Automated Recovery of Compressedly Observed Sparse Signals From Smooth Background

Zhaofu Chen, Rafael Molina, *Member, IEEE*, and Aggelos K. Katsaggelos, *Fellow, IEEE*

***Abstract*—We propose a Bayesian based algorithm to recover sparse signals from compressed noisy measurements in the presence of a smooth background component. This problem is closely related to robust principal component analysis and compressive sensing, and is found in a number of practical areas. The proposed algorithm adopts a hierarchical Bayesian framework for modeling, and employs approximate inference to estimate the unknowns. Numerical examples demonstrate the effectiveness of the proposed algorithm and its advantage over the current state-of-the-art solutions.**

***Index Terms*—Bayesian algorithm, compressive sensing, robust principal component analysis.**

I. Introduction

CONSIDER the measurement system expressed as

$$\mathbf{Y} = \mathbf{\Phi X} + \mathbf{B} + \mathbf{N}, \tag{1}$$

where the signal of interest $\mathbf{X} \in \mathbb{R}^{M\times N}$ undergoes a transformation $\mathbf{\Phi} \in \mathbb{R}^{L\times M}$ and is corrupted by both noise $\mathbf{N}$ and a smooth background $\mathbf{B}$. The signal $\mathbf{X}$ is assumed to have sparse columns, i.e., $\|\mathbf{x}_{\cdot i}\|_0 \ll M$ for $i = 1, \ldots, N$, where $\mathbf{x}_{\cdot i}$ denotes the ith column of $\mathbf{X}$ and $\|\cdot\|_0$ is the ℓ_0-(pseudo)norm. The smooth background $\mathbf{B}$ is a low-rank matrix. The transformation $\mathbf{\Phi}$ in general has the effect of compression, i.e., $L \leq M$.

The model in (1) is found in a number of applications. In network anomaly detection [1], $\mathbf{X}$ consists of the temporal snapshots of flow anomalies, $\mathbf{\Phi}$ represents the network routing operation, and $\mathbf{B}$ contains the smooth link measurements resulted from the normal traffic flows, respectively. As another application, in video surveillance from compressed measurements [2], $\mathbf{X}$ denotes the moving objects in the foreground, $\mathbf{\Phi}$ is a known measurement matrix, and $\mathbf{B}$ is the compressed version of the background.

Manuscript received February 25, 2014; revised April 16, 2014; accepted April 26, 2014. Date of publication April 30, 2014; date of current version May 15, 2014. This work supported in part by the U.S. Department of Energy under Grant DE-NA0000457, the Spanish Ministry of Economy and Competitiveness under Project TIN2010-15137, the European Regional Development Fund (FEDER) , and by the CEI BioTic at the Universidad de Granada. The associate editor coordinating the review of this manuscript and approving it for publication was Prof. Jun Fang.

Z. Chen and A. K. Katsaggelos are with the Department of Electrical Engineering and Computer Science, Northwestern University, Evanston, IL, 60208 USA (e-mail: zhaofuchen2014@u.northwestern.edu; aggk@eecs.northwestern.edu).

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

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/LSP.2014.2321256

The model in (1) is also closely related to Compressive Sensing (CS) [3]–[5] and Robust Principal Component Analysis (RPCA) [6]. For CS, where $\mathbf{B}$ is not present, algorithms generally employ regularized optimization (e.g., in [7]–[9]) or Bayesian approaches (e.g., in [10]–[12]). For RPCA, where $\mathbf{\Phi} = \mathbf{I}$, regularized optimization problems are solved with proper convex relaxation to sparsity and rank [13]–[15]. Alternatively, Bayesian approaches model the sparse and low-rank components with appropriate prior structures and employ approximate inference techniques for estimation [16], [17].

Recently, the model in (1) has received much attention from the signal processing community. [1] provides the conditions for identifiability and recoverability. Algorithms based on convex optimization have been proposed in [1], [18], [19]. The optimization-based algorithms usually require proper selection of user parameters. In contrast, [20] takes a Bayesian perspective and automatically estimates all model parameters. A limitation of [20], though, is the memory and computational requirement incurred by the use of Hessian information.

In this letter, we incorporate the Laplace prior to model the sparse signal. As explained in [21], this prior has been shown to promote sparsity to a higher level than the Sparse Bayesian Learning (SBL) prior used in [20]. In addition, in order to reduce memory consumption, we develop a constructive approach based on the principles in [22] and [12] that essentially replaces the demanding matrix inversion with a sequence of efficient rank-one updates. The algorithm proposed herein is free of user parameters, making it amenable to be deployed for automatic operation.

This letter is organized as follows. In Section II we introduce the hierarchical Bayesian model. The inference procedure with the constructive algorithm is outlined in Section III. Numerical examples demonstrating the effectiveness of the proposed algorithm are provided in Section IV. Finally, we draw concluding remarks in Section V.

II. Hierarchical Bayesian Model

A. Modeling Additive Noise

$\mathbf{N} = \{n_{ij}\}_{ij}$ in (1) contains uncorrelated noise. We employ an independent and identically distributed (i.i.d.) Gaussian distribution $\mathrm{p}(n_{ij}|\beta) = \mathcal{N}(n_{ij}|0, \beta^{-1})$, with precision β modeled using the conjugate Gamma prior, i.e., $\mathrm{p}(\beta|a_0, b_0) = \mathcal{G}(\beta|a_0, b_0)$, where the hyperparameters a_0 and b_0 are fixed at small values to approximate the non-informative Jeffreys prior.

