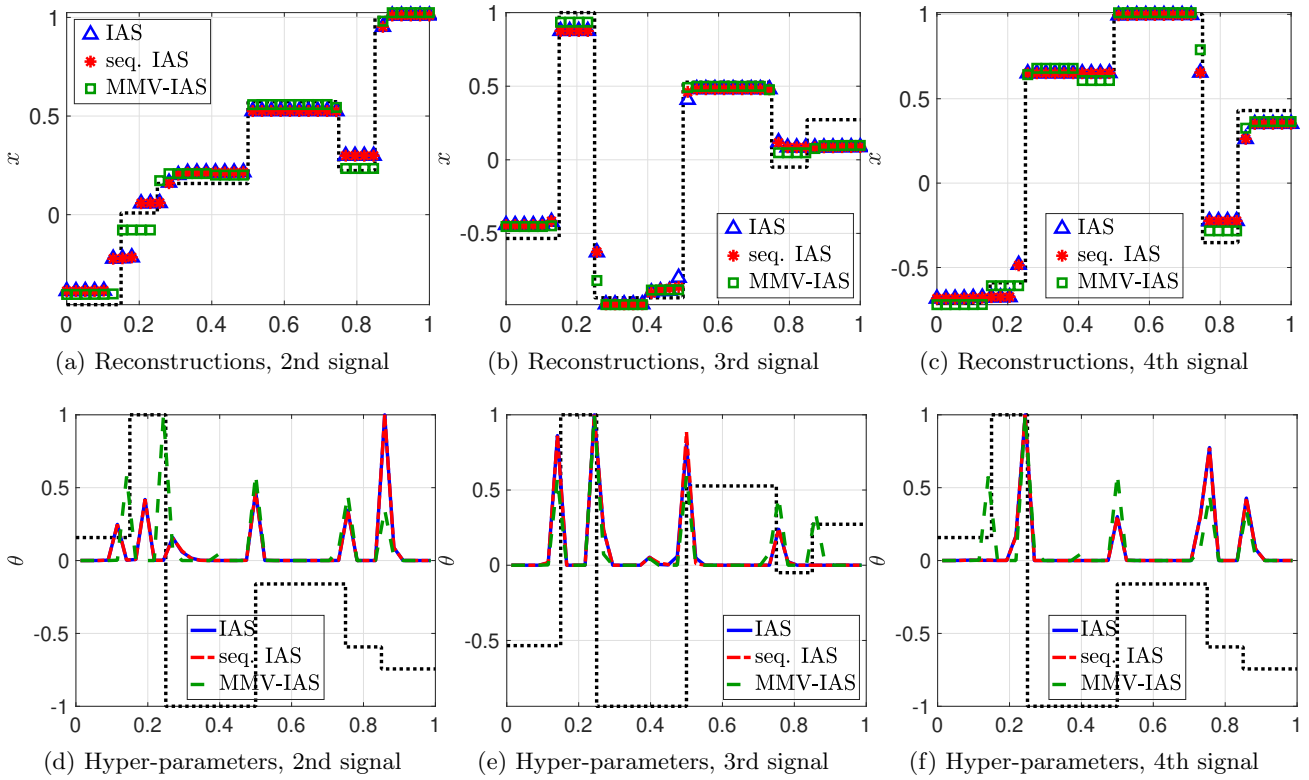
$$\left[\nabla_{\boldsymbol{\theta}} \nabla_{\mathbf{x}_l} \mathcal{G} \right]_{k,n} = \partial_{\theta_k} \left[R^T D_{\boldsymbol{\theta}}^{-1} R\mathbf{x}_l \right]_n = -\theta_k^{-2} [R]_{k,n} [R\mathbf{x}_l]_k \tag{A.5}$$

for $k = 1, \dots, K$ and $n = 1, \dots, N$. ∎

The next lemma provides a lower bound in terms of the Hessian of the objective function G, allowing us to investigate its convexity.

