SCISPACE
formerly Typeset

Open access • Journal Article • DOI:10.1049/IET-SPR.2014.0157

Compressive sensing for cluster structured sparse signals: variational Bayes approach — Source link

Lei Yu, Chen Wei, Jinyuan Jia, Hong Sun

Published on: 05 Sep 2016 - Iet Signal Processing (IET)

Topics: Sparse approximation, Compressed sensing, Nyquist–Shannon sampling theorem, Bayes' theorem and Bayesian probability

Related papers:

- Bayesian compressive sensing for cluster structured sparse signals
- Bayesian Compressive Sensing
- Compressed sensing
- Exploiting Structure in Wavelet-Based Bayesian Compressive Sensing
- Enhanced ISAR Imaging by Exploiting the Continuity of the Target Scene

Share this paper:

View more about this paper here: https://typeset.io/papers/compressive-sensing-for-cluster-structured-sparse-signals-2zibl0r8yo

HAL
open science

Compressive Sensing for Cluster Structured Sparse Signals: Variational Bayes Approach

Lei Yu, Jean-Pierre Barbot, Gang Zheng, Hong Sun

To cite this version:

Lei Yu, Jean-Pierre Barbot, Gang Zheng, Hong Sun. Compressive Sensing for Cluster Structured Sparse Signals: Variational Bayes Approach. 2011. hal-00573953

HAL Id: hal-00573953

https://hal.archives-ouvertes.fr/hal-00573953

Preprint submitted on 6 Mar 2011

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Compressive Sensing for Cluster Structured Sparse Signals: Variational Bayes Approach

Lei Yu, Jean-Pierre Barbot, Gang Zheng, and Hong Sun,

Abstract

Compressive Sensing (CS) provides a new paradigm of sub-Nyquist sampling which can be considered as an alternative to Nyquist sampling theorem. In particular, providing that signals are with sparse representations in some known space (or domain), information can be perfectly preserved even with small amount of measurements captured by random projections. Besides sparsity prior of signals, the inherent structure property underline some specific signals is often exploited to enhance the reconstruction accuracy and promote the ability of recovery. In this paper, we are aiming to take into account the cluster structure property of sparse signals, of which the nonzero coefficients appear in clustered blocks. By modeling simultaneously both sparsity and cluster prior within a hierarchical statistical Bayesian framework, a nonparametric algorithm can be obtained through variational Bayes approach to recover original sparse signals. The proposed algorithm could be slightly considered as a generalization of Bayesian CS, but with a consideration on cluster property. Consequently, the performance of the proposed algorithm is at least as good as Bayesian CS, which is verified by the experimental results.

Index Terms

Compressive Sensing, Cluster Structure, Variational Bayes.

I. Introduction

Compressive Sensing (CS) is recently developed [9], [10], [15], and then attracts lots of researchers. It provides a new paradigm of sub-Nyquist sampling which can be considered as an alternative to Nyquist sampling theorem. In particular, providing that signals are with sparse representations in some known space (or domain), information can be perfectly preserved by random projection measurements.

To reconstruct the original signals, sparse prior is generally exploited into the deficient linear inverse problem, which results in lots of algorithms, Basis Pursuit (BP) [14], [8], Orthogonal Matching Pursuit (OMP) [26], CoSaMP

This work was supported by "Bourses Doctorales en Alternance", PEPS-A2SDC of INSIS-CNRS, NSFC (No. 60872131) and FEDER through CPER 2007-2013 with INRIA Lille-Nord Europe.

L.Yu and H. Sun are with Signal Processing Laboratory, Electronic and Information School, Wuhan University, 430079, Wuhan, China. L. Yu and J-P. Barbot are with ECS-lab ENSEA, 6 Avenue du Ponceau, 95014 Cergy-Pontoise, France and EPI ALIEN, INRIA, France. G. Zheng is with INRIA Lille-Nord Europe, 40 Avenue Halley, 59650, Villeneuve d'Ascq, France.

Email address: yuleiwhu@gmail.com, barbot@ensea.fr, gang.zheng@inria.fr, hongsun@whu.edu.cn.

TABLE I

COMPARISON BETWEEN DIFFERENT CS RECOVERY ALGORITHMS.

Algorithms	Is for Cluster?	Num. of Clusters	Size of Clusters	Fixed Cluster Positions	Sparsity
Greedy algorithms (CoSaMP, OMP, etc)	No	-	-	-	✓
Linear Programming (BP, etc)	No	-	-	-	✓
Iterative Thresholding (IHT, IST)	No	-	-	-	X
Bayesian CS	No	-	-	-	X
Block-CoSaMP [2], [16]	Yes	✓	✓	✓	✓
Dynamical Programming [13]	Yes	✓	X	X	✓
LaMP [12]	Yes	X	X	X	✓
Struct OMP [20]	Yes	✓	X	X	✓
CluSS MCMC [27]	Yes	X	X	X	X

[1] "✓" denotes necessary for the corresponding algorithm.