B. Modeling Smooth Background

Consider the factorization of the smooth background $\mathbf{B} = \mathbf{PQ}^{\mathrm{T}}$, where $\mathbf{P}$ and $\mathbf{Q}$ are $L \times K$ and $N \times K$ matrices, respectively, and K is a loose upper bound for the rank of $\mathbf{B}$. Smoothness of $\mathbf{B}$ is resulted when most of the K outer products in $\mathbf{PQ}^{\mathrm{T}}$ are

zeros. To achieve this, common sparsity promoting priors are simultaneously assigned to the columns of $\mathbf{P}$ and $\mathbf{Q}$, as in [20]

$$\begin{aligned}\mathrm{p}(\mathbf{P}|\boldsymbol{\gamma}) &= \prod_{i=1}^{K} \mathcal{N}(\mathbf{p}_{\cdot i}|\mathbf{0}, \gamma_i^{-1}\mathbf{I}) \\ \mathrm{p}(\mathbf{Q}|\boldsymbol{\gamma}) &= \prod_{i=1}^{K} \mathcal{N}(\mathbf{q}_{\cdot i}|\mathbf{0}, \gamma_i^{-1}\mathbf{I}),\end{aligned} \tag{2}$$

where the precisions $\{\gamma_i\}_i$ are assigned Gamma priors, i.e., $\mathrm{p}(\gamma_i|a_1, b_1) = \mathcal{G}(\gamma_i|a_1, b_1)$. The hyperparameters a_1 and b_1 are fixed at small values to yield broad distributions.

C. Modeling Sparse Signal

In least squares fitting problems ℓ_1-norm is commonly used for regularization so that sparse solutions are preferred. This is equivalent to adopting a Laplace prior on the sparse signals

$$\mathrm{p}(\mathbf{x}_{\cdot j}|\lambda_j) = \lambda_j^{M/2}/2^M \times \exp(-\sqrt{\lambda_j}\|\mathbf{x}_{\cdot j}\|_1). \tag{3}$$

The Laplace prior, though being sparsity promoting, is not conjugate to the Gaussian model. To overcome this limitation, a hierarchical model is established as in [23], where

$$\begin{aligned}\mathrm{p}(\mathbf{x}_{\cdot j}|\boldsymbol{\omega}_{\cdot j}) &= \prod_{i=1}^{M} \mathcal{N}(x_{ij}|0, \omega_{ij}) \\ \mathrm{p}(\omega_{ij}|\lambda_j) &= \lambda_j/2 \times \exp(-\lambda_j\omega_{ij}/2).\end{aligned} \tag{4}$$

Finally, λ_j is modeled as $\mathrm{p}(\lambda_j|a_2, b_2) = \mathcal{G}(\lambda_j|a_2, b_2)$, where a_2 and b_2 are fixed at small values.

D. Complete System Model

By combining the observation and prior models, the joint distribution is expressed as

$$\begin{aligned}&\mathrm{p}(\mathbf{Y}, \mathbf{X}, \mathbf{P}, \mathbf{Q}, \boldsymbol{\Omega}, \boldsymbol{\lambda}, \boldsymbol{\gamma}, \beta|a_0, b_0, a_1, b_1, a_2, b_2) \\ &= \mathrm{p}(\mathbf{Y}|\mathbf{X}, \mathbf{P}, \mathbf{Q}, \beta)\mathrm{p}(\beta|a_0, b_0)\mathrm{p}(\mathbf{P}|\boldsymbol{\gamma})\mathrm{p}(\mathbf{Q}|\boldsymbol{\gamma}) \\ &\quad \times \mathrm{p}(\boldsymbol{\gamma}|a_1, b_1)\mathrm{p}(\mathbf{X}|\boldsymbol{\Omega})\mathrm{p}(\boldsymbol{\Omega}|\boldsymbol{\lambda})\mathrm{p}(\boldsymbol{\lambda}|a_2, b_2),\end{aligned} \tag{5}$$

where $\boldsymbol{\Omega} = [\boldsymbol{\omega}_{\cdot 1}, \cdots, \boldsymbol{\omega}_{\cdot N}]$, $\boldsymbol{\lambda} = [\lambda_1, \ldots, \lambda_N]$, and $\boldsymbol{\gamma} = [\gamma_1, \ldots, \gamma_k]$.

III. Approximate Bayesian Inference

Bayesian approaches seek the posterior distributions of the unknowns given the observations. Approximations are usually employed since the exact posterior distributions are analytically intractable. Common approximate approaches include (1) point estimation, such as Maximum Likelihood (ML) and Maximum *A Posteriori* (MAP) estimation, (2) sampling approaches, such as Gibbs Sampler (GS), and (3) Variational Bayes (VB), etc.

In this letter, we employ VB to approximate the posterior distributions of $\mathbf{P}$, $\mathbf{Q}$ and $\boldsymbol{\gamma}$. For $\mathbf{X}$, $\boldsymbol{\Omega}$ and $\boldsymbol{\lambda}$, we introduce a constructive algorithm based on marginalization. Finally, the noise precision β is estimated via VB.