Lemma A.2. *Let $r \in \mathbb{R} \setminus \{0\}$ and $\beta, \vartheta_k > 0$ for $k = 1, \dots, K$. Moreover, let*

$$H = H(\mathbf{x}_{1:L}, \boldsymbol{\theta}) = \begin{bmatrix} \nabla_{\mathbf{x}_{1:L}} \nabla_{\mathbf{x}_{1:L}} \mathcal{G} & \nabla_{\mathbf{x}_{1:L}} \nabla_{\boldsymbol{\theta}} \mathcal{G} \\ \nabla_{\boldsymbol{\theta}} \nabla_{\mathbf{x}_{1:L}} \mathcal{G} & \nabla_{\boldsymbol{\theta}} \nabla_{\boldsymbol{\theta}} \mathcal{G} \end{bmatrix} \tag{A.6}$$



(a) Reconstructions, 2nd signal (b) Reconstructions, 3rd signal (c) Reconstructions, 4th signal

(d) Hyper-parameters, 2nd signal (e) Hyper-parameters, 3rd signal (f) Hyper-parameters, 4th signal

Figure 13: Reconstructions (top row) and normalized hyper-parameter estimates θ (bottom row) for the last three of four piecewise constant signals with a common edge profile. We compare the IAS algorithm (blue triangles), the sequential IAS algorithm (red stars), and the MMV-IAS algorithm (green squares). All methods use a generalized gamma hyper-prior with $r = 1$, ensuring globally convex objective functions.

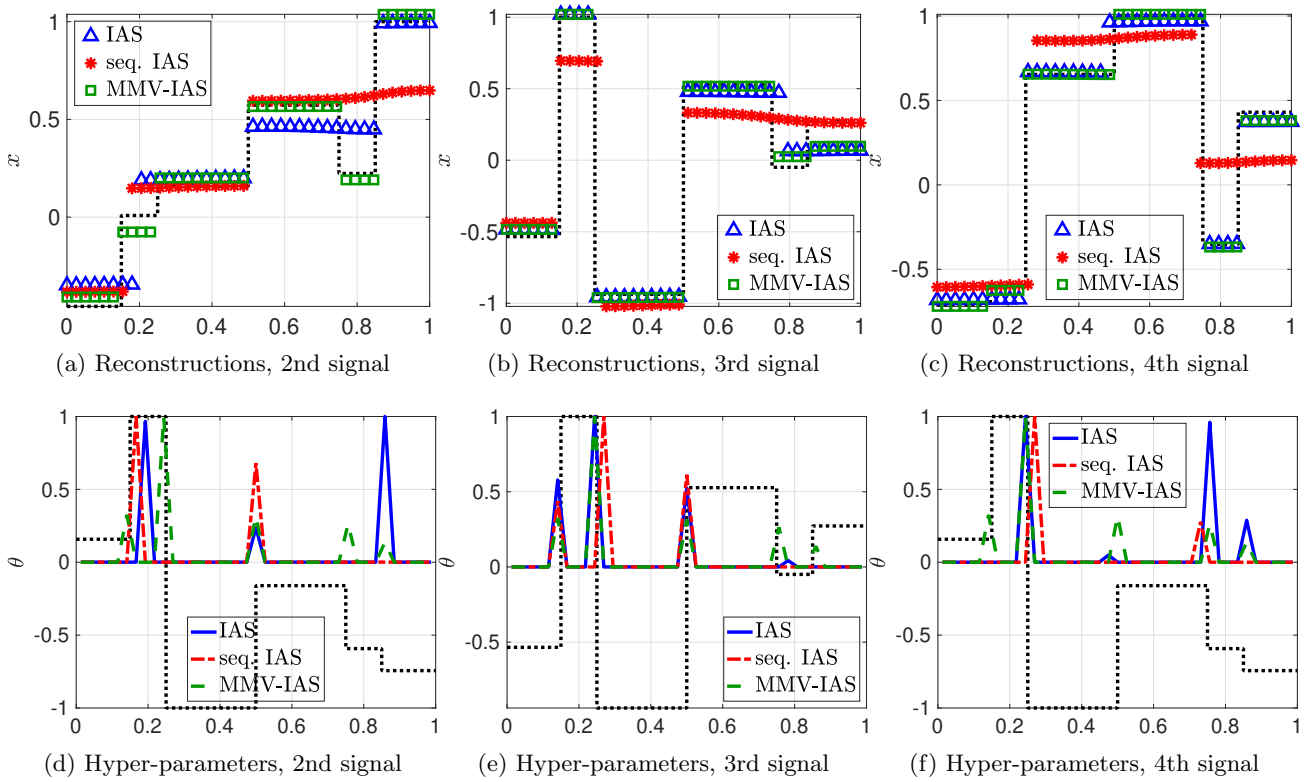
be the Hessian of the objective function $\mathcal{G}$ *in* (3.3) *and let* $\mathbf{u} = [\mathbf{v}_{1:L}; \mathbf{w}]$ *with* $\mathbf{v}_l \in \mathbb{R}^N$, $l = 1, \ldots, L$, *and* $\mathbf{w} \in \mathbb{R}^K$. *Then,*

