Bayesian Simultaneous Sparse Approximation With Smooth Signals

Martin Luessi, *Member, IEEE*, S. Derin Babacan, *Member, IEEE*, Rafael Molina, *Member, IEEE*, and Aggelos K. Katsaggelos, *Fellow, IEEE*

***Abstract*—In the simultaneous sparse approximation problem, several latent vectors corresponding to independent random realizations from a common sparsity profile are recovered from an undercomplete set of measurements. In this paper, we address an extension of this problem, where in addition to the common sparsity profile, the vectors of interest are assumed to have a high correlation among each other. Specifically, we consider the case when the non-zero rows in the combined latent signal vectors are considered to be temporally smooth signals. We present a Bayesian formulation of the problem and develop a greedy inference algorithm based on sparse Bayesian learning for independent observations. A difficulty is that unlike for existing greedy methods, there is no closed form expression for the maximizer of the objective function in the greedy algorithm when row correlations are introduced. We derive two methods to maximize the objective function, one based on the EM algorithm and another on a fixed-point iteration. Empirical results show that the proposed method provides better reconstruction results compared to existing methods, especially when the signal-to-noise ratio is low and the latent signal vectors are highly correlated. We also demonstrate the application of the proposed method to source localization in magnetoencephalography, where it obtains a temporally smooth solution with accurate localization of the brain activity.**

***Index Terms*—Bayesian methods, sparse Bayesian learning, sparse signal recovery, sparsity, MEG/EEG source localization.**

I. Introduction

SUPPOSE we have the following measurement system

$$\mathbf{y}_i = \mathbf{\Phi}\mathbf{w}_i + \boldsymbol{\eta}_i, \quad i = 1, \ldots, L, \tag{1}$$

where $\mathbf{y}_i$ are the measurements of length M, $\mathbf{w}_i$ is the latent vector of length N, $\boldsymbol{\eta}_i$ is the noise, and $\mathbf{\Phi} \in \mathbb{R}^{M \times N}$ is a known fixed forward operator. Depending on the application, this forward operator can be a dictionary (e.g., in the sparse representation problems [1]), or a measurement/degradation system (e.g., in compressed sensing [2], [3] or image deconvolution [4]).

The system in (1) can be jointly represented for all L latent vectors as

$$\mathbf{Y} = \mathbf{\Phi}\mathbf{W} + \boldsymbol{\eta}, \tag{2}$$

where $\mathbf{Y} \in \mathbb{R}^{M \times L}$ is the measurement matrix, and $\mathbf{W} \in \mathbb{R}^{N \times L}$ is the latent variable matrix. In general, the number of measurements is smaller than the number of latent variables in each vector $\mathbf{w}_i$ ($M \ll N$). In this case, the problem of estimating $\mathbf{W}$ from $\mathbf{Y}$ is ill-posed and additional constraints are necessary to render the solution unique. It is now well established that if the vectors $\mathbf{w}_i$ are sparse, then $\mathbf{W}$ can be recovered even when $M \ll N$ using ℓ_p-norm regularization with $p \leq 1$ [1].

An interesting special case of this problem is the one where the columns in $\mathbf{W}$ are jointly sparse, i.e., they have non-zero entries at the same locations. In this case, better reconstruction performance can be attained by exploiting the joint sparsity through solving an $\ell_p\ell_2$-norm regularized optimization problem

$$\hat{\mathbf{W}} = \arg\min_{\mathbf{W}} \|\mathbf{Y} - \mathbf{\Phi}\mathbf{W}\|_{\mathcal{F}}^2 + \tau \left(\sum_{i=1}^{N} \left(\|\mathbf{w}_{i\cdot}\|_2 \right)^p \right)^{\frac{1}{p}}, \tag{3}$$

where $\mathbf{w}_{i\cdot}$ denotes a column vector with the elements of the i-th row of $\mathbf{W}$, τ is a regularization parameter, and $\|\cdot\|_{\mathcal{F}}$ denotes the Frobenius norm. A number of algorithms have been proposed for solving this problem for $p \leq 1$; the most common ones are the M-OMP [5], [6] and the M-FOCUSS algorithms [7]. The problem also lends itself to an empirical Bayesian approach based on sparse Bayesian learning (SBL) [8], as in the M-SBL algorithm [9] and the multitask compressive sensing algorithm [10], which is based on greedy SBL [11]. The SBL and M-SBL methods model the rows of $\mathbf{W}$ using independent normal-gamma scale mixture priors [12]. Detailed information on sparse linear regression from a Bayesian perspective can be found in [13]. Note that (1) can be considered a latent factor model with a fixed loadings matrix. Sparse latent factor models have been developed in the statistics literature, e.g., in [14] where a model with sparsity-inducing "spike-and-slab" priors [15] is proposed. One advantage of normal-gamma priors is that they allow for computationally efficient inference whereas "spike-and-slab" priors require sampling methods, which can

Manuscript received March 29, 2011; revised June 17, 2012, November 28, 2012, and March 28, 2013; accepted August 21, 2013. Date of publication September 05, 2013; date of current version October 16, 2013. The associate editor coordinating the review of this manuscript and approving it for publication was Prof. Sofia Charlotta Olhede. This work was partially supported by the NSFGrant 0958669, U.S. Department of Energy Grant DENA0000457, by the "Ministerio de Ciencia e Innovación" under contract TIN2010-15137, and the CEI BioTic at the Universidad de Granada.

M. Luessi was with the Department of Electrical Engineering and Computer Science, Northwestern University, Evanston, IL 60201 USA. He is now with the Department of Radiology, Athinoula A. Martinos Center for Biomedical Imaging, Massachusetts General Hospital, Harvard Medical School, Boston, MA 02138 USA (e-mail: mluessi@nmr.mgh.harvard.edu).

S. D. Babacan is with Google, Inc., Mountain View, CA 94043 USA (e-mail: dbabacan@gmail.com).

R. Molina is with the Departamento de Ciencias de la Computación e I.A. Universidad de Granada, Granada 18071, Spain (e-mail: rms@decsai.ugr.es).

A. K. Katsaggelos is with the Department of Electrical Engineering and Computer Science, Northwestern University, Evanston, IL 60201 USA (e-mail: aggk@eecs.northwestern.edu).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TSP.2013.2280441

1053-587X © 2013 IEEE

be prohibitive for higher dimensions. An in depth discussion of normal-gamma and "spike-and-slab" priors and the properties of the resulting posterior distributions can be found in [16].

While methods based on (3) induce row-sparsity in $\mathbf{W}$, i.e., most rows are all zero, and obtain a solution with few active rows, they do not fully exploit all prior information about $\mathbf{W}$. Namely, in many problems the coefficients in a row may be strongly correlated. For instance, if $\mathbf{w}_{i\cdot}$ are signals varying over time, information about the nature of change over time can be exploited in the reconstruction. Source localization in electroencephalography (EEG) and magnetoencephalography (MEG) [17] falls into this category. Specifically, in the M/EEG source localization problem, each $\mathbf{w}_{i\cdot}$ is a temporally smooth signal, as it corresponds to the time course of an equivalent electrical current representing the activity of a group of neurons. Temporal correlations play an important role when analyzing neuronal activity and in [18] a factor analysis (FA) method is proposed to summarize multiunit recordings of neuronal activity where temporal smoothness is imposed using Gaussian process priors. Other related problems are source localization [19], [20], and distributed compressed sensing [21], where neighboring elements in $\mathbf{w}_{i\cdot}$ are highly correlated and taking these correlations into account is of high importance towards more effective reconstruction.

In this paper, we propose a recovery algorithm which exploits this kind of dependencies. Specifically, in addition to row-sparsity, we utilize the *a priori* knowledge that non-zero rows in the coefficient matrix $\mathbf{W}$ correspond to smooth waveforms. The goal herein is to enforce row-sparsity and to penalize non-smooth solutions during the reconstruction. Although we focus on the smoothness constraint, the proposed formulation is very flexible and can be used for the recovery of latent signals with an arbitrary intra-row correlation structure. Moreover, notice that (3) can be considered a special case of a structured sparsity formulation [22], [23], where arbitrary groups of latent variables are assumed to be jointly sparse. Similarly, our formulation could be generalized to model correlations among arbitrary groups of latent variables.

Following ideas from sparse Bayesian learning [8], we employ a Bayesian formulation of the problem and first develop a global method in which we obtain an approximation to the posterior distribution of all unknowns based on the evidence procedure [24], also known as empirical Bayes [25], type-II maximum likelihood approach [26], and generalized maximum likelihood approach [27]. However, as this method ends up to be computationally very demanding, we derive a greedy (constructive) inference scheme which is computationally more efficient and lends itself to a parallel execution on distributed computing systems such as clusters.

We demonstrate the performance of the proposed method by means of an empirical evaluation with simulated data. We show that if we know that the latent signals are smooth, we can use the proposed method to penalize non-smooth solutions, resulting in a significantly lower reconstruction error, especially when the signal-to-noise ratio is low. In a second experiment, we demonstrate the ability of the method to recover latent signals generated by an autoregressive process, which is commonly used to model signals in numerous signal processing applications (e.g., speech and audio processing, seismic signal processing). This experiment demonstrates the flexibility of the proposed method as it can be used for the recovery of a broad class of signals if we have prior information about their correlation structure. Finally, to demonstrate the application to a particular problem by applying the proposed method to magnetoencephalography (MEG) data, where it accurately localizes the activity and obtains a solution with smooth waveforms.

This paper is organized as follows. The Bayesian model underlying the proposed method is presented in Section II. In Section III we derive the greedy Bayesian inference algorithm. Results from the empirical evaluation with simulated data are presented in Section IV. In Section V, we demonstrate the application of the proposed method to MEG source localization. Finally, the paper is summarized and conclusions are drawn in Section VI.

Throughout this work we use boldface capital letters to denote matrices, while $\mathbf{a}_{i\cdot}$ and $\mathbf{a}_{\cdot i}$ denote column vectors with the contents of the i-th row and the i-th column of matrix $\mathbf{A}$, respectively. Similarly, the i-th element of vector $\mathbf{a}$ is denoted by a_i.

II. Bayesian Modeling

We assume that the observation noise $\boldsymbol{\eta}$ is zero-mean independent identically distributed (i.i.d.) Gaussian with precision $\beta = 1/\sigma^2$, where σ^2 is the variance of the noise. Therefore, the conditional distribution of the observations is Gaussian

$$\mathrm{p}(\mathbf{Y}|\mathbf{W},\beta) = \prod_{i=1}^{L} \mathcal{N}(\mathbf{y}_{\cdot i}|\boldsymbol{\Phi}\mathbf{w}_{\cdot i}, \beta^{-1}\mathbf{I}). \tag{4}$$

We place a gamma prior on β, i.e.,

$$\mathrm{p}(\beta) = \left(\frac{b^a}{\Gamma(a)}\right)\beta^{a-1}\exp(-b\beta), \tag{5}$$

where $\Gamma(\cdot)$ denotes the gamma function and a and b are the shape and scale parameters, respectively. We proceed by assigning a singular multinormal distribution (see [28]) as a prior to each row in $\mathbf{W}$. The prior for the i-th row is given by

$$\mathrm{p}(\mathbf{w}_{i\cdot}|\gamma_i) = \frac{\left(\prod_{j=1}^{R}\lambda_j\right)^{-\frac{1}{2}}}{(2\pi\gamma_i)^{\frac{R}{2}}}\exp\left(-\frac{1}{2\gamma_i}\mathbf{w}_{i\cdot}^T\mathbf{P}^{+}\mathbf{w}_{i\cdot}\right), \tag{6}$$

where $\mathbf{P}^{+}$ is the generalized inverse of a known $L \times L$ matrix $\mathbf{P}$ with rank R and non-zero eigenvalues $\{\lambda_1, \ldots, \lambda_R\}$ and γ_i is an unknown variance hyperparameter. The motivation for using a singular multinormal distribution instead of a multivariate Gaussian distribution is that $\mathbf{P}$ is not restricted to be full rank. For the case when $\mathbf{P}$ is full rank, i.e., $R = L$, (6) becomes a multivariate Gaussian distribution, i.e., $\mathrm{p}(\mathbf{w}_{i\cdot}|\gamma_i) = \mathcal{N}(\mathbf{w}_{i\cdot}|\mathbf{0}, \gamma_i\mathbf{P})$. Note that this prior is very similar to the Gaussian process prior used in [18] to impose temporal smoothness in factor analysis. By combining the priors of all rows we obtain

$$\mathrm{p}(\mathbf{W}|\boldsymbol{\gamma}) = \prod_{i=1}^{N}\mathrm{p}(\mathbf{w}_{i\cdot}|\gamma_i), \tag{7}$$

where $\boldsymbol{\gamma}$ is a vector containing all N variance hyperparameters. Notice that for $\mathbf{P} = \mathbf{I}$ this model is equivalent to the M-SBL formulation [9]. Our approach is more flexible as arbitrary positive semi-definite matrices can be used for $\mathbf{P}$. One option is to assume that the signals are smooth and use $\mathbf{P}^+ = \mathbf{T}^T\mathbf{T}$, where $\mathbf{T}$ is a matrix implementing a discrete second order derivative operator. Another option is to obtain the $\mathbf{P}$ matrix using prior information about the process which generated the signals (an example where the signals are generated by an autoregressive process is shown in Section IV-B). If we do not have precise information about the generative process, it could still be the case that we know that latent variables in only certain subsets of column locations are highly correlated. This information can be used to construct a custom $\mathbf{P}$ matrix which models our knowledge about correlations between columns for the problem at hand.

