Enhanced Sparse Bayesian Learning via Statistical Thresholding for Signals in Structured Noise

Martin Hurtado, *Member, IEEE*, Carlos H. Muravchik, *Senior Member, IEEE*, and Arye Nehorai, *Fellow, IEEE*

***Abstract*—In this paper we address the problem of sparse signal reconstruction. We propose a new algorithm that determines the signal support applying statistical thresholding to accept the active components of the model. This adaptive decision test is integrated into the sparse Bayesian learning method, improving its accuracy and reducing convergence time. Moreover, we extend the formulation to accept multiple measurement sequences of signal contaminated by structured noise in addition to white noise. We also develop analytical expressions to evaluate the algorithm estimation error as a function of the problem sparsity and indeterminacy. By simulations, we compare the performance of the proposed algorithm with respect to other existing methods. We show a practical application processing real data of a polarimetric radar to separate the target signal from the clutter.**

***Index Terms*—Bayesian estimation, constant false alarm rate (CFAR), probabilistic framework, radar, radar detection, sparse model, sparse signal reconstruction, statistical thresholding.**

I. Introduction

In recent years, many techniques for solving the sparse inverse problem have been proposed. The sudden growth of this topic came about because sparse models can be used in several areas of signal processing, including image processing, bioengineering, communications, and remote sensing. To name but a few interesting problems, sparse signal reconstruction has been successfully applied to signal denoising [1], [2]; DNA micro-arrays [3]; EEG/MEG source imaging [4], [5]; sub-Nyquist sampling [6]–[8]; channel estimation [9], [10]; source localization with sensor arrays [11]; radar detection and estimation [12]–[14]; MIMO radar [15], [16]; and radar imaging [17], [18]. In this paper, we present a new technique to reconstruct sparse signals by applying statistical signal processing tools.

The canonical form of the sparse representation is a linear regression model. The observed signal accepts a representation over a dictionary, which is a collection of known waveforms (atoms) [19]. The regression coefficients provide instruction on how to combine the atoms linearly. The representation is sparse if only a few coefficients are significant. The same problem can be alternatively discussed in the context of component analysis for source separation [20]. Frequently, the number of available observations is smaller than the number of atoms in the dictionary. Then, this linear problem becomes under-determined and it cannot be solved by conventional means. Nevertheless, when the vector of regression coefficients is sparse, it is still possible to solve the inverse problem, even if its support is unknown [21], [22]. In theory, it is possible to recover the sparsest vector of coefficients by proving all possible sparse supports and retaining the solution that minimizes sparsity and the error between the model and the observations. This constrained minimization of the ℓ_0 norm is a combinatorial optimization problem, which is unfeasible even for vectors of reduced size [21].

Fortunately, several tractable methods have been proposed for retrieving signal representations in over-complete dictionaries. One popular approach is greedy pursuit. A greedy algorithm approximates the solution iteratively. At each stage, it computes the residual between the observations and the former solution; then, it updates the support, searching for the dictionary atom that correlates best with the residual. Examples of methods from this group are matching pursuit (MP) [23] and orthogonal matching pursuit (OMP) [24], which can produce fast results with moderate accuracy. Variants that can improve the accuracy by proper tuning are stagewise orthogonal matching pursuit (StOMP) [25] and compressive sampling matching pursuit (CoSaMP) [26]. A different strategy to solve the problem is to relax the cost function of the minimization by replacing the ℓ_0 norm with ℓ_p norms for some $p \in (0, 1]$. An example is the focal underdetermined system solver (FOCUSS)[27], which applies iterative re-weighted least squares (IRLS) to solve the ℓ_p norm. When the ℓ_1 norm is applied, it results in a convex optimization problem with a unique global solution. The new solution can be achieved by linear programming but with considerable computational burden. For instance, basis pursuit (BP) solves the noise-free problem [19]. To account for measurement noise, other relaxation methods are available, such as basis pursuit denoising (BPDN) [19] (also known as LASSO [28]) and Dantzig selector (DS) [29]. The methods called sparse reconstruction by separable approximation (SpaRSA) [30] and Nesterov's algorithm (NESTA) [31] are efficient approaches for solving the optimization problem, suitable for large sparse problems.

Manuscript received September 11, 2012; revised January 24, 2013, May 12, 2013, and July 29, 2013; accepted August 04, 2013. Date of publication August 15, 2013; date of current version September 27, 2013. The associate editor coordinating the review of this manuscript and approving it for publication was Prof. Ljubisa Stankovic. The work of M. Hurtado and C. H. Muravchik was supported by ANPCyT (PICT 0909, PICT PRH 97), CONICET (PIP 114-200901-00391), UNLP (11-I-166), and CICpBA. The work of A. Nehorai was supported by an NSF Grant CCF-1014908, AFOSR Grant FA9550-11-1-0210, and ONR Grant N000141310050.

M. Hurtado and C. H. Muravchik are with the Department of Electrical Engineering, National University of La Plata, Argentina (e-mail: martin.hurtado@ing.unlp.edu.ar; carlosm@ing.unlp.edu.ar).

A. Nehorai is with the Department of Electrical and Systems Engineering, Washington University, St. Louis, MO 63130 USA (e-mail: nehorai@ese.wustl.edu).

This paper has supplementary downloadable multimedia material available at http://ieeexplore.ieee.org provided by the authors. This includes the MATLAB code to reproduce the presented results. This material is 200 KB in size.

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.2013.2278811

In this work, we are interested in techniques developed under the probabilistic framework. Most of the probabilistic methods for sparse signal reconstruction are Bayesian and exploit priors on the regression coefficients for regularizing the under-determined problem [32]. These techniques outperform other approaches when the probabilistic model is a reasonable representation of the physical process that generates the observations. Furthermore, the Bayesian formulation provides more information by estimating the posterior distribution of the coefficients instead of their point estimate.

One method that has attractive properties is sparse Bayesian learning (SBL). It was developed for the relevance vector machine (RVM) [33] and adapted for basis selection from overcomplete dictionaries [34]. Later, it was applied to the inversion of compressive measurements and renamed Bayesian compressive sensing (BCS) [35]. The SBL algorithm assumes that the regression coefficients are independent random variables with zero-mean Gaussian distribution. The variance of the coefficients is treated as a hyper-parameter that is learned from the observations by maximizing the likelihood function, usually using the expectation-maximization (EM) algorithm. Theoretical analysis demonstrated that the likelihood function of the SBL hyper-parameters achieves a global maximum at the sparsest solution and that the local maxima are also sparse [34]. The shrinking procedure that brings to zero irrelevant hyper-parameters has substantial complexity and is time demanding. The BCS method reduces the computation time by adopting a fast RVM algorithm [36] but at the expense of some loss of performance.

Nevertheless, there exist other probabilistic algorithms that propose different formulations. For instance, automatic double overrelaxation (ADORE) employs model selection for determining the signal sparsity and then applies hard thresholding to ensure that the signal meets the estimated sparsity [37]. On the other hand, the fast Bayesian matching pursuit algorithm (FBMP) applies model averaging instead of model selection, considering a Gaussian mixture as the prior of the unknown coefficients [38], similar to the work in [32]. Additionally, several greedy and relaxation algorithms have a Bayesian interpretation in which the prior on the regression coefficients is a distribution promoting sparsity [39], [40].

A generalization of the former sparse problem arises when several measurements are available and it is assumed that they can be represented by the same atoms of the dictionary, sharing a common sparsity profile. This formulation is a realistic representation of multiple snapshots of a signal corrupted by noise. This new problem is known as sparse representation of multiple measurement vectors (MMV) [41] or simultaneous sparse approximation (SSA) [42]. Several methods for the single measurement case have been extended to solve the MMV problem. For instance, MMP and MOMP are extensions of the standard greedy pursuit algorithms [43], [44]. Among the convex relaxation class, it is found the MFOCUSS algorithm [43], as well as the Rx-Penalty and Rx-Error approaches [45]. The MBSL method is the multiple response counterpart of sparse Bayesian learning [46]. Additionally, these six algorithms were extended for complex-valued measurements, dictionaries, and weights [43]–[46].

Most of the algorithms for sparse signal reconstruction consider the observed data as a linear signal contaminated by additive noise. However, in many situations it is more realistic to introduce a third term in the model to account for interfering signals [47]. This term is known as structured noise because it presents a structure dependent on the system mapping the interference into the measurements [48]. Nevertheless, it is given different names for specific applications. The interference is generally called clutter in radar, reverberation in active sonar, and multipath in wireless communications. Any conventional sparse method could address this kind of problem by merging the interference and the additive (unstructured) noise. The performance of the algorithms could be improved if it were possible to incorporate the information of the interference structure. Applying a stochastic representation under the probabilistic framework seems to be the proper approach for this purpose.

In this paper, we develop a new algorithm for recovering multiple sparse signals from corrupted measurements based on the SBL method. Our contribution herein is two-fold. First, we propose a more general probabilistic model to account for data corresponding to signal plus interference plus noise. Therefore, the algorithm gains knowledge on the interference structure, improving its capability to discriminate the signal from the contamination sources. Second, we introduce a pruning procedure during the optimization of the likelihood function. The parameter space is reduced progressively, accelerating the learning process and decreasing the estimation error. For this sparsification of the model, we design a statistical decision test that differentiates active from idle atoms of the dictionary. The statistic derived from the test has a well-known distribution that depends neither on the power of the signal, the interference, nor the noise. Then, the sparsification test has a constant false alarm rate (CFAR), which makes it robust against possible changes in the operating conditions. Moreover, exploiting the probability of detection and false alarm of the atoms, it is possible to fairly predict the estimation error induced by the proposed algorithm.

Other Bayesian approaches also apply pruning to improve their convergence rate. In [33], [46], the model components are removed when their weights fall below a fixed threshold. In [37], only the largest components are held to reach a certain sparsity level. The choice of the threshold has a significant effect on the estimation error (see [49]); then, these methods require fine tuning of the threshold value. The pruning procedures in [36] and [50] which retain components above a specific signal-to-noise ratio seem to be more robust; however, no detailed analysis was elaborated regarding this issue.

The paper is organized as follows. In Section II, we present a description of the sparse problem and summarize the proposed solution. Readers interested in the specifics of the algorithm should refer to Sections III and IV. In Section V, we develop analytical expressions to evaluate the estimation error. The algorithm performance is also numerically compared to other techniques in Section VI. In Section VII, we assess the algorithm's capabilities in a real polarimetric radar problem, detecting and estimating a small target in the presence of sea clutter. Finally, we provide concluding remarks in Section VIII.

Notation

We adopt the following notation. A scalar variable is denoted x, a vector is $\boldsymbol{x}$ and a matrix is X. The vectors $\boldsymbol{x}_{\cdot j}$ and $\boldsymbol{x}_{i\cdot}$ represent the j^{th} column and i^{th} row of matrix X, respectively. The scalar x_{ij} is the ij^{th} entry of matrix X, and x_i is the i^{th} element of vector $\boldsymbol{x}$. The identity matrix of size p is I_p. The conjugate transpose of a complex vector and matrix is $\boldsymbol{x}^H$ and X^H. The trace of a square matrix X is $\mathrm{tr}(X)$ and the block trace is $\mathrm{btr}(X)$, as defined in Appendix A. The operator $\otimes$ denotes the Kronecker product.

The random scalar $x \sim \chi^2_n$ has chi-squared distribution with n degrees of freedom. The random vector $\boldsymbol{x} \sim \mathcal{CN}_p(\boldsymbol{\mu}, \Sigma)$ is a circularly symmetric complex Gaussian vector of dimension p with mean $\boldsymbol{\mu}$ and covariance Σ.