A. Inference of Smooth Background

The inference of $\mathbf{P}$, $\mathbf{Q}$ and $\boldsymbol{\gamma}$ follows from the invocation of the mean-field approximation and variational calculus. Omitting the details that can be found in [20], the posterior means of $\mathbf{P}$ and $\mathbf{Q}$ are given by

$$\bar{\mathbf{P}} = \bar{\beta}\left(\mathbf{Y} - \boldsymbol{\Phi}\bar{\mathbf{X}}\right)\bar{\mathbf{Q}}\boldsymbol{\Sigma}^p, \bar{\mathbf{Q}} = \bar{\beta}\big(\mathbf{Y} - \boldsymbol{\Phi}\bar{\mathbf{X}}\big)^{\mathrm{T}}\bar{\mathbf{P}}\boldsymbol{\Sigma}^q, \tag{6}$$

where $\bar{\beta}$ denotes the posterior mean of β, and the covariance matrices $\boldsymbol{\Sigma}^p$ and $\boldsymbol{\Sigma}^q$ are inter-related as

$$\boldsymbol{\Sigma}^p = \left(\bar{\beta}\bar{\mathbf{Q}}^{\mathrm{T}}\bar{\mathbf{Q}} + N\bar{\beta}\boldsymbol{\Sigma}^q + \bar{\boldsymbol{\Gamma}}\right)^{-1}, \boldsymbol{\Sigma}^q = \left(\bar{\beta}\bar{\mathbf{P}}^{\mathrm{T}}\bar{\mathbf{P}} + L\bar{\beta}\boldsymbol{\Sigma}^p + \bar{\boldsymbol{\Gamma}}\right)^{-1}.$$

The term $\bar{\boldsymbol{\Gamma}}$ above is a diagonal matrix constructed from $\bar{\boldsymbol{\gamma}}$. Similarly, the approximate posterior mean of γ_i is found to be

$$\bar{\gamma}_i = \frac{L + N + 2a_1}{\|\bar{\mathbf{p}}_{\cdot i}\|_2^2 + \|\bar{\mathbf{Q}}_{\cdot i}\|_2^2 + L\sigma_{ii}^p + N\sigma_{ii}^q + 2b_1}, \tag{7}$$

where σ_{ii}^p and σ_{ii}^q are the ith elements on the diagonals of $\boldsymbol{\Sigma}^p$ and $\boldsymbol{\Sigma}^q$, respectively. In the iterations, the posterior means in (6) and (7) are used as the estimates of the corresponding variables.

B. Inference of Sparse Signal

If we continued applying VB inference we would have to find, within the iterative process, the distributions $\mathrm{q}(\boldsymbol{\lambda})$, $\mathrm{q}(\boldsymbol{\Omega})$, and $\mathrm{q}(\mathbf{X})$ minimizing the Kullback-Leibler divergence $\mathrm{KL}(\mathrm{q}(\boldsymbol{\lambda})\mathrm{q}(\boldsymbol{\Omega})\mathrm{q}(\mathbf{X})|\mathrm{p}(\mathbf{G}|\mathbf{X}, \beta)\mathrm{p}(\mathbf{X}|\boldsymbol{\Omega})\mathrm{p}(\boldsymbol{\Omega}|\boldsymbol{\lambda})\mathrm{p}(\boldsymbol{\lambda})\mathrm{p}(\beta))$, where $\mathbf{G} = \mathbf{Y} - \bar{\mathbf{P}}\bar{\mathbf{Q}}^{\mathrm{T}}$, $\mathrm{p}(\mathbf{G}|\mathbf{X}, \beta) = \prod_j \mathcal{N}(\mathbf{g}_{\cdot j}|\boldsymbol{\Phi}\mathbf{x}_{\cdot j}, \beta^{-1}\mathbf{I})$, and $\mathrm{q}(\boldsymbol{\lambda})$ and $\mathrm{q}(\boldsymbol{\Omega})$ are degenerate distributions. Then we would have

$$\mathrm{q}(\mathbf{X}) = \mathrm{p}(\mathbf{X}|\mathbf{G}, \beta, \boldsymbol{\omega}) = \prod_{j=1}^{N} \mathrm{p}(\mathbf{x}_{\cdot j}|\mathbf{g}_{\cdot j}, \beta, \boldsymbol{\omega}_{\cdot j}), \tag{8}$$

and with some algebra it follows that this conditional distribution is a multivariate Gaussian $\mathcal{N}(\mathbf{x}_{\cdot j}|\bar{\mathbf{x}}_{\cdot j}, \boldsymbol{\Sigma}_j^x)$ with

$$\bar{\mathbf{x}}_{\cdot j} = \beta\boldsymbol{\Sigma}_j^x\boldsymbol{\Phi}^{\mathrm{T}}\mathbf{g}_{\cdot j}, \quad \boldsymbol{\Sigma}_j^x = \left(\beta\boldsymbol{\Phi}^{\mathrm{T}}\boldsymbol{\Phi} + \boldsymbol{\Omega}_j^{-1}\right)^{-1}, \tag{9}$$

where $\boldsymbol{\Omega}_j = \mathrm{diag}([\omega_{1j}, \cdots, \omega_{Mj}])$.

Two observations follow from (9). First, $\boldsymbol{\Sigma}_j^x$ involves the inversion of an $M \times M$ matrix, which is memory and computationally intensive. Second, if an ω_{ij} is zero, then the corresponding x_{ij} must be zero (see (4)), and then the i^{th} column and row in $\boldsymbol{\Sigma}_j^x$ are eliminated by removing ω_{ij}^{-1} from the diagonal of Ω_j^{-1} and the i^{th} column of $\boldsymbol{\Phi}$. Therefore, the dimension of $\boldsymbol{\Sigma}_j^x$ is determined by the number of nonzeros in $\boldsymbol{\omega}_{\cdot j}$. Since $\mathbf{x}_{\cdot j}$ is sparse, most of its entries are expected to be zero with zero variance.