Finally, we assign priors to the variance hyperparameters γ_i. When the modeling is performed in terms of precision hyperparameters $\alpha_i = (\gamma_i)^{-1}$, a common choice is a gamma prior $\mathrm{p}(\alpha_i|a_\alpha, b_\alpha) = \Gamma(\alpha_i|a_\alpha, b_\alpha)$, resulting in a normal-gamma scale mixture prior for the rows of $\mathbf{W}$. In [8] the hyperparameter optimization is performed in the logarithmic domain. In this case, the prior for α_i can be made noninformative (uniform in the logarithmic domain) by using $a_\alpha = b_\alpha = 0$. As pointed out in [8], a noninformative prior has the advantage that it is invariant to scaling of $\mathbf{P}$, $\boldsymbol{\Phi}$, and $\mathbf{Y}$. Here, we use a uniform prior for γ_i. While this prior is improper, we found that it works well in practice. Note that while this prior is scale invariant, i.e., we do not need to adjust the prior parameters when the scale of the observations or the matrix $\mathbf{P}$ changes, one problem encountered in practice is that γ_i can attain very small values, which causes numerical problems as some matrices become ill-conditioned. We address this problem by heuristically assuming $\gamma_i = 0$ if the value is small enough for the corresponding signal to be zero for all practical purposes; this issue is further explained in Section III-A.

III. Bayesian Inference

In this section we develop an efficient Bayesian inference algorithm based on the model introduced in the previous section. We wish to draw inference based on the posterior distribution given by

$$\mathrm{p}(\mathbf{W}, \boldsymbol{\gamma}, \beta|\mathbf{Y}) = \frac{\mathrm{p}(\mathbf{Y}|\mathbf{W}, \beta)\mathrm{p}(\mathbf{W}|\boldsymbol{\gamma})\mathrm{p}(\beta)\mathrm{p}(\boldsymbol{\gamma})}{\mathrm{p}(\mathbf{Y})}. \quad (8)$$

However, estimating the full posterior is intractable since

$$\mathrm{p}(\mathbf{Y}) = \iiint \mathrm{p}(\mathbf{W}, \boldsymbol{\gamma}, \beta, \mathbf{Y}) d\mathbf{W} d\boldsymbol{\gamma} d\beta \quad (9)$$

cannot be calculated analytically. Numerous methods have been developed to address this difficulty. In this work, we adopt the evidence procedure [24] to obtain an approximation to the posterior. This procedure is based on the following posterior distribution decomposition

$$\mathrm{p}(\mathbf{W}, \boldsymbol{\gamma}, \beta|\mathbf{Y}) = \mathrm{p}(\mathbf{W}|\mathbf{Y}, \boldsymbol{\gamma}, \beta)\mathrm{p}(\boldsymbol{\gamma}, \beta|\mathbf{Y}). \quad (10)$$

The distribution $\mathrm{p}(\mathbf{W}|\mathbf{Y}, \boldsymbol{\gamma}, \beta)$ is found to be Gaussian $\mathcal{N}(\mathrm{vec}(\mathbf{W})|\boldsymbol{\mu}, \boldsymbol{\Sigma})$ with

$$\boldsymbol{\mu} = \mathrm{vec}\left(\langle \mathbf{W} \rangle\right) = \beta \boldsymbol{\Sigma} \boldsymbol{\Psi}^T \mathrm{vec}(\mathbf{Y}), \quad (11)$$

$$\boldsymbol{\Sigma} = \left(\beta \boldsymbol{\Psi}^T \boldsymbol{\Psi} + \boldsymbol{\Lambda}\right)^{-1}, \quad (12)$$

where the operator $\mathrm{vec}(\cdot)$ vectorizes a matrix by stacking its columns and $\langle \cdot \rangle$ denotes the expectation operator. The matrices $\boldsymbol{\Psi} \in \mathbb{R}^{ML \times LN}$ and $\boldsymbol{\Lambda} \in \mathbb{R}^{LN \times LN}$ are given by

$$\boldsymbol{\Psi} = (\mathbf{I}_L \otimes \boldsymbol{\Phi}), \quad \boldsymbol{\Lambda} = \left(\mathbf{P}^+ \otimes \mathrm{Diag}(\boldsymbol{\gamma})^{-1}\right), \quad (13)$$

where $\otimes$ denotes the Kronecker product, $\mathrm{Diag}(\boldsymbol{\gamma})$ is a diagonal matrix with $\boldsymbol{\gamma}$ on its main diagonal, and $\mathbf{I}_L$ denotes the $L \times L$ identity matrix. The distribution $\mathrm{p}(\boldsymbol{\gamma}, \beta, \mathbf{Y}) \propto \mathrm{p}(\boldsymbol{\gamma}, \beta|\mathbf{Y})$ is obtained from $\mathrm{p}(\boldsymbol{\gamma}, \beta, \mathbf{W}, \mathbf{Y})$ by marginalizing over $\mathbf{W}$, i.e.,

$$\begin{aligned}\mathrm{p}(\boldsymbol{\gamma}, \beta, \mathbf{Y}) &= \int \mathrm{p}(\mathbf{Y}|\mathbf{W}, \beta)\mathrm{p}(\mathbf{W}|\boldsymbol{\gamma})\mathrm{p}(\boldsymbol{\gamma})\mathrm{p}(\beta) d\mathbf{W} \\ &= \mathcal{N}\left(\mathrm{vec}(\mathbf{Y})|\mathbf{0}, \mathbf{C}\right)\mathrm{p}(\beta), \end{aligned} \quad (14)$$

and $\mathbf{C}$ is given by

$$\mathbf{C} = \beta^{-1}\mathbf{I}_{ML} + \boldsymbol{\Psi}\boldsymbol{\Lambda}^{-1}\boldsymbol{\Psi}^T. \quad (15)$$

In order to draw inference we wish to maximize $\mathrm{p}(\boldsymbol{\gamma}, \beta, \mathbf{Y})$ with respect to the hyperparameters $\{\beta, \boldsymbol{\gamma}\}$, which is equivalent to maximizing the logarithm of the distribution, given by

$$\mathcal{L} = -\frac{1}{2}\log|\mathbf{C}| - \frac{1}{2}\mathrm{vec}(\mathbf{Y})^T\mathbf{C}^{-1}\mathrm{vec}(\mathbf{Y}) + (a-1)\log\beta - b\beta. \quad (16)$$

To maximize this objective function, we calculate the derivative with respect to each hyperparameter and set it to zero, resulting in

$$\gamma_i = \frac{\langle \mathbf{w}_{i\cdot} \rangle^T \mathbf{P}^+ \langle \mathbf{w}_{i\cdot} \rangle + \mathrm{tr}\left(\mathrm{cov}(\mathbf{w}_{i\cdot})\mathbf{P}^+\right)}{R}, \quad (17)$$

$$\beta = \frac{\frac{ML}{2} + a - 1}{\frac{1}{2}\left\|\mathrm{vec}(\mathbf{Y}) - \boldsymbol{\Psi}\boldsymbol{\mu}\right\|_2^2 + b}, \quad (18)$$

where $\mathrm{cov}(\mathbf{w}_{i\cdot})$ is the covariance matrix of the i-th row of $\mathbf{W}$ (this matrix can easily be extracted from $\boldsymbol{\Sigma}$, i.e., $\mathrm{cov}(\mathbf{w}_{i\cdot})_{r,s} = \boldsymbol{\Sigma}_{i+(r-1)L, i+(s-1)L}$). Detailed derivations for the maximization with respect to γ_i is given in Appendix A.

We note here that the noise precision estimate can be extremely inaccurate due to a identifiability problem [9] and it is generally preferable to use a fixed β, which has been estimated by other means or is known *a priori*. Using the above update rules we can obtain a simple iterative Bayesian inference algorithm. At each iteration, the algorithm updates the distribution $\mathrm{p}(\mathbf{W}|\mathbf{Y}, \boldsymbol{\gamma}, \beta)$ using (11) and (12) and then maximizes $\mathrm{p}(\boldsymbol{\gamma}, \beta|\mathbf{Y})$ with respect to the hyperparameters $\boldsymbol{\gamma}$ and (optionally) β using (17) and (18), respectively. Notice that the maximization with respect to $\boldsymbol{\gamma}$ and β in the second step of the algorithm corresponds to the M-step of the expectation maximization (EM) algorithm [29] with $\mathbf{W}$ being treated as a hidden variable.

For large-scale problems the EM algorithm may converge slowly and a faster algorithm can be obtained by replacing the M-step with an update rule that is obtained by equating the

derivative to zero and forming a fixed-point equation, a method known as the MacKay update [9], [24]. However, using the faster MacKay update does not solve a fundamental problem with the described algorithm: The algorithm requires the inversion of an $NL \times NL$ matrix at each iteration in order to calculate $\mathbf{\Sigma}$ in (12). This problem can be somewhat alleviated by using the Woodbury identity to solve (12), but the algorithm still requires the inversion of an $ML \times ML$ matrix, which can be large in practice. Another option is to neglect the covariance component in (17) and solve (11) using an efficient linear system solver, such as the conjugate gradient algorithm. However, due to omitting the covariance component this method is potentially less accurate and it still does not scale well to large-scale problems.

The scalability problem is not as severe in the M-SBL algorithm [9] where $\mathbf{P} = \mathbf{I}$, as in this case $\mathbf{\Sigma}^{-1}$ is a block diagonal matrix with identical $N \times N$ blocks. Therefore, $\mathbf{\Sigma}$ can be computed using a single $N \times N$ inversion, or even a single $M \times M$ inversion when the Woodbury identity is used. To obtain a computationally tractable inference algorithm for large-scale problems when $\mathbf{P} \neq \mathbf{I}$, we exploit the row sparsity of $\mathbf{W}$ and in the following develop a greedy inference algorithm which only includes signals in the model with $\gamma_i > 0$.

A. Greedy Inference

Row sparsity in $\mathbf{W}$ implies that most γ_i will be zero since $\mathbf{w}_{i\cdot} = \mathbf{0} \Rightarrow \gamma_i = 0$. A signal with $\gamma_i = 0$ is equivalent to removing it from the model by removing the i-th column of $\mathbf{\Phi}$, the i-th row of $\mathbf{W}$, and the i-th element of $\boldsymbol{\gamma}$. This observation can be exploited by starting with a full model and removing signals as the corresponding γ_i approach zero, which leads to an acceleration of the algorithm since N becomes smaller as the algorithm progresses. However, for large-scale problems this approach is still computationally infeasible as the algorithm has to start with a full model. Therefore, we go the opposite route and develop a greedy algorithm which starts with an empty model ($\boldsymbol{\gamma} = \mathbf{0}$) and only adds signals for which $\gamma_i > 0$. Doing so leads to significant computational advantages as the size of the matrix $\mathbf{\Sigma}$ becomes $Ln \times Ln$, where n denotes the number of signals currently included in the model.

To develop the greedy algorithm, first note that we can write $\mathbf{C}$ as

$$\begin{aligned}\mathbf{C} &= \beta^{-1}\mathbf{I}_{ML} + \mathbf{\Psi}\mathbf{\Lambda}^{-1}\mathbf{\Psi}^T \\ &= \beta^{-1}\mathbf{I}_{ML} + (\mathbf{I}_L \otimes \mathbf{\Phi})\,(\mathbf{P} \otimes \mathrm{Diag}(\boldsymbol{\gamma}))\,(\mathbf{I}_L \otimes \mathbf{\Phi}^T) \\ &= \beta^{-1}\mathbf{I}_{ML} + \sum_{i=1}^{N} \gamma_i \mathbf{Q}_i \mathbf{Q}_i^T = \mathbf{C}_{-i} + \gamma_i \mathbf{Q}_i \mathbf{Q}_i^T, \end{aligned} \tag{19}$$

where we decompose $\mathbf{P}$ as $\mathbf{P} = \mathbf{V}\mathbf{V}^T$ using its eigenvalue decomposition and the $ML \times L$ matrix $\mathbf{Q}_i$ is defined as $\mathbf{Q}_i = (\mathbf{I}_L \otimes \mathbf{\Phi}_{\cdot i})\mathbf{V}$. The notation $\mathbf{C}_{-i}$ denotes that the contribution from the i-th row has not been included. This allows the application of the Woodbury identity to obtain

$$\mathbf{C}^{-1} = \mathbf{C}_{-i}^{-1} - \mathbf{C}_{-i}^{-1}\mathbf{Q}_i \left(\gamma_i^{-1}\mathbf{I}_L + \mathbf{Q}_i^T \mathbf{C}_{-i}^{-1}\mathbf{Q}_i\right)^{-1} \mathbf{Q}_i^T \mathbf{C}_{-i}^{-1}. \tag{20}$$

Additionally using the determinant identity [28] we have

$$|\mathbf{C}| = |\mathbf{C}_{-i}|\left|\mathbf{I}_L + \gamma_i \mathbf{Q}_i^T \mathbf{C}_{-i}^{-1}\mathbf{Q}_i\right|. \tag{21}$$