II. Problem Description

A. Data Model

In order to represent measurements that correspond to signal plus interference plus noise, we consider the class of linear mixed models with K independent random factors,

$$\boldsymbol{y} = X\boldsymbol{b} + \sum_{k=1}^{K} Z_k \boldsymbol{u}_k + \boldsymbol{e}, \tag{1}$$

where X and Z_k are the matrices of regressors, $\boldsymbol{y}$ is the measurement vector of dimension N, and $\boldsymbol{b}$ is a vector of fixed effects of dimension M, to be estimated. The k^{th} component of the random effects is the vector $\boldsymbol{u}_k \sim \mathcal{CN}_Q(0, \Sigma_u)$. The vector of residuals is $\boldsymbol{e} \sim \mathcal{NN}_N(0, \sigma I_N)$. The observed data can be represented in a more compact form,

$$\boldsymbol{y} = X\boldsymbol{b} + Z\boldsymbol{u} + \boldsymbol{e}, \tag{2}$$

by stacking the K random effects in a single vector $\boldsymbol{u} \sim \mathcal{CN}_{KQ}(0, I_K \otimes \Sigma_u)$ and concatenating the corresponding matrices in $Z = [Z_1, \ldots, Z_K]$. Now, considering that a sequence of D independent measurement vectors are available, the data model becomes

$$Y = XB + ZU + E, \tag{3}$$

where matrices Y, B, U, and E are built using the respective vectors from (2) as their columns; for instance $Y = [\boldsymbol{y}_{\cdot 1}, \ldots, \boldsymbol{y}_{\cdot D}]$. All matrices in (3) are complex.

B. Sparse Formulation

In the problem depicted by expressions (2) and (3), matrix X is a known dictionary featuring a number M of atoms (columns) significantly larger than the dimension N of the observed signal. We assume that only a few columns of the over-complete dictionary are needed to recreate the uncorrupted signal. Then, the vectors $\boldsymbol{b}_{\cdot d}$ are sparse, given that most of their M entries are zero. We consider it is likely that the D observations of the sequence follow the same behavior. Then, these vectors share the same support, although the weight values can change from one observation to the other. Hence, matrix B has a small number of nonzero rows. The goal is to estimate the columns of B under the condition of common sparsity profile.

In the term corresponding to the interference, matrices Z_k are also known, which is a common assumption in communications and remote sensing. We also consider that these matrices are full column rank. This implies that $N \geq KQ$. However, the covariance Σ_u of the interference sources is unknown. The variance σ of the noise is likewise unknown. These two quantities are considered nuisance parameters, and they will be estimated from the observed data.

C. Proposed Solution

In order to solve the former inverse problem, herein we adapt the SBL formulation to accommodate the model for signal plus interference plus noise. In addition, we introduce an adaptive sparsification procedure that helps to speed up the convergence and to improve the estimation of the weighting coefficients.

Following the SBL framework [34], [46], we treat the columns of B as vectors with zero-mean Gaussian distribution

$$p(\boldsymbol{b}_{\cdot d}; \boldsymbol{\gamma}) = \prod_{m=1}^{M} \frac{1}{\pi\gamma_m} \exp\left(-\frac{|b_{md}|^2}{\gamma_m}\right) = \mathcal{CN}_M(0, \Gamma), \tag{4}$$

for $d = 1, \ldots, D$, where $\Gamma = \mathrm{diag}(\boldsymbol{\gamma})$ and $\boldsymbol{\gamma} = [\gamma_1, \ldots, \gamma_M]^T$ is the vector of M hyper-parameters controlling which components of the dictionary are active. Additionally, the rows of B are $p(\boldsymbol{b}_{m\cdot}; \gamma_m) = \mathcal{CN}_D(0, \gamma_m I_D)$. Then, the prior distribution of matrix B is

$$p(B; \boldsymbol{\gamma}) = \prod_{d=1}^{D} p(\boldsymbol{b}_{\cdot d}; \boldsymbol{\gamma}) = \prod_{m=1}^{M} p(\boldsymbol{b}_{m\cdot}; \gamma_m). \tag{5}$$

If the set of parameters $\Theta = \{\boldsymbol{\gamma}, \Sigma_u, \sigma\}$ is known, by *Lemma 2* in Appendix A the posterior density of the weights given the measurements is

$$p(B/Y; \Theta) = \prod_{d=1}^{D} \mathcal{CN}_M(\hat{\boldsymbol{b}}_{\cdot d}, \Sigma_{B/Y}), \tag{6}$$

where the posterior covariance and mean are, respectively,

$$\Sigma_{B/Y} = \Gamma - \Gamma X^H \Sigma_y^{-1} X \Gamma \tag{7}$$

$$\hat{B} = [\hat{\boldsymbol{b}}_{\cdot 1}, \ldots, \hat{\boldsymbol{b}}_{\cdot D}] = \Gamma X^H \Sigma_y^{-1} Y \tag{8}$$

with $\Sigma_y = X\Gamma X^H + Z(I_K \otimes \Sigma_u)Z^H + \sigma I_N$. The latter Bayesian mean provides a point estimate of the weights. Nevertheless, since these parameters are unknown a priori, the SBL formulation suggests they should be estimated by the maximization of the likelihood function [33], [34]:

$$\begin{aligned} p(Y; \Theta) &= \int p(Y/B; \Sigma_u, \sigma) p(B; \boldsymbol{\gamma}) dB \\ &= |\pi\Sigma_y|^{-D} \exp\left[-D\mathrm{tr}(\Sigma_y^{-1} S_y)\right], \end{aligned} \tag{9}$$

where matrix S_y is

$$S_y = \frac{1}{D} \sum_{d=1}^{D} \boldsymbol{y}_{\cdot d} \boldsymbol{y}_{\cdot d}^H = \frac{1}{D} Y Y^H. \tag{10}$$

Since there is no closed form solution for the values of Θ that maximize (9), we apply the expectation-maximization (EM) algorithm to solve the estimation problem numerically.

- Initialization: Set $i = 0$ and

$$S_y = D^{-1} Y Y^H$$
$$\Gamma^{(0)} = I_M$$
$$C_u^{(0)} = I_K \otimes \Sigma_0$$
$$\sigma^{(0)} = \sigma_0$$

- Iteration: Increment i by 1 and perform these steps
  - EM Step:

$$\gamma_m^{(i)} = \gamma_m^{(i-1)} + (\gamma_m^{(i-1)})^2 \boldsymbol{x}_{\cdot m}^H G \boldsymbol{x}_{\cdot m}$$
$$\sigma^{(i)} = N^{-1} \mathrm{tr}\left[\sigma^{(i-1)} I_N + (\sigma^{(i-1)})^2 G\right]$$
$$\Sigma_u^{(i)} = K^{-1} \mathrm{btr}\left[C_u^{(i-1)} + C_u^{(i-1)} Z^H G Z C_u^{(i-1)}\right]$$

  - Update Step:

$$\Gamma^{(i)} = \mathrm{diag}(\gamma_1^{(i)}, \ldots, \gamma_M^{(i)})$$
$$C_u^{(i)} = I_K \otimes \Sigma_u^{(i)}$$
$$\Sigma_y^{(i)} = X \Gamma^{(i)} X^H + Z C_u^{(i)} Z^H + \sigma^{(i)} I_N$$
$$G = (\Sigma_y^{(i)})^{-1} S_y (\Sigma_y^{(i)})^{-1} - (\Sigma_y^{(i)})^{-1}$$

  - Pruning Step: Set $\gamma_m^{(i)} = 0$ if

$$\frac{2D \boldsymbol{x}_{\cdot m}^H (\Sigma_y^{(i)})^{-1} S_y (\Sigma_y^{(i)})^{-1} \boldsymbol{x}_{\cdot m}}{\boldsymbol{x}_{\cdot m}^H (\Sigma_y^{(i)})^{-1} \boldsymbol{x}_{\cdot m} [1 - \gamma_m \boldsymbol{x}_{\cdot m}^H (\Sigma_y^{(i)})^{-1} \boldsymbol{x}_{\cdot m}]} < \alpha$$

  - Stop Rule: Iterate if

$$\mathrm{Err} = \sum_{m=1}^{M} [\gamma_m^{(i)} - \gamma_m^{(i-1)}]^2 > \mathrm{Tol}$$

- Output: After i iterations,

$$\hat{B} = \Gamma^{(i)} X^H (\Sigma_y^{(i)})^{-1} Y$$

Fig. 1. Pseudo-code of the proposed algorithm.

Throughout the estimation, many elements of the vector $\boldsymbol{\gamma}$ are driven near to zero. In [33], [46], the hyper-parameters γ_m are set to zero when they reach the machine precision, or some arbitrary small value, to guarantee the sparsity of the posterior mean in (8). The pruning also avoids ill-conditioned situations and accelerates the estimation. Motivated by this result, we devised a sparsification procedure that adaptively decides which columns of the dictionary are active or idle based on the statistical features of the data, instead of using an arbitrary fixed threshold. This detection test is embedded in the EM algorithm, accelerating the convergence of the learning process. The proposed algorithm is summarized in Fig. 1. We provide details of the EM and Update Steps in Section III and the Pruning Step in Section IV.

We remark that the proposed formulation can be applied to solve more specific problems. If prior knowledge indicates that there are no interference signals or their effects can be disregarded, the matrix Z should be set equal zero. This action forces the model in (3) to represent the more common problem of signals corrupted by additive noise without disrupting the functioning of the code listed in Fig. 1. Similarly, no change is needed in the algorithm code for processing a single measurement vector. Although the problem is defined in the complex domain, it is possible to solve real-valued problems by removing the factor two of the pruning step. This factor is related to the degree of freedom of the probability distribution used to classify the signal components. This issue will become clearer after Section IV.

III. Parameter Estimation

The EM algorithm is a numerical method to determine the maximum likelihood estimator (MLE). This algorithm exploits the fact that the observed (incomplete) data Y are related to the artificial data V, called complete data, which simplify the computation of the MLE. The algorithm is iterative, with each iteration computing the following expectation and maximization steps[1]:

$$\mathrm{E-Step}: \quad Q^{(i)}(\Theta) = \mathrm{E}\left[\ln p(V;\Theta)/Y;\Theta^{(i)}\right] \tag{11}$$
$$\mathrm{M-Step}: \quad \Theta^{(i+1)} = \arg\max_{\Theta} Q^{(i)}(\Theta). \tag{12}$$

To solve the E-step, we formulate the columns of V to be $\boldsymbol{v}_{\cdot d} = [\boldsymbol{b}_{\cdot d}^H, \boldsymbol{u}_{\cdot d}^H, \mathbf{e}_{\cdot d}^H]^H$. Applying Bayes' theorem, the density of each column is

$$\begin{aligned} p(\boldsymbol{v}_{\cdot d};\Theta) &= p(\boldsymbol{b}_{\cdot d}/\boldsymbol{u}_{\cdot d}, \boldsymbol{e}_{\cdot d};\Theta) p(\boldsymbol{u}_{\cdot d}/\boldsymbol{e}_{\cdot d};\Theta) p(\boldsymbol{e}_{\cdot d};\Theta) \\ &= p(\boldsymbol{b}_{\cdot d};\boldsymbol{\gamma}) p(\boldsymbol{u}_{\cdot d};\Sigma_u) p(\boldsymbol{e}_{\cdot d};\sigma). \end{aligned} \tag{13}$$

Then, since the observations are assumed independent, the logarithmic likelihood of the complete data is

$$\begin{aligned} \ln p(V;\Theta) &= \sum_{d=1}^{D} \ln p(\boldsymbol{v}_{\cdot d};\Theta) \\ &= \sum_{d=1}^{D} \big[\ln p(\boldsymbol{b}_{\cdot d};\boldsymbol{\gamma}) + \ln p(\boldsymbol{u}_{\cdot d};\Sigma_u) \\ &\quad + \ln p(\boldsymbol{e}_{\cdot d};\sigma)\big] \\ &= \sum_{m=1}^{M} \ln p(\boldsymbol{b}_{m\cdot};\gamma_m) \\ &\quad + \ln p(U;\Sigma_u) + \ln p(E;\sigma) \end{aligned} \tag{14}$$

where the first term in the last line of (14) results from using the density (5). The expression (14) shows that the computation of one parameter is dissociated from the others. Therefore, the function Q becomes

$$Q^{(i)}(\Theta) = \sum_{m=1}^{M} Q^{(i)}(\boldsymbol{\gamma}_m) + Q^{(i)}(\Sigma_u) + Q^{(i)}(\sigma). \tag{15}$$

