SIAM/ASA J. Uncertainty Quantification
Vol. 11, No. 1, pp. 262–284

Generalized Sparse Bayesian Learning and Application to Image Reconstruction*

Jan Glaubitz†, Anne Gelb†, and Guohui Song‡

Abstract. Image reconstruction based on indirect, noisy, or incomplete data remains an important yet challenging task. While methods such as compressive sensing have demonstrated high-resolution image recovery in various settings, there remain issues of robustness due to parameter tuning. Moreover, since the recovery is limited to a point estimate, it is impossible to quantify the uncertainty, which is often desirable. Due to these inherent limitations, a sparse Bayesian learning approach is sometimes adopted to recover a posterior distribution of the unknown. Sparse Bayesian learning assumes that some linear transformation of the unknown is sparse. However, most of the methods developed are tailored to specific problems, with particular forward models and priors. Here, we present a generalized approach to sparse Bayesian learning. It has the advantage that it can be used for various types of data acquisitions and prior information. Some preliminary results on image reconstruction/recovery indicate its potential use for denoising, deblurring, and magnetic resonance imaging.

Key words. image reconstruction, sparse Bayesian learning, regularized inverse problems, Bayesian inference

MSC codes. 15A29, 62F15, 65F22, 94A08, 92C55

DOI. 10.1137/22M147236X

1. Introduction.

Many applications seek to solve the *linear inverse problem*

$$\mathbf{y} = F\mathbf{x} + \boldsymbol{\nu}, \tag{1.1}$$

where $\mathbf{y} \in \mathbb{R}^m$ is a vector of indirect measurements, $\mathbf{x} \in \mathbb{R}^n$ is the vector of unknowns, $F \in \mathbb{R}^{m\times n}$ is a known linear forward operator, and $\boldsymbol{\nu} \in \mathbb{R}^m$ corresponds to a typically unknown noise vector (see [35, 55, 39] and references therein). In particular, (1.1) can be associated with signal or image reconstruction [48, 40, 51]. In this regard it is often reasonable to assume that some linear transformation of the unknown solution $\mathbf{x}$, say $R\mathbf{x}$, is sparse. A common approach is to consider the ℓ^1*-regularized inverse problem*

$$\mathbf{x}_\lambda = \arg\min_{\mathbf{x}} \left\{ \|F\mathbf{x} - \mathbf{y}\|_2^2 + \lambda \|R\mathbf{x}\|_1 \right\}, \tag{1.2}$$

* Received by the editors January 18, 2022; accepted for publication (in revised form) August 1, 2022; published electronically March 3, 2023.
https://doi.org/10.1137/22M147236X
Funding: The work of the first and second authors was partially supported by AFOSR grant F9550-18-1-0316. The work of the second author was partially supported by NSF grants DMS-1502640 and DMS-1912685 and ONR grant N00014-20-1-2595. The work of the third author was partially supported by NSF grants DMS-1521661 and DMS-1939203.

† Department of Mathematics, Dartmouth College, Hanover, NH 03755 USA (Jan.Glaubitz@Dartmouth.edu, Anne.E.Gelb@Dartmouth.edu).
‡ Department of Mathematics and Statistics, Old Dominion University, Norfolk, VA 23529 USA (gsong@odu.edu).

where $R \in \mathbb{R}^{k\times n}$ is referred to as the *regularization operator* and $\lambda > 0$ as the *regularization parameter*. The motivation for this approach is that the ℓ^1-norm, $\|\cdot\|_1$, serves as a convex surrogate for the ℓ^0-"norm", $\|\cdot\|_0$. Thus, (1.2) balances data fidelity, noise, and the sparsity assumption on $R\mathbf{x}$, while still enabling efficient computations [26, 27, 31]. However, an often encountered difficulty for the ℓ^1-regularized inverse problem (1.2) is the selection of an appropriate regularization parameter λ. This parameter can critically influence the quality of the regularized reconstruction $\mathbf{x}_\lambda$ [34, 24, 38, 50, 44]. Partly due to this reason, many statistical approaches have been proposed for regularized inverse problems [41, 13, 51]. Another advantage in using statistical approaches is that they may allow for uncertainty quantification in the reconstructed solution [14]. For example, the hierarchical Bayesian formulation of the ℓ^1-regularized inverse problem (1.2) (see [41, 13]) is based on extending $\mathbf{x}$, $\mathbf{y}$, and all other involved parameters, which we collectively write as $\boldsymbol{\theta}$, into random variables. Consequently $\mathbf{x}$, $\mathbf{y}$, and $\boldsymbol{\theta}$ are characterized by certain density functions, as are their relationships to each other. In particular, one usually considers the following density functions:

- The *likelihood* $p(\mathbf{y}|\mathbf{x},\boldsymbol{\theta})$, which is the probability density function for $\mathbf{y}$ given $\mathbf{x}$ and $\boldsymbol{\theta}$.
- The *prior* $p(\mathbf{x}|\boldsymbol{\theta})$, which is the density function for $\mathbf{x}$ given $\boldsymbol{\theta}$.
- The *hyper-prior* $p(\boldsymbol{\theta})$, which is the probability density function for the parameters $\boldsymbol{\theta}$.
- The *posterior* $p(\mathbf{x},\boldsymbol{\theta}|\mathbf{y})$, which is the probability density function for the solution $\mathbf{x}$ and the parameters $\boldsymbol{\theta}$ given the data $\mathbf{y}$.

One can use Bayes' theorem to obtain

$$p(\mathbf{x},\boldsymbol{\theta}|\mathbf{y}) \propto p(\mathbf{y}|\mathbf{x},\boldsymbol{\theta})p(\mathbf{x}|\boldsymbol{\theta})p(\boldsymbol{\theta}), \tag{1.3}$$

where "$\propto$" means that the two sides of (1.3) are equal to each other up to a multiplicative constant that does not depend on $\mathbf{x}$ or $\boldsymbol{\theta}$. Note that the parameters $\boldsymbol{\theta}$ are now part of the problem and are no longer selected a priori. Furthermore, using an appropriate method for Bayesian inference allows one to quantify uncertainty in the reconstructed solution $\mathbf{x}$.

There are a variety of sparsity-promoting priors to choose from, including but not limited to Laplace priors [29], total variation (TV)-priors [42, 4], mixture-of-Gaussian priors [28], and hyper-Laplacian distributions based on ℓ^p-quasinorms with $0 < p < 1$ [45, 43]. In this investigation we consider the well-known class of conditionally Gaussian priors given by

$$p(\mathbf{x}|\boldsymbol{\beta}) \propto \det(B)^{1/2} \exp\left\{-\frac{1}{2}\mathbf{x}^T R^T B R\mathbf{x}\right\}, \tag{1.4}$$

where $B = \operatorname{diag}(\beta_1,\ldots,\beta_k)$ is a diagonal inverse covariance matrix. Ideas discussed in [54, 57, 19, 12, 18, 7, 5, 16, 22, 23] suggest that conditionally Gaussian priors of the form (1.4) are particularly suited to promote sparsity of $R\mathbf{x}$. For example, the model proposed in [54] is designed to recover sparse representations of kernel approximations, coining the term *sparse Bayesian learning (SBL)*. Promoting sparse solutions, as done in [54], corresponds to using $R = I \in \mathbb{R}^{n\times n}$ as a regularization operator in (1.4). Further investigations that made use of SBL to promote sparse solutions include [57, 61, 59, 16, 22]. In many applications, however, it is some linear transformation $R\mathbf{x}$ that is desired to be sparse. For example, TV-regularization is of particular interest in image recovery. Extensions of SBL for this setting

have been proposed in [19, 12, 14, 18, 7, 5, 23]. That said, since the TV-regularization operator R is singular, the prior (1.4) is improper. This prohibits the application of many of the existing SBL approaches. An often-encountered idea therefore is to make $R \in \mathbb{R}^{k\times n}$ with $k < n$ invertible by introducing additional rows that are consistent with assumptions about the underlying solution. For example, in [12, 14, 7] the additional rows encode certain boundary conditions. The same technique can be extended to higher-order TV-regularization [23]. Unfortunately, such additional information might not always be available or may be complicated to incorporate, especially in two or more dimensions. Further, different types of regularization operators must be adapted on a case-by-case basis, and the resulting prior may promote undesired artificial features in the solution when the regularization operator is not carefully modified. The approach in [19, 18], by contrast, depends on the assumption of a "commuting property" of the form $FR = RF$. Requiring such a commuting property is often unrealistic in applications, however.[1]

Our contribution. To address these issues, we present a generalized approach to SBL for "almost" general forward and regularization operators, F and R. By "almost" general, we mean that the only restriction on F and R is that their common kernel should be trivial, $\text{kernel}(F) \cap \text{kernel}(R) = \{\mathbf{0}\}$, a standard assumption in regularized inverse problems [41]. We propose an efficient numerical method for Bayesian inference that yields a full conditional posterior density $p(\mathbf{x}|\mathbf{y})$, rather than a simple point estimate, which allows for uncertainty quantification in the solution $\mathbf{x}$. The present work implies that SBL can be applied to a broader class of problems than currently known. In particular, some preliminary results on signal and image reconstruction indicate its potential use for denoising, deblurring, and magnetic resonance imaging.

Outline. The rest of this paper is organized as follows. Section 2 provides some details on the sparsity promoting hierarchical Bayesian model under consideration. In section 3, we propose an efficient numerical method for Bayesian inference. A series of numerical examples is presented in section 4 to illustrate the descriptive span of the hierarchical Bayesian model. Finally, in section 5, we provide some concluding thoughts.

2. The hierarchical Bayesian model.

We begin by reviewing the generalized hierarchical Bayesian model considered here, which is illustrated in Figure 1.

2.1. The likelihood.

The likelihood function $p(\mathbf{y}|\mathbf{x}, \boldsymbol{\alpha})$ models the connection between the solution $\mathbf{x}$, the noise parameters $\boldsymbol{\alpha}$, and the indirect measurements $\mathbf{y}$. It is often assumed that $\boldsymbol{\nu} \in \mathbb{R}^m$ in (1.1) is zero-mean i.i.d. normal noise with inverse variance $\alpha > 0$, that is, $\boldsymbol{\nu} \sim \mathcal{N}(\mathbf{0}, \alpha^{-1} I)$. This assumption yields the conditionally Gaussian likelihood function

$$p(\mathbf{y}|\mathbf{x}, \alpha) = (2\pi)^{-m/2} \alpha^{m/2} \exp\left\{-\frac{\alpha}{2}\|F\mathbf{x} - \mathbf{y}\|_2^2\right\}. \tag{2.1}$$

The likelihood function given by (2.1) was considered, for instance, in [54, 19, 5, 6, 23]. By contrast, we restrict $\boldsymbol{\nu}$ to be independent but *not* necessarily identically distributed. This translates into $\boldsymbol{\nu} \sim \mathcal{N}(\mathbf{0}, A^{-1})$ with diagonal positive definite *inverse noise covariance matrix*

$$A = \text{diag}(\boldsymbol{\alpha}), \quad \boldsymbol{\alpha} = [\alpha_1, \ldots, \alpha_m]. \tag{2.2}$$

[1] The dimensions of F and R are typically not consistent.