Using (20) and (21), the objective function $\mathcal{L}$ in (16) can be written as

$$\begin{aligned}\mathcal{L}(\boldsymbol{\gamma}) =& -\frac{1}{2}\left[\log|\mathbf{C}_{-i}| + \mathrm{vec}(\mathbf{Y})^T \mathbf{C}_{-i}^{-1}\mathrm{vec}(\mathbf{Y})\right] \\ &+ \frac{1}{2}\left[\log\frac{1}{\left|\mathbf{I}_L + \gamma_i \mathbf{Q}_i^T \mathbf{C}_{-i}^{-1}\mathbf{Q}_i\right|} + \mathrm{vec}(\mathbf{Y})^T \mathbf{C}_{-i}^{-1}\mathbf{Q}_i \right. \\ &\left. \times \left(\gamma_i^{-1}\mathbf{I}_L + \mathbf{Q}_i^T \mathbf{C}_{-i}^{-1}\mathbf{Q}_i\right)^{-1} \mathbf{Q}_i^T \mathbf{C}_{-i}^{-1}\mathrm{vec}(\mathbf{Y})\right] \\ =& \mathcal{L}(\boldsymbol{\gamma}_{-i}) + \ell(\gamma_i), \end{aligned} \tag{22}$$

i.e., $\mathcal{L}(\boldsymbol{\gamma})$ is decomposed into $\mathcal{L}(\boldsymbol{\gamma}_{-i})$, the objective function for all signals except the i-th signal and $\ell(\gamma_i)$ which is the contribution of the i-th signal, given by

$$\begin{aligned}\ell(\gamma_i) =& \frac{1}{2}\left[\log\frac{1}{\left|\mathbf{I}_L + \gamma_i \mathbf{Q}_i^T \mathbf{C}_{-i}^{-1}\mathbf{Q}_i\right|} + \mathrm{vec}(\mathbf{Y})^T \mathbf{C}_{-i}^{-1}\mathbf{Q}_i \right. \\ &\left. \times \left(\gamma_i^{-1}\mathbf{I}_L + \mathbf{Q}_i^T \mathbf{C}_{-i}^{-1}\mathbf{Q}_i\right)^{-1} \mathbf{Q}_i^T \mathbf{C}_{-i}^{-1}\mathrm{vec}(\mathbf{Y})\right]. \end{aligned} \tag{23}$$

This decomposition of the objective function enables us to calculate the improvement in the objective when only one γ_i is modified and all others are held constant. This is useful since we wish to obtain a greedy algorithm, by choosing to modify only the γ_i which leads to the greatest increase in $\log \mathrm{p}(\boldsymbol{\gamma}, \beta, \mathbf{Y})$ during each iteration.

Due to the form of $\ell(\gamma_i)$ there is no closed form expression for the maximizer. Furthermore, the objective function may not be concave, i.e., there may be multiple maxima. In the following we introduce two methods to maximize this objective function. The first method uses an auxiliary probabilistic model together with the Expectation Maximization (EM) algorithm [29]. While this method benefits from the convergence guarantees of the EM algorithm, it is slow in practice as it requires the inversion of an $L \times L$ matrix in every iteration of the EM algorithm. We therefore propose a second maximization method which is based on a fixed-point iteration. The advantage of this method is that it converges considerably faster than the EM algorithm. While we have no proof for the convergence properties of the method, we empirically find that it obtains the same result as the EM algorithm; a comparison of the convergence properties is shown in Section IV-C.

1) Maximization Using EM: To maximize the objective function using the EM algorithm we introduce an auxiliary probabilistic model for which the γ_i maximizing the marginal likelihood also corresponds to the maximizer of $\ell(\gamma_i)$. The probabilistic model is given by

$$\mathrm{p}(\mathbf{z}|\gamma_i) = \frac{\left(\prod_{j=1}^{R}\lambda_j\right)^{-\frac{1}{2}}}{(2\pi\gamma_i)^{\frac{R}{2}}} \exp\left(-\frac{1}{2\gamma_i}\mathbf{z}^T\mathbf{P}^{+}\mathbf{z}\right), \tag{24}$$

$$\mathrm{p}(\mathbf{Y}|\mathbf{z}) = \mathcal{N}\left(\mathrm{vec}(\mathbf{Y})|(\mathbf{I}_L \otimes \mathbf{\Phi}_{\cdot i})\mathbf{z}, \mathbf{C}_{-i}\right), \tag{25}$$

where the vector $\mathbf{z} \in \mathbb{R}^L$. Using this model, we can obtain $\mathrm{p}(\mathbf{Y}|\gamma_i)$ by marginalizing $\mathbf{z}$, i.e.,

$$\begin{aligned}\mathrm{p}(\mathbf{Y}|\gamma_i) &= \int \mathrm{p}\left(\mathrm{vec}(\mathbf{Y})|\mathbf{z}\right)\mathrm{p}(\mathbf{z}|\gamma_i)d\mathbf{z} \\ &= \mathcal{N}\left(\mathrm{vec}(\mathbf{Y})|\mathbf{0}, \mathbf{C}_{-i} + \gamma_i \mathbf{Q}_i \mathbf{Q}_i^T\right). \end{aligned} \tag{26}$$

The estimate γ_i^* can be found by maximizing the marginal likelihood function of the auxiliary probabilistic model

$$\begin{aligned}\gamma_i^* &= \arg\max_{\gamma_i} \mathrm{p}(\mathbf{Y}|\gamma_i) \\ &= \arg\max_{\gamma_i}\Bigg[\log\frac{1}{\left|\mathbf{I}_L + \gamma_i \mathbf{Q}_i^T \mathbf{C}_{-i}^{-1}\mathbf{Q}_i\right|} + \mathrm{vec}(\mathbf{Y})^T\mathbf{C}_{-i}^{-1}\mathbf{Q}_i \\ &\qquad \times\left(\gamma_i^{-1}\mathbf{I}_L + \mathbf{Q}_i^T\mathbf{C}_{-i}^{-1}\mathbf{Q}_i\right)^{-1}\mathbf{Q}_i^T\mathbf{C}_{-i}^{-1}\mathrm{vec}(\mathbf{Y})\Bigg], \end{aligned} \tag{27}$$

from which it is clear that $\gamma_i^* = \arg\max_{\gamma_i} \ell(\gamma_i)$. However, instead of maximizing the marginal likelihood function directly we can now employ the EM algorithm together with the introduced auxiliary probabilistic model to find γ_i^*. In order to do so, we treat $\mathbf{z}$ as a hidden variable and apply the EM algorithm. In the E-step during the k-th iteration of the EM algorithm we use the current hyperparameter estimate γ_i^k to update the sufficient statistics of $\mathrm{p}(\mathbf{z}|\mathbf{Y}, \gamma_i^k) = \mathcal{N}(\mathbf{z}|\boldsymbol{\mu}_\mathbf{z}^k, \boldsymbol{\Sigma}_\mathbf{z}^k)$ where

$$\boldsymbol{\Sigma}_\mathbf{z}^k = \left((\mathbf{I}_L \otimes \boldsymbol{\Phi}_{\cdot i})^T\mathbf{C}_{-i}^{-1}(\mathbf{I}_L \otimes \boldsymbol{\Phi}_{\cdot i}) + \left(\gamma_i^k\right)^{-1}\mathbf{P}^+\right)^{-1}, \tag{28}$$

$$\boldsymbol{\mu}_\mathbf{z}^k = \boldsymbol{\Sigma}_\mathbf{z}(\mathbf{I}_L \otimes \boldsymbol{\Phi}_{\cdot i})^T\mathbf{C}_{-i}^{-1}\mathrm{vec}(\mathbf{Y}). \tag{29}$$

In the proceeding M-step we obtain γ_i^{k+1} by maximizing the expectation of complete-data log-likelihood under $\mathrm{p}(\mathbf{z}|\mathbf{Y}, \gamma_i^k)$ with respect to γ_i

$$\begin{aligned}\gamma_i^{k+1} &= \arg\max_{\gamma_i}\left[\mathbf{E}_{\mathrm{p}\left(\mathbf{z}|\mathbf{Y},\gamma_i^k\right)}\left[\log \mathrm{p}(\mathbf{z}, \mathbf{Y}|\gamma_i)\right]\right] \\ &= \frac{\left(\boldsymbol{\mu}_\mathbf{z}^k\right)^T\mathbf{P}^+\left(\boldsymbol{\mu}_\mathbf{z}^k\right) + \mathrm{tr}\left(\boldsymbol{\Sigma}_\mathbf{z}^k\mathbf{P}^+\right)}{R}, \end{aligned} \tag{30}$$

where the maximization is performed by equating the derivative with respect to γ_i to zero. To summarize, the EM algorithm consists of the E-step where $\boldsymbol{\Sigma}_\mathbf{z}$ and $\boldsymbol{\mu}_\mathbf{z}$ are updated using (28) and (29), respectively, followed by the M-step where the expectation of complete-data log-likelihood under $\mathrm{p}(\mathbf{z}|\mathbf{Y}, \gamma_i^k)$ with respect to γ_i is maximized by applying (30). The interleaved E-steps and M-steps are repeated until a convergence criterion, e.g., $|\gamma_i^k - \gamma_i^{k-1}|/\gamma_i^{k-1} \leq \epsilon$, is satisfied. While the EM algorithm is guaranteed to converge to a (local) maximum of $\ell(\gamma_i)$, it is computationally expensive as it requires the inversion of an $L \times L$ matrix to calculate $\boldsymbol{\Sigma}_\mathbf{z}^k$ in (28) at every iteration.

2) Maximization Using Fixed-Point Iteration: Due to the computational burden of the EM algorithm, we propose a computationally efficient iterative procedure based on a fixed-point iteration for the maximization of $\ell(\gamma_i)$. First, note that we can use the eigendecomposition of a symmetric positive semi-definite matrix to obtain

$$\mathbf{U}\mathbf{D}\mathbf{U}^T = \mathbf{Q}_i^T\mathbf{C}_{-i}^{-1}\mathbf{Q}_i, \tag{31}$$

where $\mathbf{U}$ is a matrix of size $L \times K$ with orthonormal columns and $\mathbf{D}$ is a diagonal $K \times K$ matrix with positive eigenvalues $\lambda_1, \ldots, \lambda_K$ on its main diagonal. Furthermore, we define

$$\mathbf{a} = \mathbf{U}^T\mathbf{Q}_i^T\mathbf{C}_{-i}^{-1}\mathrm{vec}(\mathbf{Y}). \tag{32}$$

Using the properties of the determinant and the matrix inverse together with (31) and (32) we can write the objective function (23) as

$$\ell(\gamma_i) = -\frac{1}{2}\sum_{k=1}^K \log(1 + \gamma_i\lambda_k) + \frac{1}{2}\sum_{k=1}^K \frac{a_k^2\gamma_i}{1 + \gamma_i\lambda_k}. \tag{33}$$

To find the stationary points of (33) we calculate its derivative with respect to γ_i and equate it to zero, to obtain

$$\frac{d\ell(\gamma_i)}{d\gamma_i} = \frac{1}{2}\sum_{k=1}^K \frac{-\gamma_i\lambda_k^2 - \lambda_k + a_k^2}{(1 + \gamma_i\lambda_k)^2} = 0. \tag{34}$$

Clearly, one way of finding the extrema of $\ell(\gamma_i)$ is solving (34) analytically. However, doing so requires finding the roots of a polynomial of order $2(K-1)+1$, which is computationally prohibitive. Hence, we propose a more efficient method for solving (34). First, note that we can write (34) as

$$\gamma_i\sum_{l=1}^K \frac{\lambda_l^2}{(1 + \gamma_i\lambda_l)^2} = \sum_{k=1}^K \frac{\lambda_k^2}{(1 + \gamma_i\lambda_k)^2}\left[\frac{a_k^2}{\lambda_k^2} - \frac{1}{\lambda_k}\right], \tag{35}$$

and therefore

$$\gamma_i = \sum_{k=1}^K \frac{\frac{\lambda_k^2}{(1+\gamma_i\lambda_k)^2}}{\sum_{l=1}^K \lambda_l^2/(1 + \gamma_i\lambda_l)^2}\left[\frac{a_k^2}{\lambda_k^2} - \frac{1}{\lambda_k}\right]. \tag{36}$$

Using this we obtain an iterative procedure in which the value γ_i^{old} obtained in the previous iteration is used to calculate the value for the current iteration as follows

$$\gamma_i^{\mathrm{new}} = \sum_{k=1}^K \mu_k\left(\gamma_i^{\mathrm{old}}\right)\left[\frac{a_k^2}{\lambda_k^2} - \frac{1}{\lambda_k}\right], \tag{37}$$

where

$$\mu_k\left(\gamma_i^{\mathrm{old}}\right) = \frac{\frac{\lambda_k^2}{\left(1+\gamma_i^{\mathrm{old}}\lambda_k\right)^2}}{\sum_{l=1}^K \lambda_l^2/\left(1 + \gamma_i^{\mathrm{old}}\lambda_l\right)^2}. \tag{38}$$

The maximizer of $\ell(\gamma_i)$ is found when the iterative procedure converges; we use $|\gamma_i^{\mathrm{new}} - \gamma_i^{\mathrm{old}}|/\gamma_i^{\mathrm{old}} \leq \epsilon$ with $\epsilon = 10^{-6}$ to decide whether the procedure has converged throughout this work. This procedure is computationally very efficient. After the eigendecomposition in (31), which typically has a computational complexity of $O(L^3)$, every iteration is of complexity

$O(K)$ and the procedure converges in a smaller number of iterations than the EM algorithm (an empirical comparison of the methods is shown in Section IV-C). Notice, however, that this is an *ad hoc* maximization procedure for which convergence has not been established.