The expectation terms on the right-hand side are explicitly

$$\begin{aligned} Q^{(i)}(\boldsymbol{\gamma}_m) &= -D\left(\ln\pi + \ln\gamma_m + \gamma_m^{-1}\omega_{b_m/Y}^{(i)}\right) \\ Q^{(i)}(\Sigma_u) &= -DK\left[\ln\pi + \ln|\Sigma_u| + \mathrm{tr}\left(\Sigma_u^{-1}\Omega_{U/Y}^{(i)}\right)\right] \\ Q^{(i)}(\sigma) &= -DN\left(\ln\pi + \ln\sigma + \sigma^{-1}\omega_{E/Y}^{(i)}\right) \end{aligned} \tag{16}$$

[1]The classical notation for the EM algorithm explicitly defines Q as a function of the parameter Θ and the parameter evaluated at the i^{th} iteration. To shorten the notation, we redefine this function as $Q^{(i)}(\Theta) = Q(\Theta, \Theta^{(i)})$.

where $\omega_{b_m/Y}^{(i)}$, $\Omega_{U/Y}^{(i)}$, and $\omega_{E/Y}^{(i)}$ are the second-order moment conditioned to the incomplete data Y. Applying *Lemma 2* in Appendix A, the second-order moment of $\boldsymbol{b}_{m\cdot}$ is

$$\begin{aligned}\omega_{b_m/Y}^{(i)} &= \frac{1}{D}\sum_{d=1}^{D}\mathrm{E}\left[|b_{md}|^2/Y;\Theta^{(i)}\right] \\ &= \gamma_m^{(i)} + (\gamma_m^{(i)})^2 \boldsymbol{x}_{\cdot m}^H G \boldsymbol{x}_{\cdot m}\end{aligned} \tag{17}$$

where, to shorten the expression, we defined the matrix $G = (\Sigma_y^{(i)})^{-1} S_y (\Sigma_y^{(i)})^{-1} - (\Sigma_y^{(i)})^{-1}$, with the covariance of the incomplete data $\Sigma_y^{(i)}$ built using the parameter estimates available at the i^{th} iteration. Similarly, the second-order moments of E and U are, respectively,

$$\begin{aligned}\omega_{E/Y}^{(i)} &= \frac{1}{N}\mathrm{tr}\left[\sigma^{(i)} I_N + (\sigma^{(i)})^2 G\right] \\ \Omega_{U/Y}^{(i)} &= \frac{1}{K}\mathrm{btr}\left[C_u^{(i)} + C_u^{(i)} Z' G Z C_u^{(i)}\right]\end{aligned} \tag{18}$$

where $C_u^{(i)} = I_K \otimes \Sigma_u^{(i)}$.

For the M-step, we maximize expression (15) with respect to the unknown parameters. Because each term of this expression depends only on γ_m, σ, or Σ_u, the maximization is separated into three decoupled optimization problems with closed-form solution. The maximization of $Q^{(i)}(\boldsymbol{\gamma}_m)$ and $Q^{(i)}(\sigma)$ is a simple, straightforward procedure solved by equating their derivatives to zero, and for the maximization of $Q^{(i)}(\Sigma_u)$, we apply *Lemma 1* stated in Appendix A. These results correspond to the EM Step and the Update Step of the proposed algorithm summarized in Fig. 1.

The maximization of the likelihood function using the EM algorithm requires to update and invert the covariance of the incomplete data Σ_y at each iteration. Because the inversion of this matrix involves $\mathcal{O}(N^3)$ operations, this procedure becomes computationally expensive for large-scale problems. More efficient variations of the SBL algorithm have been proposed, for example, applying the belief propagation algorithm [51]. Extending our method to manage problems for large values of the dimensions M and N is beyond the scope of the present paper. However, it will be the subject of future research.

This algorithm is recursive and requires initial values of the unknown parameters to begin the estimation. We initiate the hyper-parameters γ_m using a non-negative value larger than the contribution of the interference and noise to the measurements. For the interference and noise, a first guess of their statistical parameters is useful to start the algorithm. In many communications problems, such as radar, it is reasonable to assume that there are available signal-free data from which the interference and noise parameters can be estimated using, for example, the stochastic MLE [52].

IV. Sparsification Process

The former EM algorithm aims to produce an estimate of the row-sparse matrix B by generating a few significant hyper-parameters γ_m and several small ones. Therefore, we devise a rule to decide which hyper-parameters are negligible. The detection problem is to choose between these two hypotheses

$$\begin{cases}\mathcal{H}_0: & \gamma_j = 0 \\ \mathcal{H}_1: & \gamma_j > 0\end{cases} \tag{19}$$

for $j = 1, \ldots, M$. Under $\mathcal{H}_0$, the posterior probability (6) satisfies $\Pr[\boldsymbol{b}_{j\cdot} = 0/Y; \mathcal{H}_0] = 1$, ensuring that the posterior mean of row $\boldsymbol{b}_{j\cdot}$ is zero [34], [46]. Therefore, choosing $\mathcal{H}_0$ as the true hypothesis is equivalent to pruning column $\boldsymbol{x}_{\cdot j}$ from the model.

At the i^{th} iteration of the EM algorithm, we have the estimate $\Theta^{(i)}$ of the unknown parameters. Then, to compare the two hypotheses, we propose a test based on the logarithmic likelihood ratio of the complete data:

$$\begin{aligned}\mathcal{L}(V) &= \ln p(V;\Theta_1^{(i)}) - \ln p(V;\Theta_0^{(i)}) \\ &= \ln p(\boldsymbol{b}_{j\cdot};\gamma_j^{(i)})\end{aligned} \tag{20}$$

where $\Theta_1^{(i)}$ and $\Theta_0^{(i)}$ are the model parameters under each respective hypothesis. The second line in (20) results from (14) and the fact tha under $\mathcal{H}_0$ we remove from the model the component corresponding to the hyper-parameter γ_j. Taking a monotonically increasing function of the former expression generates an equivalent test. We apply the expectation over the conditional distribution of the complete data V given the incomplete data Y with parameter $\Theta^{(i-1)}$,

$$\begin{aligned}\mathcal{L}'(Y) &= \int \mathcal{L}(V) p(V/Y;\Theta^{(i-1)}) dV = Q^{(i-1)}(\gamma_j^{(i)}) \\ &= -D\left(\ln\pi + \ln\omega_{b_j/Y}^{(i)} + 1\right)\end{aligned} \tag{21}$$

where this result is a consequence of the M-step, defined in (12), evaluated for the first equation in (16). Using (17) and keeping only the term that depends on the observed data, the test statistic becomes

$$\mathcal{T}(Y) = \boldsymbol{t}_j^H S_y \boldsymbol{t}_j, \tag{22}$$

where $\boldsymbol{t}_j = (\Sigma_y^{(i)})^{-1}\boldsymbol{x}_{\cdot j}$. To make a decision about the model hypotheses, we have to establish a threshold for comparing the test evaluated at the data Y. This threshold is fixed to achieve a specific performance, such as the probability of false alarm P_{FA}.

A. Test Performance

To determine the performance of the test defined in (22), we note that it is a quadratic form in S_y which follows a chi-squared distribution. Using *Theorem 1* in Appendix A, the normalized test is

$$\text{Under } \mathcal{H}_0: T_0 = \frac{2D\mathcal{T}(Y)}{\boldsymbol{t}_j^H C_y^{(i)} \boldsymbol{t}_j} \sim \chi_{2D}^2 \tag{23}$$

$$\text{Under } \mathcal{H}_1: T_1 = \frac{2D\mathcal{T}(Y)}{\boldsymbol{t}_j^H \Sigma_y^{(i)} \boldsymbol{t}_j} \sim \chi_{2D}^2 \tag{24}$$

where $C_y^{(i)}$ and $\Sigma_y^{(i)}$ are the covariance of the data Y under the respective hypotheses, which are related by

$$\Sigma_y^{(i)} = C_y^{(i)} + \gamma_j^{(i)} \boldsymbol{x}_{\cdot j} \boldsymbol{x}_{\cdot j}^H. \tag{25}$$

Additionally, the denominator in (24) is

$$\begin{aligned} \boldsymbol{t}_j^H \Sigma_y^{(i)} \boldsymbol{t}_j &= \boldsymbol{x}_{\cdot j}^H (\Sigma_y^{(i)})^{-1} \boldsymbol{x}_{\cdot j} \\ &= \boldsymbol{x}_{\cdot j}^H \boldsymbol{t}_j = \boldsymbol{t}_j^H \boldsymbol{x}_{\cdot j}. \end{aligned} \tag{26}$$

Combining (25) and (26), the denominator in (23) is

$$\begin{aligned} \boldsymbol{t}_j^H C_y^{(i)} \boldsymbol{t}_j &= \boldsymbol{t}_j^H \Sigma_y^{(i)} \boldsymbol{t}_j - \gamma_j^{(i)} \boldsymbol{t}_j^H \boldsymbol{x}_{\cdot j} \boldsymbol{x}_{\cdot j}^H \boldsymbol{t}_j \\ &= \boldsymbol{t}_j^H \Sigma_y^{(i)} \boldsymbol{t}_j \left[1 - \gamma_j^{(i)} \boldsymbol{x}_{\cdot j}^H (\Sigma_y^{(i)})^{-1} \boldsymbol{x}_{\cdot j} \right]. \end{aligned} \tag{27}$$

The matrices $C_y^{(i)}$ and $\Sigma_y^{(i)}$ are positive definite since they are covariance matrices. Then, the scalars $\boldsymbol{t}_j^H C_y^{(i)} \boldsymbol{t}_j$, $\boldsymbol{t}_j^H \Sigma_y^{(i)} \boldsymbol{t}_j$, and $\boldsymbol{x}_{\cdot j}^H (\Sigma_y^{(i)})^{-1} \boldsymbol{x}_{\cdot j}$ are also positive. Therefore, the scalar factor in (27) satisfies

$$0 < 1 - \gamma_j^{(i)} \boldsymbol{x}_{\cdot j}^H (\Sigma_y^{(i)})^{-1} \boldsymbol{x}_{\cdot j} \leq 1. \tag{28}$$

According to these remarks, we decide that $\mathcal{H}_0$ is true if the test statistic (23) falls below the threshold α. The threshold is set for the test to meet a fixed probability of false alarm:

$$P_{\mathrm{FA}} = \Pr\left[T_0 > \alpha; \mathcal{H}_0 \right] = 1 - F_{\chi^2_{2D}}(\alpha), \tag{29}$$

where F represents the cumulative function of the distribution χ^2_{2D}. Hence, the threshold follows from the inverse

$$\alpha = F^{-1}_{\chi^2_{2D}} \left(1 - P_{\mathrm{FA}} \right). \tag{30}$$

Then, the probability of detecting an active component is

$$\begin{aligned} P_D &= \Pr\left[T_0 > \alpha; \mathcal{H}_1 \right] \\ &= \Pr\left[T_1 > \frac{\boldsymbol{t}_j^H C_y^{(i)} \boldsymbol{t}_j}{\boldsymbol{t}_j^H \Sigma_y^{(i)} \boldsymbol{t}_j} \alpha; \mathcal{H}_1 \right]. \end{aligned} \tag{31}$$

Applying (27),

$$\begin{aligned} P_D &= \Pr\left[T_1 > \left(1 - \gamma_j^{(i)} \boldsymbol{x}_{\cdot j}^H (\Sigma_y^{(i)})^{-1} \boldsymbol{x}_{\cdot j} \right) \alpha; \mathcal{H}_1 \right] \\ &= 1 - F_{\chi^2_{2D}} \left[\left(1 - \gamma_j^{(i)} \boldsymbol{x}_{\cdot j}^H (\Sigma_y^{(i)})^{-1} \boldsymbol{x}_{\cdot j} \right) \alpha \right]. \end{aligned} \tag{32}$$

The decision test that allows the detection of idle components and their removal from the model is under the Pruning Step of the proposed algorithm in Fig. 1. We note that the expression of the detection threshold α in (30) does not depend on the interference covariance Σ_u and the variance noise σ, nor on matrices X and Z. Therefore, this decision test has the constant false-alarm rate (CFAR) property. The implication of this property is that once the threshold α is set for a required probability of false alarm P_{FA}, the detection test is completely set and needs no further adjustment even if the power of the noise or interference changes. In radar community, detectors supporting this attribute are termed adaptive.