Let us now find the nonzero components of $\boldsymbol{\omega}_{\cdot j}$. We fix $\mathrm{q}(\mathbf{X})$ to (8), where $\boldsymbol{\Omega}$, $\boldsymbol{\lambda}$, and β are estimated from

$$\begin{aligned}\mathrm{p}(\mathbf{G}, \boldsymbol{\Omega}, \boldsymbol{\lambda}, \beta) &= \mathrm{p}(\beta)\prod_{j=1}^{N} \mathrm{p}(\mathbf{g}_{\cdot j}, \boldsymbol{\omega}_{\cdot j}, \lambda_j|\beta) \\ &= \mathrm{p}(\beta)\prod_{j=1}^{N} \mathcal{N}(\mathbf{g}_{\cdot j}|\mathbf{0}, \mathbf{C}_j)\mathrm{p}(\boldsymbol{\omega}_{\cdot j}|\lambda_j)\mathrm{p}(\lambda_j),\end{aligned} \tag{10}$$

where $\mathbf{C}_j = \beta^{-1}\mathbf{I} + \boldsymbol{\Phi}\Omega\boldsymbol{\Phi}^{\mathrm{T}}$.

To determine the optimal $\boldsymbol{\omega}_{\cdot j}$, we take the logarithm of (10), drop the terms independent of $\boldsymbol{\omega}_{\cdot j}$ and maximize

$$\mathcal{L}(\boldsymbol{\omega}_{\cdot j}) = -\frac{1}{2}\log|\mathbf{C}_j| - \frac{1}{2}\mathbf{g}_{\cdot j}^{\mathrm{T}}\mathbf{C}_j^{-1}\mathbf{g}_{\cdot j} - \frac{\lambda_j}{2}\sum_{i=1}^{M}\omega_{ij}, \tag{11}$$

Focusing on a single entry ω_{ij} of $\boldsymbol{\omega}_{\cdot j}$, it follows that

$$\begin{aligned}\mathbf{C}_j &= \beta^{-1}\mathbf{I} + \sum\nolimits_{k\neq i}\omega_{kj}\boldsymbol{\phi}_{\cdot k}\boldsymbol{\phi}_{\cdot k}^{\mathrm{T}} + \omega_{ij}\boldsymbol{\phi}_{\cdot i}\boldsymbol{\phi}_{\cdot i}^{\mathrm{T}} \\ &=^{-i} \mathbf{C}_j + \omega_{ij}\boldsymbol{\phi}_{\cdot i}\boldsymbol{\phi}_{\cdot i}^{\mathrm{T}},\end{aligned} \tag{12}$$

where $^{-i}\mathbf{C}_j$ denotes the portion of $\mathbf{C}_j$ with the contribution from ω_{ij} excluded. Utilizing properties of determinant, we have

$$\begin{aligned}\log|\mathbf{C}_j| &= \log|^{-i}\mathbf{C}_j| + \log(1 + \omega_{ij}\boldsymbol{\phi}_{\cdot i}^{\mathrm{T}}(^{-i}\mathbf{C}_j)^{-1}\boldsymbol{\phi}_{\cdot i}) \\ &= \log|^{-i}\mathbf{C}_j| + \log(1 + \omega_{ij}r_{ij}),\end{aligned} \tag{13}$$

where we define $r_{ij} = \boldsymbol{\phi}_{\cdot i}^{\mathrm{T}}({}^{-i}\mathbf{C}_j)^{-1}\boldsymbol{\phi}_{\cdot i}$ for notational clarity. Invoking matrix inversion lemma, we have

$$\mathbf{C}_j^{-1} = ({}^{-i}\mathbf{C}_j)^{-1} - \frac{\omega_{ij}({}^{-i}\mathbf{C}_j)^{-1}\boldsymbol{\phi}_{\cdot i}\boldsymbol{\phi}_{\cdot i}^{\mathrm{T}}({}^{-i}\mathbf{C}_j)^{-1}}{1+\omega_{ij}r_{ij}} \quad (14)$$

and

$$\begin{aligned}\mathbf{g}_{\cdot j}^{\mathrm{T}}\mathbf{C}_j^{-1}\mathbf{g}_{\cdot j} &= \mathbf{g}_{\cdot j}^{\mathrm{T}}({}^{-i}\mathbf{C}_j)^{-1}\mathbf{g}_{\cdot j} - \frac{\omega_{ij}\left(\boldsymbol{\phi}_{\cdot i}^{\mathrm{T}}({}^{-i}\mathbf{C}_j)^{-1}\mathbf{g}_{\cdot j}\right)^2}{1+\omega_{ij}r_{ij}} \\ &= \mathbf{g}_{\cdot j}^{\mathrm{T}}({}^{-i}\mathbf{C}_j)^{-1}\mathbf{g}_{\cdot j} - \frac{\omega_{ij}s_{ij}^2}{1+\omega_{ij}r_{ij}}, \end{aligned} \quad (15)$$

where $s_{ij} = \boldsymbol{\phi}_{\cdot i}^{\mathrm{T}}({}^{-i}\mathbf{C}_j)^{-1}\mathbf{g}_{\cdot j}$ is defined for notational clarity.