Note that both methods, the EM algorithm and the fixed-point iteration, will converge to a local maximum of $\ell(\gamma_i)$. Hence, the value of the maximizer found depends on the starting point γ_i^0 used. One option is to find the smallest positive maximizer by starting the procedure with a very small positive value (e.g., $\gamma_i^0 = 10^{-9}$). A better performance can be achieved when a multi-start method is employed, i.e., the maximization is performed multiple times with a different starting point for each repetition and the maximizer corresponding to the largest local maximum is retained as the final solution. We adopt the multi-start method and use the following starting points throughout this work $\gamma_i^0 = \{10^{-9}, 10^{-7}, 10^{-5}, 10^{-3}, 10^{-1}\}$. Note that the objective function $\ell(\gamma_i)$ can have maxima for $\gamma_i < 0$. Hence, depending on the starting point used, the fixed-point iteration can converge to an infeasible solution with $\gamma_i < 0$. In this case, we use $\gamma_i = 0$, which is the closest feasible solution. The EM algorithm does not suffer from this problem, as γ_i is a variance hyperparameter and therefore cannot attain negative values. Even though we do not have convergence guarantees for the fixed-point procedure, we have experimentally found that it provides the same results as the EM algorithm (refer to Section IV-C for an example). Hence, the fixed-point procedure is adopted for the maximization of $\ell(\gamma_i)$ in the remainder of this paper.

Having a method for maximizing $\ell(\gamma_i)$ enables us to obtain a greedy inference algorithm in which we only modify the γ_i hyperparameter of one signal during each iteration. The greedy algorithm is outlined in Fig. 1. It starts with an empty model, i.e., $\boldsymbol{\gamma} = \mathbf{0}$. During each iteration, the improvement of the objective function $\ell(\gamma_i^*)$ resulting from adding any of the signals currently not included in the model ($i \notin \mathcal{S} \Leftrightarrow \gamma_i = 0$) is evaluated. For active signals ($i \in \mathcal{S} \Leftrightarrow \gamma_i > 0$), we calculate the improvement of the objective function when updating the γ_i of the signal.

After the improvement of the objective function has been evaluated for all possible actions, the algorithm selects the action leading to the largest improvement of the objective function and updates the model accordingly. If the algorithm updates a signal currently included in the model with a new value $\gamma_i = 0$, the signal is removed from the model. It should be noted that for numerical reasons we assume $\gamma_i = 0$ if $\gamma_i < 10^{-12}$ throughout the algorithm. Note that this means the proposed cannot detect active signals with $\gamma_i < 10^{-12}$. For data with very small scales where active signals can correspond to $\gamma_i < 10^{-12}$, it is therefore necessary to adjust this parameter accordingly. The algorithm maintains the matrix $\mathbf{C}_{\mathcal{S}}^{-1}$ for all signals currently included in the model, i.e., the matrix corresponds to $\mathbf{C}_{-i}^{-1} \forall i \notin \mathcal{S}$ in the above derivation. When the algorithm needs access to the matrix $\mathbf{C}_{-i}^{-1}$ for a signal that is currently included in the model, the matrix is calculated using the Woodbury identity similarly to (20), i.e.,

$$\mathbf{C}_{-i}^{-1} = \mathbf{C}_{\mathcal{S}}^{-1} - \mathbf{C}_{\mathcal{S}}^{-1}\mathbf{Q}_i \left(\mathbf{Q}_i^T \mathbf{C}_{\mathcal{S}}^{-1} \mathbf{Q}_i - \gamma_i^{-1}\mathbf{I}_L\right)^{-1} \mathbf{Q}_i^T \mathbf{C}_{\mathcal{S}}^{-1}. \tag{39}$$

1: INPUTS: $\boldsymbol{\Phi}$, $\mathbf{Y}$, β
2: OUTPUTS: $\boldsymbol{\mu}_{\mathcal{S}}$, $\boldsymbol{\Sigma}_{\mathcal{S}}$
3: Initialization: $\mathcal{S} \leftarrow \emptyset$, $\boldsymbol{\gamma} \leftarrow \mathbf{0}$
4: **while** convergence criterion not met **do**
5: **for all** $i \in \{1, 2, 3, \ldots, N\} \setminus \mathcal{S}$ **do** {Inactive signals}
6: $\mathtt{obj}[i] \leftarrow \ell(\gamma_i^*)$ $\{\gamma_i^* = \mathrm{argmax}_{\gamma_i} \ell(\gamma_i)\}$
7: **end for**
8: **for all** $i \in \mathcal{S}$ **do** {Active signals}
9: $\mathtt{obj}[i] \leftarrow \ell(\gamma_i^*) - \ell(\gamma_i)$ $\{\gamma_i^* = \mathrm{argmax}_{\gamma_i} \ell(\gamma_i)\}$
10: **end for**
11: $i \leftarrow \mathrm{argmax}\, \mathtt{obj}[i]$ {Greedy maximization of $\mathcal{L}(\boldsymbol{\gamma})$}
12: **if** $i \notin \mathcal{S}$ **then**
13: Add: $\gamma_i \leftarrow \gamma_i^*$, $\mathcal{S} \leftarrow \mathcal{S} \cup \{i\}$
14: **else if** $i \in \mathcal{S}$ **and** $\gamma_i^* = 0$ **then**
15: Remove: $\gamma_i \leftarrow 0$, $\mathcal{S} \leftarrow \mathcal{S} \setminus \{i\}$
16: **else if** $i \in \mathcal{S}$ **and** $\gamma_i^* > 0$ **then**
17: Update: $\gamma_i \leftarrow \gamma_i^*$
18: **end if**
19: Update $\mathbf{C}_{\mathcal{S}}^{-1}$ and (optional) $\boldsymbol{\Sigma}_{\mathcal{S}}$
20: **end while**
21: Calculate $\boldsymbol{\mu}_{\mathcal{S}}$

Fig. 1. Greedy Bayesian inference algorithm.

Updating the covariance matrix $\boldsymbol{\Sigma}$ for signals currently included in the model can be done efficiently using a procedure similar to the one used in [11]. The updating procedure only requires the inversion of a single $L \times L$ matrix during each iteration and is described in the Appendix. If one is not interested in the posterior covariance matrix $\boldsymbol{\Sigma}$, instead of updating $\boldsymbol{\Sigma}$ during each iteration, one can calculate the posterior mean $\boldsymbol{\mu}_{\mathcal{S}}$, which includes only the signals with $\gamma_i > 0$, after $\boldsymbol{\gamma}$ has been determined by the algorithm by solving a linear system of order nL, where n denotes the number of active signals, i.e., $n = |\mathcal{S}|$. The linear system is given by

$$\left(\beta \boldsymbol{\Psi}_{\mathcal{S}}^T \boldsymbol{\Psi}_{\mathcal{S}} + \boldsymbol{\Lambda}_{\mathcal{S}}\right) \boldsymbol{\mu}_{\mathcal{S}} = \beta \boldsymbol{\Psi}_{\mathcal{S}}^T \mathrm{vec}(\mathbf{Y}), \tag{40}$$

where

$$\boldsymbol{\Psi}_{\mathcal{S}} = \left(\mathbf{I}_L \otimes [\boldsymbol{\Phi}_{\cdot i} \; \forall \, i \in \mathcal{S}]\right), \tag{41}$$

$$\boldsymbol{\Lambda}_{\mathcal{S}} = \left(\mathbf{P} \otimes \mathrm{Diag}\left([\gamma_i \; \forall \, i \in \mathcal{S}]\right)^{-1}\right). \tag{42}$$

Note that $\boldsymbol{\Psi}_{\mathcal{S}}$ and $\boldsymbol{\Lambda}_{\mathcal{S}}$ are defined analogously to $\boldsymbol{\Psi}$ and $\boldsymbol{\Lambda}$, but only include signals with non-zero γ_i. Finally, the estimate $\hat{\mathbf{W}}$ is obtained from $\boldsymbol{\mu}_{\mathcal{S}}$ by re-arranging the elements and setting all rows corresponding to signals with $\gamma_i = 0$ to zero.

IV. Simulation Experiments

In this section, we demonstrate the performance of the proposed method using an empirical evaluation with simulated data. We present simulations with two types of signals. In the first experiment we generate smooth signals using windowed sinusoids and use a $\mathbf{P}^+$ matrix constructed from second order

derivative operators. In the second experiment, the latent signals are realizations of a first-order autoregressive (AR) process, which enables us to include our knowledge about the generative process by setting $\mathbf{P}$ equal to a scaled version of the covariance matrix of the AR process. For both experiments we use random measurement matrices $\mathbf{\Phi}$ corresponding to a uniform spherical ensemble, i.e., the columns of $\mathbf{\Phi}$ are drawn from a uniform distribution on the M-sphere with radius 1.

The following methods are included in our evaluation: The proposed greedy method (denoted by M-SBL-S), M-OMP [5], [6], M-FOCUSS [7] (with $p = 0.8$), M-SBL [9], and the multitask compressive sensing (MT-CS) method [10], which can be considered a greedy version of M-SBL. For the M-SBL and MT-CS methods, MATLAB implementations obtained from the authors' websites were used while MATLAB implementations from the "Multiple-Spars Toolbox"[1] were used for M-BP and M-FOCUSS. For M-BP and M-FOCUSS the regularization parameter used for each run is selected by executing the algorithms for 20 logarithmically equally spaced regularization parameter values in the interval $[10^{-3} - 10]$ and then retaining the result with the lowest relative reconstruction error, calculated as $\|\hat{\mathbf{W}} - \mathbf{W}\|_{\mathcal{F}}^2 / \|\mathbf{W}\|_{\mathcal{F}}^2$.

A. Sinusoidal Signals

In this experiment, each active signal with $L = 50$ points was generated using a sinusoid weighted by a Hann window (sometimes also called Hanning window). The phase of the sine wave was drawn from a uniform distribution between 0 and π and the number of periods was uniformly distributed between 1 and 3; see Fig. 3(a). The smoothness of the signals was modeled by the proposed algorithm by using $\mathbf{P}^+ = \mathbf{T}^T\mathbf{T}$, where the $L \times L$ matrix $\mathbf{T}$ implements a discrete second order derivative operator given by

$$\mathbf{T}_{ij} = \begin{cases} -2 & \text{if } i = j, \\ 1 & \text{if } j = i \pm 1, \\ 0 & \text{otherwise.} \end{cases} \tag{43}$$

We used two different degrees of sparsity by using $\mathbf{W}$ matrices with either 5 or 10 non-zero rows out of $N = 200$. The simulated measurement $\mathbf{Y}$ data were generated using (2) with zero-mean, i.i.d., Gaussian noise with a noise variance corresponding to a peak signal to noise ratio (PSNR) of 15 dB; the PSNR is defined as $\text{PSNR} = 10\log(\|\text{vec}(\mathbf{\Phi W})\|_\infty^2/\sigma^2)$. While N, L, and the degree of sparsity were kept constant, we varied the number of measurements M from 20 to 100 in steps of 5, i.e., from 10% to 50% of N.

Reconstruction errors calculated over 100 runs are shown in Fig. 2. Clearly, the proposed method (denoted by M-SBL-S) outperforms existing methods by a considerable margin and achieves reconstruction error scores that are typically more than 50% lower than the scores obtained by existing methods, except for 10 active signals and $M < 30$, where all methods fail to give good results. It can also be seen that except for experiments with few measurements, the standard deviation of the reconstruction error is typically lower than that of existing methods, which shows that the proposed method is robust in terms of providing a reliable reconstruction performance.

[1]http://asi.insa-rouen.fr/enseignants/arakotom/code/SSAindex.html.

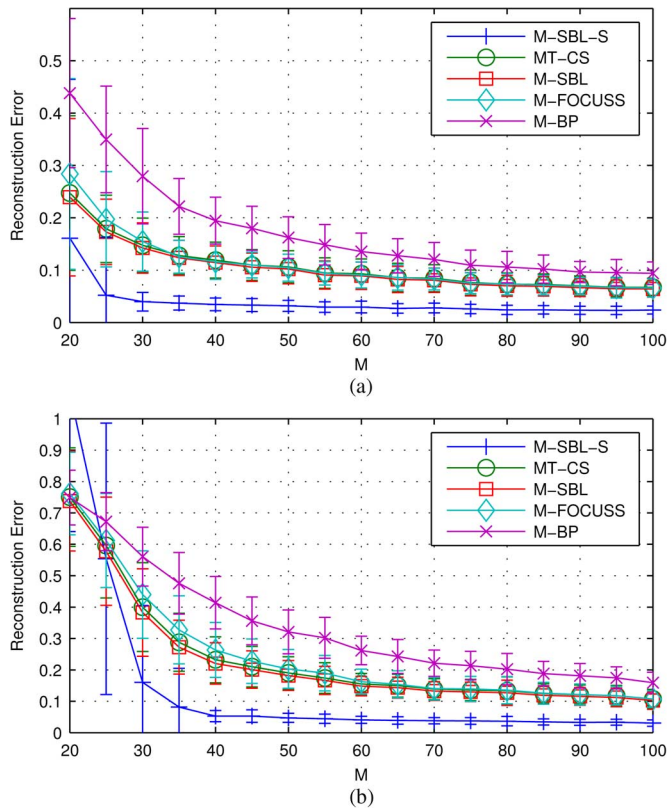


Fig. 2. Reconstruction error versus number of measurements M for smooth signals with $N = 200$ and $\text{PSNR} = 15$ dB. a) Results for 5 non-zero rows in $\mathbf{W}$, (b) results for 10 non-zero rows in $\mathbf{W}$. The lines show the average error computed over 100 simulations and the error bars indicate the standard deviation.