It is worth to mention that the probability of false alarm P_{FA} behaves as a tuning parameter of our algorithm. Similarly to the regularization parameter in FOCUSS and the trade-off parameter in LASSO, it balances the model residual and the solution sparsity. Setting P_{FA} close to one lowers the threshold α and loosens the pruning. We remark that SBL tends to track the noise by selecting spurious atoms of the dictionary. On the other hand, a low P_{FA} promotes sparser solutions. Nevertheless, the advantage of using the probability of false alarm is twofold. First, the parameter P_{FA} is clearly related to the performance of the test, whereas the counterpart parameter in FOCUSS and LASSO lacks such an intuitive insight. P_{FA} quantifies the confidence in the test. Second, it is more robust because of the CFAR property. This last statement will be validated by simulations in Section VI.

Interestingly, other versions of the SBL method devised pruning rules similar to the expression (23), but starting from different approaches. In [36] the authors solve the maximization of the marginal likelihood function by sequential optimization of the individual model hyper-parameters. This procedure results in a pruning condition which can be reinterpreted as removing those components whose squared weights are below the noise power. In [50], the former work is generalized using the variational approach to solve the SBL problem. The authors briefly mention that their pruning condition follows a χ^2 distribution when the actual weight is zero, but it is a noncentral χ^2 distribution when the weight is not zero. Unfortunately, deeper analysis of this test was not developed.

B. Implementation Issues

The test statistic presented in (22) depends on the model parameters that are being estimated. Since the EM algorithm improves the estimates sequentially, the results of the first iterations may not constitute an accurate approximation to the true parameters. By trial and error, we learned that at least ten iterations of the EM algorithm must be run before engaging the pruning step of the proposed algorithm.

V. Performance Analysis

When developing our algorithm, we have found relatively compact expressions for both the estimation and sparsification steps. Nevertheless, they are not simple enough to provide insight about the algorithm's behavior under different conditions. In this section, we look for closed-form expressions of the algorithm's performance in terms of the signal power as well as model sparsity and indeterminacy. To cover much of the previous work on sparse models, we specialize the problem assuming that the entries of matrix X are $x_{ij} \sim \mathcal{CN}(0, M^{-1})$, independently distributed. At first, we avoid the interference to simplify the problem; at the end of this section, we reintroduce this term and generalize the results.

A. Component Detection

To analyze the probability of detecting an active component, we consider that the hyper-parameters in the support of the columns of B have the same value γ. Then, the inverse of the data covariance is

$$\begin{aligned} \Sigma_y^{-1} &= \left(\gamma X_{\mathbb{S}} X_{\mathbb{S}}^H + \sigma I_N \right)^{-1} \\ &= \frac{1}{\sigma} \left[I_N - X_{\mathbb{S}} \left(X_{\mathbb{S}}^H X_{\mathbb{S}} + \frac{\sigma}{\gamma} I_S \right)^{-1} X_{\mathbb{S}}^H \right] \end{aligned} \tag{33}$$

where $X_{\mathbb{S}}$ is a sub-matrix of X formed by the S columns that correspond to the support of $\boldsymbol{b}_{\cdot d}$, denoted by $\mathbb{S} = \mathrm{Supp}\{\boldsymbol{b}_{\cdot d}\}$. Note that $S \ll M$. The second line in the former expression results from using the inverse matrix identity. Applying *Lemma 3* in Appendix A to solve the product $X_{\mathbb{S}}^H X_{\mathbb{S}}$, the inverse of the covariance is, asymptotically for N large,

$$\Sigma_y^{-1} = \frac{1}{\sigma}\left[I_N - \left(\frac{M\gamma}{N\gamma + M\sigma}\right) X_{\mathbb{S}} X_{\mathbb{S}}^H\right]. \tag{34}$$

Replacing (34) in (32) and again using the same lemma, the probability of detecting the component whose index is $j \in \mathbb{S}$ becomes

$$P_D = 1 - F_{\chi^2_{2D}}\left(\alpha \frac{M\sigma}{N\gamma + M\sigma}\right). \tag{35}$$

We define the signal power $P_s = \mathrm{tr}(\Gamma) = S\gamma$, the noise power $P_n = N\sigma$, and the signal-to-noise ratio $\mathrm{SNR} = P_s/P_n$. Then, the detection probability is

$$\begin{aligned} P_D &= 1 - F_{\chi^2_{2D}}\left[\alpha\left(1 + \mathrm{SNR}\frac{N^2}{SM}\right)^{-1}\right] \\ &= 1 - F_{\chi^2_{2D}}\left[\frac{\alpha}{1 + \mathrm{SNR}\delta/\rho}\right] \end{aligned} \tag{36}$$

where $\delta = N/M$ is a normalized measure of indeterminacy and $\rho = S/N$ is a normalized measure of sparsity [53]. We observe that the probability of detecting an active component increases with the signal-to-noise ratio and the indeterminacy, but it decreases with the value of the sparsity ρ.

B. The Oracle Estimator

The oracle estimator assumes that the support of the columns of B and the statistical parameters of the model are known. Although it would produce the best result, this estimator is unfeasible. In this case, it is a common practice to consider the oracle as a reference of performance to compare against.

Under the former assumptions, the problem becomes the Bayesian linear model described by *Lemma 2* in Appendix A. Then, the mean-squared error produced by the oracle in the estimation of B is

$$\begin{aligned} \mathrm{MSE}^{\mathrm{o}} &= \sum_{m=1}^{M}\sum_{d=1}^{D} \mathrm{E}\left[|\hat{b}_{md} - b_{md}|^2/Y\right] \\ &= \sum_{d=1}^{D} \mathrm{E}\left[\|\hat{\boldsymbol{b}}_{\cdot d} - \boldsymbol{b}_{\cdot d}\|^2/Y\right] = D\mathrm{tr}\left(\Sigma_{B/Y}\right), \end{aligned} \tag{37}$$

where, different from (7), the posterior covariance $\Sigma_{B/Y}$ corresponds only to the components in its support. Using first (A.4) and then *Lemma 3*, the posterior covariance of the oracle estimator is given by

$$\Sigma_{B/Y} = \left(\frac{I_S}{\gamma} + \frac{X_{\mathbb{S}}^H X_{\mathbb{S}}}{\sigma}\right)^{-1} = \left(\frac{M\sigma\gamma}{N\gamma + M\sigma}\right) I_S. \tag{38}$$

Replacing (38) in (37), the mean-squared error becomes

$$\mathrm{MSE}^{\mathrm{o}} = D\frac{SM\sigma\gamma}{N\gamma + M\sigma} = D\frac{P_s}{1 + \mathrm{SNR}\delta/\rho}. \tag{39}$$

We note that, similar to the detection probability, the oracle performance improves with the signal-to-noise ratio and the indeterminacy, and it worsens with the value of the sparsity ρ.

C. Performance of the Proposed Algorithm

The oracle and our proposed algorithm have in common the Bayesian framework for the estimation of the unknown matrix B. If the proposed sparsification step detects the correct active components, it will produce an estimate of the support of the columns of B. Once the support is known, our algorithm will perform similar to the oracle. Nevertheless, a faulty estimation of the support degrades the estimation of B. Therefore, we can develop an analytical expression for the performance of our method by introducing the error induced by the estimation of the support.

In the presence of contaminated data, the sparsification step can fail in the following two ways. First, active components may not be detected and then pruned from the support. Second, idle components may be improperly classified as active. Thus, we can sort the components of the estimated support into three disjoint groups: $\mathbb{T}$ represents the set of T active components properly detected, $\bar{\mathbb{T}}$ is the set of $\bar{T}$ active components not detected, and $\mathbb{F}$ is the set of F idle components falsely detected. Note that $\mathbb{S} = \mathbb{T} \cup \bar{\mathbb{T}}$ and $S = T + \bar{T}$. We still consider that the hyper-parameters corresponding to the components in $\mathbb{S}$ have the same value γ, and we also assume that those in $\mathbb{F}$ have the value ν. Based on this partition, we compute the mean-squared error as the contribution of three terms:

$$\mathrm{MSE} = \mathrm{MSE}_{\mathbb{T}} + \mathrm{MSE}_{\bar{\mathbb{T}}} + \mathrm{MSE}_{\mathbb{F}}. \tag{40}$$

The first term represents the error produced by our algorithm when estimating the weights of the T detected components. It is computed similarly to the oracle performance:

$$\begin{aligned} \mathrm{MSE}_{\mathbb{T}} &= \sum_{m\in\mathbb{T},d} \mathrm{E}\left[|\hat{b}_{md} - b_{md}|^2/Y\right] \\ &= D\frac{TM\sigma\gamma}{N\gamma + M\sigma} = D\frac{P_D P_s}{1 + \mathrm{SNR}\delta/\rho} \end{aligned} \tag{41}$$

where we approximate the ratio T/S by the detection probability calculated in (36). The second term in (40) considers the missed components for which our algorithm assigns a null weight:

$$\begin{aligned} \mathrm{MSE}_{\bar{\mathbb{T}}} &= \sum_{m\in\bar{\mathbb{T}},d} \mathrm{E}\left[|b_{md}|^2/Y\right] \\ &= D\bar{T}\gamma + D\gamma^2 \sum_{m\in\bar{\mathbb{T}}} \boldsymbol{x}_{\cdot m}^H(\Sigma_y^{-1} S_y \Sigma_y^{-1} - \Sigma_y^{-1})\boldsymbol{x}_{\cdot m} \\ &= D\bar{T}\gamma = D(S - T)\gamma = D(1 - P_D)P_s. \end{aligned} \tag{42}$$

The former expression results from first using (A.5) to solve the conditional second-order moment and then replacing the matrix S_y by its expected value Σ_y. The last term in (40) represents the error of estimating the weights of the idle components:

$$\begin{aligned}\mathrm{MSE}_{\mathbb{F}} &= \sum_{m\in\mathbb{F},d} \mathrm{E}\left[|\hat{b}_{md}|^2/Y\right] = \sum_{m\in\mathbb{F}} \|\hat{\boldsymbol{b}}_{m\cdot}\|^2 \\ &= D\nu^2 \sum_{m\in\mathbb{F}} \boldsymbol{x}_{\cdot m}^H \Sigma_y^{-1} \boldsymbol{x}_{\cdot m} \\ &= D(M-S)P_{\mathrm{FA}}\nu^2 \frac{N}{M\sigma}. \end{aligned} \tag{43}$$

This expression is the consequence of using (A.3) and again replacing S_y by Σ_y. Then, applying (34) and *Lemma 3*, we find $\boldsymbol{x}_{\cdot m}^H \Sigma_y^{-1} \boldsymbol{x}_{\cdot m} = N/(M\sigma)$, because $\boldsymbol{x}_{\cdot m}^H X_{\mathbb{S}}$ is a null vector for $m \notin \mathbb{S}$. Finally, we replace $F = (M-S)P_{\mathrm{FA}}$. In order to compute $\mathrm{MSE}_{\mathbb{F}}$, we still have to find the value of ν that deceives the hypothesis test (19), causing the algorithm to accept an idle component. To calculate this value, we first replace S_y by Σ_y in (23) and use (27) to get the following relation for $m \in \mathbb{F}$:

$$\frac{2D}{1-\nu \boldsymbol{x}_{\cdot m}^H \Sigma_y^{-1} \boldsymbol{x}_{\cdot m}} = \alpha. \tag{44}$$

Then, we solve for ν:

$$\nu = \frac{M\sigma}{N}\left(1 - \frac{2D}{\alpha}\right). \tag{45}$$

Using (45) in (43), the last term of the mean-squared error becomes

$$\mathrm{MSE}_{\mathbb{F}} = D(M-S)P_{\mathrm{FA}} \frac{MP_s}{N^2 \mathrm{SNR}} \left(1 - \frac{2D}{\alpha}\right)^2, \tag{46}$$

where, for convenience, we write σ in terms of the signal-to-noise ratio and the signal power.