$$\mathbf{u}^T H \mathbf{u} \geq \sum_{k=1}^{K} \theta_k^{-2} w_k^2 \left(\theta_k^r \left(\frac{r(r-1)}{\vartheta_k^r} \right) + \eta \right), \tag{A.7}$$

where $\eta = r\beta - (L/2 + 1)$.

Proof. We start by noting that

$$\mathbf{u}^T H \mathbf{u} = \sum_{l=1}^{L} \mathbf{v}_l^T \left(\nabla_{\mathbf{x}_l} \nabla_{\mathbf{x}_l} \mathcal{G} \right) \mathbf{v}_l + 2 \sum_{l=1}^{L} \mathbf{w}^T \left(\nabla_{\boldsymbol{\theta}} \nabla_{\mathbf{x}_l} \mathcal{G} \right) \mathbf{v}_l + \mathbf{w}^T \left(\nabla_{\boldsymbol{\theta}} \nabla_{\boldsymbol{\theta}} \mathcal{G} \right) \mathbf{w}. \tag{A.8}$$



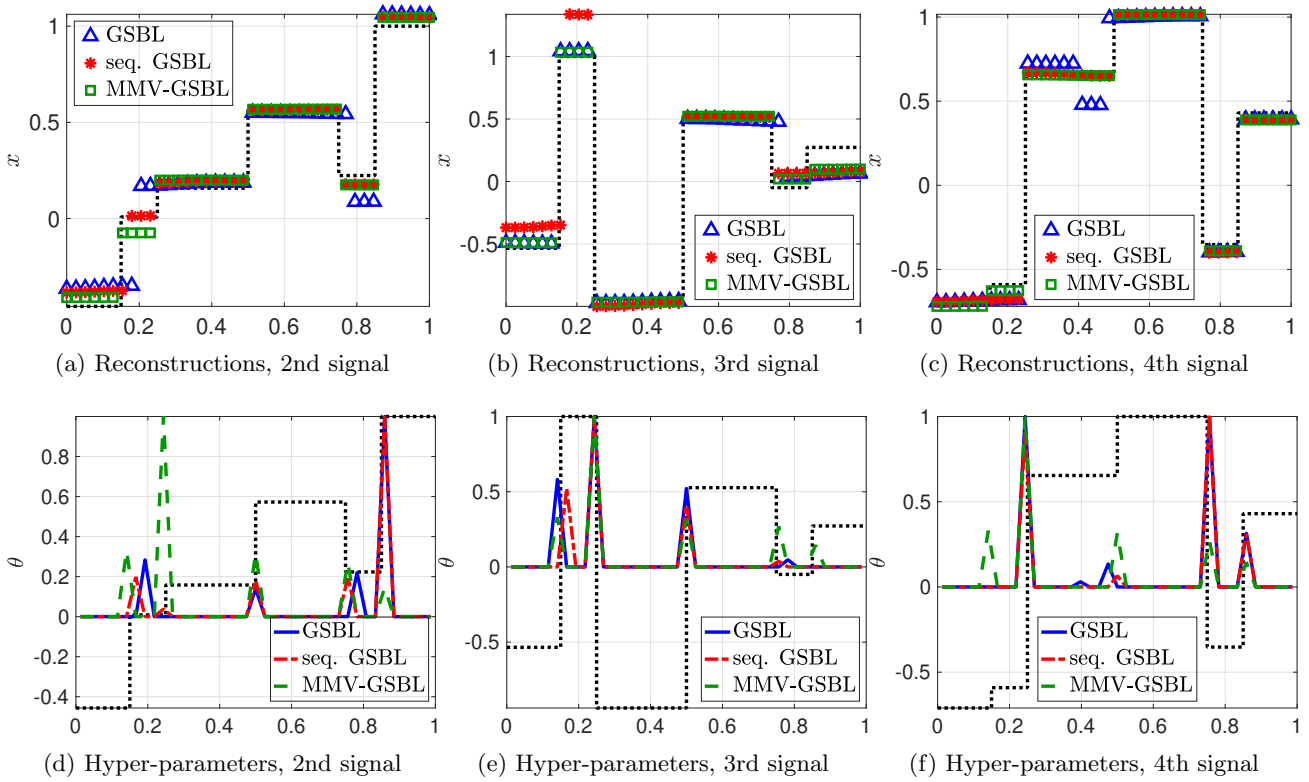
(a) Reconstructions, 2nd signal (b) Reconstructions, 3rd signal (c) Reconstructions, 4th signal