The advantage of incorporating smoothness into the reconstruction process is even more compelling when comparing the original and the reconstructed signals, as depicted in Fig. 3. Note that we only include the M-SBL method in the comparison since it performed best among the existing methods and the reconstructions obtained by the other evaluated methods are similarly noisy. It can be seen that the signals reconstructed by the proposed method are much smoother and closer to the original signals than the signals reconstructed by the M-SBL method, which are very noisy. Notice that both the proposed method and M-SBL find a number of spurious signals with very small amplitudes (smooth signals for the proposed method, noisy signals for M-SBL). The γ_i parameters corresponding to these signals have very small values and the spurious signals could easily be removed using a thresholding procedure.

Given that the reconstructions obtained by existing methods are very noisy it is of interest to analyze how the reconstruction error is affected by the noise variance. In order to do so we repeated the above experiment for $M = 40$ and a row-support of 5 of $\mathbf{W}$ and varied the noise variance σ^2. Note that a PSNR of 15 dB in the previous experiment typically corresponds to $\sigma^2 \approx 0.015$. The results are shown in Fig. 4. As intuitively expected, the difference in terms of reconstruction error between the proposed method and existing methods becomes smaller

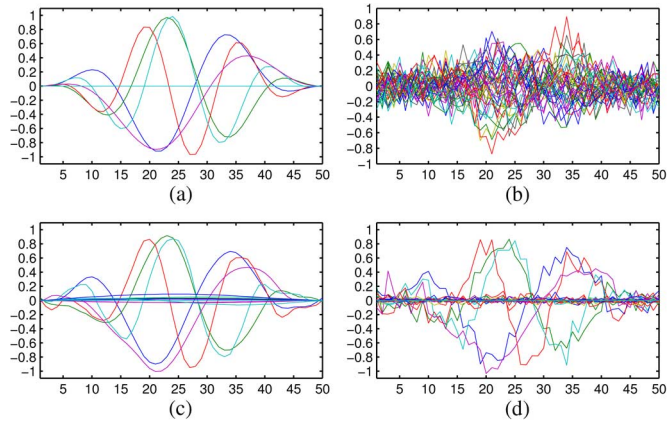


Fig. 3. Reconstruction of smooth signals with $N = 200$, $M = 40$, $L = 50$. (a) Original signal, (b) noisy observation ($\mathrm{PSNR} = 15$ dB), (c) reconstruction by proposed method (error: 0.027), (d) reconstruction by M-SBL method (error: 0.108).

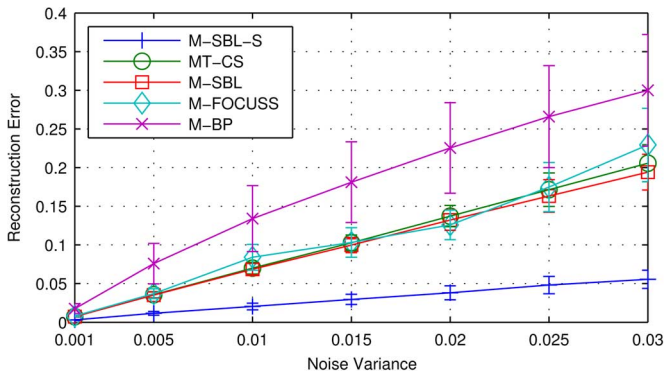


Fig. 4. Reconstruction error versus noise variance for $N = 200$, $M = 40$, and 5 smooth sinusoidal signals. The lines show the average error computed over 100 simulations and the error bars indicate the standard deviation.

when the noise variance is reduced. Hence it can be concluded that for noise free situations the use of smoothness priors offers little benefit over existing methods. On the other hand, at low SNRs, the proposed method is far superior to existing methods.

B. Signals Generated by an AR Process

In the previous section we demonstrated the usefulness of M-SBL-S in the reconstruction of smooth signals by utilizing a $\mathbf{P}^+$ matrix which penalizes non-smooth solutions. However, the proposed method allows for the reconstruction of a much broader class of signals than just smooth signals, i.e., the method can be used to recover latent signals with an arbitrary intra-row correlation structure, which we can model by the covariance matrix $\mathbf{P}$. In the specific example shown here, we assumed that the latent signals in the rows of $\mathbf{W}$ are realizations of a first order autoregressive (AR) process. We generated the signal in the i-th row according to

$$w_{i,t} = s_i(t) = a s_i(t-1) + \epsilon_i(t), \quad 0 \leq a < 1, \tag{44}$$

where a is the AR model parameter controlling the amount of correlation and $\epsilon_i(t) \sim \mathcal{N}(0, \sigma_i^2)$. To generate row-sparsity, we set σ_i^2 for all rows to equal to zero, except for 5 rows where we

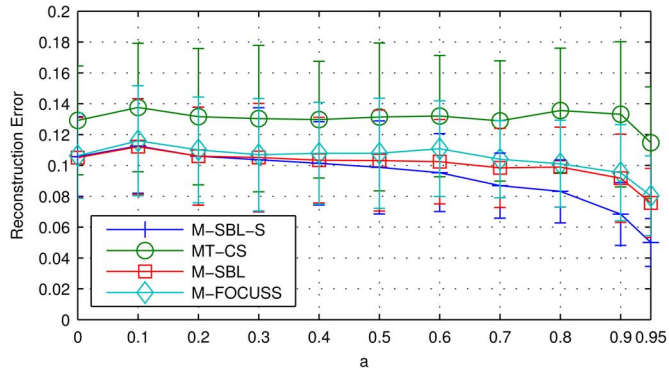


Fig. 5. Reconstruction error versus a parameter of the AR process. Results for M-BP are omitted in order to show the differences between the other methods more clearly (as in the other experiments, M-BP has the highest reconstruction error). The lines show the average error computed over 100 simulations and the error bars indicate the standard deviation.

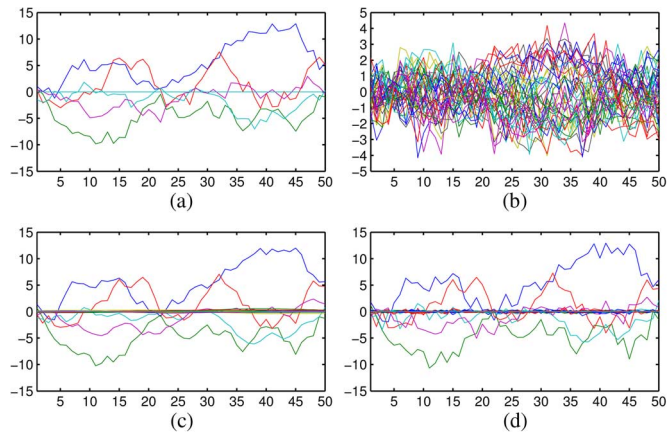


Fig. 6. Reconstructions of signals generated by an AR process with $a = 0.95$ and $N = 200$, $M = 40$, $L = 50$. (a) Original signals, (b) noisy observations ($\mathrm{PSNR} = 15$ dB), (c) reconstruction by proposed method (error: 0.028), (d) reconstruction by M-SBL method (error: 0.046).

drew σ_i^2 from a uniform distribution in the range [1–3]. From the properties of AR processes, we know that the i-th row can be considered to be drawn from a zero-mean multivariate Gaussian distribution with a covariance matrix

$$\mathrm{cov}(w_{i\cdot})_{r,s} = \frac{\sigma_i^2}{1-a} a^{|r-s|}. \tag{45}$$

This motivated us to use

$$\mathbf{P}_{i,j} = \frac{1}{1-a} a^{|i-j|}, \tag{46}$$

i.e., γ_i estimated by the algorithm corresponds to σ_i^2. Note that if the signals of interest are always AR, it would be advantageous to derive a method with a parametric AR model for the rows of $\mathbf{W}$. However, such a method would be limited to the estimation of AR signals while the proposed method is more flexible and allows the use of arbitrary $\mathbf{P}$ matrices.

We evaluated the performance for values of a in the range [0–0.95] together with $N = 200$, $M = 40$, $L = 50$, $\mathrm{PSNR} = 15$ dB; the results are shown in Fig. 5. It can be seen that when a approaches one, the proposed method clearly outperforms

existing methods. Note that for $a = 0$ the rows in $\mathbf{W}$ are zero-mean, i.i.d., Gaussian distributed, i.e., $w_{ij} \sim \mathcal{N}(0, \sigma_i^2)$. From (46) it can be seen that we use $\mathbf{P} = \mathbf{I}$ if $a = 0$, which is the underlying assumption in existing Bayesian methods (M-SBL, MT-CS). As expected, the proposed method performs similarly to existing methods in this case. Reconstructions for one simulation are shown in Fig. 6. While the difference between the methods is smaller than when sinusoidal signals are used, it can still be seen that the signals estimated by the proposed method are more similar to the original signals. This experiment demonstrates how for high correlation values the inclusion of information about the generative process of the latent signal through the matrix $\mathbf{P}$ enables the proposed method to outperform existing methods which do not have this option.

C. Comparison of the EM and Fixed-Point Methods

We introduced two methods for maximizing the objective function (23) in the greedy algorithm. Namely a method based on the EM algorithm and a method using a fixed point iteration. Due to its computational efficiency, we use the fixed point method for all experiments in this section. While we have no proof that the fixed-point method converges to the same solution, in our experiments we observed that it always converges to the same solution as the EM algorithm. In the following experiment we analyze the convergence of the methods for a simulation from Section IV-A with $N = 200$, $M = 40$ and 5 sinusoidal signals. In Fig. 7, the evolution of γ_i for the signal that is added during the first iteration of greedy algorithm is shown. It can be seen that both methods obtain to the same solution but the fixed-point method requires far fewer iterations to reach convergence. The runtime of the algorithms is 1.7 ms for the fixed-point method and 64 ms for the EM algorithm (on a 2.8 GHz Xeon X5660 CPU). As the optimization needs to be performed for every signal and starting point during each iteration of the greedy algorithm, using the fixed-point method results in drastically lower computation times.

D. Computational Requirements

To conclude this experimental evaluation, we compare the computational requirements of the proposed method with those of existing methods. The computationally demanding parts of the proposed method are the search over the inactive signals, which has time complexity of $O(NL^3)$ (search over signals and eigendecomposition in (31)), and the update of $\mathbf{C}^{-1}$ in (20), which has a time complexity of $O(M^3L^3)$. To evaluate the computational requirements in practice, we computed the average execution time and the number of iterations for the experiment in Section IV-A with 5 active signals. The results are shown in Fig. 8. It can be seen that while the proposed method, due to its greedy nature, requires a small number of iterations to reach convergence, its execution time is higher than that of other evaluated methods. Clearly, the higher execution time limits the problem size for which the method can be used in practice. As noted above, the time needed critically depends on the number measurements M and the number of time points L. Clearly, in practice the user has to decide whether the superior reconstruction quality of the proposed method warrants the higher execution time. In the next section, we show the feasibility of applying the proposed method to a large-scale problem with $N = 20'471$, $M = 64$, $L = 91$.

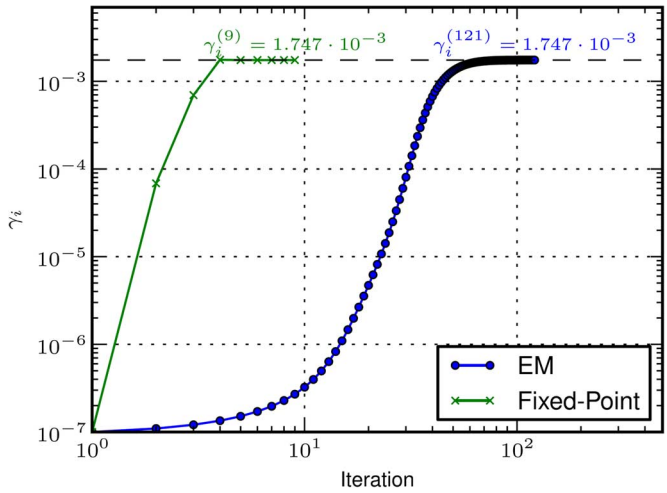


Fig. 7. Example of the evolution of γ_i during the maximization of $\ell(\gamma_i)$ using the proposed fixed-point method and the EM algorithm. The labels indicate the final value of γ_i and the number of iterations for each method.

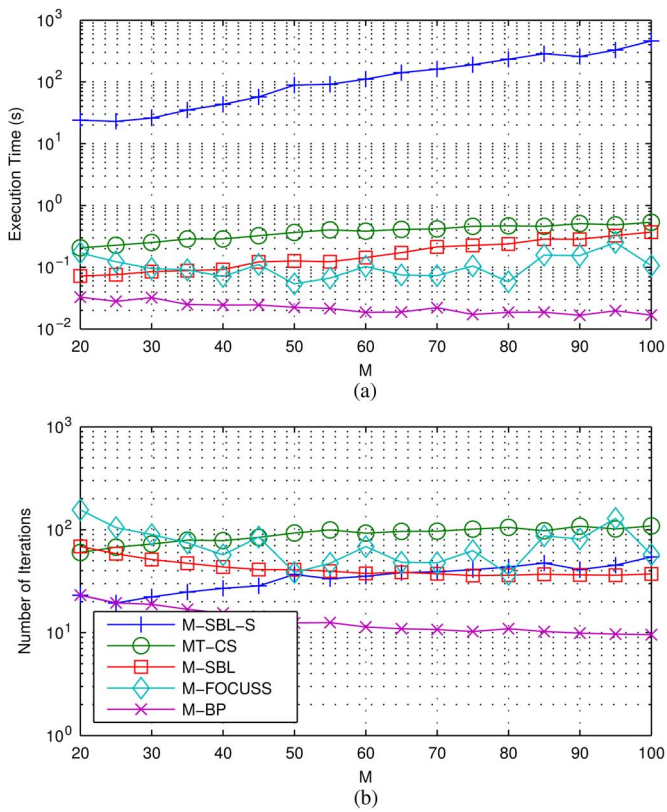


Fig. 8. Average execution time (a) and number of iterations (b) for the evaluated methods and the experiment from Section IV-A with 5 active signals.