D. Interference Effect

Until now, we have considered only the effect of the noise. To introduce the interference, we consider the particular case in which the entries of matrix Z are also $z_{ij} \sim \mathcal{CN}(0, M^{-1})$, independently distributed. Additionally, we consider a single factor of interference ($K = 1$). We represent the interference covariance through an equivalent diagonal matrix with the same power: $\Sigma_u = \eta I_Q$, where $\eta = \mathrm{tr}(\Sigma_u)/Q$. The interference plus noise is $\boldsymbol{w}_{\cdot d} = Z\boldsymbol{u}_{\cdot d} + \boldsymbol{e}_{\cdot d}$, whose covariance is $\Sigma_w = \eta Z Z^H + \sigma I_N$. Then, the power of the interference plus noise is

$$\begin{aligned} P_w &= \mathrm{tr}(\Sigma_w) = \eta \mathrm{tr}(Z^H Z) + \sigma N \\ &= P_n\left(1 + \frac{\eta Q}{\sigma M}\right) = P_n\left(1 + \mathrm{INR}\frac{Q}{M}\right) \end{aligned} \tag{47}$$

where we get $Z^H Z = I_Q N/M$ by *Lemma 3*. We also define the interference-to-noise ratio $\mathrm{INR} = \eta/\sigma$. Then, the signal-to-interference-plus-noise ratio (SINR) is

$$\mathrm{SINR} = \frac{P_s}{P_w} = \frac{\mathrm{SNR}}{1 + \mathrm{INR}\frac{Q}{M}}. \tag{48}$$

This expression shows how the signal-to-noise ratio is decreased due to the existence of the interference. Therefore, we can take into account the interference effect by replacing SNR by SINR in previous expressions of this section.

This procedure is not valid for multiple interference factors ($K > 1$), because our algorithm exploits the structure in the interference covariance $I_K \otimes \Sigma_u$ to produce a better estimate of Σ_u, which in turn improves the rejection of the interference sources. Unfortunately, we have not yet been able to find a simple analytical expression as a function of K that can represent this behavior of the algorithm.

VI. Simulation Results

In this section, we propose a set of different cases to study the performance of our algorithm, which we will call Enhanced Sparse Bayesian Learning or ESBL for short. The reported results correspond to the average of 1000 independent realizations. For each realization, we randomly generated matrices X and Z as described in the former section. We also randomly chose the covariance Σ_u and the support and weights of matrix B in each realization. The number of rows of B was fixed to $M = 60$. We set the probability of false alarm to $P_{\mathrm{FA}} = 10^{-4}$ for the pruning step of our algorithm. We have included the MATLAB code needed to reproduce the presented results. This material will be available at http://ieeexplore.ieee.org.

First, we compare the performance of ESBL with respect to the oracle estimator, using both the averaged empirical results and the analytical expression presented in Section V. For this experiment, we considered that the number of available measurements is $D = 5$, and there exists a single interference factor ($K = 1$) of size $Q = 5$ and the interference-to-noise ratio is $\mathrm{INR} = 10$ dB. We changed the length N of the measurement vector and the length S of the support of B to generate different operating conditions. Fig. 2 shows the mean-squared error as a function of the signal-to-interference-plus-noise ratio. We observe that when the SINR is high, our algorithm matches the performance of the oracle because it is able to detect the true support of B with high probability, as stated by expression (36). However, when SINR decreases, our algorithm starts failing to estimate the support. This shortcoming is translated into a larger estimation error with respect to the oracle, due to an increase of missed components and those falsely detected. This behavior is fairly well predicted by expressions (42) and (46).

Additionally, we compare the performance of our method to the following algorithms: MOMP [43], regularized MFOCUSS (for $p = 0.8$) [43], Rx-Penalty [45], Rx-Error [45], and MBSL [46]. As it is mentioned in the Introduction, these five algorithm were devised to process multiple measurement vectors and complex-value data. Some of the algorithms we want to evaluate have parameters to be tuned: MFOCUSS and Rx-Penalty have a trade-off parameter; Rx-Error requires a residual constraint; and ESBL depends on the probability of false alarm. We propose to adjust those parameters to minimize the mean squared error of each algorithm. For the new experiment, we set the length of the measurement vector to $N = 30$, the cardinality of the solution to $S = 5$, the number of measurement vectors to $D = 5$, the

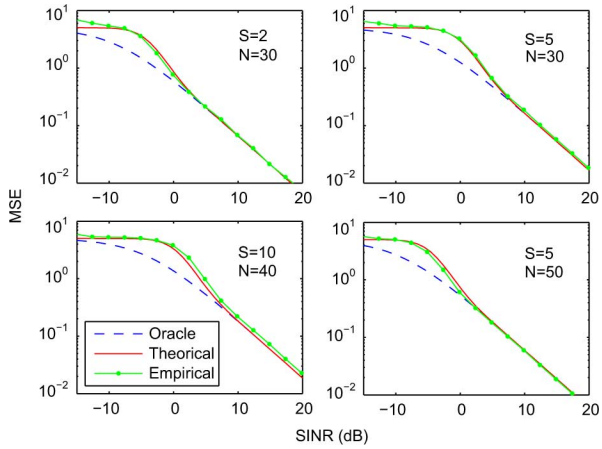


Fig. 2. Mean-squared error of the ESBL algorithm as a function of the signal-to-interference-plus-noise ratio for different cases of sparsity and indeterminacy.

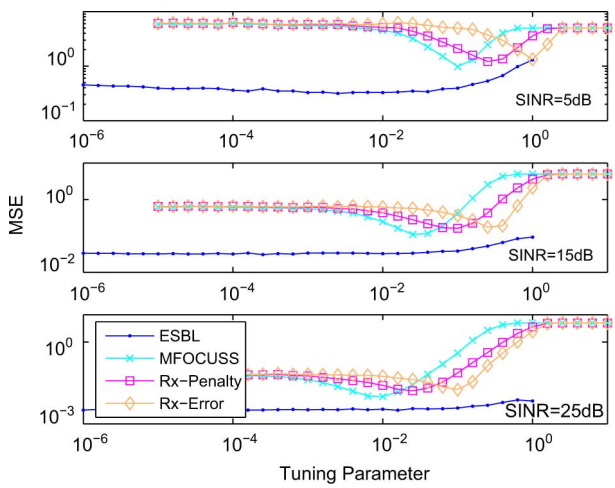


Fig. 3. Performance of ESBL, MFOCUSS, Rx-Penalty, and Rx-Error as a function of their tuning parameter for different SINR conditions.

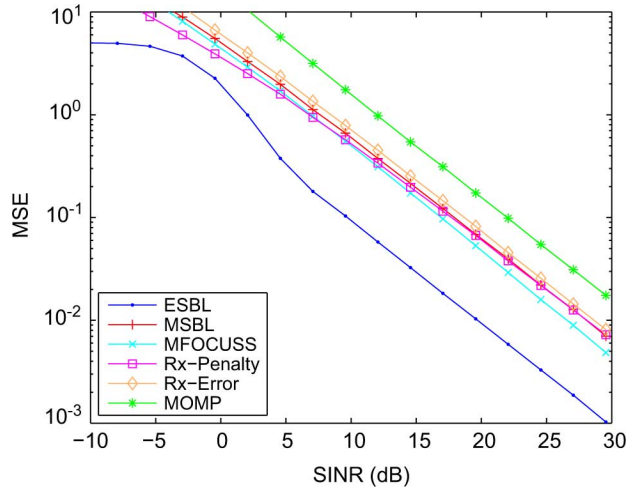


Fig. 4. Mean-squared error when processing multiple measurement vectors corrupted by interference plus noise.

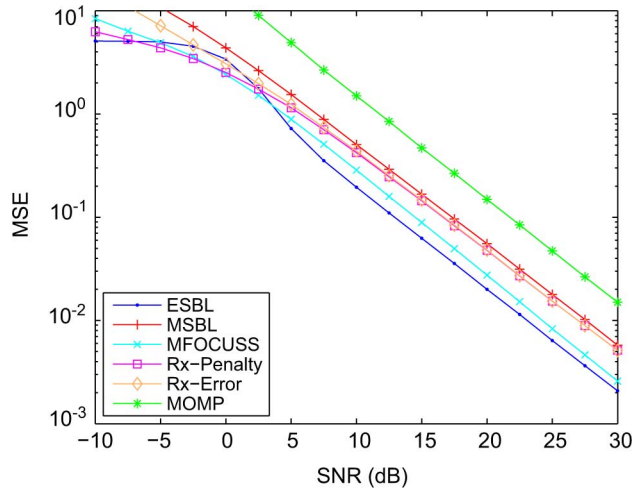


Fig. 5. Mean-squared error when processing multiple measurement vectors corrupted only by additive noise.

number of interference factors to $K = 3$, the dimension of each interference factor to $Q = 5$, and the interference-to-noise ratio to $\mathrm{INR} = 10\,\mathrm{dB}$. Fig. 3 shows examples of the tuning procedure for different scenarios. We observe that MFOCUSS, Rx-Penalty and Rx-Error are very sensitive to the choice of their parameters; and their optimal values vary with the power of the noise and the interference. On the other hand, ESBL performance depicts a wide valley which becomes flatter as the SINR increases. Therefore, for proper comparison of these algorithms, we calibrated MFOCUSS, Rx-Penalty and Rx-Error for each value of SINR, but for ESBL we fixed the probability of false alarm to $P_{\mathrm{FA}} = 10^{-4}$.

After the tuning process, we were able to run simulations in order to compare the different algorithms. Fig. 4 plots the mean-squared error as a function of the SINR. This figure shows that MSBL, MFOCUSS, Rx-Penalty and Rx-Error have similar performance; but MOMP is slightly worse. We also observe that ESBL gains at least 7dB in the signal power with respect to the other methods. ESBL outperforms the other methods because it exploits the knowledge of the interference structure incorporated in the modeling. Then, the magnitude of the improvement depends on the parameters K, Q, and INR.

We also studied the case in which the interference contribution was neglected ($\mathrm{INR} = 0$) and the multiple measurement vectors were corrupted only by noise. The parameters N, S, and D remained as in the former experiment. To handle this particular problem with the ESBL algorithm, we set the matrix Z equal zero forcing our formulation to avoid the interference component. Under this configuration all the algorithms share the same model, but they differ in the approach to solve the inverse sparse problem. Fig. 5 plots the performance results for this scenario. Now the gap between ESBL and the other methods is reduced. However, ESBL is still better than the others; and MFOCUSS results the second-best. By contrasting ESBL and MSBL, we observe the enhancement that results from the pruning step embedded in the EM algorithm.

Moreover, there exist algorithms that can process complex-valued data but they were not extended for multiple measurement vectors. Examples of these methods are DS [29], SpaRSA [30], NESTA [31], and FBMP [38]. In this experiment, we maintained the scenario free of interference ($\mathrm{INR} = 0$) and reduced

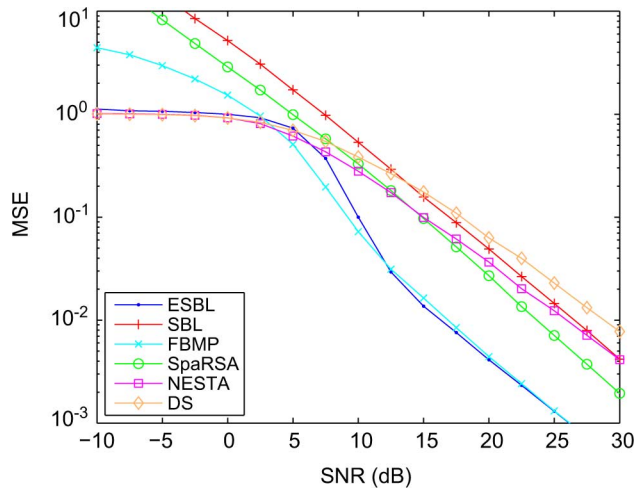


Fig. 6. Mean-squared error when processing a single measurement vector corrupted only by additive noise.