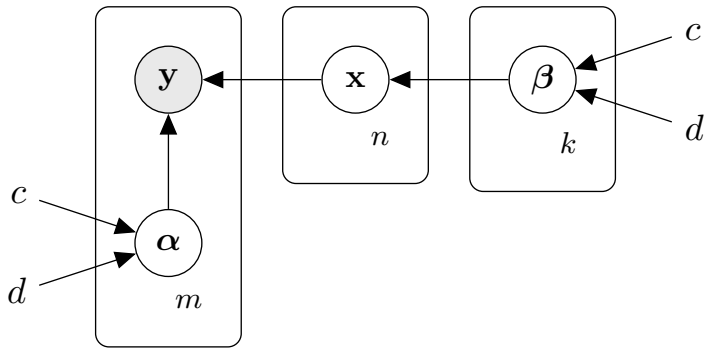


Figure 1. *Graphical representation of the hierarchical Bayesian model. Nodes denoted within circles correspond to random variables, while nodes without a circle correspond to parameters. Shaded circles represent observed random variables, while plain circles represent hidden random variables.*

The linear data model (1.1) then yields the generalized likelihood function

$$p(\mathbf{y}|\mathbf{x},\boldsymbol{\alpha}) = (2\pi)^{-m/2}\det(A)^{1/2}\exp\left\{-\frac{1}{2}(F\mathbf{x}-\mathbf{y})^T A(F\mathbf{x}-\mathbf{y})\right\}, \tag{2.3}$$

which reduces to (2.1) if the inverse variances $\alpha_1,\dots,\alpha_m$ are all equal to α. We note that conditionally Gaussian likelihoods of the form (2.3) were considered in [15, Example 3.4] in combination with smoothness promoting priors to address data outliers. Example 2.1 below motivates the weaker assumption $\boldsymbol{\nu}\sim\mathcal{N}(\mathbf{0},A^{-1})$ for sparsity promoting priors in the context of data fusion [37] and multisensor acquisition systems [36, 21].

Example 2.1. Assume we have a collection of measurements $\mathbf{y}^{(d)}\in\mathbb{R}^{m_d}$, $d=1,\dots,D$, generated from the same source $\mathbf{x}$ from D different sensors. The corresponding data models are

$$\mathbf{y}^{(d)} = F^{(d)}\mathbf{x}+\boldsymbol{\nu}^{(d)}, \quad d=1,\dots,D. \tag{2.4}$$

Further, assume that the noise in the measurements from the same sensor is i.i.d., that is, $\boldsymbol{\nu}^{(d)}\sim\mathcal{N}(\mathbf{0},\alpha_1^{-1}I)$. However, the noise variance might differ from sensor to sensor, so that $\alpha_1\neq\cdots\neq\alpha_D$. If we combine the different measurements and consider the joint data model

$$\underbrace{\begin{bmatrix}\mathbf{y}^{(1)}\\ \vdots\\ \mathbf{y}^{(D)}\end{bmatrix}}_{=\mathbf{y}} = \underbrace{\begin{bmatrix}F^{(1)}\\ \vdots\\ F^{(D)}\end{bmatrix}}_{=F}\mathbf{x}+\underbrace{\begin{bmatrix}\boldsymbol{\nu}^{(1)}\\ \vdots\\ \boldsymbol{\nu}^{(D)}\end{bmatrix}}_{=\boldsymbol{\nu}}, \tag{2.5}$$

the stacked noise vector $\boldsymbol{\nu}$ cannot be assumed to be i.i.d., which we cannot appropriately model using the likelihood function (2.1). However, using the more general likelihood function (2.3), we can model (2.5) by choosing a diagonal inverse noise covariance matrix A of the form

$$A = \operatorname{diag}(\alpha_1 I_1,\dots,\alpha_D I_D), \tag{2.6}$$

where $I_d \in \mathbb{R}^{m_d \times m_d}$, $d = 1, \dots, D$, denotes the identity matrix with dimensions matching the number of measurements provided by the dth sensor.

Remark 2.2. We note that in [60] it was pointed out that for classical SBL algorithms, even when the exact inverse noise variance α (or A) is known, using this fixed value instead of a variable Gamma hyper-prior can yield suboptimal reconstructions.

2.2. The prior. The prior function $p(\mathbf{x}|\boldsymbol{\beta})$ models our prior belief about the unknown solution $\mathbf{x}$. Assume that some linear transformation of $\mathbf{x}$, say $R\mathbf{x}$, is sparse. The SBL approach promotes this assumed sparsity by using a conditionally Gaussian prior function,

$$p(\mathbf{x}|\boldsymbol{\beta}) = \det(B)^{1/2} \exp\left\{-\frac{1}{2}\mathbf{x}^T R^T B R \mathbf{x}\right\}, \tag{2.7}$$

where $B = \operatorname{diag}(\beta_1, \dots, \beta_k)$ is referred to as the *inverse prior convariance matrix*. See [54, 57, 19, 12, 18, 7, 5, 16, 22, 23] and references therein. The conditionally Gaussian prior (2.7) can be justified by its asymptotic behavior [12]. If we assume that the inverse variances $\beta_1, \dots, \beta_k$ are all equal, then (2.7) favors solutions $\mathbf{x}$ for which $R\mathbf{x}$ is equal to or close to zero,[2] since these solutions have a higher probability. For instance, when $R\mathbf{x}$ corresponds to the total variation of $\mathbf{x}$, $[R\mathbf{x}]_j = x_{j+1} - x_j$, then (2.7) promotes solutions $\mathbf{x}$ that have no or little variation. However, if one of the inverse variances, say β_j, is significantly smaller than the remaining ones, a jump between x_j and x_{j+1} becomes more likely. In this way, (2.7) promotes sparsity of $R\mathbf{x}$.

Remark 2.3 (improper priors). If $\operatorname{kernel} R \neq \{\mathbf{0}\}$, then $R^T B R$ is singular and (2.7) becomes an improper prior. Most existing SBL algorithms are infeasible in this case, thus motivating us to propose an alternative method in section 3. In particular, the resulting difficulties for the evidence approach, which was used in the original investigation [54] and later in [5], are addressed in Appendix A.

2.3. The hyper-prior. From the discussion above it is evident that the inverse variances $\beta_1, \dots, \beta_k$ must be allowed to have distinctly different values for the conditionally Gaussian prior (2.7) to promote sparsity of $R\mathbf{x}$. This can be achieved by treating $\beta_1, \dots, \beta_k$ as random variables with uninformative density functions. A common choice is the gamma distribution with probability density function

$$\Gamma(x|c,d) = \frac{d^c}{\Gamma(c)} x^{c-1} e^{-dx}, \tag{2.8}$$

where c and d are positive shape and rate parameters. Furthermore, $\Gamma(\cdot)$ on the right-hand side of (2.8) denotes the usual gamma function [3]. Note that a gamma-distributed random variable, $X \sim \Gamma(c,d)$, respectively, has mean $E[X] = c/d$ and variance $V[X] = c/d^2$. In particular, $c \to 1$ and $d \to 0$ implies $E[X], V[X] \to \infty$, making (2.8) an uninformative prior. We therefore choose the inverse noise and prior variances, $\boldsymbol{\alpha}$ and $\boldsymbol{\beta}$, to be gamma-distributed:

$$\begin{aligned} p(\alpha_i) &= \Gamma(\alpha_i|c,d), \quad i = 1, \dots, m, \\ p(\beta_j) &= \Gamma(\beta_j|c,d), \quad j = 1, \dots, k. \end{aligned} \tag{2.9}$$

[2] For this prior, $R\mathbf{x}$ being close to zero means that $R\mathbf{x}$ has a small (unweighted) ℓ^2-norm, $\|R\mathbf{x}\|_2$.

By setting $c = 1$ and $d \approx 0$, $\boldsymbol{\alpha}$ and $\boldsymbol{\beta}$ are free from the moderating influence of the hyper-prior and allowed to "vary wildly" following the data. In our numerical tests we used $d = 10^{-4}$ for all one-dimensional problems (signals) and $d = 10^{-2}$ for all two-dimensional problems (images), which is similar to the choices in [54, 5, 6]. Future investigations will elaborate on the influence of these parameters. A few remarks are in order.

Remark 2.4 (conjugate hyper-priors). Choosing the hyper-priors $p(\alpha_i)$ and $p(\beta_j)$ as gamma distributions is convenient since the gamma distribution is a conjugate[3] (see [33, 30, 47]) for the conditionally Gaussian distributions (2.3) and (2.7).

Remark 2.5 (informative hyper-priors). For simplicity we use the same hyper-prior $\Gamma(\cdot|c,d)$ and parameters c, d for all components of $\boldsymbol{\alpha}$, $\boldsymbol{\beta}$. If one has a reasonable a priori notion of what $\boldsymbol{\alpha}$ or $\boldsymbol{\beta}$ should be, the choice for hyper-prior could be modified correspondingly [6, 16].

Remark 2.6 (generalized gamma hyper-priors). The use of generalized gamma distributions was recently investigated in [11] and merged into a hybrid solver in [10]. Although generalized gamma hyper-priors were demonstrated to promote sparsity more strongly in some cases to not exceed the scope of the present work, we limit our discussion to usual gamma hyper-priors.

3. Bayesian inference.

We now propose a Bayesian inference method for the generalized hierarchical Bayesian model from section 2.

3.1. Preliminary observations.

The conditionally Gaussian prior (2.7) and the gamma hyper-prior (2.8) were intentionally chosen because of their conditional conjugacy relationship. Some especially important implications include the following (see [33]):

$$p(\mathbf{y}|\mathbf{x},\boldsymbol{\alpha})p(\mathbf{x}|\boldsymbol{\beta}) \propto \mathcal{N}(\mathbf{x}|\boldsymbol{\mu}, C), \tag{3.1}$$

$$p(\mathbf{y}|\mathbf{x},\boldsymbol{\alpha})p(\boldsymbol{\alpha}) \propto \prod_{i=1}^{m} \Gamma(\alpha_i|1/2+c, [F\mathbf{x}-\mathbf{y}]_i^2/2+d), \tag{3.2}$$

$$p(\mathbf{x}|\boldsymbol{\beta})p(\boldsymbol{\beta}) \propto \prod_{j=1}^{k} \Gamma(\beta_j|1/2+c, [R\mathbf{x}]_j^2/2+d). \tag{3.3}$$

Here the covariance matrix C and the mean $\boldsymbol{\mu}$ in (3.1) are, respectively, given by

$$C = \left(F^T A F + R^T B R\right)^{-1}, \quad \boldsymbol{\mu} = C F^T A \mathbf{y}, \tag{3.4}$$

$[F\mathbf{x}-\mathbf{y}]_i$ denotes the ith entry of the vector $F\mathbf{x}-\mathbf{y} \in \mathbb{R}^m$, and $[R\mathbf{x}]_j$ denotes the jth entry of the vector $R\mathbf{x} \in \mathbb{R}^k$. Note that the two sides of (3.1), (3.2), and (3.3) are equal up to a multiplicative constant that does not depend on $\mathbf{x}$, $\boldsymbol{\alpha}$, and $\boldsymbol{\beta}$, respectively. Finally, we stress that (3.1) only holds if the forward operator $F \in \mathbb{R}^{m\times n}$ and the regularization operator $R \in \mathbb{R}^{k\times n}$ satisfy the *common kernel condition*:

$$\operatorname{kernel}(F) \cap \operatorname{kernel}(R) = \{\mathbf{0}\}, \tag{3.5}$$

[3] Recall that $p(\theta)$ is a *conjugate* for $p(z|\theta)$ if the posterior $p(\theta|z)$ is in the same class of densities (in this case corresponding to gamma distributions) as $p(\theta)$.