V. Application to MEG Data

A potential application for the proposed method is source localization in electroencephalography (EEG) and magne-

toencephalography (MEG). In the distributed formulation of the source localization problem [17], $\mathbf{Y}$ represents recordings from M sensors (scalp electrodes in EEG and magnetic field detectors in MEG) and $\mathbf{W}$ represents the unknown time courses of N current dipoles distributed over the cortical surface. The gain matrix $\mathbf{\Phi}$ can be calculated using knowledge about tissue conductivities and the head geometry, either using spherical approximations to the head geometry [30], [31], which allows for an analytical solution, or using more realistic head models by employing boundary-element or finite-element methods [32]. Numerous M/EEG source localization methods have been proposed in the literature. One assumption that is often used is that the measurements can be explained by a small number of current dipoles which are localized in the active brain regions. In the minimum-current method [33], [34] this sparsity assumption is formalized using ℓ_1-norm regularization at each time point separately thus computing the estimate of each column of $\mathbf{W}$ separately. However, this approach can cause the set of active dipoles to change from time instant to time instant resulting in discontinuous current estimates. This problem can be alleviated by employing simultaneous sparse approximation, as in [35], [36] where a deterministic $\ell_1\ell_2$-norm regularization term is used. A related method was recently proposed in [37], where in addition to $\ell_1\ell_2$-norm regularization, ℓ_1 regularization within a Gabor dictionary is used to model non-stationarity and impose temporal smoothness.

To demonstrate the use of the proposed method for source localization, we apply it to real MEG data from an auditory experiment where a pure tone stimulus is applied to the left ear. The dataset is publicly available as part of the MNE software.[2] The data were acquired using a 306-channel Elekta Neuromag Vectorview MEG system using a sampling rate of 600 Hz. We applied the following preprocessing steps to the data: Artifacts were reduced using the Signal Space Separation (SSS) method [38] implemented in the Elekta Neuromag MaxFilter 2.0 software. As a next step, a low-pass filter of 100 Hz was applied and the data were downsampled to 300 Hz. An evoked response was computed by averaging 68 epochs (responses to the stimulus) from 0 ms to 300 ms after stimulus onset, resulting in a evoked response matrix $\mathbf{Y}$ of size 306×91. The data from -200 ms to 0 ms before stimulus presentation can be considered noise and we used this data segment from all epochs to estimate a noise covariance matrix $\mathbf{\Sigma}_n$ of size 306×306. The maximum-likelihood method was used for the estimation, which due to the large number of time samples works well in this case. Next, we computed a dimensionality reduction matrix which can be used to project the MEG data to the 64 spatial dimensions kept by the MaxFilter software. This was done as follows: We used the singular value decomposition (SVD) to obtain $\mathbf{M} = \mathbf{USV}^T$ where $\mathbf{M}$ is a 306×83400 matrix with the MEG data after the application of the SSS method, $\mathbf{U}$ is a 306×64 matrix with orthonormal columns, $\mathbf{S}$ is a 64×64 matrix with the non-zero singular values on its diagonal, and $\mathbf{V}$ is a 83400×64 matrix. Using this, we reduced dimension of the evoked response and the noise covariance matrix to 64 as follows

$$\tilde{\mathbf{Y}} = \mathbf{U}^T\mathbf{Y}, \quad \tilde{\mathbf{\Sigma}}_n = \mathbf{U}^T\mathbf{\Sigma}_n\mathbf{U}. \tag{47}$$

[2]http://www.martinos.org/mne/.

It is important to note that since the SSS method only retains 64 spatial dimensions, no information is lost by this step; it simply means that we project the data to 64 independent virtual sensors. The smaller number of sensors reduces the computational requirements when performing source localization. The computed noise covariance matrix was used to whiten the evoked response using

$$\bar{\mathbf{Y}} = \tilde{\mathbf{\Sigma}}_n^{-1/2}\tilde{\mathbf{Y}}. \tag{48}$$

Note that after whitening, the additive noise in $\bar{\mathbf{Y}}$ is i.i.d. Gaussian distributed with unit variance. As mentioned above, in MEG source localization, the matrix $\mathbf{\Phi}$ relates the currents on the cortical surface to the sensor measurements. Here, we used 20471 dipoles evenly distributed over the surface of the neocortex (spacing approximately 3 mm) and the MNE software with a subject specific BEM model to compute the gain matrix $\mathbf{\Phi}$ of size 306×20471. Finally, the dimensionality reduction and whitening operations that were applied to the evoked response were also applied to the gain matrix as follows

$$\bar{\mathbf{\Phi}} = \tilde{\mathbf{\Sigma}}_n^{-1/2}\mathbf{U}^T\mathbf{\Phi}, \tag{49}$$

resulting in a matrix $\bar{\mathbf{\Phi}}$ of size 64×20471.

The evoked response $\bar{\mathbf{Y}}$ and the gain matrix $\bar{\mathbf{\Phi}}$ are used as observation and forward operator matrix, respectively, when applying the reconstruction algorithms. Note that due to the whitening, the additive noise in $\hat{\mathbf{Y}}$ has unit variance (precision). However, as the data does not exactly fit the model it is necessary to assume a lower noise precision in order to obtain a sparse solution. From previous work with the same data [36], we know that the solution should contain dipoles in the left and right auditory cortices. Hence, we force the solution to be sparse with only two active dipoles, one in each auditory cortex, by adjusting the noise precision. Consequently, we use $\beta = 7 \cdot 10^{-3}$ for the proposed method and $\beta = 12 \cdot 10^{-3}$ for the M-SBL method. Using larger values for the noise precision leads to less sparse solutions with additional "spurious" active dipoles. For the proposed method, we model the temporal smoothness of the cortical currents by using the temporal covariance matrix $\mathbf{P}$ of an AR process computed using (46) with $a = 0.6$.

Results are shown in Fig. 9. It can be seen that both, the proposed method and the M-SBL method, correctly localize the active dipoles in the left and right auditory cortices. The results are also similar to results in [36], where $\ell_1\ell_2$ regularization was used. Notice that the locations of the active sources estimated by M-SBL are slightly more superficial, which explains the difference in current amplitudes between the methods, as more superficial currents produce stronger magnetic fields at the sensors. While the locations of the active dipoles are similar for both methods, the modeling employed in the proposed method leads to current waveforms which exhibit a greater degree of temporal smoothness, which is (presumably) closer to the waveforms of the true cortical currents. Clearly, the degree of temporal smoothness depends on the value of the AR parameter a. For the problem at hand, we found that $a = 0.6$ results in an appropriately smooth solution. Using larger values for a increases the temporal smoothness and can lead to overly smooth solutions. Given the amount of temporal correlation in MEG

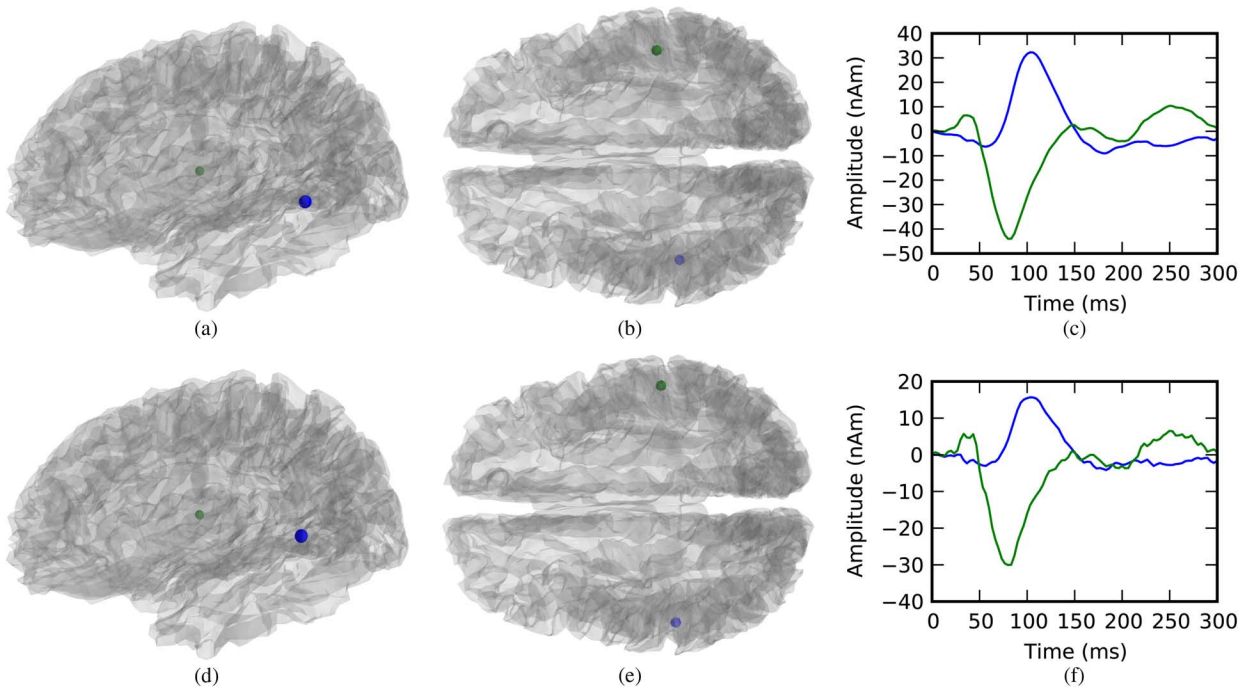


Fig. 9. Results for the MEG experiment showing the location of the active dipoles and the corresponding current waveforms obtained by the proposed method (top row) and the M-SBL method (bottom row). (a) Proposed method: lateral view; (b) proposed method: superior view; (c) proposed method: currents; (d) M-SBL: lateral view; (e) M-SBL: superior view; (f) M-SBL: currents.

data, using a $\mathbf{P}$ matrix based on an AR model with $a = 0.6$ works well. However, for other types of data, the a parameter, or more generally the $\mathbf{P}$ matrix, has to be adjusted based on knowledge about the temporal correlation of the underlying signals in order to obtain solutions with an appropriate degree of temporal smoothness. Note that it is also possible to treat a as a hyperparameter and learn it from the data together with other model hyperparameters using Bayesian inference. However, doing so would require substantial changes to the modeling and inference procedure employed here. Hence, we chose not to pursue this approach and instead adjusted a manually.

VI. Conclusions

In this paper we proposed a method for simultaneous sparse approximation which in addition to row-sparsity also takes the correlation among latent signal vectors into account. We used a Bayesian formulation of the problem and incorporate row-smoothness priors into the sparse Bayesian learning (SBL) formulation for simultaneous sparse approximation [9]. A major challenge of taking signal correlations into account is that the resulting inference procedure is computationally very demanding, even for moderately sized problems. Hence, we developed a more efficient greedy Bayesian inference algorithm [10], [11]. The algorithm starts with an empty model and at each iteration in a greedy fashion selects the variance parameter to update which leads to the largest increase of the objective function. As there is no closed form expression for the maximizer of the objective function, we put forward two iterative procedures which can be used to find the maximizer efficiently.

The modeling assumptions in our method are flexible and thus cover broad class of latent signals to be recovered from undercomplete measurements. We demonstrated the flexibility of the method using both an empirical evaluation with simulated data and real MEG data from an auditory experiment. In the first simulation experiment, the signals were composed of windowed sinusoidals. We incorporated our prior knowledge about the temporal smoothness by constructing the row precision matrix from derivative operators, such that the method penalizes non-smooth solutions while simultaneously promoting row-sparsity. The more accurate prior modeling of the latent signals enabled the proposed method to recover the signals much more accurately than existing methods, especially in situations with low signal-to-noise ratios. In the second simulation experiment we demonstrated the ability of our method to recover latent signals generated by an autoregressive (AR) process. In this case we encoded our prior knowledge about the generative process into the structure of the row precision matrix. As expected, our method outperformed existing methods with an increasing margin as the correlation between the latent signal vectors increases while offering the same performance as existing methods when the latent signal vectors are in fact uncorrelated. Finally, we demonstrated the utility of the proposed method for MEG source localization using real MEG data from an auditory experiment. The proposed method correctly localized the activity in the left and right auditory cortices and due to the employed modeling, the temporal waveforms of the currents are temporally smoother than those estimated by existing methods. These are just three examples of signals which can be recovered by our method. In all cases, use of additional information enabled the proposed method to reconstruct the latent signals

more faithfully than existing methods which are based on the assumption that the latent signal vectors are uncorrelated.