Substituting (13) and (15) into (11), and retaining only terms dependent on ω_{ij}, it follows that

$$l(\omega_{ij}) = \frac{1}{2}\left[\log\frac{1}{1+\omega_{ij}r_{ij}} + \frac{\omega_{ij}s_{ij}^2}{1+\omega_{ij}r_{ij}} - \lambda_j\omega_{ij}\right]. \quad (16)$$

It is clear that the ω_{ij} maximizing (16) also maximizes (11). Note in (16) both r_{ij} and s_{ij} are independent of ω_{ij}. The univariate function $l(\omega_{ij})$ can be maximized by examining its derivative and taking into account the domain of $l(\omega_{ij})$. With some algebra, the optimal ω_{ij} is found to be

$$\omega_{ij}^{\star} = \begin{cases} \frac{-(r_{ij}+2\lambda_j)+\sqrt{r_{ij}^2+4\lambda_j s_{ij}^2}}{2\lambda_j r_{ij}}, & s_{ij}^2 - r_{ij} > \lambda_j \\ 0, & \text{otherwise} \end{cases} \quad (17)$$

As discussed above, a zero-valued ω_{ij} indicates that the corresponding row and column in $\boldsymbol{\Sigma}_j^x$ are all zeros. Therefore, (17) provides a guideline as how the model should be adjusted. In the iterative procedure, depending on the previous estimate of ω_{ij} and the condition $s_{ij}^2 - r_{ij} \gtrless \lambda_j$, there are three possible adjustments to the model, namely

$$\begin{cases} \text{Add } \omega_{ij}, & \text{if previously excluded and } s_{ij}^2 - r_{ij} > \lambda_j \\ \text{Remove } \omega_{ij}, & \text{if previously included and } s_{ij}^2 - r_{ij} \le \lambda_j \\ \text{Update } \omega_{ij}, & \text{if previously included and } s_{ij}^2 - r_{ij} > \lambda_j \end{cases} \quad (18)$$

Note that, for a fixed j, all these updates only involve efficient rank-one modifications to $\boldsymbol{\Sigma}_j^x$ and $\bar{\mathbf{x}}_{\cdot j}$, rather than requiring matrix inversions (see [22] for details). Also note that the effective dimension of $\boldsymbol{\Sigma}_j^x$ and $\bar{\mathbf{x}}_{\cdot j}$ is much smaller than M, as only a small subset of $\{\omega_{ij}\}_i$ are included at any time in the iterative process. In this way the constructive approach alleviates the memory and computational requirement. The selection of an ω_{ij} for update can be done either randomly, or such that the largest increase in the log-likelihood is obtained.

For a fixed j, whenever the model is adjusted, $\{r_{ij}\}_i$ and $\{s_{ij}\}_i$ need to be updated efficiently. Invocation of matrix inversion lemma yields

$$\mathbf{C}_j^{-1} = \beta\mathbf{I} - \beta^2\boldsymbol{\Phi}\Sigma_j^x\boldsymbol{\Phi}^{\mathrm{T}}, \quad (19)$$

which is now used.

For r_{ij}, using (14), it follows

$$r_{ij} = \boldsymbol{\phi}_{\cdot i}^{\mathrm{T}}\mathbf{C}_j^{-1}\boldsymbol{\phi}_{\cdot i} + \frac{\omega_{ij}r_{ij}^2}{1+\omega_{ij}r_{ij}} = R_{ij} + \frac{\omega_{ij}r_{ij}^2}{1+\omega_{ij}r_{ij}}, \quad (20)$$

where

$$R_{ij} = \boldsymbol{\phi}_{\cdot i}^{\mathrm{T}}\mathbf{C}_j^{-1}\boldsymbol{\phi}_{\cdot i} = \beta\boldsymbol{\phi}_{\cdot i}^{\mathrm{T}}\boldsymbol{\phi}_{\cdot i} - \beta^2\boldsymbol{\phi}_{\cdot i}^{\mathrm{T}}\boldsymbol{\Phi}\Sigma_j^x\boldsymbol{\Phi}^{\mathrm{T}}\boldsymbol{\phi}_{\cdot i} \quad (21)$$

from (19). Since the effective dimension of $\boldsymbol{\Sigma}_j^x$ in (21) is much smaller than M, R_{ij} can be efficiently updated. Solving (20) for r_{ij}, we have

$$r_{ij} = R_{ij}/(1-\omega_{ij}R_{ij}), i = 1,\ldots,M. \quad (22)$$

The introduction of the auxiliary variables $\{R_{ij}\}_i$ eliminates $M-1$ matrix inversions for the update of $\{r_{ij}\}_i$. Similarly, by introducing

$$S_{ij} = \boldsymbol{\phi}_{\cdot i}^{\mathrm{T}}\mathbf{C}_j^{-1}\mathbf{g}_{\cdot j} = \beta\boldsymbol{\phi}_{\cdot i}^{\mathrm{T}}\mathbf{g}_{\cdot j} - \beta^2\boldsymbol{\phi}_{\cdot i}^{\mathrm{T}}\boldsymbol{\Phi}\Sigma_j^x\boldsymbol{\Phi}^{\mathrm{T}}\mathbf{g}_{\cdot i}, \quad (23)$$

the update of s_{ij} can be performed via

$$s_{ij} = S_{ij}/(1-\omega_{ij}R_{ij}), i = 1,\ldots,M. \quad (24)$$

Finally, the update of λ_j is done by maximizing the logarithm of (10) with respect to λ_j to obtain

$$\lambda_j^{\star} = (M-1+a_2)/(\textstyle\sum_i \omega_{ij}/2 + b_2). \quad (25)$$