(d) Hyper-parameters, 2nd signal (e) Hyper-parameters, 3rd signal (f) Hyper-parameters, 4th signal

Figure 14: Reconstructions (top row) and normalized hyper-parameter estimates θ (bottom row) for the last three of four piecewise constant signals with a common edge profile. We compare the IAS algorithm (blue triangles), the sequential IAS algorithm (red stars), and the MMV-IAS algorithm (green squares). All methods use a generalized gamma hyper-prior with $r = -1$, leading to non-convex objective functions.

Substituting the second derivatives (A.1) from Lemma A.1 yields

$$
\begin{aligned}
\mathbf{v}_l^T \left(\nabla_{\mathbf{x}_l} \nabla_{\mathbf{x}_l} \mathcal{G}\right) \mathbf{v}_l &= \sum_{m=1}^{M} \left[F_l \mathbf{v}_l\right]_m^2 + \sum_{k=1}^{K} \theta_k^{-1} \left[R \mathbf{v}_l\right]_k^2 , \\
\mathbf{w}^T \left(\nabla_{\boldsymbol{\theta}} \nabla_{\mathbf{x}_l} \mathcal{G}\right) \mathbf{v}_l &= -\sum_{k=1}^{K} \theta_k^{-2} w_k \left[R \mathbf{x}_l\right]_k \left[R \mathbf{v}_l\right]_k , \\
\mathbf{w}^T \left(\nabla_{\boldsymbol{\theta}} \nabla_{\boldsymbol{\theta}} \mathcal{G}\right) \mathbf{w} &= \sum_{k=1}^{K} w_k^2 \left(\theta_k^{-3} \left(\sum_{l=1}^{L} [R \mathbf{x}_l]_k^2 \right) + \theta_k^{r-2} \left(\frac{r(r-1)}{\vartheta_k^r} \right) + \theta_k^{-2} \eta \right) .
\end{aligned}
\tag{A.9}
$$



(a) Reconstructions, 2nd signal (b) Reconstructions, 3rd signal (c) Reconstructions, 4th signal

(d) Hyper-parameters, 2nd signal (e) Hyper-parameters, 3rd signal (f) Hyper-parameters, 4th signal

Figure 15: Reconstructions (top row) and normalized hyper-parameter estimates θ (bottom row) for the last three of four piecewise constant signals with a common edge profile. We compare the GSBL algorithm (blue triangles), the sequential GSBL algorithm (red stars), and the MMV-GSBL algorithm (green squares).

Furthermore, substituting (A.9) into (A.8) we obtain

$$\begin{aligned}\mathbf{u}^T H\mathbf{u} = &\sum_{l=1}^{L}\sum_{m=1}^{M}\left[F_l\mathbf{v}_l\right]_m^2 \\ &+\sum_{l=1}^{L}\sum_{k=1}^{K}\left(\theta_k^{-1}\left[R\mathbf{v}_l\right]_k^2 - 2\theta_k^{-2}\left[R\mathbf{v}_l\right]_k w_k\left[R\mathbf{x}_l\right]_k + \theta_k^{-2}w_k^2\left[R\mathbf{x}_l\right]_k^2\right) \\ &+\sum_{k=1}^{K}\theta_k^{-2}w_k^2\left(\theta_k^r\left(\frac{r(r-1)}{\vartheta_k^r}\right)+\eta\right).\end{aligned} \tag{A.10}$$