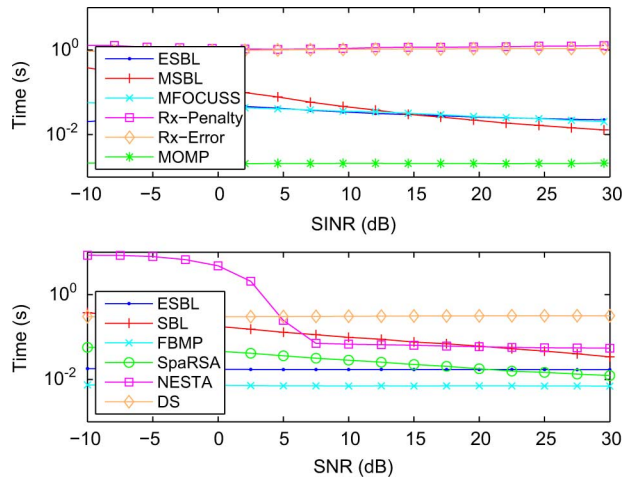


Fig. 7. Elapsed time for solving the simulations: multiple measurement vectors corrupted by interference plus noise (upper plot) and single measurement vector corrupted only by additive noise (lower plot).

the number of measurement vectors to $D = 1$. The parameters N and S remained as before. Beside these four algorithms, we also used SBL for complex data. For ESBL, we set again the matrix Z to zero and the probability of false alarm to $P_{\mathrm{FA}} = 10^{-4}$. For DS, we constrained the residual to $\varepsilon = \sqrt{2\sigma \log M}$. The algorithms SpaRSA, NESTA, and FBMP have tuning parameters which we optimized by trial and error as in the former experiments. Fig. 6 plots the new results. The simulation shows that DS, SpaRSA, NESTA, and SBL have similar performance. We also note that ESBL and FBMP are close, although FBMP is slightly better for moderate SNR. These two algorithms gain approximately 10dB in the signal power with respect to the other methods.

For further characterization of the algorithms, we recorded and averaged the time required to solve the former experiments. We emphasize that an accurate comparison is not possible because the algorithms applied to solve the experiments are based on different approaches and do not use the same stopping condition. Nevertheless, their run-time, depicted in Fig. 7, provides the order of magnitude of their computational complexity. We observe that ESBL runs somewhere in-between the slow relaxation methods and the fast, but inaccurate, greedy pursuit. We also point out the efficiency of FBMP.

Finally, we compute the phase transition diagram to analyze the capabilities of our proposed method. This type of diagram plots the breakdown of an algorithm as a function of indeterminacy and sparsity. The phase above the curve in the diagram indicates failure and the phase below the line is success. Hence, an algorithm performs better than the others if the line of its phase diagram is higher. We estimated the empirical phase transition following the procedure described in [53, SectionIV]. For these simulations, we considered again the scenario free of interference ($\mathrm{INR} = 0$). We set the signal-to-noise ratio to $\mathrm{SNR} = 50$ dB to ensure that the failure of the algorithms would be related to the problem complexity and not because of a weak signal. The upper plot of Fig. 8 shows the diagrams for the algorithms solving the problem of multiple measurement vectors

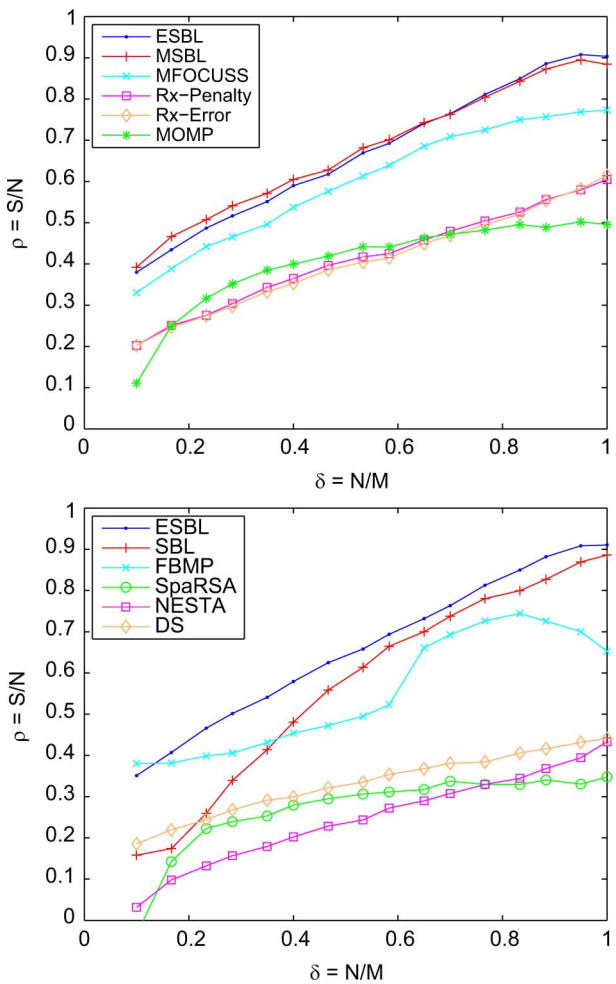


Fig. 8. Empirical phase transitions: algorithms for multiple measurement vectors (upper plot) and for a single measurement vector (lower plot).

($D = 5$), and the lower plot shows the results for the algorithms processing a single measurement vector ($D = 1$). We observe that ESBL has the best performance, followed closely by MSBL.

In summary, we learned from these simulation examples that ESBL outperforms other algorithms when solving the sparse

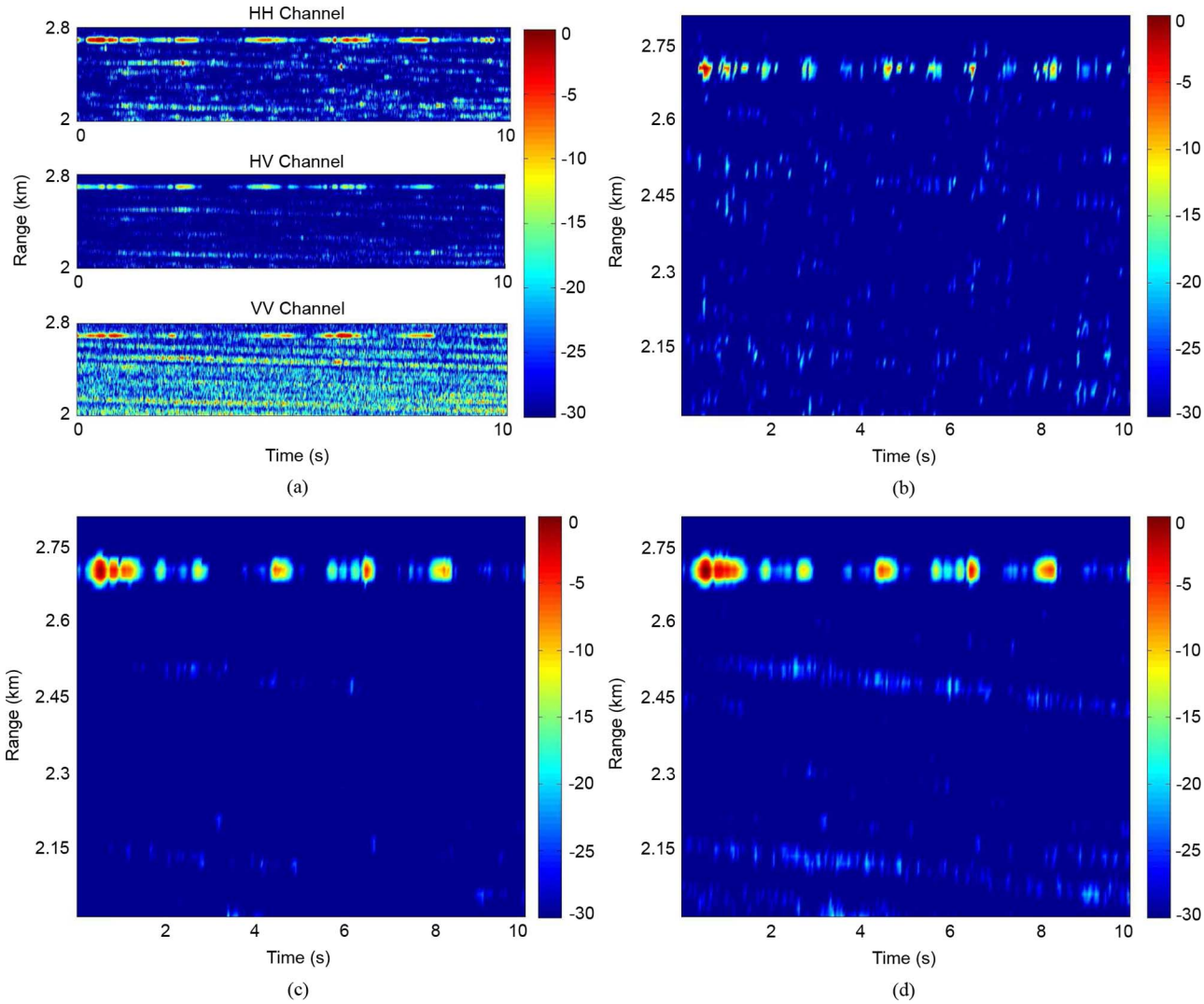


Fig. 9. Radar images: (a) Raw data from different polarimetric channels. (b) PST-GLR algorithm. (c) ESBL algorithm. (d) MFOCUSS algorithm.

inverse problem for multiple measurement vectors corrupted by interference plus noise. This result was fairly expected since ESBL was devised for that specific purpose. However, we showed that ESBL is also highly competitive when there are no interference signals, for the case of multiple measurement vectors as well as for a single measurement vector.

VII. Real Radar Data

The true motivation for developing our algorithm is the detection and estimation of radar targets in the presence of clutter. In radar, clutter denotes the reflections produced by the nearby environment of the target. Radar signal processing in the presence of clutter becomes a challenging problem because it can mask the target reflections and trigger false alarms. Nevertheless, proper accounting of the clutter effects improves target detection. Therefore, in this section we evaluate our algorithm with real polarimetric radar data by representing the clutter as the structured noise in the proposed formulation. Additionally, we briefly overview related work of sparse modeling for radar.

Assuming that the targets are scarce and not densely located in the scene, sparse models and techniques become an interesting alternative to address the radar problem. Most of the previous work involves the development of the dictionary that can be used to represent the radar data as a sparse signal. The usual goal is to improve the system resolution that would lead to better target estimation [12], [15], [16] and radar imaging [17], [18]. Moreover, radar sparse models can be combined with compressive sensing techniques to reduce the amount of data to be processed and stored without significant loss of performance. Once the formulation is defined, authors have applied different conventional techniques to solve the sparse inverse problem. For instance, BP was used to implement a high resolution radar [12], estimate target parameters with a MIMO radar [16], and focus SAR data [17]; DS was also applied to MIMO radar [15]; and OMP was used to reconstruct under-sampled SAR images [18]. Depending on the scenario, clutter can be merely ignored, see for example [15]. When the clutter power is substantial, in [13] it is suggested to pre-process the radar data by using a projection into a clutter-free subspace. This procedure would succeed only if the clutter is known a priori, and its features are stationary and spatially homogeneous. To the best of our knowledge, no previous technique includes the clutter in the sparse inverse problem for reconstructing the target signal.

To test our algorithm, we used data collected with the IPIX polarimetric radar which belongs to McMaster University [54]. We processed the dataset *stare0* recorded on Nov. 11 of 1993 at Dartmouth, Nova Scotia, Canada. The data correspond to a beachball wrapped with aluminum foil floating on the sea surface in mild weather (wave height was 0.67m and wind speed was 21km/h).

For the sparse modeling of the radar data, we generated the overcomplete dictionary by allowing the presence of a target in each range cell that forms the radar footprint. A potential target was represented using Krogager's decomposition of the scattering matrix. We considered nine components: a sphere, left and right helices, and six diplanes with different orientations. In [55], we provided more details about the design of matrices X and Z. We used these two matrices and the radar data to feed our EBSL algorithm. For comparison, we also applied the regularized MFOCUSS algorithm [43] to solve the same problem.