Algorithm 3.1 BCD algorithm for the mean

1: Initialize $\boldsymbol{\alpha}^0$, $\boldsymbol{\beta}^0$, and $l=0$
2: **repeat**
3: Update $\mathbf{x}$ by setting $\mathbf{x}^{l+1}=E[\mathbf{x}|\boldsymbol{\alpha}^l,\boldsymbol{\beta}^l,\mathbf{y}]$.
4: Update $\boldsymbol{\alpha}$ by setting $\boldsymbol{\alpha}^{l+1}=E[\boldsymbol{\alpha}|\mathbf{x}^{l+1},\boldsymbol{\beta}^l,\mathbf{y}]$.
5: Update $\boldsymbol{\beta}$ by setting $\boldsymbol{\beta}^{l+1}=E[\boldsymbol{\beta}|\mathbf{x}^{l+1},\boldsymbol{\alpha}^{l+1},\mathbf{y}]$.
6: Increase $l\to l+1$.
7: **until** convergence or maximum number of iterations is reached

which is a standard assumption in regularized inverse problems [41, 53]. Indeed, (3.5) can be interpreted as the prior (regularization) introducing a sufficient amount of complementary information to the likelihood (the given measurements) to make the problem well posed. This indicates that the hierarchical Bayesian model proposed in section 2 does not require R to be invertible as long as (3.5) is satisfied.

3.2. Proposed method: Bayesian coordinate descent. We are now in a position to formulate a Bayesian inference method for the generalized hierarchical Bayesian model from section 2. This method is motivated by the coordinate descent approaches [32, 58] and solves for a descriptive quantity (mode, mean, variance, etc.) of the posterior density function $p(\mathbf{x},\boldsymbol{\alpha},\boldsymbol{\beta}|\mathbf{y})$ by alternatingly updating this quantity for $\mathbf{x}$, $\boldsymbol{\alpha}$, and $\boldsymbol{\beta}$. Henceforth, we refer to this method as the *Bayesian coordinate descent (BCD) algorithm*.

Assume that we are interested in the expected value (mean) of the posterior, $E[\mathbf{x},\boldsymbol{\alpha},\boldsymbol{\beta}|\mathbf{y}]$. The BCD algorithm for this case is described in Algorithm 3.1.

In Algorithm 3.1 and henceforth, all variables with superscripts are treated as fixed parameters. That is, the expected values in Algorithm 3.1 are computed w.r.t. $\mathbf{x}$, $\boldsymbol{\alpha}$, and $\boldsymbol{\beta}$, respectively. Algorithm 3.1 is simple to implement because of the particular decomposition of the posterior density function $p(\mathbf{x},\boldsymbol{\alpha},\boldsymbol{\beta}|\mathbf{y})$ provided by Bayes' theorem (see (1.3)):

$$p(\mathbf{x},\boldsymbol{\alpha},\boldsymbol{\beta}|\mathbf{y})\propto p(\mathbf{y}|\mathbf{x},\boldsymbol{\alpha})p(\mathbf{x}|\boldsymbol{\beta})p(\boldsymbol{\alpha})p(\boldsymbol{\beta}). \tag{3.6}$$

By (3.1)–(3.3), we therefore have

$$p(\mathbf{x}|\boldsymbol{\alpha}^l,\boldsymbol{\beta}^l,\mathbf{y})\propto p(\mathbf{y}|\mathbf{x},\boldsymbol{\alpha}^l)p(\mathbf{x}|\boldsymbol{\beta}^l)\propto\mathcal{N}(\mathbf{x}|\boldsymbol{\mu},C), \tag{3.7}$$

$$p(\boldsymbol{\alpha}|\mathbf{x}^{l+1},\boldsymbol{\beta}^l,\mathbf{y})\propto p(\mathbf{y}|\mathbf{x}^{l+1},\boldsymbol{\alpha})p(\boldsymbol{\alpha})\propto\prod_{i=1}^{m}\Gamma(\alpha_i|1/2+c,[F\mathbf{x}^{l+1}-\mathbf{y}]_i^2/2+d), \tag{3.8}$$

$$p(\boldsymbol{\beta}|\mathbf{x}^{l+1},\boldsymbol{\alpha}^{l+1},\mathbf{y})\propto p(\mathbf{x}^{l+1}|\boldsymbol{\beta})p(\boldsymbol{\beta})\propto\prod_{j=1}^{k}\Gamma(\beta_j|1/2+c,[R\mathbf{x}^{l+1}]_j^2/2+d), \tag{3.9}$$

where the covariance matrix C and the mean $\boldsymbol{\mu}$ in (3.7) are given as in (3.4) with $A=\operatorname{diag}(\boldsymbol{\alpha}^l)$ and $B=\operatorname{diag}(\boldsymbol{\beta}^l)$. Thus, the update step for $\mathbf{x}$ in Algorithm 3.1 reduces to solving the linear system

$$\left(F^TAF+R^TBR\right)\mathbf{x}^{l+1}=F^TA\mathbf{y} \tag{3.10}$$

for the mean $\mathbf{x}^{l+1}$, and the subsequent update steps for $\boldsymbol{\alpha}$ and $\boldsymbol{\beta}$ yield

$$\alpha_i^{l+1} = \frac{1+2c}{[F\mathbf{x}^{l+1}-\mathbf{y}]_i^2 + 2d}, \quad i = 1,\dots,m, \tag{3.11}$$

$$\beta_j^{l+1} = \frac{1+2c}{[R\mathbf{x}^{l+1}]_j^2 + 2d}, \quad j = 1,\dots,k, \tag{3.12}$$

respectively. Hence, Algorithm 3.1 consists of alternating between (3.10)–(3.12).

Remark 3.1. For i.i.d. noise, that is, the likelihood function is (2.1) rather than (2.3), the linear system (3.10) will be simplified to

$$\left(\alpha F^T F + R^T B R\right)\mathbf{x}^{l+1} = \alpha F^T \mathbf{y}, \tag{3.13}$$

and the update step (3.11) correspondingly reduces to

$$\alpha^{l+1} = \frac{m+2c}{\|F\mathbf{x}^{l+1}-\mathbf{y}\|_2^2 + 2d}. \tag{3.14}$$

Remark 3.2. It was demonstrated in [57] that the cost function of classic SBL, which can be recovered from the generalized model in section 2 for $R = I$, is nonconvex with potentially many local minima that are achieved at a sparse solution. Further, the cost function has a global minimum that can produce the maximally sparse solution at the posterior mean and the classic SBL algorithm based on evidence maximization is globally convergent. While we numerically observed similar properties in the context of generalized sparse Bayesian learning (GSBL) and other regularization operators R (with sparsity holding for $R\mathbf{x}$ instead of $\mathbf{x}$), a detailed analysis exceeds the scope of the present paper.

3.3. Efficient implementation of the x-update. If the common kernel condition (3.5) is satisfied, then the coefficient matrix on the left-hand side of (3.10) is symmetric and positive definite (SPD). For sufficiently small problems, (3.10) can therefore be solved efficiently using a preconditioned conjugate gradient (PCG) method [49]. However, the coefficient matrix may become prohibitively large in some cases. To avoid any potential storage and computational issues, we implemented our method using gradient descent for the imaging problems described in section 4.

Let $G = F^T A F + R^T B R$ and $\mathbf{b} = F^T A \mathbf{y}$ be the SPD coefficient matrix and the right-hand side of the linear system (3.10), respectively. The solution of (3.10) then corresponds to the unique minimizer of the quadratic functional

$$J(\mathbf{x}) = \mathbf{x}^T G \mathbf{x} - 2\mathbf{x}^T \mathbf{b} \quad \text{with} \quad \nabla J(\mathbf{x}) = 2\left(G\mathbf{x} - \mathbf{b}\right). \tag{3.15}$$

For this functional, line search minimization can be performed analytically to find the locally optimal step size γ in every iteration. This allows us to use the classical gradient descent method described in Algorithm 3.2 to approximate the solution $\mathbf{x}^{l+1}$ of (3.10).

It is important to note that the gradient in (3.15) can be computed efficiently and without having to store the whole coefficient matrix G, which might be prohibitively large. To show this, assume that the unknown solution $\mathbf{x} \in \mathbb{R}^{n^2}$ corresponds to a vectorized matrix $X \in \mathbb{R}^{n\times n}$

Algorithm 3.2 Gradient descent method

1: Set $\mathbf{r} = \mathbf{b} - G\mathbf{x}$
2: **repeat**
3: Compute $G\mathbf{r}$ according to (3.21).
4: Compute the step size: $\gamma = \mathbf{r}^T\mathbf{r}/\mathbf{r}^T G\mathbf{r}$.
5: Update the solution: $\mathbf{x} + \gamma\mathbf{r}$.
6: Update the difference: $\mathbf{r} = \mathbf{r} - \gamma G\mathbf{r}$.
7: **until** convergence or maximum number of iterations is reached

and that the forward operator F corresponds to applying the same one-dimensional forward operator F_1 to the matrix X in x- and y-direction:

$$F\mathbf{x} = \mathbf{y} \iff F_1 X F_1^T = Y, \tag{3.16}$$

where $F = F_1 \otimes F_1$, $\mathbf{x} = \operatorname{vec}(X)$, and $\mathbf{y} = \operatorname{vec}(Y)$. We furthermore assume that the regularization operator R is defined by

$$R\mathbf{x} = \begin{bmatrix} I \otimes R_1 \\ R_1 \otimes I \end{bmatrix} \operatorname{vec}(X) = \begin{bmatrix} \operatorname{vec}(R_1 X) \\ \operatorname{vec}(X R_1^T) \end{bmatrix}, \tag{3.17}$$

which corresponds to anisotropic regularization. Using some basic properties of the Kronecker product and the elementwise Hadamard product $\odot$, it can be shown that

$$F^T A F\mathbf{x} = \operatorname{vec}\left(F_1^T \left[\tilde{A} \odot F_1 X F_1^T\right] F_1\right), \tag{3.18}$$

$$R^T B R\mathbf{x} = \operatorname{vec}\left(\left[\tilde{B}_1 \odot X R_1^T\right] R_1\right) + \operatorname{vec}\left(R_1^T \left[\tilde{B}_2 \odot R_1 X\right]\right), \tag{3.19}$$

$$\mathbf{b} = \operatorname{vec}\left(F_1^T \left[\tilde{A} \odot X\right] F_1\right), \tag{3.20}$$

where $\tilde{A}$, $\tilde{B}_1$, and $\tilde{B}_2$ are such that $\operatorname{vec}(\tilde{A}) = \boldsymbol{\alpha}$, $\operatorname{vec}(\tilde{B}_1) = \boldsymbol{\beta}^1$, and $\operatorname{vec}(\tilde{B}_2) = \boldsymbol{\beta}^2$, with $\boldsymbol{\beta} = [\boldsymbol{\beta}^1, \boldsymbol{\beta}^2]$. Combining (3.18)–(3.20) yields

$$G\mathbf{x} = \operatorname{vec}\left(F_1^T \left[\tilde{A} \odot F_1 X F_1^T\right] F_1\right) + \operatorname{vec}\left(\left[\tilde{B}_1 \odot X R_1^T\right] R_1\right) + \operatorname{vec}\left(R_1^T \left[\tilde{B}_2 \odot R_1 X\right]\right), \tag{3.21}$$

and therefore,

$$\begin{aligned} \nabla J(\mathbf{x}) = 2\Bigg[& \operatorname{vec}\left(F_1^T \left[\tilde{A} \odot F_1 X F_1^T\right] F_1\right) + \operatorname{vec}\left(\left[\tilde{B}_1 \odot X R_1^T\right] R_1\right) \\ & + \operatorname{vec}\left(R_1^T \left[\tilde{B}_2 \odot R_1 X\right]\right) - \operatorname{vec}\left(F_1^T \left[\tilde{A} \odot X\right] F_1\right)\Bigg]. \end{aligned} \tag{3.22}$$

Observe that all of the matrices in (3.21) and (3.22) are significantly smaller than F and R.