The updates in (18) and (21) to (25) are iterated until convergence. In each iteration an ω_{ij} is selected for update. One criterion for convergence is that the relative change in log-likelihood (11) falls below a pre-defined threshold. In practice, we find that a small number of iterations (e.g., 20) is usually sufficient for good performance. When the iterations converge, the approximate posterior mean $\bar{\mathbf{x}}_{\cdot j}$ is used as an estimate of $\mathbf{x}_{\cdot j}$.

C. Inference of noise power

Via mean-field approximation, the approximate posterior distribution of the noise precision β is found to be Gamma, with mean

$$\bar{\beta} = \frac{LN + a_0}{\langle\|\mathbf{Y} - \boldsymbol{\Phi}\mathbf{X} - \mathbf{P}\mathbf{Q}^{\mathrm{T}}\|_{\mathrm{F}}^2\rangle + b_0}, \quad (26)$$

where

$$\begin{aligned}\langle\|\mathbf{Y} - \boldsymbol{\Phi}X - \mathbf{P}\mathbf{Q}^{\mathrm{T}}\|_{\mathrm{F}}^2\rangle &= \|\mathbf{Y} - \boldsymbol{\Phi}\bar{\mathbf{X}} - \bar{\mathbf{P}}\bar{\mathbf{Q}}^{\mathrm{T}}\|_{\mathrm{F}}^2 \\ &+ N\times\mathrm{trace}(\bar{\mathbf{P}}^{\mathrm{T}}\bar{\mathbf{P}}\boldsymbol{\Sigma}^q) + L\times\mathrm{trace}(\bar{\mathbf{Q}}^{\mathrm{T}}\bar{\mathbf{Q}}\boldsymbol{\Sigma}^p) \\ &+ LN\times\mathrm{trace}(\boldsymbol{\Sigma}^p\boldsymbol{\Sigma}^q) + \sum_{j=1}^{N}\mathrm{trace}(\boldsymbol{\Phi}\Sigma_j^x\boldsymbol{\Phi}^{\mathrm{T}}). \end{aligned} \quad (27)$$

The iterative procedure described above continues until convergence is reached. One criterion for convergence is that the relative change in the variables, e.g., $\mathbf{X}$ or $\mathbf{B}$, falls below a pre-defined threshold.

IV. Numerical Results

A. Simulation

1) Recovery Accuracy: We first demonstrate the performance of the proposed algorithm on simulated data. The data generation is described as follows. We consider problems of varying scales $M \in \{100, 200, 400, 600\}$. For each scale, $L = M/2$ and $N = M$. The ground truth sparsity is fixed at $\|\mathbf{X}\|_0 = 5\%MN$, and nonzeros of $\mathbf{X}$ are independently drawn from uniform $\mathcal{U}(-10, 10)$ distribution. The ground truth rank of $\mathbf{B}$ is fixed at $r = 0.05L$. The smooth component $\mathbf{B}$ is generated as the product of an $L \times r$ matrix and an $r \times N$ matrix, whose elements are drawn from i.i.d. $\mathcal{N}(0, 100/L)$ and $\mathcal{N}(0, 100/N)$ distributions, respectively. Additive noise with standard deviation $\sigma_n = 0.05$ is added to the measurement. The transformation $\boldsymbol{\Phi}$

TABLE I
COMPARISON OF PERFORMANCE FOR VARYING PROBLEM SCALES

Ground truth			Proposed				VBSE				ADMM			
M	$r(\mathbf{B})$	$\|\mathbf{X}\|_0$	$r(\hat{\mathbf{B}})$	$\|\hat{\mathbf{X}}\|_0$	ϵ^x	Time	$r(\hat{\mathbf{B}})$	$\|\hat{\mathbf{X}}\|_0$	ϵ^x	Time	$r(\hat{\mathbf{B}})$	$\|\hat{\mathbf{X}}\|_0$	ϵ^x	Time
100	3	500	3	500	8.2e-3	10.4	3	500	7.8e-3	22.7	34	2163	2.5e-2	4.7
200	5	2000	5	2000	8.7e-3	80.4	5	2000	8.4e-3	171.2	69	9149	2.2e-2	15.6
400	10	8000	10	8000	8.4e-3	604.6	10	8000	8.0e-3	1098.7	145	37898	2.1e-2	79.7
600	15	18000	15	18000	8.2e-3	2114.5	15	18000	8.0e-3	4819.6	221	86733	2.1e-2	231.6

is generated similarly to that in [1], which consists of random orthonormal rows.

The proposed algorithm is applied to recover the sparse signal and smooth background, with estimates denoted as $\hat{\mathbf{X}}$ and $\hat{\mathbf{B}}$, respectively. All hyperparameters in the model are set to a small value (e.g., 10^{-6}), making the algorithm fully automated. As a comparison, we show the performance on the same test data of two alternative algorithms, namely the Variational Bayes Sparse Estimator (VBSE) proposed in [20], and the state-of-the-art Alternating Directions Method of Multipliers (ADMM) proposed in [1]. For the problem considered herein, VBSE is the only available Bayesian approach to our knowledge and ADMM is reported to have high recovery accuracy in the literature.