As a reference for the sparse approach, we also processed the data using the polarization-space-time generalized likelihood ratio (PST-GLR) algorithm [56]. This is a well-known radar algorithm for polarimetric data processing. It was derived under the assumption of homogeneous Gaussian clutter, and it employs secondary data to estimate the clutter covariance.

Fig. 9(a) shows different polarimetric channels of the raw data. Note that the VH channel was processed but not displayed here because of its similarity to the HV channel. We remark that the target is located at approximately 2.7km and that there is strong contribution from maritime clutter, especially in the VV channel. Fig. 9(b) plots the result when applying PST-GLR with $P_{\rm FA} = 10^{-2}$. This algorithm recovers the target signature. However, it suffers from a significantly large number of false detections because the sea clutter deviates from the homogeneous Gaussian model. Setting a lower probability of false alarm reduces the number of false detection but also eliminates the target signal. Fig. 9(c) shows the output of ESBL with $P_{\rm FA} = 10^{-8}$ in the pruning step. We note that the target signal is recovered with only a few false detections generated by the strong clutter. However, the magnitude of the false detections is 20dB weaker than the target. Finally, in Fig. 9(d) we observe that MFOCUSS is also able to reconstruct the target signal. It cannot reject as much clutter as ESBL because its model is missing the information of the interference structure.

VIII. Conclusions

In this paper, we proposed a novel algorithm devised from the combination of the SBL method and statistical thresholding. We adopted the Bayesian framework to incorporate structured noise into the model, extending the scope of sparse signal reconstruction to problems of signals corrupted by interference plus noise. In our algorithm, we integrated an adaptive pruning step into the iterative estimation procedure of the regular SBL algorithm, creating a more precise and faster method. In comparison to other existing techniques for sparse signals, our algorithm shows very good tradeoff between accuracy and time efficiency. The computed phase diagrams also demonstrated that, similarly to SBL, our algorithm can assess problems with larger values of sparsity ρ and indeterminacy δ. Moreover, we presented analytical expressions for the algorithm's estimation error, which we developed from the probability of correctly detecting the signal support.

To demonstrate a practical application of the proposed algorithm, we processed real polarimetric radar data that correspond to a target in the presence of strong sea clutter. The results proved that our algorithm effectively rejects the clutter, producing a clean image of the target response.

Further research will include a formal analysis of the convergence of the algorithm. In addition, we will employ our algorithm in the design of adaptive compressive sensing systems that can reduce the number of measurements but also filter out undesired interference signals.

Appendix
Mathematical and Statistical Tools

Definition: Let A be a $(KQ \times KQ)$ matrix whose $(Q \times Q)$ blocks are denoted A_{ij}, for $i, j = 1, \ldots, K$; then the block trace operation is [57]

$$\mathrm{btr}(A) = \sum_{k=1}^{K} A_{kk}. \tag{A.1}$$

Notice that $\mathrm{btr}(A)$ is a $(Q \times Q)$ matrix, not a scalar.

Lemma 1: Let Σ be a $Q \times Q$ positive definite matrix. Then the following inequality holds for any positive definite matrix F:

$$\ln|\Sigma| + \mathrm{tr}(\Sigma^{-1}F) \geq Q + \ln|F|. \tag{A.2}$$

The equality is achieved when $\Sigma = F$.

Proof: See [52].

Lemma 2: Let the observed data be modeled as a Bayesian general linear model: $\boldsymbol{y} = X\boldsymbol{b} + \boldsymbol{w}$, where X is a known matrix of size $N \times M$, $\boldsymbol{b} \sim \mathcal{CN}_M(0, \Sigma_b)$ and $\boldsymbol{w} \sim \mathcal{CN}_N(0, \Sigma_w)$ are independent random vectors. Then, $\boldsymbol{y}$ is complex Gaussian with zero mean and covariance $\Sigma_y = X\Sigma_b X^H + \Sigma_w$. The posterior distribution of $\boldsymbol{b}$ is also complex Gaussian with the following mean and covariance:

$$\boldsymbol{\mu}_{b/y} = \Sigma_b X^H \Sigma_y^{-1} \boldsymbol{y}, \tag{A.3}$$

$$\begin{aligned}\Sigma_{b/y} &= \Sigma_b - \Sigma_b X^H \Sigma_y^{-1} X \Sigma_b \\ &= (\Sigma_b^{-1} + X^H \Sigma_w^{-1} X)^{-1}.\end{aligned} \tag{A.4}$$

The posterior second-order moment is

$$\begin{aligned}\Omega_{b/y} &= \mathrm{E}\left[\boldsymbol{b}\boldsymbol{b}^H / \boldsymbol{y}\right] \\ &= \Sigma_{b/y} + \boldsymbol{\mu}_{b/y}\boldsymbol{\mu}_{b/y}^H \\ &= \Sigma_b + \Sigma_b X^H (\Sigma_y^{-1}\boldsymbol{y}\boldsymbol{y}^H\Sigma_y^{-1} - \Sigma_y^{-1}) X \Sigma_b.\end{aligned} \tag{A.5}$$

Proof: It follows from *Theorem 10.3* in [58].

Lemma 3: Let $X \in \mathbb{C}^{N \times M}$ be a matrix whose elements are $x_{ij} \sim \mathcal{CN}(0, M^{-1})$, independently distributed, and $X_{\mathbb{S}} \in \mathbb{C}^{N \times S}$ be a sub-matrix of X formed by the set $\mathbb{S}$ of its columns. Then, $X_{\mathbb{S}}$ satisfies $\mathrm{E}[X_{\mathbb{S}}^H X_{\mathbb{S}}] = (N/M) I_S$ and, asymptotically, $X_{\mathbb{S}}^H X_{\mathbb{S}} = (N/M) I_S$ for N large.

Proof: The first identity follows from computing the expectation of the entries of the matrix $X_{\mathbb{S}}^H X_{\mathbb{S}}$,

$$\mathrm{E}[\boldsymbol{x}_{\cdot i}^H \boldsymbol{x}_{\cdot j}] = \sum_{n=1}^{N} \mathrm{E}[x_{ni}^* x_{nj}] = \begin{cases} N/M & i = j \\ 0 & i \neq j \end{cases}, \tag{A.6}$$

where $i, j \in \mathbb{S}$. Additionally, note that each row of $X_{\mathbb{S}}$ is $\mathcal{CN}_S(0, M^{-1} I_S)$ and the sample covariance is $X_{\mathbb{S}}^H X_{\mathbb{S}}/N$. Then, the second identity is met when N is large.

Theorem 1: Let $\boldsymbol{y}_d$, for $d = 1, \ldots, D$, be a random sample from the distribution $\mathcal{CN}_N(0, \Sigma_y)$ and denote matrix S_y, defined in (10), as the estimate of the covariance Σ_y. Then, the following quadratic form in S_y has a chi-squared distribution,

$$2D \frac{\boldsymbol{t}^H S_y \boldsymbol{t}}{\boldsymbol{t}^H \Sigma_y \boldsymbol{t}} \sim \chi_{2D}^2, \tag{A.7}$$

where $\boldsymbol{t} \in \mathbb{C}^N - \{0\}_N$.

Proof: First, consider the quadratic form in S_y,

$$D \boldsymbol{t}^H S_y \boldsymbol{t} = \sum_{d=1}^{D} \boldsymbol{t}^H \boldsymbol{y}_d \boldsymbol{y}_d^H \boldsymbol{t} = \sum_{d=1}^{D} |\boldsymbol{t}^H \boldsymbol{y}_d|^2, \tag{A.8}$$

where $\boldsymbol{t}^H \boldsymbol{y}_d \sim \mathcal{CN}_1(0, \boldsymbol{t}^H \Sigma_y \mathbf{t})$. Then, define the complex Gaussian random variable,

$$v = \boldsymbol{t}^H \boldsymbol{y}_d \sqrt{\frac{2}{\boldsymbol{t}^H \Sigma_y \boldsymbol{t}}}. \tag{A.9}$$

By definition of circular symmetry, its real and imaginary parts are $[v_r; v_i] \sim \mathcal{N}_2(0, I_2)$. Then, $|v|^2 \sim \chi_2^2$, because it is the sum of two squared independent standard Gaussian variables. Finally, the distribution of (A.7) is the result of the summation of D independent random variables with distribution χ_2^2.

REFERENCES

[1] J. Haupt and R. Nowak, "Signal reconstruction from noisy random projections," *IEEE Trans. Inf. Theory*, vol. 52, no. 9, pp. 4036–4048, Sep. 2006.

[2] M. Protter, I. Yavneh, and M. Elad, "Closed-form MMSE estimation for signal denoising under sparse representation modeling over a unitary dictionary," *IEEE Trans. Signal Process.*, vol. 58, no. 7, pp. 3471–3484, Jul. 2010.

[3] F. Parvaresh, H. Vikalo, S. Misra, and B. Hassibi, "Recovering sparse signals using sparse measurement matrices in compressed DNA microarrays," *IEEE J. Sel. Topics Signal Process.*, vol. 2, no. 3, pp. 275–285, Jun. 2008.

[4] D. Wipf and S. Nagarajan, "A unified Bayesian framework for MEG/EEG source imaging," *NeuroImage*, vol. 44, no. 3, pp. 947–966, Feb. 2009.

[5] N. v. Ellenrieder, M. Hurtado, and C. Muravchik, "Electromagnetic source imaging for sparse cortical activation patterns," in *Proc. IEEE Int. Conf. Eng. in Medicine and Biology Soc. EMBC*, Sep. 2010, pp. 4316–4319.

[6] J. Tropp, J. Laska, M. Duarte, J. Romberg, and R. Baraniuk, "Beyond Nyquist: Efficient sampling of sparse bandlimited signals," *IEEE Trans. Inf. Theory*, vol. 56, no. 1, pp. 520–544, Jan. 2010.

[7] M. Mishali and Y. Eldar, "From theory to practice: Sub-Nyquist sampling of sparse wideband analog signals," *IEEE J. Sel. Topics Signal Process.*, vol. 4, no. 2, pp. 375–391, Apr. 2010.

[8] A. Harms, W. Bajwa, and R. Calderbank, "Beating Nyquist through correlations: A constrained random demodulator for sampling of sparse bandlimited signals," in *Proc. IEEE Int. Conf. Acoust., Speech and Signal Processing ICASSP*, May 2011, pp. 5968–5971.

[9] W. Bajwa, J. Haupt, A. Sayeed, and R. Nowak, "Compressed channel sensing: A new approach to estimating sparse multipath channels," *Proc. IEEE*, vol. 98, no. 6, pp. 1058–1076, Jun. 2010.

[10] D. Shutin and B. Fleury, "Sparse variational Bayesian SAGE algorithm with application to the estimation of multipath wireless channels," *IEEE Trans. Signal Process.*, vol. 59, no. 8, pp. 3609–3623, Aug. 2011.

[11] D. Malioutov, M. Cetin, and A. Willsky, "A sparse signal reconstruction perspective for source localization with sensor arrays," *IEEE Trans. Signal Process*, vol. 53, no. 8, pp. 3010–3022, Aug. 2005.

[12] M. Herman and T. Strohmer, "High-resolution radar via compressed sensing," *IEEE Trans. Signal Process*, vol. 57, no. 6, pp. 2275–2284, Jun. 2009.

[13] J. Ender, "On compressive sensing applied to radar," *Signal Processing*, vol. 90, no. 5, pp. 1402–1414, May 2010.

[14] L. Anitori, A. Maleki, M. Otten, R. Baraniuk, and P. Hoogeboom, "Design and analysis of compressed sensing radar detectors," in *IEEE Trans. Signal Process.* : .

[15] Y. Yu, A. Petropulu, and H. Poor, "MIMO radar using compressive sampling," *IEEE J. Sel. Topics Signal Process*, vol. 4, no. 1, pp. 146–163, Feb. 2010.

[16] S. Gogineni and A. Nehorai, "Target estimation using sparse modeling for distributed MIMO radar," *IEEE Trans. Signal Process.*, vol. 59, no. 11, pp. 5315–5325, Nov. 2011.