3.4. Uncertainty quantification. The proposed BCD algorithm has the advantage of allowing for uncertainty quantification in the reconstructed solution $\mathbf{x}$. For fixed $\boldsymbol{\alpha}$ and $\boldsymbol{\beta}$, Bayes' theorem and the conjugacy relationship (3.1) yield

$$p(\mathbf{x}|\mathbf{y}) \propto p(\mathbf{y}|\mathbf{x})p(\mathbf{x}) \propto \mathcal{N}(\mathbf{x}|\boldsymbol{\mu}, C), \tag{3.23}$$

where the mean $\boldsymbol{\mu}$ and the covariance matrix C are again given by (3.4). We can then sample from the normal distribution $\mathcal{N}(\boldsymbol{\mu}, C)$ to obtain, for instance, credible intervals for every component of the solution $\mathbf{x}$. At the same time, we stress that this only allows for uncertainty quantification in $\mathbf{x}$ for given hyper-parameters $\boldsymbol{\alpha}$ and $\boldsymbol{\beta}$. The above approach does not include uncertainty in $\boldsymbol{\alpha}$ and $\boldsymbol{\beta}$ when these are treated as random variables themselves. This might be achieved by employing a computationally more expensive sampling approach [6], which we will investigate in future work.

3.5. Relationship to current methodology. We now address the connection between the proposed BCD algorithm and some existing methods.

3.5.1. Iterative alternating sequential algorithm. There are both notable similarities and key distinctions between the proposed BCD algorithm and the iterative alternating sequential (IAS) algorithm, developed in [13, 12] and further investigated in [9, 16]. Both algorithms estimate the unknown $\mathbf{x}$ and other involved parameters by alternatingly updating them. However, in contrast to the BCD method, the IAS algorithm assumes that the noise covariance matrix A is known, which then allows the restriction to white Gaussian noise $\boldsymbol{\nu} \sim \mathcal{N}(\mathbf{0}, I)$; see [16, section 2]. Moreover, the IAS algorithm builds upon a conditionally Gaussian prior for which the elements of the diagonal covariance matrix are gamma-distributed, rather than the elements of the diagonal *inverse* covariance matrix as done here, which does not result in a conjugate hyper-prior. This makes the update steps for $\mathbf{x}$ and the hyper-parameters of the prior more complicated. Finally, the IAS algorithm solves for the MAP estimate of the posterior, which does not provide uncertainty quantification in the reconstructed solution. By contrast, the proposed BCD method grants access to the solution posterior $p(\mathbf{x}|\mathbf{y})$ for fixed hyper-parameters.

3.5.2. Iteratively reweighted least squares. The update steps (3.10)–(3.12) resulting from Algorithm 3.1 can be interpreted as an iteratively reweighted least squares (IRLS) algorithm [25]. The idea behind the IRLS algorithms is to recover, for instance, a sparse solution by penalizing the components of $\mathbf{x}$ by weighting them individually and iteratively updating these weights. Indeed, the update steps (3.10)–(3.12) resemble reweighted Tikhonov-regularization strategies. In this regard, the BCD method provides a solid Bayesian interpretation for commonly used reweighting choices and might be used to tailor these weights to specific statistical assumptions on the underlying problem.

3.5.3. ARD/SBL optimization via iteratively reweighted ℓ^1-minimization. The first SBL algorithms used the same $\mathbf{x}$-update as in Algorithm 3.1, but updated the noise and prior parameters α, $\boldsymbol{\beta}$ using the evidence approach (expectation maximization) or the fixed-point approach, [54, 46]. Although these methods can yield sparse solutions, they have no convergence guarantees and become prohibitively slow for large problems. Subsequently, in [56] it was demonstrated that the (type-II) evidence approach can be interpreted as a (type-I)

MAP approach with a special nonfactorable prior. With this insight in hand, a more efficient algorithm was then proposed to update β based on reweighted ℓ^1-minimization, which provably converges to a local maximum of the evidence $p(\mathbf{y}|\boldsymbol{\alpha},\boldsymbol{\beta})$ (see (A.3)) with respect to $\boldsymbol{\beta}$. For the "almost" general regularization operators considered here, we cannot use the algorithm proposed in [56] since the evidence becomes improper if kernel$(R) \neq \{\mathbf{0}\}$ (see Appendix A). By contrast, the α- and $\boldsymbol{\beta}$-updates in Algorithm 3.1 are decoupled and based on maximizing the full conditional posteriors (3.8) and (3.9), respectively (if we solve for the mode of the posterior $p(\mathbf{x},\boldsymbol{\alpha},\boldsymbol{\beta}|\mathbf{y})$) or computing the mean of the full conditional posteriors (3.8) and (3.9) (if we solve for the mean of the posterior $p(\mathbf{x},\boldsymbol{\alpha},\boldsymbol{\beta}|\mathbf{y})$). We were able to derive explicit and efficient formulas for these based on the conditionally conjugate relationships between the likelihood, prior, and hyper-priors.

4. Numerical results.

The MATLAB code used to generate the numerical tests presented here is open access and can be found at GitHub.[4]

4.1. Computational complexity.

We start by addressing the computational complexity of the proposed BCD algorithm (Algorithm 3.1) for Bayesian inference. Assume that Algorithm 3.1 stops after L iterations, either because the algorithm has converged or reached the maximum number of iterations. In every iteration, the algorithm performs the $\mathbf{x}$-update (3.10), the α-update (3.11), and the $\boldsymbol{\beta}$-update (3.12). Denoting their computational complexity by $\mathcal{O}(h_x)$, $\mathcal{O}(h_\alpha)$, and $\mathcal{O}(h_\beta)$, respectively, the total computational complexity of the BCD method is $\mathcal{O}(L(h_x + h_\alpha + h_\beta))$.

The $\mathbf{x}$-update. If $\mathbf{x} \in \mathbb{R}^n$ represents a one-dimensional signal and the $\mathbf{x}$-update (3.10) is solved using the PCD method, then the computational complexity of this update is $\mathcal{O}(\tilde{n})$, where $\tilde{n}$ is the number of the nonzero elements of the coefficient matrix $G \in \mathbb{R}^{n\times n}$ on the left-hand side of (3.10).[5] On the other hand, if $\mathbf{x} = \mathrm{vec}(X) \in \mathbb{R}^{n^2}$ is the vectorized representation of an image $X \in \mathbb{R}^{n\times n}$ and the coefficient matrix $G \in \mathbb{R}^{n^2\times n^2}$ is dense. In this case we solve the $\mathbf{x}$-updated (3.10) using the efficient gradient descent approach described in subsection 3.3. This method has a computational complexity of $\mathcal{O}(n^3)$ for a fixed number of iterations.[6] We thus have $h_x = \max\{n^3, \tilde{n}\}$.

The α- and β-updates. If $\mathbf{x} \in \mathbb{R}^n$, $F \in \mathbb{R}^{m\times n}$, and $R \in \mathbb{R}^{k\times n}$, then $\boldsymbol{\alpha}$, $\boldsymbol{\beta}$ in (3.11) and (3.12) can be computed in $\mathcal{O}(nm)$ and $\mathcal{O}(nk)$, respectively. Assuming that F and R only contain $\tilde{n}_F$ and $\tilde{n}_R$ elements, then the computational complexity of the $\boldsymbol{\alpha}$- and $\boldsymbol{\beta}$-updates reduces to $\mathcal{O}(\tilde{n}_F)$ and $\mathcal{O}(\tilde{n}_R)$, respectively. We thus have $h_\alpha = \max\{nm, \tilde{n}_F\}$ and $h_\beta = \max\{nk, \tilde{n}_R\}$.

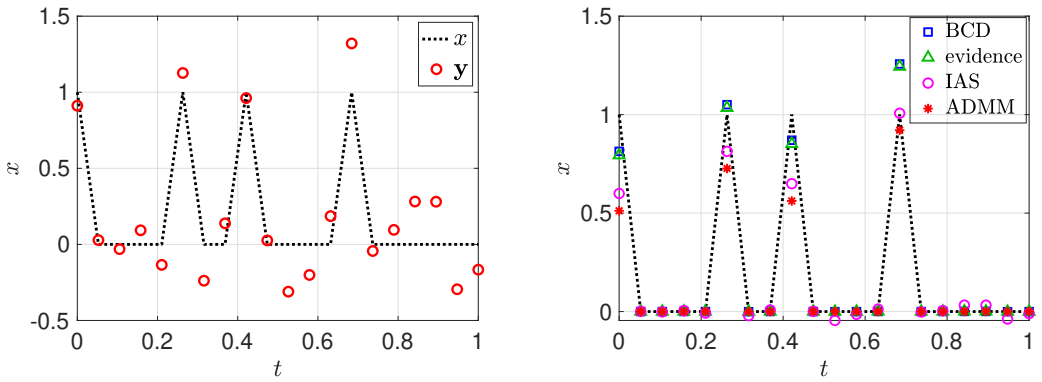
4.2. Denoising a sparse signal.

Consider the sparse nodal values $\mathbf{x}$ of a signal $x : [0,1] \to \mathbb{R}$ at $n = 20$ equidistant points. All of the values in $\mathbf{x}$ are zero except at four randomly selected locations, where the values were set to 1. We are given noisy observations $\mathbf{y}$ which result from adding i.i.d. zero-mean normal noise with variance $\sigma^2 = 5 \cdot 10^{-2}$ to the exact values $\mathbf{x}$. The signal-to-noise ratio (SNR), defined as $E[\mathbf{x}^2]/\sigma^2$ with $E[\mathbf{x}^2] = (x_1^2 + \cdots + x_n^2)/n$, is 4.

[4] See https://github.com/jglaubitz/generalizedSBL.

[5] This assumes that the coefficient matrix itself is computed in $\mathcal{O}(\tilde{n})$.

[6] In our implementation we used five gradient descent steps for each $\mathbf{x}$-update.



(a) Signal x and noisy observations $\mathbf{y}$ (b) Reconstructions by different methods

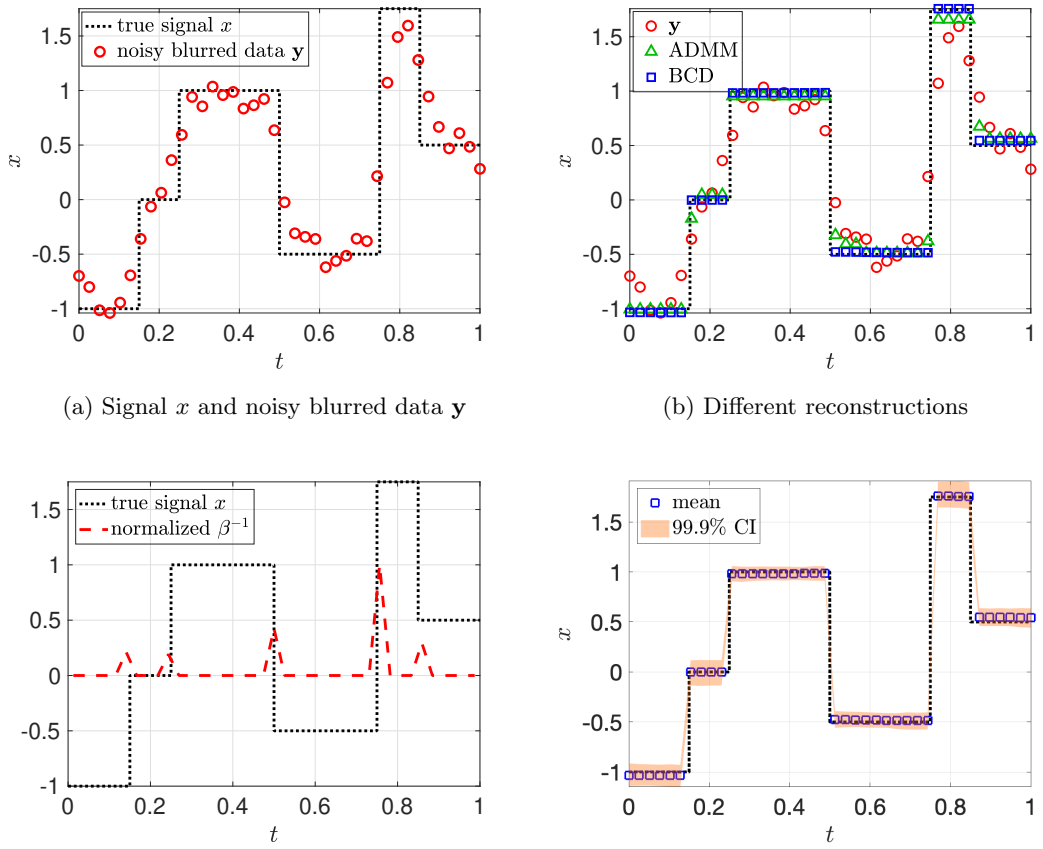
Figure 2. *The sparse signal x and noisy observations $\mathbf{y}$ at $n = 20$ equidistant points, and reconstructions by different methods.*