Note that $\theta_k^{-2}\left[R\mathbf{v}_l\right]_k w_k\left[R\mathbf{x}_l\right]_k + \theta_k^{-2}w_k^2\left[R\mathbf{x}_l\right]_k^2 = \theta_k^{-3}\left(\theta_k\left[R\mathbf{v}_l\right]_k - w_k\left[R\mathbf{x}_l\right]_k\right)^2$. Hence, we can rewrite (A.10) as

$$\begin{aligned}\mathbf{u}^T H\mathbf{u} = &\sum_{l=1}^{L}\sum_{m=1}^{M}\left[F_l\mathbf{v}_l\right]_m^2 + \sum_{l=1}^{L}\sum_{k=1}^{K}\theta_k^{-3}\left(\theta_k\left[R\mathbf{v}_l\right]_k - w_k\left[R\mathbf{x}_l\right]_k\right)^2 \\ &+\sum_{k=1}^{K}\theta_k^{-2}w_k^2\left(\theta_k^r\left(\frac{r(r-1)}{\vartheta_k^r}\right)+\eta\right).\end{aligned} \tag{A.11}$$

Finally, note that the first two sums on the right-hand side of (A.11) are non-negative, which yields the assertion. ∎

We are now positioned to prove Theorem 4.1.

Proof of Theorem 4.1. Recall that $\mathcal{G}$ is convex if and only if its Hessian H satisfies $\mathbf{u}^T H\mathbf{u} \geq 0$ for all $\mathbf{u} = [\mathbf{v}_{1:L}; \mathbf{w}]$. Let $\mathbf{u} = [\mathbf{v}_{1:L}; \mathbf{w}]$, then Lemma A.2 implies

$$\mathbf{u}^T H\mathbf{u} \geq \sum_{k=1}^{K} \theta_k^{-2} w_k^2 \left(\theta_k^r \left(\frac{r(r-1)}{\vartheta_k^r} \right) + \eta \right). \tag{A.12}$$

The right-hand side of (A.12) is positive if

$$\theta_k^r \left(\frac{r(r-1)}{\vartheta_k^r} \right) > -\eta, \quad k = 1, \dots, K. \tag{A.13}$$

The proof for the different cases follows by enforcing condition (A.13). ∎

Acknowledgements. This work was partially supported by AFOSR #F9550-22-1-0411, DOD (ONR MURI) #N00014-20-1-2595, DOE ASCR #DE-ACO5-000R22725, and NSF DMS #1912685.

REFERENCES

[1] B. Adcock, A. Gelb, G. Song, and Y. Sui, *Joint sparse recovery based on variances*, SIAM Journal on Scientific Computing, 41 (2019), pp. A246–A268.
[2] M. Asim, M. Daniels, O. Leong, A. Ahmed, and P. Hand, *Invertible generative models for inverse problems: mitigating representation error and dataset bias*, in International Conference on Machine Learning, PMLR, 2020, pp. 399–409.
[3] S. D. Babacan, R. Molina, and A. K. Katsaggelos, *Sparse Bayesian image restoration*, in 2010 IEEE International Conference on Image Processing, IEEE, 2010, pp. 3577–3580.
[4] A. Beck, *First-Order Methods in Optimization*, SIAM, 2017.
[5] A. Beck and M. Teboulle, *A fast iterative shrinkage-thresholding algorithm for linear inverse problems*, SIAM Journal on Imaging Sciences, 2 (2009), pp. 183–202.
[6] D. Calvetti, A. Pascarella, F. Pitolli, E. Somersalo, and B. Vantaggi, *A hierarchical Krylov–Bayes iterative inverse solver for MEG with physiological preconditioning*, Inverse Problems, 31 (2015), p. 125005.
[7] D. Calvetti, F. Pitolli, E. Somersalo, and B. Vantaggi, *Bayes meets Krylov: Statistically inspired preconditioners for CGLS*, SIAM Review, 60 (2018), pp. 429–461.
[8] D. Calvetti, M. Pragliola, and E. Somersalo, *Sparsity promoting hybrid solvers for hierarchical Bayesian inverse problems*, SIAM Journal on Scientific Computing, 42 (2020), pp. A3761–A3784.
[9] D. Calvetti, M. Pragliola, E. Somersalo, and A. Strang, *Sparse reconstructions from few noisy data: analysis of hierarchical Bayesian models with generalized gamma hyperpriors*, Inverse Problems, 36 (2020), p. 025010.
[10] D. Calvetti and E. Somersalo, *A Gaussian hypermodel to recover blocky objects*, Inverse Problems, 23 (2007), p. 733.
[11] D. Calvetti and E. Somersalo, *An Introduction to Bayesian Scientific Computing: Ten Lectures on Subjective Computing*, vol. 2, Springer Science & Business Media, 2007.
[12] D. Calvetti and E. Somersalo, *Computationally efficient sampling methods for sparsity promoting hierarchical Bayesian models*, arXiv preprint arXiv:2303.16988, (2023).
[13] D. Calvetti, E. Somersalo, and A. Strang, *Hierachical Bayesian models and sparsity: ℓ_2-magic*, Inverse Problems, 35 (2019), p. 035003.
[14] E. J. Candes, M. B. Wakin, and S. P. Boyd, *Enhancing sparsity by reweighted ℓ_1 minimization*, Journal of Fourier Analysis and Applications, 14 (2008), pp. 877–905.
[15] C. M. Carvalho, N. G. Polson, and J. G. Scott, *Handling sparsity via the horseshoe*, in Artificial Intelligence and Statistics, PMLR, 2009, pp. 73–80.
[16] G. K. Chantas, N. P. Galatsanos, and A. C. Likas, *Bayesian restoration using a new nonstationary edge-preserving image prior*, IEEE Transactions on Image Processing, 15 (2006), pp. 2987–2997.