APPENDIX A
MAXIMIZATION OF (16) WITH RESPECT TO γ_i

To maximize (16) with respect to γ_i, we calculate the derivative with respect to γ_i and set it to zero. In order to do so, we first rewrite $|\mathbf{C}|$ using the matrix determinant lemma as follows

$$\begin{aligned}|\mathbf{C}| &= |\beta^{-1}\mathbf{I}_{ML}||\mathbf{I}_{ML} + \beta\mathbf{\Psi}\mathbf{\Lambda}^{-1}\mathbf{\Psi}^T| \\ &= |\beta^{-1}\mathbf{I}_{ML}||\mathbf{I}_{NL} + \beta\mathbf{\Lambda}^{-1}\mathbf{\Psi}^T\mathbf{\Psi}| \\ &= |\beta^{-1}\mathbf{I}_{ML}||\mathbf{\Lambda}^{-1}||\mathbf{\Sigma}^{-1}|, \end{aligned} \tag{50}$$

and using this we obtain

$$\log|\mathbf{C}| = -\log|\mathbf{\Lambda}| - ML\log\beta - \log|\mathbf{\Sigma}|. \tag{51}$$

Second, we use the Woodbury identity to obtain

$$\mathbf{C}^{-1} = \beta\mathbf{I}_{ML} - \beta\mathbf{\Psi}\mathbf{\Sigma}\mathbf{\Psi}^T\beta \tag{52}$$

and therefore

$$\begin{aligned}&\mathrm{vec}(\mathbf{Y})^T\mathbf{C}^{-1}\mathrm{vec}(\mathbf{Y}) \\ &\quad= \beta\mathrm{vec}(\mathbf{Y})^T\mathrm{vec}(\mathbf{Y}) - \beta\mathrm{vec}(\mathbf{Y})^T\mathbf{\Psi}\mathbf{\Sigma}\mathbf{\Psi}^T\mathrm{vec}(\mathbf{Y})\beta \\ &\quad= \beta\mathrm{vec}(\mathbf{Y})^T\left[\mathrm{vec}(\mathbf{Y}) - \beta\mathbf{\Psi}\mathbf{\Sigma}\mathbf{\Psi}^T\mathrm{vec}(\mathbf{Y})\right] \\ &\quad= \beta\left\|\mathrm{vec}(\mathbf{Y}) - \mathbf{\Psi}\boldsymbol{\mu}\right\|^2 + \beta\boldsymbol{\mu}^T\mathbf{\Sigma}^{-1}\boldsymbol{\mu} - \beta\boldsymbol{\mu}^T\mathbf{\Psi}^T\mathbf{\Psi}\boldsymbol{\mu} \\ &\quad= \beta\left\|\mathrm{vec}(\mathbf{Y}) - \mathbf{\Psi}\boldsymbol{\mu}\right\|^2 + \boldsymbol{\mu}^T\mathbf{\Lambda}\boldsymbol{\mu}. \end{aligned} \tag{53}$$

By using these identities we obtain the derivative of $\mathcal{L}$ with respect to γ_i as

$$\frac{d\mathcal{L}}{d\gamma_i} = \frac{1}{2}\left[\frac{\left\langle \mathbf{w}_{i\cdot}^T\mathbf{P}^+\mathbf{w}_{i\cdot}\right\rangle}{\gamma_i^2} - \frac{R}{\gamma_i}\right]. \tag{54}$$

Hence, the maximum is attained at

$$\begin{aligned}\gamma_i &= \frac{\left\langle \mathbf{w}_{i\cdot}^T\mathbf{P}^+\mathbf{w}_{i\cdot}\right\rangle}{R} \\ &= \frac{\langle\mathbf{w}_{i\cdot}\rangle^T\mathbf{P}^+\langle\mathbf{w}_{i\cdot}\rangle + \mathrm{tr}\left(\mathrm{cov}(\mathbf{w}_{i\cdot})\mathbf{P}^+\right)}{R}, \end{aligned} \tag{55}$$

where $\mathrm{cov}(\mathbf{w}_{i\cdot})$ is the covariance matrix of the i-th row of $\mathbf{W}$ (this matrix can easily be extracted from $\mathbf{\Sigma}$, i.e., $\mathrm{cov}(\mathbf{w}_{i\cdot})_{r,s} = \mathbf{\Sigma}_{i+(r-1)L,i+(s-1)L}$).

APPENDIX B
UPDATING $\mathbf{\Sigma}$

In this appendix we provide the update equations for the covariance matrix $\mathbf{\Sigma}$ when adding, updating, and removing signals in the proposed greedy algorithm. Note that the update operation can be performed by combining the two basic operations for removing and adding a signal. More specifically, in order to update the variance parameter γ_i belonging to the i-th signal from the value γ_i^{old} to a new value γ_i^{new} we can remove the i-th signal from the model and then add the same signal with $\gamma_i = \gamma_i^{\mathrm{new}}$ to the model. Therefore, we only show the update equations for $\mathbf{\Sigma}$ when adding and removing signals.

a) Adding a signal: In the following we assume that the model currently contains n signals. The covariance matrix is then given by

$$\mathbf{\Sigma} = \left(\beta(\mathbf{I}_L \otimes \bar{\mathbf{\Phi}}^T\bar{\mathbf{\Phi}}) + \left(\mathbf{P}^+ \otimes \mathrm{Diag}(\bar{\boldsymbol{\gamma}})^{-1}\right)\right)^{-1}, \tag{56}$$

where $\bar{\mathbf{\Phi}}$ and $\bar{\boldsymbol{\gamma}}$ only contain the atoms and the corresponding variance parameters for which $\gamma_i > 0$. Now, assume we would like to include the i-th signal with atom $\Phi_{\cdot i}$ and variance parameter γ_i in the model. The updated covariance matrix can be written as

$$\mathbf{\Sigma}_+ = \left(\beta\left(\mathbf{I}_L \otimes [\bar{\mathbf{\Phi}}\ \Phi_{\cdot i}]^T[\bar{\mathbf{\Phi}}\ \Phi_{\cdot i}]\right) + \mathbf{\Lambda}_+\right)^{-1}, \tag{57}$$

where

$$\mathbf{\Lambda}_+ = \left(\mathbf{P}^+ \otimes \mathrm{Diag}\left([\bar{\boldsymbol{\gamma}}\ \gamma_i]\right)^{-1}\right). \tag{58}$$

By examining the structure of $(\mathbf{\Sigma}_+)^{-1}$, one can see that the newly introduced elements only affect the columns and rows with indices $n+1, 2(n+1), \ldots, L(n+1)$ of $(\mathbf{\Sigma}_+)^{-1}$, with a total of L affected rows and L affected columns. Therefore, we can rearrange the rows and columns such that the new elements only affect the L last rows and columns. The rearranged matrix is given by

$$\begin{aligned}\mathbf{A}_+ &= \mathbf{R}_{(L,n+1)}^T\mathbf{\Sigma}_+^{-1}\mathbf{R}_{(L,n+1)} \\ &= \beta\left([\bar{\mathbf{\Phi}}\ \Phi_{\cdot i}]^T[\bar{\mathbf{\Phi}}\ \Phi_{\cdot i}] \otimes \mathbf{I}_L\right) + \left(\mathrm{Diag}\left([\bar{\boldsymbol{\gamma}}\ \gamma_i]\right)^{-1} \otimes \mathbf{P}^+\right), \end{aligned} \tag{59}$$

where the $pq \times pq$ permutation matrix $\mathbf{R}_{(p,q)}$ has the property $\mathbf{R}_{(p,q)}\mathrm{vec}(\mathbf{X}) = \mathrm{vec}(\mathbf{X}^T)$ for any $p \times q$ matrix $\mathbf{X}$. Note that since $\mathbf{R}_{(p,q)}^{-1} = \mathbf{R}_{(p,q)}^T = \mathbf{R}_{(q,p)}$ we have

$$\mathbf{A}_+^{-1} = \mathbf{R}_{(L,n+1)}^T\mathbf{\Sigma}_+\mathbf{R}_{(L,n+1)}, \tag{60}$$

and thus

$$\mathbf{\Sigma}_+ = \mathbf{R}_{(n+1,L)}^T\mathbf{A}_+^{-1}\mathbf{R}_{(n+1,L)}. \tag{61}$$

Therefore, we can invert $\mathbf{A}_+$ to calculate $\mathbf{\Sigma}_+$, which can be done efficiently as shown in the following. First, note that we can write $\mathbf{A}_+$ as

$$\mathbf{A}_+ = \begin{bmatrix}\mathbf{A} & \mathbf{B} \\ \mathbf{B}^T & \mathbf{C}\end{bmatrix}, \tag{62}$$

where the $nL \times nL$ matrix $\mathbf{A}$ is given by

$$\mathbf{A} = \left(\beta(\bar{\mathbf{\Phi}}^T\bar{\mathbf{\Phi}} \otimes \mathbf{I}_L) + \left(\mathrm{Diag}(\bar{\boldsymbol{\gamma}})^{-1} \otimes \mathbf{P}^+\right)\right). \tag{63}$$

The matrices $\mathbf{B}$ and $\mathbf{C}$ are given by

$$\mathbf{B} = \beta(\bar{\mathbf{\Phi}}^T\Phi_{\cdot i} \otimes \mathbf{I}_L), \tag{64}$$

$$\mathbf{C} = \beta\Phi_{\cdot i}^T\Phi_{\cdot i} \cdot \mathbf{I}_L + \gamma_i^{-1}\mathbf{P}^+. \tag{65}$$

Using 2×2 block matrix inversion and the Woodbury identity, $\mathbf{A}_+^{-1}$ is given by

$$\mathbf{A}_+^{-1} = \begin{bmatrix}\mathbf{A}^{-1} + \mathbf{A}^{-1}\mathbf{B}\mathbf{D}\mathbf{B}^T\mathbf{A}^{-1} & -\mathbf{A}^{-1}\mathbf{B}\mathbf{D} \\ -\mathbf{D}\mathbf{B}^T\mathbf{A}^{-1} & \mathbf{D}\end{bmatrix}, \tag{66}$$

where

$$\mathbf{D} = \left(\mathbf{C} - \mathbf{B}^T\mathbf{A}^{-1}\mathbf{B}\right)^{-1}. \quad (67)$$

Note that no matrix inversion is required to calculate $\mathbf{A}^{-1}$; it can be obtained from $\boldsymbol{\Sigma}$, i.e., the covariance matrix without the i-th signal included, using

$$\mathbf{A}^{-1} = \mathbf{R}_{(L,n)}^T\boldsymbol{\Sigma}\mathbf{R}_{(L,n)}. \quad (68)$$

Therefore, calculating $\mathbf{A}_+^{-1}$ only requires the inversion of an $L \times L$ matrix to calculate $\mathbf{D}$. After $\mathbf{A}_+^{-1}$ has been calculated, the updated covariance matrix $\boldsymbol{\Sigma}_+$ is obtained using (61).

b) Removing a signal: For simplicity, first assume that we would like to remove the last signal from the model, i.e., the signal which corresponds to the last L columns and rows in the $nL \times nL$ matrix $\mathbf{A}$, which can be written as

$$\mathbf{A} = \begin{bmatrix} \mathbf{A}_- & \mathbf{B} \\ \mathbf{B}^T & \mathbf{C} \end{bmatrix}, \quad (69)$$

where $\mathbf{A}_-$ is the $\mathbf{A}$ matrix with the last signal removed and thus we are interested in $\mathbf{A}_-^{-1}$ since the covariance matrix we would like to calculate can be obtained by

$$\boldsymbol{\Sigma}_- = \mathbf{R}_{(n-1,L)}^T\mathbf{A}_-^{-1}\mathbf{R}_{(n-1,L)}. \quad (70)$$

The form of $\mathbf{A}^{-1}$ is given in (68), which we rewrite as

$$\mathbf{A}^{-1} = \begin{bmatrix} \mathbf{E} & \mathbf{G} \\ \mathbf{G}^T & \mathbf{F} \end{bmatrix}, \quad (71)$$

where $\mathbf{E}$ is of size $(n-1)L \times (n-1)L$, $\mathbf{G}$ is of size $(n-1)L \times L$, and $\mathbf{F}$ is of size $L \times L$. By using

$$\mathbf{E} = \mathbf{A}_-^{-1} + \mathbf{A}_-^{-1}\mathbf{B}\mathbf{F}\mathbf{B}^T\mathbf{A}_-^{-1}, \quad (72)$$

and

$$\mathbf{G}\mathbf{F}^{-1} = -\mathbf{A}_-^{-1}\mathbf{B}, \quad (73)$$

we obtain

$$\mathbf{A}_-^{-1} = \mathbf{E} - \mathbf{G}\mathbf{F}^{-1}\mathbf{G}^T. \quad (74)$$

Hence, removing the last signal only required the inversion of an $L \times L$ matrix to calculate $\mathbf{F}^{-1}$. Now, assume that we would like matrix $\mathbf{A}_-^{-1}$ to be of size $nL \times nL$ as well, i.e., in the algorithm we remove the rows and columns corresponding to the signal being removed after $\mathbf{A}_-^{-1}$ has been calculated. In this case we can write

$$\mathbf{A}_-^{-1} = \mathbf{A}^{-1} - (\mathbf{A}^{-1})_{:,\mathbf{r}}\left((\mathbf{A}^{-1})_{\mathbf{r},\mathbf{r}}\right)^{-1}(\mathbf{A}^{-1})_{\mathbf{r},:}, \quad (75)$$

where $\mathbf{r}$ denotes a range of L consecutive indices, i.e., $\mathbf{r} = (i-1)L+1, \ldots, iL$ for the i-th signal currently included in the model. Therefore, in order to calculate $\boldsymbol{\Sigma}_-$, we first calculate $\mathbf{A}_-^{-1}$ using (75), remove the rows and columns corresponding to the signal being removed and then calculate $\boldsymbol{\Sigma}_-$ using (70). Like the procedure for adding a signal, this procedure only requires the inversion of one $L \times L$ matrix.