Figure 2a illustrates the exact values of x and the noisy observations $\mathbf{y}$. The corresponding data model and regularization operator are

$$\mathbf{y} = \mathbf{x} + \boldsymbol{\nu}, \quad R = I. \tag{4.1}$$

This simple test case allows us to compare the proposed BCD algorithm with some existing methods, some of which assume $\mathbf{x}$ itself to be sparse ($R = I$). Figure 2b provides a comparison of the BCD algorithm with (1) SBL using the evidence approach [54], (2) the IAS method [12, 13] solving for the MAP estimate of the posterior, and (3) the alternating direction method of multipliers (ADMM) [8] solving the deterministic ℓ^1-regularized problem (1.2). The free parameters of the IAS algorithm were fine-tuned by hand and chosen as $\beta = 1.55$ and $\theta_j^* = 5 \cdot 10^{-2}$ for $j = 1, \ldots, n$; see [16] for more details on these parameters. The regularization parameter λ in (1.2) was also fine-tuned by hand and set to $\lambda = 2\sigma^2 \|\mathbf{x}\|_0$. Finally, for the proposed BCD algorithm and the evidence approach, we assumed the noise variance σ^2 to be unknown, which therefore had to be estimated by the method as well. We can see in Figure 2b that for this example all of the SBL-based methods perform similarly. On the other hand, the ADMM yeilds a more regularized reconstruction, which might be explained by the uniform nature of the ℓ^1-regularization term in (1.2). This is in contrast to the hierarchical Bayesian model which allows for spatially varying regularization. In this regard we note that there are weighted ℓ_1-regularization methods [17, 20, 1] that incorporate spatially varying regularization parameters. While such techniques can improve the resolution near the nonzero values in sparse signals, as well as near the internal edges in images, they are still point estimates and thus do not provide additional uncertainty information. Hence in the current investigation we simply employ the standard ADMM with a fine-tuned nonvarying regularization parameter as a reasonable comparison.

4.3. Deconvolution of a piecewise constant signal. We next consider deconvolution of the piecewise constant signal $x : [0, 1] \to \mathbb{R}$ illustrated in Figure 3. The corresponding data model and regularization operator are given by



(a) Signal x and noisy blurred data $\mathbf{y}$

(b) Different reconstructions

(c) Normalized covariance parameter β^{-1}

(d) Mean and 99.9% credible intervals

Figure 3. *Deconvolution of a piecewise constant signal x from noisy blurred data $\mathbf{y}$ with i.i.d. zero-mean normal noise with variance $\sigma^2 = 10^{-2}$.*

$$\mathbf{y} = F\mathbf{x} + \boldsymbol{\nu}, \quad R = \begin{bmatrix} -1 & 1 & & \\ & \ddots & \ddots & \\ & & -1 & 1 \end{bmatrix} \in \mathbb{R}^{(n-1)\times n}, \tag{4.2}$$

respectively, where $\boldsymbol{\nu} \sim \mathcal{N}(\mathbf{0}, \sigma^2 I)$ with $\sigma^2 = 10^{-2}$ (SNR ≈ 80) and F is obtained by applying the midpoint quadrature to the convolution equation

$$y(s) = \int_0^1 k(s - s')x(s)\,\mathrm{d}s'. \tag{4.3}$$

We assume a Gaussian convolution kernel of the form

$$k(s) = \frac{1}{2\pi\gamma^2} \exp\left(-\frac{s^2}{2\gamma^2}\right) \tag{4.4}$$

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

$$[F]_{ij} = hk(h[i-j]), \quad i, j = 1, \ldots, n, \tag{4.5}$$

where $h = 1/n$ is the distance between consecutive grid points. Note that F has full rank but quickly becomes ill-conditioned.

Figure 3a illustrates the true signal x as well as the given noisy blurred data $\mathbf{y}$ at $n = 40$ equidistant points. Figure 3b provides the reconstructions using the SBL-based BCD algorithm and the ADMM ℓ^1-regularized inverse problem (1.2). The regularization parameter λ in (1.2) was again fine-tuned by hand and chosen as $\lambda = 2\sigma^2\|R\mathbf{x}\|_0$. We do not include any of the existing SBL algorithms considered before (the evidence approach and IAS algorithm) since they cannot be applied to the nonquadratic regularization operator R in (4.2) without modifying this operator first. Figure 3c illustrates the normalized prior covariance parameters β^{-1} which are estimated as part of the BCD algorithm. Observe that the values are significantly larger at the locations of the jump discontinuities. This allows the reconstruction to "jump" and highlights the nonuniform character of regularization in the hierarchical Bayesian model suggested in section 2. Finally, Figure 3d demonstrates the possibility to quantify uncertainty when using the BCD algorithm by providing the 99.9% credible intervals of the solution posterior $p(\mathbf{x}|\mathbf{y})$ for the final estimates of α and $\boldsymbol{\beta}$. Note that these credible intervals, especially their width, indicate the amount of uncertainty in the reconstruction.

The results displayed in Figure 4 are for the same model with the noise variance, increased by 500%, to $\sigma^2 = 5 \cdot 10^{-2}$ (SNR ≈ 16). The BCD algorithm now yields a less accurate reconstruction, especially between $t = 0.15$ and $t = 0.25$. This is also reflected in the corresponding normalized prior covariance parameters β^{-1}, which can be found Figure 4c. Observe that the second peak around $t = 0.25$ is underestimated and therefore causes the block associated with the region $[0.15, 0.25]$ to be drawn towards the subsequent block associated with the region $[0.25, 0.5]$. The increased uncertainty of the reconstruction is indicated by the 99.9% credible intervals in Figure 4d. In particular, we note the increased width of the credible interval in the region $[0.15, 0.25]$.

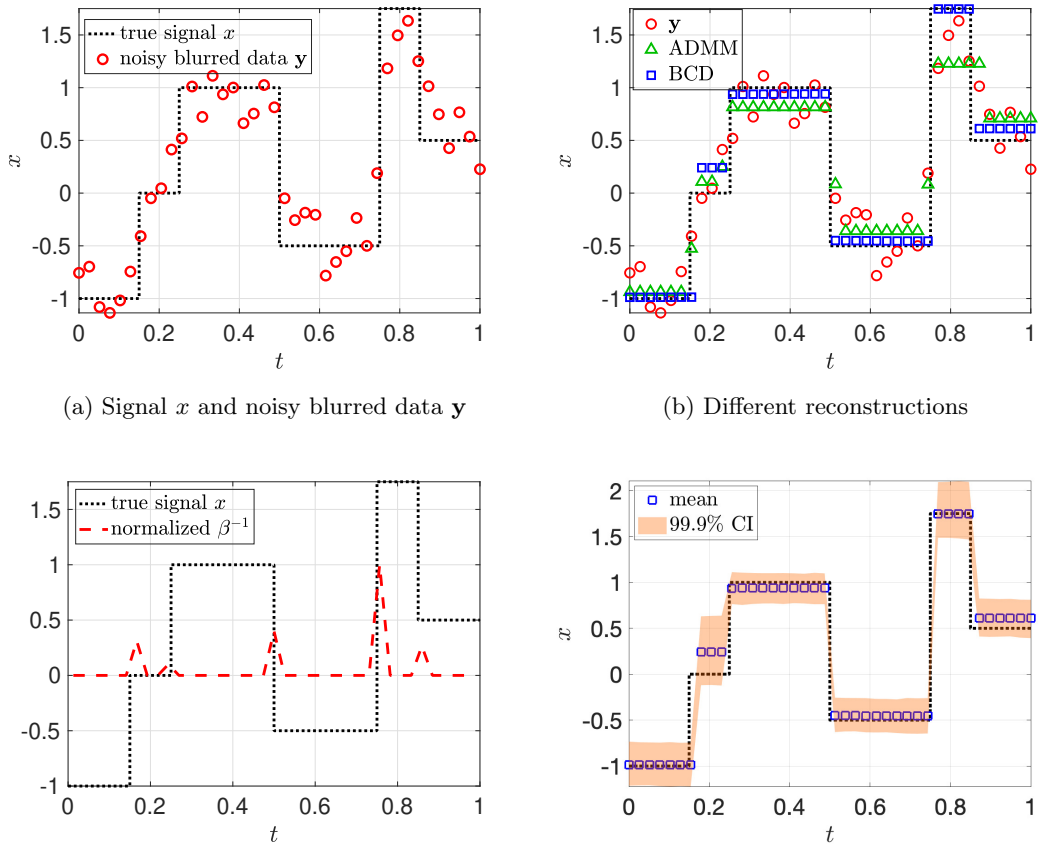
4.4. Combining different regularization operators. We next demonstrate that generalized SBL allows us to consider combinations of different regularization operators. Consider the signal $x : [0, 1] \to \mathbb{R}$ illustrated in Figure 5a, which is piecewise constant on $[0, 0.5]$ and piecewise linear on $[0.5, 1]$. The corresponding data model is the same as before with convolution parameter $\gamma = 10^{-2}$ and i.i.d. zero-mean normal noise with variance $\sigma^2 = 10^{-2}$ (SNR ≈ 40). Figure 5b illustrates the reconstructions obtained by the BCD algorithm using a first- and second-order TV-regualrization operator,

$$R_1 = \begin{bmatrix} -1 & 1 & & \\ & \ddots & \ddots & \\ & & -1 & 1 \end{bmatrix}, \quad R_2 = \begin{bmatrix} -1 & 2 & -1 & & \\ & \ddots & \ddots & \ddots & \\ & & -1 & 2 & -1 \end{bmatrix}, \tag{4.6}$$

which promote piecewise constant and piecewise linear solutions, respectively. Observe that neither R_1 nor R_2 is even square, meaning that both would have to be modified by introducing additional rows to apply a standard SBL approach, which can become increasingly complicated for higher orders and multiple dimensions.

It is evident from Figure 5b that using first-order TV-regularization yields a less accurate reconstruction in $[0.5, 1]$, where the signal is piecewise linear,[7] while using second-order

[7] This well-known artifact of first-order TV-regularization is often called the "staircasing" effect and motivates using higher order TV-regularization [2, 52].