[17] R. Chartrand and W. Yin, *Iteratively reweighted algorithms for compressive sensing*, in 2008 IEEE International Conference on Acoustics, Speech and Signal Processing, IEEE, 2008, pp. 3869–3872.
[18] I. Y. Chun and B. Adcock, *Compressed sensing and parallel acquisition*, IEEE Transactions on Information Theory, 63 (2017), pp. 4860–4882.
[19] I. Y. Chun, B. Adcock, and T. M. Talavage, *Efficient compressed sensing SENSE pMRI reconstruction with joint sparsity promotion*, IEEE Transactions on Medical Imaging, 35 (2015), pp. 354–368.
[20] V. Churchill and A. Gelb, *Detecting edges from non-uniform Fourier data via sparse Bayesian learning*, Journal of Scientific Computing, 80 (2019), pp. 762–783.
[21] S. F. Cotter, B. D. Rao, K. Engan, and K. Kreutz-Delgado, *Sparse solutions to linear inverse problems with multiple measurement vectors*, IEEE Transactions on Signal Processing, 53 (2005), pp. 2477–2488.
[22] I. Daubechies, R. DeVore, M. Fornasier, and C. S. Güntürk, *Iteratively reweighted least squares minimization for sparse recovery*, Communications on Pure and Applied Mathematics: A Journal Issued by the Courant Institute of Mathematical Sciences, 63 (2010), pp. 1–38.
[23] D. L. Donoho, *Compressed sensing*, IEEE Transactions on Information Theory, 52 (2006), pp. 1289–1306.
[24] Y. C. Eldar and G. Kutyniok, *Compressed Sensing: Theory and Applications*, Cambridge University Press, 2012.
[25] Y. C. Eldar and M. Mishali, *Robust recovery of signals from a structured union of subspaces*, IEEE Transactions on Information Theory, 55 (2009), pp. 5302–5316.
[26] G. Evensen, *Data Assimilation: The Ensemble Kalman Filter*, vol. 2, Springer, 2009.
[27] S. Foucart and H. Rauhut, *A mathematical introduction to compressive sensing*, Bull. Am. Math, 54 (2017), pp. 151–165.
[28] A. Gelb and T. Scarnati, *Reducing effects of bad data using variance based joint sparsity recovery*, Journal of Scientific Computing, 78 (2019), pp. 94–120.
[29] J. Glaubitz, A. Gelb, and G. Song, *Generalized sparse Bayesian learning and application to image reconstruction*, SIAM/ASA Journal on Uncertainty Quantification, 11 (2023), pp. 262–284.
[30] M. Guerquin-Kern, L. Lejeune, K. P. Pruessmann, and M. Unser, *Realistic analytical phantoms for parallel magnetic resonance imaging*, IEEE Transactions on Medical Imaging, 31 (2011), pp. 626–636.
[31] J. Kaipio and E. Somersalo, *Statistical and Computational Inverse Problems*, vol. 160, Springer Science & Business Media, 2006.
[32] H. Kim, D. Sanz-Alonso, and A. Strang, *Hierarchical ensemble Kalman methods with sparsity-promoting generalized gamma hyperpriors*, Foundations of Data Science, 5 (2023), pp. 366–388.
[33] C. Li, M. Dunlop, and G. Stadler, *Bayesian neural network priors for edge-preserving inversion*, Inverse Problems and Imaging, 16 (2022), pp. 1229–1254.
[34] R. M. Neal, *Priors for infinite networks*, in Bayesian Learning for Neural Networks, Springer, 1996, pp. 29–53.