In Table I four metrics are used to evaluate the performance: rank of $\hat{\mathbf{B}}$, number of nonzeros in $\hat{\mathbf{X}}$, reconstruction error $\epsilon^x = \|\hat{\mathbf{X}} - \mathbf{X}\|_{\mathrm{F}}/\|\mathbf{X}\|_{\mathrm{F}}$ and runnig time (Intel Core i5-3210M CPU@2.50 GHz, 4 GB RAM, MATLAB R2012b). Both the proposed algorithm and VBSE are free of user parameters. For ADMM we manually tune the two user parameters to yield empirically smallest ϵ^x. For tuning the parameters in ADMM, scanning is performed over a 2-D logarithmic grid followed by a 2-D linear grid. As the table shows, both Bayesian based approaches correctly estimate the rank and sparsity, as well as yield significantly lower recovery error than ADMM. Despite being manually tuned, ADMM suffers substantial performance degradation with the presence of even moderate level of noise.

Regarding computational cost, the following observations are in line. First of all, we see that the proposed algorithm reduces running time by about 50% compared with VBSE. In addition, note that the proposed algorithm, thanks to its constructive nature, has lower memory requirement than VBSE, making it scale better with problem dimensions. Last but not least, although the numbers seem to imply that ADMM is close to an order of magnitude more efficient than the proposed algorithm, we would like to point out that the numbers shown here do not include the overhead of parameter tuning. Parameter tuning may take significantly longer than the running time shown in the table.

2) Robustness to Noise: To examine robustness of the proposed algorithm to noise, we fix $L = 100$, $M = N = 200$, $r(\mathbf{B}) = 5$, $\|\mathbf{X}\|_0 = 2000$, and vary σ_n from 0.01 to 1. Three parameter settings for ADMM are considered, namely ADMM-1 that yields the lowest ϵ^x, ADMM-2 that gives correct $r(\hat{\mathbf{B}})$, and ADMM-3 that gives almost correct $\|\hat{\mathbf{B}}\|_0$.

It is clear from Fig. 1 that the proposed algorithm is robust to additive noise, thanks to its automatic estimation of noise level. In particular, it accurately recovers the nonzeros in $\mathbf{X}$ and tracks the rank of $\mathbf{B}$ even at very low Signal-to-Noise-Ratio (SNR) conditions (e.g., SNR < 10 for $\sigma_n = 1$). The ADMM algorithm is most sensitive to noise due to its inherent modeling assumption. Moreover, it requires careful tuning of user parameters to achieve reasonable results.

3) Ability to Identify Nonzero Signal Elements: To investigate the ability of the proposed algorithm in detecting nonzeros of $\mathbf{X}$

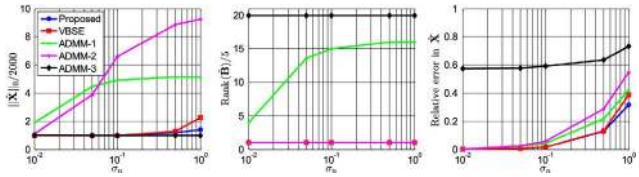

Fig. 1. Recovery performance with respect to noise level. (Note: in the middle figure, the proposed, VBSE and ADMM-2 all correctly estimate the rank, hence their curves overlap with each other).

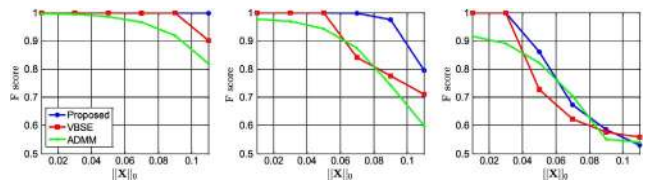

Fig. 2. Comparison of abilities to detect nonzeros in $\mathbf{X}$ ($r(\mathbf{B}) = 5$, 10, and 15 from left to right).

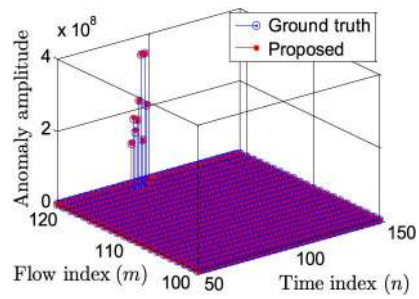

Fig. 3. Detection of network anomalies in Internet2 data.

under different conditions, we consider $\|\mathbf{X}\|_0$ from 1% to 11% and $r(\mathbf{B}) \in \{5, 10, 15\}$. σ_n is fixed at 0.01. The data generation is similar to that above.

Fig. 2 shows the F-scores at various conditions. The F-score, defined as the harmonic mean of precision and recall, is a balanced indicator of detection performance. From the figure it is seen the proposed algorithm gives reliable detections over a wider range of operational conditions than its alternatives.

B. Network Anomaly Detection Example

To further validate its effectiveness, the proposed algorithm is applied to the real life Internet2 flow traffic data [24]. In the data there are $M = 121$ origin-destination flows carried by $L = 41$ links. The routing matrix $\mathbf{\Phi}$ is given in the data set. Fig. 3 shows the proposed algorithm detects the anomalous flows from the data and moreover, accurately estimates their magnitudes. Moreover, the proposed algorithm is amenable for automated deployment since it requires no parameter tuning.