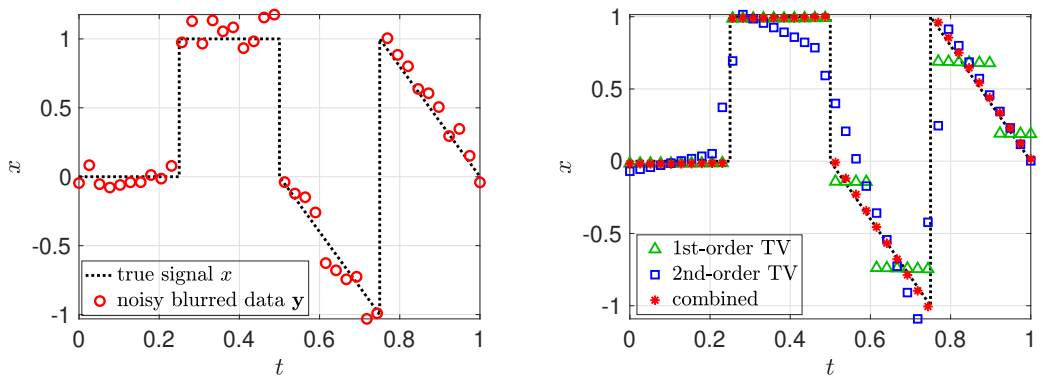
(a) Signal x and noisy blurred data $\mathbf{y}$

(b) Different reconstructions

(c) Normalized covariance parameter β^{-1}

(d) Mean and 99.9% credible intervals

Figure 4. *Deconvolution of a piecewise constant signal x from noisy blurred data $\mathbf{y}$ with i.i.d. zero-mean normal noise with variance $\sigma^2 = 5 \cdot 10^{-2}$.*



(a) Signal x and noisy observations $\mathbf{y}$

(b) Reconstructions using different regualrizations

Figure 5. *Signal x and noisy observations $\mathbf{y}$ at $n = 20$ equidistant points, and reconstructions by different methods.*

TV-regularization yields a less accurate reconstruction in $[0, 0.5]$, where the signal is piecewise constant. However, generalized SBL and the proposed BCD algorithm allows us to consider the combined regularization operator

$$R = \begin{bmatrix} -1 & 1 & & & & & \\ & \ddots & \ddots & & & & \\ & & -1 & 1 & & & \\ & & -1 & 2 & -1 & & \\ & & & \ddots & \ddots & \ddots & \\ & & & & -1 & 2 & -1 \end{bmatrix} \in \mathbb{R}^{(n-3)\times n}. \tag{4.7}$$

Assuming $n = 2q$, the first $k-1$ rows correspond to first-order TV-regularization while the last $k-2$ rows correspond to second-order TV-regularization. The advantage of using this nonstandard regularization operator in the BCD algorithm is demonstrated by the red stars in Figure 5b.

4.5. Image deconvolution. We next consider the reference image X in Figure 6a and its noisy blurred version Y in Figure 6b. Y results from X by applying the discrete one-dimensional convolution operator (4.5) in the two canonical coordinate directions and then adding i.i.d. zero-mean normal noise. The corresponding forward model is $Y = FXF^T + N$ or, equivalently,

$$\mathbf{y} = G\mathbf{x} + \boldsymbol{\nu}, \tag{4.8}$$

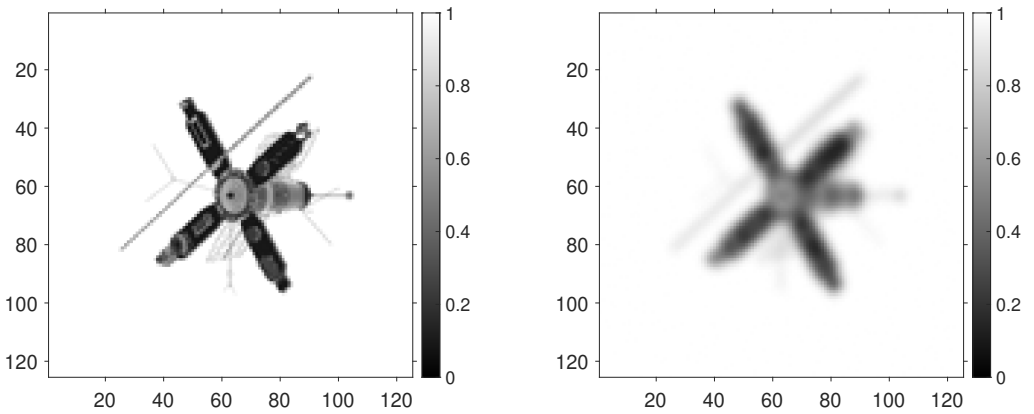
after vectorization. Here, $\mathbf{z} = \mathrm{vec}(Z)$ denotes the $mn \times 1$ column vectors obtained by stacking the columns of the $m \times n$ matrix Z on top of one another, and $G = F \otimes F$. Further, the blurring parameter and noise variance were chosen as $\gamma = 1.5 \cdot 10^{-2}$ and $\sigma^2 = 10^{-5}$ ($\mathrm{SNR} \approx 4 \cdot 10^3$) to make the test case comparable to the one in [6, section 4.2].

Figures 6c and 6d show the reconstructions obtained by the ADMM applied to (1.2) and the SBL-based BCD algorithm with an anisotropic second-order TV operator

$$R = \begin{bmatrix} I \otimes D \\ D \otimes I \end{bmatrix} \quad \text{with} \quad D = \begin{bmatrix} -1 & 2 & -1 & & \\ & \ddots & \ddots & \ddots & \\ & & -1 & 2 & -1 \end{bmatrix} \in \mathbb{R}^{(n-2)\times n}. \tag{4.9}$$

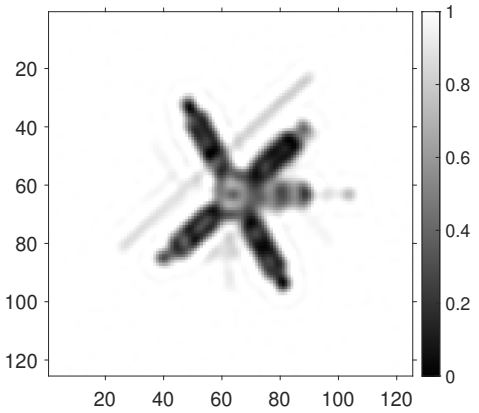
The regularization parameter λ in (1.2) was again fine-tuned by hand and set to $\lambda = 10^{-5}$. The BCD algorithm provides a sharper reconstruction (see Figure 6) than the ADMM applied to the ℓ^1-regularized inverse problem (1.2). Further parameter tuning might increase the accuracy of the reconstruction by the ADMM. By contrast, it is important to stress that the BCD algorithm requires no such exhaustive parameter tuning.

4.6. Noisy and incomplete Fourier data. We next address the reconstruction of images based on noisy and incomplete Fourier data, which is common in applications such as magnetic resonance imaging (MRI) and synthetic aperture radar (SAR). The popular prototype Shepp–Logan phantom test image is displayed in Figure 7a.

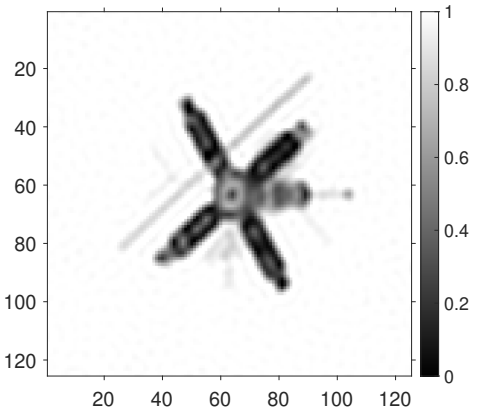


(a) Reference image

(b) Noisy blurred image



(c) ℓ^1-regularization by ADMM



(d) SBL by the BCD algorithm

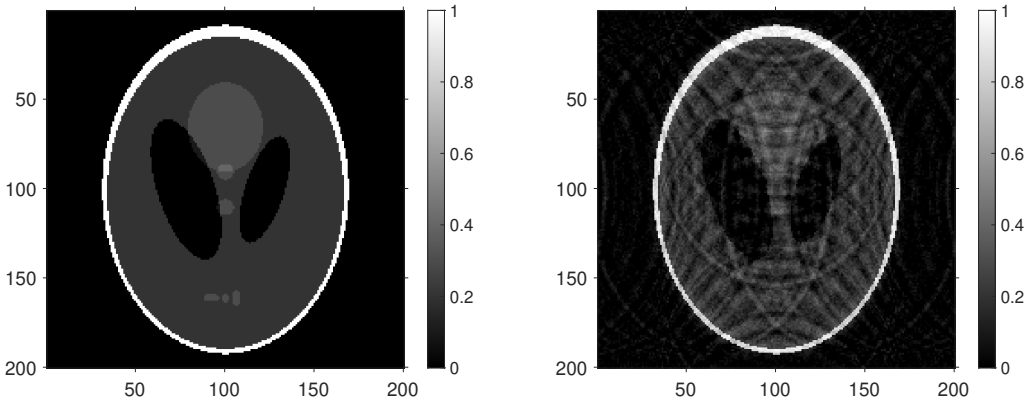
Figure 6. *The reference image, the corresponding noisy blurred image, and reconstructions using the ADMM and the BCD algorithm, Algorithm 3.1.*

The indirect data $\mathbf{y} = \mathrm{vec}(Y)$ is given by applying the two-dimensional discrete Fourier transform to the reference image X, removing certain frequencies, and adding noise. Since in this investigation we are assuming $\mathbf{x} \in \mathbb{R}^n$, we consider the data model

$$\begin{bmatrix} \mathrm{Re}(\mathbf{y}) \\ \mathrm{Im}(\mathbf{y}) \end{bmatrix} = \begin{bmatrix} \mathrm{Re}(G) \\ \mathrm{Im}(G) \end{bmatrix} \mathbf{x} + \boldsymbol{\nu} \tag{4.10}$$

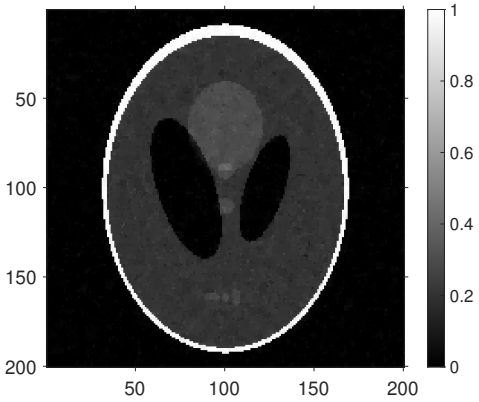
with $\mathrm{Re}(\mathbf{y})$ and $\mathrm{Im}(\mathbf{y})$, respectively, denoting the real and imaginary part of $\mathbf{y} \in \mathbb{C}^m$.[8] Further, $\boldsymbol{\nu} \in \mathbb{R}^{2m}$ corresponds to i.i.d. zero-mean normal noise with variance $\sigma^2 = 10^{-3}$ (SNR ≈ 60)

[8]Our technique is not limited to real-valued solutions, and we will consider complex-valued solutions, such as those occurring in SAR, in future work.

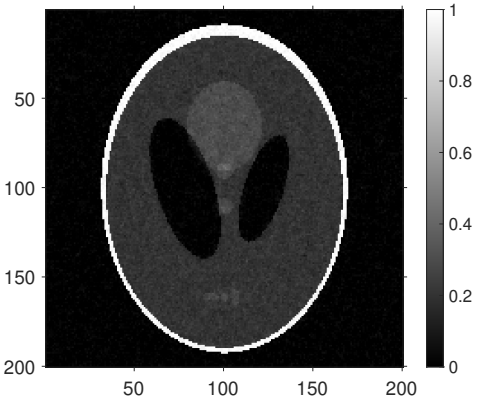


(a) Reference image

(b) ML/LS estimate



(c) ℓ^1-regularization by ADMM



(d) SBL by the BCD algorithm

Figure 7. (a) *The Shepp–Logan phantom test image;* (b) *the ML/LS estimate, and reconstructions using* (c) *the ADMM applied to* (1.2)*; and* (d) *the SBL-based BCD algorithm.*

and $G = F \otimes F$, where F denotes the one-dimensional discrete Fourier transform with missing frequencies, which we impose to mimic the situation where the system is underdetermined and some data must for some reason be discarded. The removed frequencies were determined by sampling 100 logarithmically spaced integers between 10 and 200. Finally, because the image is piecewise constant, we used first-order TV-regularization.