[17] V. Patel, G. Easley, D. Healy, and R. Chellappa, "Compressed synthetic aperture radar," *IEEE Trans. Signal Process.*, vol. 4, no. 2, pp. 244–254, Apr. 2010.

[18] M. Tello Alonso, P. L. Dekker, and J. Mallorqui, "A novel strategy for radar imaging based on compressive sensing," *IEEE Trans. Geosci. Remote Sens.*, vol. 48, no. 12, pp. 4285–4295, Dec. 2010.

[19] S. Chen, D. Donoho, and M. Saunders, "Atomic decomposition by basis pursuit," in *SIAM Rev.* : , 2001.

[20] H. Zayyani, M. Babaie-Zadeh, and C. Jutten, "An iterative Bayesian algorithm for sparse component analysis in presence of noise," *IEEE Trans. Signal Process.*, vol. 57, no. 11, pp. 4378–4390, Nov. 2009.

[21] E. Candes, 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, Feb. 2006.

[22] D. Donoho, "Compressed sensing," *IEEE Trans. Inf. Theory*, vol. 52, no. 4, pp. 1289–1306, Apr. 2006.

[23] S. Mallat and Z. Zhang, "Matching pursuits with time-frequency dictionaries," *IEEE Trans. Signal Process*, vol. 41, no. 12, pp. 3397–3415, Dec. 1993.

[24] J. Tropp and A. Gilbert, "Signal recovery from random measurements via orthogonal matching pursuit," *IEEE Trans. Inf. Theory*, vol. 53, no. 12, pp. 4655–4666, Dec. 2007.

[25] D. Donoho, Y. Tsaig, T. Drori, and J. Starck, Sparse solution of underdetermined linear equations by stagewise orthogonal matching pursuit, Tech. Rep 2006 [Online]. Available: http://www-stat.stanford.edu/~donoho/Reports

[26] D. Needell and J. Tropp, "CoSaMP iterative signal recovery from incomplete and inaccurate samples," *Appl. Computat. Harmon. Anal.*, vol. 26, no. 3, pp. 301–321, 2009.

[27] I. Gorodnitsky and B. Rao, "Sparse signal reconstruction from limited data using FOCUSS a re-weighted minimum norm algorithm," *IEEE Trans Signal Process.*, vol. 45, no. 3, pp. 600–616, Mar. 1997.

[28] R. Tibshirani, "Regression shrinkage and selection via the LASSO," *J. Roy. Stat. Soc., Ser. B*, vol. 58, no. 1, pp. 267–288, 1996.

[29] E. Candes and T. Tao, "The Dantzig selector: Statistical estimation when p is much larger than n," *Ann. Statist.*, vol. 35, no. 6, pp. 2313–2351, 2007.

[30] S. Wright, R. Nowak, and M. Figueiredo, "Sparse reconstruction by separable approximation," *IEEE Trans. Signal Process.*, vol. 57, no. 7, pp. 2479–2493, Jul. 2009.

[31] S. Becker, J. Bobin, and E. Candes, "NESTA a fast and accurate first-order method for sparse recovery," *SIAM J. Imag. Sci.*, vol. 4, no. 1, pp. 1–39, 2011.

[32] E. G. Larsson and Y. Selen, "Linear regression with a sparse parameter vector," *IEEE Trans. Signal Process.*, vol. 55, no. 2, pp. 451–460, Feb. 2007.

[33] M. Tipping, "Sparse Bayesian learning and the relevance vector machine," *J. Mach. Learn. Res.*, vol. 1, pp. 211–244, Sep. 2001.
[34] D. Wipf and B. Rao, "Sparse Bayesian learning for basis selection," *IEEE Trans. Signal Process.*, vol. 52, no. 8, pp. 2153–2164, Aug. 2004.
[35] S. Ji, Y. Xue, and L. Carin, "Bayesian compressive sensing," *IEEE Trans. Signal Process.*, vol. 56, no. 6, pp. 2346–2356, Jun. 2008.
[36] M. Tipping and A. Faul, "Fast marginal likelihood maximization for sparse Bayesian models," in *Proc. Int. Workshop Artif. Intell. Statist.*, 2003.
[37] K. Qui and A. Dogandzic, "Sparse signal reconstruction via ECME hard thresholding," *IEEE Trans. Signal Process.*, vol. 60, no. 9, pp. 4551–4569, Sep. 2012.
[38] P. Schniter, L. Potter, and J. Ziniel, "Fast Bayesian matching pursuit," in *Proc. Inf. Theory Appl. Workshop*, Feb. 2008, pp. 326–333.
[39] B. Rao, K. Engan, S. Cotter, J. Palmer, and K. Kreutz-Delgado, "Subset selection in noise based on diversity measure minimization," *IEEE Trans. Signal Process.*, vol. 51, no. 3, pp. 760–770, Mar. 2003.
[40] M. Elad and I. Yavneh, "A plurality of sparse representations is better than the sparsest one alone," *IEEE Trans. Inf. Theory*, vol. 55, no. 10, pp. 4701–4714, Oct. 2009.
[41] 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, Dec. 2006.
[42] A. Rakotomamonjy, "Surveying and comparing simultaneous sparse approximation or group-LASSO algorithms," *Signal Process.*, vol. 91, no. 7, pp. 1505–1526, 2011.
[43] S. Cotter, B. 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.
[44] J. Tropp, A. Gilbert, and M. Strauss, "Algorithms for simultaneous sparse approximation. Part I: Greedy pursuit," *Signal Process.*, vol. 86, no. 3, pp. 572–588, 2006.
[45] J. Tropp, "Algorithms for simultaneous sparse approximation. Part II: Convex relaxation," *Signal Process.*, vol. 86, no. 3, pp. 589–602, 2006.
[46] D. Wipf and B. Rao, "An empirical Bayesian strategy for solving the simultaneous sparse approximation problem," *IEEE Trans. Signal Process.*, vol. 55, no. 7, pp. 3704–3716, Jul. 2007.
[47] L. Scharf and M. McCloud, "Blind adaptation of zero forcing projections and oblique pseudo-inverses for subspace detection and estimation when interference dominates noise," *IEEE Trans. Signal Process.*, vol. 50, no. 12, pp. 2938–2946, Dec. 2002.
[48] R. Behrens and L. Scharf, "Signal processing applications of oblique projection operators," *IEEE Trans. Signal Process.*, vol. 42, no. 6, pp. 1413–1424, Jun. 1994.
[49] Z. Zhang and B. Rao, Clarify some issues on the sparse Bayesian learning for sparse signal recovery, Tech. Rep. San Diego, Univ. Calif., Sep. 2011 [Online]. Available: https://sites.google.com/site/researchbyzhang/
[50] D. Shutin, T. Buchgraber, S. Kulkarni, and H. Poor, "Fast variational sparse Bayesian learning with automatic relevance determination for superimposed signals," *IEEE Trans. Signal Process.*, vol. 59, no. 12, pp. 6257–6261, 2011.
[51] X. Tan and J. Li, "Computationally efficient sparse Bayesian learning via belief propagation," *IEEE Trans. Signal Process.*, vol. 58, no. 4, pp. 2010–2021, Apr. 2010.
[52] P. Stoica and A. Nehorai, "On the concentrated stochastic likelihood function in array signal processing," *Circuits, Syst., Signal Process.*, vol. 14, pp. 669–674, Sep. 1995.
[53] A. Maleki and D. L. Donoho, "Optimally tuned iterative reconstruction algorithms for compressed sensing," *IEEE J. Sel. Topics Signal Process.*, vol. 4, no. 2, pp. 330–341, Apr. 2010.
[54] S. Haykin, C. Krasnor, T. Nohara, B. Currie, and D. Hamburger, "A coherent dual-polarized radar for studying the ocean environment," *IEEE Trans. Geosci. Remote Sens.*, vol. 29, no. 1, pp. 189–191, Jan. 1991.
[55] M. Hurtado, N. v. Ellenrieder, C. Muravchik, and A. Nehorai, "Sparse modeling for polarimetric radar," in *Proc. IEEE Statist. Signal Process. Workshop*, Jun. 2011, pp. 17–20.
[56] H. Park, J. Li, and H. Wang, "Polarization-space-time domain generalized likelihood ratio detection of radar targets," *Signal Process.*, vol. 41, no. 2, pp. 153–164, Jan. 1995.
[57] A. Edelman and H. Murakami, "Polynomial roots from companion matrix eigenvalues," *Math. Comp.*, vol. 64, pp. 763–776, 1995.
[58] S. M. Kay, *Fundamentals of Statistical Signal Processing: Estimation Theory*. Englewood Cliffs, NJ: Prentice–Hall, 1993.

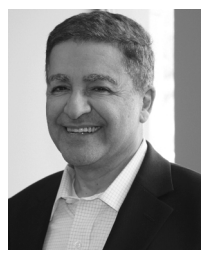
Martin Hurtado (M'11) received the B.Eng. and M.Sc. degrees in electrical engineering from the National University of La Plata, Argentina, in 1996 and 2001, respectively. He received the Ph.D. degree in electrical engineering from Washington University, St. Louis, MO, in 2007.

Currently, he is a Research Associate of the National Council of Scientific and Technical Research of Argentina and an adjunct professor in the Department of Electrical Engineering at National University of La Plata. His research interests are in the area of statistical signal processing, detection and estimation theory, and their applications in sensor arrays, communications, and remote sensing systems.

Carlos H. Muravchik (S'81–M'83–SM'99) was born in Argentina, June 11, 1951. He graduated as an electronics engineer from the National University of La Plata, Argentina, in 1973. He received the M.Sc. degree in statistics (1983) and the M.Sc. (1980) and Ph.D. (1983) degrees in electrical engineering, from Stanford University, Stanford, CA.

He is a Professor at the Department of the Electrical Engineering of the National University of La Plata and chairman of its Industrial Electronics, Control and Instrumentation Laboratory (LEICI). He is also a member of the Comision de Investigaciones Cientificas de la Pcia. de Buenos Aires. He was a Visiting Professor at Yale University, New Haven, CT, in 1983 and 1994, at the University of Illinois at Chicago in 1996, 1997, 1999, and 2003, and at Washington University, St. Louis, MO, in 2006 and 2010. His research interests are in the area of statistical and array signal processing with biomedical, communications and control applications, and in nonlinear control systems.

Dr. Muravchik has been a member of the Advisory Board of the journal *Latin American Applied Research* since 1999 and was an Associate Editor of the IEEE TRANSACTIONS ON SIGNAL PROCESSING (2003–2006).



Arye Nehorai (S'80–M'83–SM'90–F'94) received the B.Sc. and M.Sc. degrees from the Technion, Israel, and the Ph.D. degree from Stanford University, CA.

He is the Eugene and Martha Lohman Professor and Chair of the Preston M. Green Department of Electrical and Systems Engineering (ESE) at Washington University, St. Louis (WUSTL), MO. He is also Professor in the Division of Biology and Biomedical Studies (DBBS) and Director of the Center for Sensor Signal and Information Processing at WUSTL. Earlier, he was a faculty member at Yale University, New Haven, CT, and the University of Illinois at Chicago. Under his leadership as department chair, the undergraduate enrollment has more than tripled in the last four years.

Dr. Nehorai served as Editor-in-Chief of the IEEE TRANSACTIONS ON SIGNAL PROCESSING from 2000 to 2002. From 2003 to 2005, he was the Vice President (Publications) of the IEEE Signal Processing Society (SPS), the Chair of the Publications Board, and a member of the Executive Committee of this Society. He was the founding editor of the special columns on Leadership Reflections in the IEEE SIGNAL PROCESSING MAGAZINE from 2003 to 2006. He received the 2006 IEEE SPS Technical Achievement Award and the 2010 IEEE SPS Meritorious Service Award. He was elected Distinguished Lecturer of the IEEE SPS for a term lasting from 2004 to 2005. He received best paper awards in IEEE journals and conferences. In 2001, he was named University Scholar of the University of Illinois. He was the Principal Investigator of the Multidisciplinary University Research Initiative (MURI) project titled Adaptive Waveform Diversity for Full Spectral Dominance from 2005 to 2010. He is a Fellow of the Royal Statistical Society since 1996 and of AAAS since 2012.