ACKNOWLEDGMENT

The authors would like to thank Matti S. Hämäläinen and Alexandre Gramfort for their helpful comments.

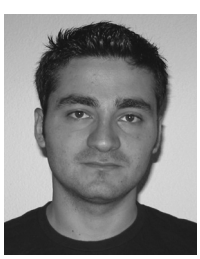
REFERENCES

[1] D. L. Donoho, M. Elad, and V. N. Temlyakov, "Stable recovery of sparse overcomplete representations in the presence of noise," *IEEE Trans. Inf. Theory*, vol. 52, no. 1, pp. 6–18, 2005.
[2] E. J. Candès, J. Romberg, and T. Tao, "Robust uncertainty principles: Exact signal reconstruction from highly incomplete frequency information," *IEEE Trans. Inf. Theory*, vol. 52, no. 2, pp. 489–509, 2006.
[3] D. L. Donoho, "Compressed sensing," *IEEE Trans. Inf. Theory*, vol. 52, no. 4, pp. 1289–1306, Apr. 2006.
[4] , A. K. Katsaggelos, Ed., *Digital Image Restoration*, ser. Springer Series in Information Sciences. New York, NY, USA: Springer-Verlag, 1991, vol. 23.
[5] J. A. Tropp, A. C. Gilbert, and M. J. Strauss, "Algorithms for simultaneous sparse approximation. Part i: Greedy pursuit," *Signal Process.*, vol. 86, no. 3, pp. 572–588, 2006.
[6] J. Chen and X. Huo, "Theoretical results on sparse representations of multiple-measurement vectors," *IEEE Trans. Signal Process.*, vol. 54, no. 12, pp. 4634–4643, 2006.
[7] S. F. Cotter, B. D. Rao, K. Engan, and K. Kreutz-Delgado, "Sparse solutions to linear inverse problems with multiple measurement vectors," *IEEE Trans. Signal Process.*, vol. 53, no. 7, pp. 2477–2488, Jul. 2005.
[8] M. E. Tipping, "Sparse Bayesian learning and the relevance vector machine," *J. Mach. Learn. Res.*, vol. 1, pp. 211–244, 2001.
[9] D. P. Wipf and B. D. Rao, "An empirical Bayesian strategy for solving the simultaneous sparse approximation problem," *IEEE Trans. Signal Process.*, vol. 55, no. 7, pp. 3704–3716, 2007.
[10] S. Ji, D. Dunson, and L. Carin, "Multitask compressive sensing," *IEEE Trans. Signal Process.*, vol. 57, no. 1, pp. 92–106, 2009.
[11] M. E. Tipping and A. Faul, "Fast marginal likelihood maximisation for sparse Bayesian models," in *Proc. 9th Int. Workshop Artif. Intell. Statist.*, 2003, vol. 8, pp. 3–6.
[12] M. West, "On scale mixtures of normal distributions," *Biometrika*, vol. 74, no. 3, pp. 646–648, 1987.
[13] T. Park and G. Casella, "The bayesian lasso," *J. Amer. Statist. Assoc.*, vol. 103, no. 482, pp. 681–686, 2008.
[14] M. West, "Bayesian factor regression models in the "large p, small n" paradigm," *Bayesian Statist.*, vol. 7, no. 2003, pp. 723–732, 2003.
[15] T. Mitchell and J. Beauchamp, "Bayesian variable selection in linear regression," *J. Amer. Statist. Assoc.*, vol. 83, no. 404, pp. 1023–1032, 1988.
[16] J. Griffin and P. Brown, "Inference with normal-gamma prior distributions in regression problems," *Bayesian Anal.*, vol. 5, no. 1, pp. 171–188, 2010.
[17] M. Hämäläinen, R. Hari, R. J. Ilmoniemi, J. Knuutila, and O. V. Lounasmaa, "Magnetoencephalography-theory, instrumentation, and applications to noninvasive studies of the working human brain," *Rev. Mod. Phys.*, vol. 65, no. 2, pp. 413–497, Apr. 1993.
[18] B. M. Yu, J. P. Cunningham, G. Santhanam, S. I. Ryu, K. V. Shenoy, and M. Sahani, "Gaussian-process factor analysis for low-dimensional single-trial analysis of neural population activity," *J. Neurophysiol.*, vol. 102, no. 1, pp. 614–35, Jul. 2009.
[19] D. Malioutov, M. Cetin, and A. S. Willsky, "A sparse signal reconstruction perspective for source localization with sensor arrays," *IEEE Trans. Signal Process.*, vol. 53, no. 8, pp. 3010–3022, 2005.
[20] T. Yardibi, J. Li, P. Stoica, and L. N. Cattafesta, "Sparsity constrained deconvolution approaches for acoustic source mapping," *J. Acoust. Soc. Amer.*, vol. 123, no. 5, pp. 2631–2642, 2008.
[21] D. Baron, M. B. Wakin, M. F. Duarte, S. Sarvotham, and R. G. Baraniuk, "Distributed compressed sensing," Rice Univ., Houston, TX, USA, Tech. Rep. TREE-0612, 2005.
[22] L. Jacob, G. Obozinski, and J. P. Vert, "Group lasso with overlap and graph lasso," in *Proc. ACM 26th Ann. Int. Conf. Mach. Learn.*, 2009, pp. 433–440.
[23] J. Huang, T. Zhang, and D. Metaxas, "Learning with structured sparsity," in *Proc. ACM 26th Ann. Int. Conf. Mach. Learn.*, 2009, pp. 417–424.
[24] D. J. C. MacKay, "Bayesian interpolation," *Neural Computat.*, vol. 4, no. 3, pp. 415–447, 1992.
[25] J. O. Berger, *Statistical Decision Theory and Bayesian Analysis*. New York, NY, USA: Springer, 1985.

[26] I. J. Good, *The Estimation of Probabilities*. Cambridge, MA, USA: MIT Press, 1965.
[27] G. Wahba, "A comparison of GCV and GML for choosing the smoothing parameter in the generalized spline smoothing problem," *Ann. Statist.*, pp. 1378–1402, 1985.
[28] K. V. Mardia, J. T. Kent, and J. M. Bibby *et al., Multivariate Analysis*. London, U.K.: Academic, 1979.
[29] A. P. Dempster, N. M. Laird, and D. B. Rubin *et al.*, "Maximum likelihood from incomplete data via the EM algorithm," *J. Roy. Statist. Soc. Ser. B (Methodolog.)*, vol. 39, no. 1, pp. 1–38, 1977.
[30] J. C. De Munck, "The potential distribution in a layered spheroidal volume conductor," *J. Appl. Phys.*, vol. 64, pp. 464–470, 1988.
[31] Z. Zhang, "A fast method to compute surface potentials generated by dipoles within multilayer anisotropic spheres," *Phys. Med. Biol.*, vol. 40, p. 335, 1995.
[32] J. C. Mosher, R. M. Leahy, and P. S. Lewis, "EEG and MEG: Forward solutions for inverse methods," *IEEE Trans. Biomed. Eng.*, vol. 46, no. 3, pp. 245–259, 2002.
[33] K. Matsuura and Y. Okabe, "Selective minimum-norm solution of the biomagnetic inverse problem," *IEEE Trans. Biomed. Eng.*, vol. 42, no. 6, pp. 608–615, 1995.
[34] K. Uutela, M. Hämäläinen, and E. Somersalo, "Visualization of magnetoencephalographic data using minimum current estimates," *NeuroImage*, vol. 10, no. 2, pp. 173–180, 1999.
[35] W. Ou, M. S. Hämäläinen, and P. Golland, "A distributed spatio-temporal EEG/MEG inverse solver," *NeuroImage*, vol. 44, no. 3, pp. 932–946, 2009.
[36] A. Gramfort, M. Kowalski, and M. Hämäläinen, "Mixed-norm estimates for the M/EEG inverse problem using accelerated gradient methods," *Phys. Med. Biol.*, no. 7, pp. 1937–1961, Mar. 2012.
[37] A. Gramfort, D. Strohmeier, J. Haueisen, M. Hamalainen, and M. Kowalski, "Functional brain imaging with m/eeg using structured sparsity in time-frequency dictionaries," in *Information Processing in Medical Imaging*, ser. Lecture Notes in Comput. Sci., G. Székely and H. Hahn, Eds. Berlin, Germany: Springer, 2011, vol. 6801, pp. 600–611.
[38] S. Taulu and M. Kajola, "Presentation of electromagnetic multichannel data: The signal space separation method," *J. Appl. Phys.*, no. 12, p. 124905, 2005.

Martin Luessi (S'04–M'12) received the Ing. FH degree in electrical engineering from the Hochschule fuer Technik Rapperswil (HSR), Rapperswil, St. Gallen, Switzerland, in 2006, and the M.S and Ph.D. degrees in electrical engineering from the Department of Electrical Engineering and Computer Science, Northwestern University, Evanston, IL, in 2007 and 2011, respectively.

In 2011, he joined the Athinoula A. Martinos Center for Biomedical Imaging, Massachusetts General Hospital (MGH), Boston, MA, as a Research Fellow. He also holds an appointment as a Research Fellow at Harvard Medical School. His current research focuses on the development of signal processing methods for neuroimaging, with focus on methods for magnetoencephalography (MEG) and functional magnetic resonance imaging (fMRI). Other research interests are, Bayesian modeling and inference, numerical optimization, sparse signal processing, and machine learning.



S. Derin Babacan (M'10) received the B.Sc. degree from the Electrical and Electronics Department, Bogazici University, Turkey, in 2004 and the M.Sc. and Ph.D. degrees from the Department of Electrical Engineering and Computer Science, Northwestern University, in 2006 and 2009, respectively.

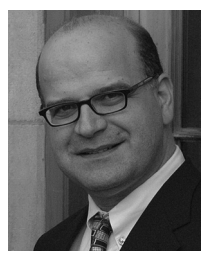
Between 2010–2012, he was a Beckman Postdoctoral Fellow at the Beckman Institute for Advanced Science and Technology at the University of Illinois at Urbana-Champaign, and he is currently at Google, Inc. His primary research interests are inverse problems in image processing, compressive sensing and computational photography.

Dr. Babacan is the recipient of an IEEE International Conference on Image Processing Paper Award (2007).

Rafael Molina (M88) was born in 1957. He received the degree in mathematics (statistics) in 1979 and the Ph.D. degree in optimal design in linear models in 1983.

He became Professor of Computer Science and Artificial Intelligence at the University of Granada, Granada, Spain, in 2000. Former Dean of the Computer Engineering School at the University of Granada (1992–2002) and head of the Computer Science and Artificial Intelligence department of the University of Granada (2005–2007). His research interest focuses mainly in using Bayesian modeling and inference in problems like image restoration (applications to astronomy and medicine), super resolution of images and video, blind deconvolution, computational photography, source recovery in medicine, compressive sensing, low rank matrix decomposition, active learning and classification. See http://decsai.ugr.es/~rms for publications, funded projects and grants.

Dr. Molina serves the IEEE and other Professional Societies: *Applied Signal Processing*, Associate Editor (2005–2007), IEEE TRANSACTIONS ON IMAGE PROCESSING, Associate Editor (2010-), *Progress in Artificial Intelligence*, Associate Editor (2011-) and *Digital Signal Processing*, Area Editor (2011-). He is the recipient of an IEEE International Conference on Image Processing Paper Award (2007), an ISPA Best Paper Award (2009) and coauthor of a paper awarded the runner-up prize at Reception for early-stage researchers at the House of Commons.



Aggelos K. Katsaggelos (F'98) received the Diploma degree in electrical and mechanical engineering from the Aristotelian University of Thessaloniki, Greece, in 1979, and the M.S. and Ph.D. degrees in electrical engineering from the Georgia Institute of Technology, in 1981 and 1985, respectively.

In 1985, he joined the Department of Electrical Engineering and Computer Science, Northwestern University, where he is currently a Professor holder of the AT&T chair. He was previously the holder of the Ameritech Chair of Information Technology (1997–2003). He is also the Director of the Motorola Center for Seamless Communications, a member of the Academic Staff, NorthShore University Health System, an affiliated faculty at the Department of Linguistics and he has an appointment with the Argonne National Laboratory. He has published extensively in the areas of multimedia signal processing and communications (over 180 journal papers, 450 conference papers, and 40 book chapters) and he holds 20 international patents. He is the coauthor of *Rate-Distortion Based Video Compression* (New York: Kluwer, 1997), *Super-Resolution for Images and Video* (Claypool, 2007) and Joint *Source-Channel Video Transmission* (Claypool, 2007).

Among his many professional activities Prof. Katsaggelos was Editor-in-Chief of the IEEE SIGNAL PROCESSING MAGAZINE (1997–2002), a BOG Member of the IEEE Signal Processing Society (1999–2001), and a member of the Publication Board of the IEEE Proceedings (2003–2007). He is a Fellow of SPIE (2009) and the recipient of the IEEE Third Millennium Medal (2000), the IEEE Signal Processing Society Meritorious Service Award (2001), the IEEE Signal Processing Society Technical Achievement Award (2010), an IEEE Signal Processing Society Best Paper Award (2001), an IEEE ICME Paper Award (2006), an IEEE ICIP Paper Award (2007) and an ISPA Paper Award (2009). He was a Distinguished Lecturer of the IEEE Signal Processing Society (2007–2008).