Figure 7b shows the maximum likelihood (ML) estimate of the image, which is obtained by maximizing the likelihood function $p(\mathbf{x}|\mathbf{y})$. In this case, the ML estimate is the same as the least squares (LS) solution of the linear system (4.10). Figures 7c and 7d illustrate the reconstructions obtained by applying ADMM to the ℓ^1-regularized inverse problem (1.2) and the SBL-based BCD algorithm. The regularization parameter in (1.2) was again fine-tuned by hand and chosen as $\lambda = 4\sigma^2$. While the reconstructions in Figures 7c and 7d are comparable,

it is important to point out that we did not use any prior knowledge about the noise variance or perform any parameter tuning for the BCD algorithm.

4.7. Data fusion. As a final example we consider a data fusion problem to demonstrate the possible advantage of using the generalized noise model discussed in subsection 2.1. Recall the piecewise constant signal discussed in subsection 4.3, and assume we want to reconstruct the values of this signal at $n = 40$ equidistant grid points, denoted by $\mathbf{x}$. We are given two sets of data: $\mathbf{y}^{(1)}$ corresponds to direct observations taken at 36 randomly selected locations with added i.i.d. zero-mean normal noise $\boldsymbol{\nu}^{(1)}$ with variance $\sigma_1^2 = 5 \cdot 10^{-1}$, and $\mathbf{y}^{(2)}$ corresponds to blurred observations at 24 randomly selected locations with added i.i.d. zero-mean normal noise $\boldsymbol{\nu}^{(2)}$ with variance $\sigma_2^2 = 10^{-2}$. The blurring is again modeled using (4.5) with a Gaussian convolution kernel and convolution parameter $\gamma = 3 \cdot 10^{-2}$. Further, a first-order TV-regularization operator is employed to promote a piecewise constant reconstruction.

The separate reconstructions by the SBL-based BCD algorithm can be found in Figures 8a and 8b. Both reconstructions are of poor quality, which is due to the high noise variance in the case of $\mathbf{y}^{(1)}$ and to the missing information in the case of $\mathbf{y}^{(2)}$. In fact, the reconstruction illustrated in Figure 8b is of reasonable quality except for the region around $t = 0.2$, where a void of observations causes the reconstruction to miss the jumps at $t = 0.15$ and $t = 0.25$.

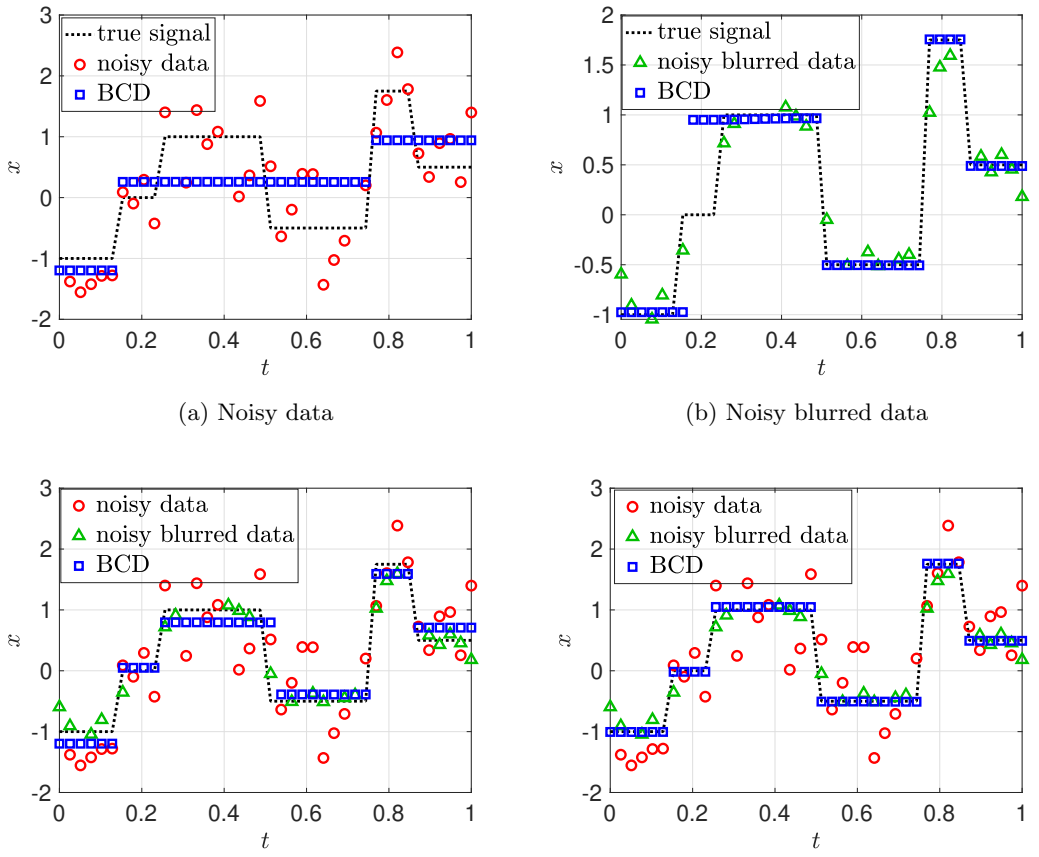
Following Example 2.1, we now fuse the two data sets by considering the joint data model

$$\underbrace{\begin{bmatrix}\mathbf{y}^{(1)}\\ \mathbf{y}^{(2)}\end{bmatrix}}_{=\mathbf{y}} = \underbrace{\begin{bmatrix}F^{(1)}\\ F^{(2)}\end{bmatrix}}_{=F}\mathbf{x} + \underbrace{\begin{bmatrix}\boldsymbol{\nu}^{(1)}\\ \boldsymbol{\nu}^{(2)}\end{bmatrix}}_{=\boldsymbol{\nu}}, \tag{4.11}$$

where $F^{(1)}$ and $F^{(2)}$ are the forward models describing how $\mathbf{x}$ is mapped to $\mathbf{y}^{(1)}$ and $\mathbf{y}^{(2)}$, respectively. Employing the usual likelihood function (2.1) would correspond to assuming that all the components of stacked noise vector $\boldsymbol{\nu}$ are i.i.d., which is not true for this example. The resulting reconstruction by the BCD algorithm can be found in Figure 8c. In contrast, utilizing the generalized likelihood function (2.3) with

$$A = \operatorname{diag}(\underbrace{\alpha_1, \ldots, \alpha_1}_{m_1 \text{ times}}, \underbrace{\alpha_2, \ldots, \alpha_2}_{m_2 \text{ times}}), \tag{4.12}$$

we can appropriately model that $\boldsymbol{\nu}^{(1)}$ and $\boldsymbol{\nu}^{(2)}$ have different variances. The corresponding reconstruction by the BCD algorithm using this generalized data model is provided in Figure 8d. Observe that the reconstruction using the generalized noise model (Figure 8d) is clearly more accurate than the one for the i.i.d. assumption (Figure 8c). This can be explained by noting that the first data set is larger than the second one, containing $m_1 = 36$ and $m_2 = 24$ observations, respectively. At the same time, the data of the first set is less accurate than of the second one, since the variances are $\sigma_1^2 = 5 \cdot 10^{-1}$ and $\sigma_2^2 = 10^{-2}$ ($\mathrm{SNR}_1 \approx 1.6$ and $\mathrm{SNR}_2 \approx 80$), respectively. Hence, when using the usual i.i.d. assumption, the first data set $\mathbf{y}^{(1)}$, which is larger but less accurate, more strongly influences the reconstruction than second



(a) Noisy data

(b) Noisy blurred data

(c) Combined, i. i. d. assumption

(d) Combined, generalized data model

Figure 8. *Data fusion example with incomplete noisy and incomplete noisy blurred data. Top row: Separate reconstructions using the SBL-based BCD algorithm. Bottom row: Combined reconstructions using the SBL-based BCD algorithm with i.i.d. assumption and using a generalized data model.*

data set, which is smaller but more accurate. Using the generalized data model, on the other hand, the BCD algorithm is able to more appropriately balance the influence of the different data sets.

5. Concluding remarks. This paper introduced a generalized approach for SBL and an efficient realization of it by the newly proposed BCD algorithm. In contrast to existing SBL methods, we are able to use any regularization operator R as long as the common kernel condition (3.5) is satisfied, a usual assumption in regularized inverse problems. Further, the BCD algorithm provides us with the full solution posterior $p(\mathbf{x}|\mathbf{y})$ for fixed hyper-parameters rather than just resulting in a point estimate, while being easy to implement and highly efficient. Future work will elaborate on sampling based methods for Bayesian inference [6], which might be computationally more expensive but would also allow sampling from the full joint posterior $p(\mathbf{x}, \boldsymbol{\alpha}, \boldsymbol{\beta}|\mathbf{y})$. This has been addressed to some extent in [14, section 6] for uncertainty quantification in regions of interest. Other research directions might include data-informed choices for the parameters c and d in (2.9) and data fusion applications. Finally,

it would be of interest to combine the proposed generalized SBL framework with generalized gamma distributions as hyper-priors [11] and the hybrid solver from [10].

Appendix A. Evidence approach. In the evidence approach [54, 5], the posterior $p(\mathbf{x}, \boldsymbol{\alpha}, \boldsymbol{\beta}|\mathbf{y})$ is decomposed as

$$p(\mathbf{x}, \boldsymbol{\alpha}, \boldsymbol{\beta}|\mathbf{y}) = p(\mathbf{x}|\mathbf{y}, \boldsymbol{\alpha}, \boldsymbol{\beta}) p(\boldsymbol{\alpha}, \boldsymbol{\beta}|\mathbf{y}). \tag{A.1}$$

The variables $\mathbf{x}$, $\boldsymbol{\alpha}$, and $\boldsymbol{\beta}$ are then alternatingly updated, with the hyper-parameters $\boldsymbol{\alpha}$ and $\boldsymbol{\beta}$ calculated as the mode (most probable value) of the hyper-parameter posterior $p(\boldsymbol{\alpha}, \boldsymbol{\beta}|\mathbf{y})$. By Bayes' law, one has

$$p(\boldsymbol{\alpha}, \boldsymbol{\beta}|\mathbf{y}) \propto p(\boldsymbol{\alpha}) p(\boldsymbol{\beta}) p(\mathbf{y}|\boldsymbol{\alpha}, \boldsymbol{\beta}), \tag{A.2}$$

where the *evidence* $p(\mathbf{y}|\boldsymbol{\alpha}, \boldsymbol{\beta})$ can be determined by marginalizing out the unknown solution $\mathbf{x}$, which yields

$$p(\mathbf{y}|\boldsymbol{\alpha}, \boldsymbol{\beta}) = \int p(\mathbf{y}|\mathbf{x}, \boldsymbol{\alpha}) p(\mathbf{x}|\boldsymbol{\beta}) \, \mathrm{d}\mathbf{x}. \tag{A.3}$$

Some basic but lengthy computations are then used to obtain

$$p(\mathbf{y}|\boldsymbol{\alpha}, \boldsymbol{\beta}) = (2\pi)^{(n-m)/2} \det(A)^{1/2} \det(B)^{1/2} \det(C)^{-1/2} \exp\left\{ -\frac{1}{2} \mathbf{y} \Sigma^{-1} \mathbf{y} \right\}, \tag{A.4}$$

where $\Sigma = A^{-1} + F(R^T B R)^{-1} F^T$. Also see [5, section 3]. However, this assumes that $R^T B R$ is invertible, which is not the case whenever $\mathrm{kernel}(R) \neq \{\mathbf{0}\}$.

Acknowledgment. The authors thank the anonymous reviewers for their helpful comments on an earlier draft of the manuscript.