V. CONCLUSIONS

In this letter we proposed a Bayesian based approach for recovery of sparse signals from compressed measurements when additive noise and a smooth background are present. A memory and computationally efficient constructive algorithm is developed under the framework of Bayesian inference. Advantages over the current state-of-the-art alternative solutions are demonstrated in numerical examples.

REFERENCES

[1] M. Mardani, G. Mateos, and G. B. Giannakis, "Recovery of low-rank plus compressed sparse matrices with application to unveiling traffic anomalies," *IEEE Trans. Inf. Theory*, vol. 59, no. 8, pp. 5186–5205, 2013.

[2] V. Cevher, A. C. Sankaranarayanan, M. Duarteand, D. Reddy, R. G. Baraniuk, and R. Chellappa, "Compressive sensing for background subtraction," in *Eur. Conf. Computer Vision*, 2008, pp. 155–168.

[3] 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.

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

[5] R. G. Baraniuk, "Compressive sensing [lecture notes]," *IEEE Signal Process. Mag.*, vol. 24, no. 4, pp. 118–121, 2007.

[6] E. Candès, X. Li, Y. Ma, and J. Wright, "Robust principal component analysis?," *J. ACM*, vol. 58, no. 3, May 2011.

[7] S. S. Chen, D. L. Donoho, and M. A. Saunders, "Atomic decomposition by basis pursuit," *SIAM Rev.*, vol. 43, no. 1, pp. 129–159, 2001.

[8] M. A. T. Figueiredo, R. D. Nowak, and S. J. Wright, "Gradient projection for sparse reconstruction: Application to compressed sensing and other inverse problems," *IEEE J. Sel. Topics Signal Process.*, vol. 1, no. 4, pp. 586–597, 2007.

[9] A. Beck and M. Teboulle, "A fast iterative shrinkage-thresholding algorithm for linear inverse problems," *SIAM J. Imag. Sci.*, vol. 2, no. 1, pp. 183–202, 2009.

[10] S. Ji, Y. Xue, and L. Carin, "Bayesian compressive sensing," *IEEE Trans. Signal Process.*, vol. 56, no. 6, pp. 2346–2356, 2008.

[11] D. Baron, S. Sarvotham, and R. G. Baraniuk, "Bayesian compressive sensing via belief propagation," *IEEE Trans. Signal Process.*, vol. 58, no. 1, pp. 269–280, 2010.

[12] S. D. Babacan, R. Molina, and A. K. Katsaggelos, "Bayesian compressive sensing using Laplace priors," *IEEE Trans. Image Process.*, vol. 19, no. 1, pp. 53–63, 2010.

[13] J. Cai, E. Candès, and Z. Shen, "A singular value thresholding algorithm for matrix completion," *SIAM J. Optim.*, vol. 20, no. 4, pp. 1956–1982, Jan. 2010.

[14] Z. Lin, A. Ganesh, J. Wright, L. Wu, M. Chen, and Y. Ma, "Fast convex optimization algorithms for exact recovery of a corrupted low-rank matrix," in *Int. Workshop on Computational Advances in Multi-Sensor Adaptive Processing*, Dec. 2009.

[15] Z. Lin, M. Chen, and Y. Ma, The augmented Lagrange multiplier method for exact recovery of corrupted low-rank matrices Univ. Illinois at Urbana-Champaign, , Tech. Rep. UILU-ENG-09-2215, 2009.

[16] X. Ding, L. He, and L. Carin, "Bayesian robust principal component analysis," *IEEE Trans. Image Process.*, vol. 20, no. 12, pp. 3419–3430, 2011.

[17] S. Babacan, M. Luessi, R. Molina, and A. Katsaggelos, "Sparse Bayesian methods for low-rank matrix estimation," *IEEE Trans. Signal Process.*, vol. 60, no. 8, pp. 3964–3977, Aug. 2012.

[18] M. Mardani, G. Mateos, and G. B. Giannakis, "Decentralized sparsity-regularized rank minimization: Algorithms and applications," *IEEE Trans. Signal Process. (to appear)*, vol. 61, no. 11, 2013.

[19] M. Mardani, G. Mateos, and G. B. Giannakis, "Dynamic anomalography: Tracking network anmalies via sparsity and low rank," *IEEE J. Sel. Topics Signal Process.*, vol. 7, no. 1, pp. 50–66, 2013.

[20] Z. Chen, R. Molina, and A. K. Katsaggelos, "A variational approach for sparse component estimation and low-rank matrix recovery," *J. Commun.*, vol. 8, no. 9, pp. 600–611, 2013.

[21] M. W. Seeger and H. Nickisch, "Compressed sensing and Bayesian experimental design," in *Proc. the 25th Int. Conf. Machine Learning*, 2008, pp. 912–919.

[22] M. Tipping and A. Faul, "Fast marginal likelihood maximisation for sparse Bayesian models," in *Proc. 9th Int. Workshop on Artificial Intelligence and Statistics*, C. M. Bishop and B. J. Frey, Eds., 2003.

[23] M. A. T. Figueiredo, "Adaptive sparseness for supervised learning," *IEEE Trans. Patt. Anal. Mach. Intell.*, vol. 25, no. 9, pp. 1150–1159, 2003.

[24] 2014, Internet2 datasets [Online]. Available: http://internet2.edu/observatory/archive/data-collections.html