[2] "X" denotes unnecessary for the corresponding algorithm.

[3] "-" means no consideration for this algorithm.

[23], Bayesian CS [22], Iterative Hard Threshold (IHT) algorithm [7], etc. While besides sparse prior, inherent structures underlying the sparse patterns have been widely employed to improve the recovery accuracy and promote the efficiency, [2], [6], [12], [13], [17], [18], [16].

In this paper, we focus on the cluster structured sparse signals, of which significant coefficients appear in clustered blocks. This kind of sparse pattern is often exploited in many concrete applications, such as multi-band signals, gene expression levels, source localization in sensor networks, MIMO channel equalization, magnetoencephalography [2], [6], [16]. Existing algorithms designed for cluster structured sparse signals always require lots of pre-defined information (Tab. I), such as (a) number of clusters; (b) size of each clusters; (c) positions where clusters are; (d) number of significant coefficients (Sparsity). However, these priors can never be known in real applications, and thus a nonparametric recovery algorithm for cluster structured sparse signal is appealed in practical problems.

A. Motivation

1) From graphical Bayesian model to CS: Considering the process of CS measurement as a hierarchical Bayesian model, namely, a graphical model [11], it provides a new framework for CS [25], [22], [1] and leads to a nonparametric recovery algorithm. In this framework, the sparse constraint is injected through some sparse priors: a Gaussian distribution together with an Inverse Gamma on the invariance, a Laplace distribution, etc. The interpretation of CS with Bayesian model provides a systematic framework, where one could conveniently consider other priors, such as structures on sparse pattern [18], dependencies between multiple signals [21], and so on. Moreover, rather than providing a point estimation for sparse signals, a full posterior density function is provided, which yields "error bars" on the estimated sparse signals. These error bars can be used to give a sense of confidence of the recovered sparse signals.

2) From latent variable model to structures: In a probabilistic, Bayesian approach, through Graphical Models (GMs) [11], [4], latent variables are often exploited to describe the dependencies (or joint probability distributions) between observations and parameters. This method is usually called latent variable model [5], and possibly, results in some non-parametric approaches to Bayesian estimators. By imposing geometrical relations underlying the sparse pattern, structures of the sparse coefficients can be conveniently described by latent variable model [12], [18].

3) From MCMC to VB: In the last work [27], a Bayesian approach to reconstruct the cluster structured sparse signals from compressed measurements has been proposed, where an MCMC re-sampling procedure is exploited for Bayesian inference. It is well known that even though MCMC is capable to find the global solution via infinite MCMC iterations, it cannot guarantee the convergence in finite iterations, which is not applicable all the time. Consequently, we turn to modifying the Bayesian model to make it conjugate and thus can lead to a deterministic solution, through a variational Bayesian method [3], of which the main idea is to optimize the lower bound of the log marginal likelihood function, and simultaneously give a maximum of the posterior distribution.

B. Contribution of this paper

The main contribution of this paper is to exploit the statistical graphical model to describe the cluster structured sparse signals and hence lead to a deterministic algorithm through the variational Bayesian method. The idea of this work is largely inspired by [19], where the resulted algorithm is dedicated to solve the tree structured sparse inverse problems. Even though, it is different from the work of [19]: the considered structures are different, where the tree structure considered in [19] is a directional graphical model but cluster structure in this paper is more likely an undirectional graphical model, and thus it results different models.

C. Outline

The following sections will introduce the proposed algorithm in detail. In section II, the Bayesian clustered sparsity model is addressed, where both the sparse prior and cluster prior are considered. Then using variational Bayes method, the inference of the introduced Bayesian model is implemented in section III. After that, in section IV, some simulations are presented to show the performance of the proposed algorithm. The paper ends with a conclusion.

II. Bayesian Clustered Sparsity Model for CS

In the framework of CS, the sampling process can be modeled as a vector $\boldsymbol{y} \in \mathbb{R}^m$, captured by a multiplication of sensing matrix $\Phi \in \mathbb{R}^{m\times n}$ and the original signals $\boldsymbol{\theta} \in \mathbb{R}^n$, then plus an error, $\boldsymbol{\epsilon}$, as follows:

$$\boldsymbol{y} = \Phi\boldsymbol{\theta} + \boldsymbol{\epsilon} \tag{1}$$

Suppose that the perturbation $\boldsymbol{\epsilon}$ is white, i.e. $\boldsymbol{\epsilon} \sim \mathcal{N}(0, \sigma_0\boldsymbol{I})$ and thus $\boldsymbol{y} \sim \mathcal{N}(\Phi\boldsymbol{\theta}, \sigma_0\boldsymbol{I})$, with $\boldsymbol{I}$ an all one vector with appropriate dimension. To infer the posterior of the noise variance σ_0, a Gamma prior is assigned on the inverse of noise variance $\alpha_0 = \sigma_0^{-1}$, i.e. $\alpha_0 \sim Gamma(c, d)$, conjugate to Gaussian distribution.