[35] A. B. Owen, *Monte Carlo Theory, Methods and Examples*, Stanford, 2013.
[36] Y. Saad, *Iterative Methods for Sparse Linear Systems*, SIAM, 2003.
[37] T. Sanders, A. Gelb, and R. B. Platte, *Composite SAR imaging using sequential joint sparsity*, Journal of Computational Physics, 338 (2017), pp. 357–370.
[38] T. Scarnati and A. Gelb, *Joint image formation and two-dimensional autofocusing for synthetic aperture radar data*, Journal of Computational Physics, 374 (2018), pp. 803–821.
[39] T. Scarnati and A. Gelb, *Accurate and efficient image reconstruction from multiple measurements of Fourier samples*, Journal of Computational Mathematics, 38 (2020), p. 797.
[40] Z. Si, Y. Liu, and A. Strang, *Path-following methods for Maximum a Posteriori estimators in Bayesian hierarchical models: How estimates depend on hyperparameters*, arXiv preprint arXiv:2211.07113, (2022).
[41] A. Spantini, R. Baptista, and Y. Marzouk, *Coupling techniques for nonlinear ensemble filtering*, SIAM Review, 64 (2022), pp. 921–953.
[42] A. M. Stuart, *Inverse problems: a Bayesian perspective*, Acta Numerica, 19 (2010), pp. 451–559.
[43] A. N. Tikhonov, A. Goncharsky, V. Stepanov, and A. G. Yagola, *Numerical Methods for the Solution of Ill-Posed Problems*, vol. 328 of Mathematics and Its Applications, Springer Science & Business Media, 2013.
[44] M. E. Tipping, *Sparse Bayesian learning and the relevance vector machine*, Journal of Machine Learning Research, 1 (2001), pp. 211–244.
[45] F. Uribe, Y. Dong, and P. C. Hansen, *Horseshoe priors for edge-preserving linear Bayesian inversion*, SIAM Journal on Scientific Computing, 45 (2023), pp. B337–B365.
[46] M. Vono, N. Dobigeon, and P. Chainais, *High-dimensional Gaussian sampling: a review and a unifying approach based on a stochastic proximal point algorithm*, SIAM Review, 64 (2022), pp. 3–56.
[47] D. P. Wipf and B. D. Rao, *Sparse Bayesian learning for basis selection*, IEEE Transactions on Signal processing, 52 (2004), pp. 2153–2164.
[48] D. P. Wipf and B. D. Rao, *An empirical Bayesian strategy for solving the simultaneous sparse approximation*

problem, IEEE Transactions on Signal Processing, 55 (2007), pp. 3704–3716.
[49] S. J. Wright, *Coordinate descent algorithms*, Mathematical Programming, 151 (2015), pp. 3–34.
[50] Y. Xiao, A. Gelb, and G. Song, *Sequential edge detection using joint hierarchical Bayesian learning*, arXiv preprint arXiv:2302.14247, (2023).
[51] Y. Xiao and J. Glaubitz, *Sequential image recovery using joint hierarchical Bayesian learning*, Journal of Scientific Computing, 96 (2023), p. 4.
[52] Y. Xiao, J. Glaubitz, A. Gelb, and G. Song, *Sequential image recovery from noisy and under-sampled Fourier data*, Journal of Scientific Computing, 91 (2022), p. 79.
[53] J. Zhang, A. Gelb, and T. Scarnati, *Empirical Bayesian inference using a support informed prior*, SIAM/ASA Journal on Uncertainty Quantification, 10 (2022), pp. 745–774.