REFERENCES

[1] B. Adcock, A. Gelb, G. Song, and Y. Sui, *Joint sparse recovery based on variances*, SIAM J. Sci. Comput., 41 (2019), pp. A246–A268, https://doi.org/10.1137/17M1155983.
[2] R. Archibald, A. Gelb, and R. B. Platte, *Image reconstruction from undersampled Fourier data using the polynomial annihilation transform*, J. Sci. Comput., 67 (2016), pp. 432–452.
[3] E. Artin, *The Gamma Function*, Dover Publications, Mineola, NY, 2015.
[4] S. D. Babacan, R. Molina, and A. K. Katsaggelos, *Parameter estimation in TV image restoration using variational distribution approximation*, IEEE Trans. Image Process., 17 (2008), pp. 326–339.
[5] S. D. Babacan, R. Molina, and A. K. Katsaggelos, *Sparse Bayesian image restoration*, in Proceedings of the 2010 IEEE International Conference on Image Processing, IEEE, 2010, pp. 3577–3580.
[6] J. M. Bardsley, *MCMC-based image reconstruction with uncertainty quantification*, SIAM J. Sci. Comput., 34 (2012), pp. A1316–A1332, https://doi.org/10.1137/11085760X.
[7] J. M. Bardsley, D. Calvetti, and E. Somersalo, *Hierarchical regularization for edge-preserving reconstruction of PET images*, Inverse Problems, 26 (2010), 035010.
[8] S. Boyd, N. Parikh, and E. Chu, *Distributed Optimization and Statistical Learning via the Alternating Direction Method of Multipliers*, Now Publishers, Delft, The Netherlands, 2011.
[9] 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), 125005.

[10] D. Calvetti, M. Pragliola, and E. Somersalo, *Sparsity promoting hybrid solvers for hierarchical Bayesian inverse problems*, SIAM J. Sci. Comput., 42 (2020), pp. A3761–A3784, https://doi.org/10.1137/20M1326246.
[11] 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), 025010.
[12] D. Calvetti and E. Somersalo, *A Gaussian hypermodel to recover blocky objects*, Inverse Problems, 23 (2007), 733.
[13] D. Calvetti and E. Somersalo, *An Introduction to Bayesian Scientific Computing: Ten Lectures on Subjective Computing*, Vol. 2, Springer, New York, 2007.
[14] D. Calvetti and E. Somersalo, *Hypermodels in the Bayesian imaging framework*, Inverse Problems, 24 (2008), 034013.
[15] D. Calvetti and E. Somersalo, *Subjective knowledge or objective belief? An oblique look to Bayesian methods*, Large-Scale Inverse Problems and Quantification of Uncertainty, Wiley Ser. Comput. Stat., Wiley, Chichester, 2011, pp. 33–70.
[16] D. Calvetti, E. Somersalo, and A. Strang, *Hierachical Bayesian models and sparsity: ℓ_2-magic*, Inverse Problems, 35 (2019), 035003.
[17] E. J. Candès, M. B. Wakin, and S. P. Boyd, *Enhancing sparsity by reweighted ℓ_1 minimization*, J. Fourier Anal. Appl., 14 (2008), pp. 877–905.
[18] G. Chantas, N. Galatsanos, A. Likas, and M. Saunders, *Variational Bayesian image restoration based on a product of t-distributions image prior*, IEEE Trans. Image Process., 17 (2008), pp. 1795–1805.
[19] G. K. Chantas, N. P. Galatsanos, and A. C. Likas, *Bayesian restoration using a new nonstationary edge-preserving image prior*, IEEE Trans. Image Process., 15 (2006), pp. 2987–2997.
[20] R. Chartrand and W. Yin, *Iteratively reweighted algorithms for compressive sensing*, in Proceedings of the 2008 IEEE International Conference on Acoustics, Speech and Signal Processing, 2008, pp. 3869–3872.
[21] I. Y. Chun and B. Adcock, *Compressed sensing and parallel acquisition*, IEEE Trans. Inform. Theory, 63 (2017), pp. 4860–4882.
[22] V. Churchill and A. Gelb, *Detecting edges from non-uniform Fourier data via sparse Bayesian learning*, J. Sci. Comput., 80 (2019), pp. 762–783.
[23] V. Churchill and A. Gelb, *Estimation and uncertainty quantification for piecewise smooth signal recovery*, J. Comput. Math., 41 (2023), pp. 246–262.
[24] D. Colton, M. Piana, and R. Potthast, *A simple method using Morozov's discrepancy principle for solving inverse scattering problems*, Inverse Problems, 13 (1997), pp. 1477–1493.
[25] I. Daubechies, R. DeVore, M. Fornasier, and C. S. Güntürk, *Iteratively reweighted least squares minimization for sparse recovery*, Comm. Pure Appl. Math., 63 (2010), pp. 1–38.
[26] D. L. Donoho, *Compressed sensing*, IEEE Trans. Inform. Theory, 52 (2006), pp. 1289–1306.
[27] Y. C. Eldar and G. Kutyniok, *Compressed Sensing: Theory and Applications*, Cambridge University Press, 2012.
[28] R. Fergus, B. Singh, A. Hertzmann, S. T. Roweis, and W. T. Freeman, *Removing camera shake from a single photograph*, ACM Trans.Graph., 25 (2006), pp. 787–794.
[29] M. A. Figueiredo, J. M. Bioucas-Dias, and R. D. Nowak, *Majorization-minimization algorithms for wavelet-based image restoration*, IEEE Trans. Image Process., 16 (2007), pp. 2980–2991.
[30] D. Fink, *A Compendium of Conjugate Priors*, 1997, available online at https://www.johndcook.com/CompendiumOfConjugatePriors.pdf.
[31] S. Foucart and H. Rauhut, *A mathematical introduction to compressive sensing*, Bull. Amer. Math. Soc. (N.S.), 54 (2017), pp. 151–165.
[32] J. Friedman, T. Hastie, and R. Tibshirani, *Regularization paths for generalized linear models via coordinate descent*, J. Stat. Softw., 33 (2010), pp. 1–22.
[33] A. Gelman, J. B. Carlin, H. S. Stern, and D. B. Rubin, *Bayesian data analysis*, 3rd ed., Chapman and Hall/CRC, Boca Raton, FL, 2021, http://www.stat.columbia.edu/∼gelman/book/.
[34] G. H. Golub, M. Heath, and G. Wahba, *Generalized cross-validation as a method for choosing a good ridge parameter*, Technometrics, 21 (1979), pp. 215–223.

[35] C. W. Groetsch and C. Groetsch, *Inverse Problems in the Mathematical Sciences*, Vol. 52, Vieweg+Teubner Verlag, Wiesbaden, 1993.
[36] M. Guerquin-Kern, L. Lejeune, K. P. Pruessmann, and M. Unser, *Realistic analytical phantoms for parallel magnetic resonance imaging*, IEEE Trans. Med. Imaging, 31 (2011), pp. 626–636.
[37] F. Gustafsson, *Statistical Sensor Fusion*, Studentlitteratur, 2010.
[38] P. C. Hansen, *The L-curve and its use in the numerical treatment of inverse problems*, Computational Inverse Problems in Electrocardiology, WT Press, 1999, pp. 119–142.
[39] P. C. Hansen, *Discrete Inverse Problems: Insight and Algorithms*, Fundam. Algorithms 7, SIAM, Philadelphia, 2010, https://doi.org/10.1137/1.9780898718836.
[40] P. C. Hansen, J. G. Nagy, and D. P. O'leary, *Deblurring Images: Matrices, Spectra, and Filterings*, Fundam. Algorithms 3, SIAM, Philadelphia, 2006, https://doi.org/10.1137/1.9780898718874.
[41] J. Kaipio and E. Somersalo, *Statistical and Computational Inverse Problems*, Appl. Math. Sci. 160, Springer-Verlag, New York, 2006.
[42] J. P. Kaipio, V. Kolehmainen, E. Somersalo, and M. Vauhkonen, *Statistical inversion and Monte Carlo sampling methods in electrical impedance tomography*, Inverse Problems, 16 (2000), 1487.
[43] D. Krishnan and R. Fergus, *Fast image deconvolution using hyper-Laplacian priors*, in Proceedings of the 22nd International Conference on Neural Information Processing Systems, 2009, pp. 1033–1041.
[44] A. Lanza, M. Pragliola, and F. Sgallari, *Residual whiteness principle for parameter-free image restoration*, Electron. Trans. Numer. Anal., 53 (2020), pp. 329–351.
[45] A. Levin, R. Fergus, F. Durand, and W. T. Freeman, *Image and depth from a conventional camera with a coded aperture*, ACM Trans. Graph., 26 (2007), pp. 70–es.
[46] D. J. MacKay, *Bayesian interpolation*, Neural Comput., 4 (1992), pp. 415–447.
[47] K. P. Murphy, *Conjugate Bayesian Analysis of the Gaussian Distribution*, 2007, https://www.cs.ubc.ca/∼murphyk/Papers/bayesGauss.pdf.
[48] F. Natterer and F. Wübbeling, *Mathematical Methods in Image Reconstruction*, SIAM Monogr. Math. Model. Comput., SIAM, Philadelphia, 2001, https://doi.org/10.1137/1.9780898718324.
[49] Y. Saad, *Iterative Methods for Sparse Linear Systems*, SIAM, Philadelphia, 2003, https://doi.org/10.1137/1.9780898718003.
[50] T. Sanders, R. B. Platte, and R. D. Skeel, *Effective new methods for automated parameter selection in regularized inverse problems*, Appl. Numer. Math., 152 (2020), pp. 29–48.
[51] H. Stark, *Image Recovery: Theory and Application*, Elsevier, 2013.
[52] W. Stefan, R. A. Renaut, and A. Gelb, *Improved total variation-type regularization using higher order edge detectors*, SIAM J. Imaging Sci., 3 (2010), pp. 232–251, https://doi.org/10.1137/080730251.
[53] A. N. Tikhonov, A. Goncharsky, V. Stepanov, and A. G. Yagola, *Numerical Methods for the Solution of Ill-Posed Problems*, Math. Appl. 328, Kluwer Academic Publishers, Dordrecht, 1995.
[54] M. E. Tipping, *Sparse Bayesian learning and the relevance vector machine*, J. Mach. Learn. Res., 1 (2001), pp. 211–244.
[55] C. R. Vogel, *Computational Methods for Inverse Problems*, Frontiers Appl. Math. 23, SIAM, Philadelphia, 2002, https://doi.org/10.1137/1.9780898717570.
[56] D. Wipf and S. Nagarajan, *A new view of automatic relevance determination*, in Proceedings of the 20th International Conference on Neural Information Processing Systems, 2007, pp. 1625–1632.
[57] D. P. Wipf and B. D. Rao, *Sparse Bayesian learning for basis selection*, IEEE Trans. Signal Process., 52 (2004), pp. 2153–2164.
[58] S. J.Wright, *Coordinate descent algorithms*, Math. Program., 151 (2015), pp. 3–34.
[59] Z. Zhang, T.-P. Jung, S. Makeig, Z. Pi, and B. D. Rao, *Spatiotemporal sparse Bayesian learning with applications to compressed sensing of multichannel physiological signals*, IEEE Trans. Neural Syst. Rehabil. Eng., 22 (2014), pp. 1186–1197.
[60] Z. Zhang and B. D. Rao, *Clarify Some Issues on the Sparse Bayesian Learning for Sparse Signal Recovery*, Tech. report, University of California, San Diego, San Diego, CA, 2011.
[61] Z. Zhang and B. D. Rao, *Sparse signal recovery with temporally correlated source vectors using sparse Bayesian learning*, IEEE J. Sel. Top. Signal Process., 5 (2011), pp. 912–926.