In order to introduce both the sparse and cluster prior inside the Bayesian model, we exploit a latent variable $\boldsymbol{z}$ to indicate whether the corresponding element of $\boldsymbol{\theta}$ is nonzero, i.e. $\boldsymbol{\theta} = \boldsymbol{w} \circ \boldsymbol{z}$, where $\circ$ is point wise multiplication and $\boldsymbol{w} \sim \mathcal{N}(0, \boldsymbol{\sigma})$. Meanwhile, Gamma prior is assigned on the inverse of weight variance $\boldsymbol{\alpha} = \boldsymbol{\sigma}^{-1}$, i.e. $\alpha_i \sim Gamma(a, b)$, with α_i the i-th element of $\boldsymbol{\alpha}$, $i \in \{1, ..., n\}$. The overall prior on $\boldsymbol{w}$ with respect to a, b can be evaluated analytically through the integration over $\boldsymbol{\alpha}$, and it corresponds to the Student-t distribution [25]. With appropriate choice of a, b, the Student-t distribution strongly peaked about $\boldsymbol{w} = 0$, and thus the overall prior on $\boldsymbol{w}$ favors sparseness.

Meanwhile, we suppose that the latent variable $\boldsymbol{z}$ is drawn from a Bernoulli distribution with probability $\boldsymbol{\pi}$, i.e. for each element of $\boldsymbol{z}$, $z_i \sim Bernoulli(\pi_i)$. To model the *cluster prior* over the coefficients $\boldsymbol{\theta}$, relations between the current component θ_i and its neighbors should be considered, called the *cluster pattern* of θ_i. Define

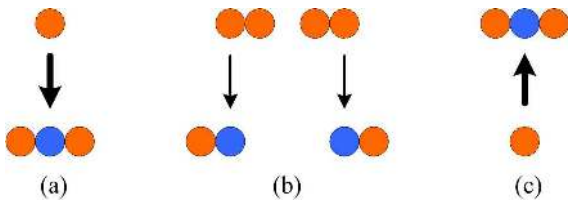


Fig. 1. Three different cluster pattern for 1D signals. (a) **Pattern 0**; (b) **Pattern 1**; (c) **Pattern 2**.

the neighborhood of location i over the coefficients $\boldsymbol{\theta}$, $\mathcal{U}_i = \{j | D(i, j) = 1, j \neq i\}$ with $D(i, j)$ the Euclidean distance between i and j. Denote $\mathcal{V}_N$ the set of all locations over the coefficients θ, $\mathcal{V}_n = \{1, ..., n\}$, then define $\mathcal{J}_i \triangleq \mathcal{U}_i \cap \mathcal{V}_n$. Hence we can denote $\theta_{\mathcal{J}_i}$ the set of components located in the neighbor of the ith coefficient θ_i. And denote $\nu_i = \sum_{j \in \mathcal{J}_i} z_j$ the number of nonzero neighbors for the ith element θ_i. Then we can use this counter ν to categorize the relations into 3 different *cluster patterns*, i.e. **Pattern 0**: "strongly eliminate", when $\nu_i = 0$, namely, the ith element should have large probability of being eliminated, see Fig. 1a; **Pattern 1**: "weakly eliminate", when $\nu_i = 1$, namely, it takes small probability of being eliminated (because of sparsity), see Fig. 1b; **Pattern 2**: "strongly plump", when $\nu_i = 2$, namely, it should be plumped up with large probability, see Fig. 1c. Then according to the cluster patterns, the mixing weight π_i is chosen by the following *pattern selection procedure*:

$$\pi_i = \begin{cases} \pi_i^0, & \text{if Pattern 0} \\ \pi_i^1, & \text{if Pattern 1} \\ \pi_i^2, & \text{if Pattern 2} \end{cases} \tag{2}$$

where π_i^0, π_i^1 and π_i^2 are drawn from different Beta distribution[1].

$$\begin{aligned} \pi_i^0 &\sim \text{Beta}(e^0, f^0) \\ \pi_i^1 &\sim \text{Beta}(e^1, f^1) \\ \pi_i^2 &\sim \text{Beta}(e^2, f^2) \end{aligned} \tag{3}$$

[1]Since Beta distribution is a conjugate prior to Bernoulli likelihood with p the model parameters.

In order to clarify the dependance between the random variables, the distributions for π could be rewritten as follows:

$$\pi_i|_{\boldsymbol{e},\boldsymbol{f},z_{\mathcal{J}_i}} \sim p(\pi_i|\boldsymbol{e},\boldsymbol{f},z_{\mathcal{J}_i}) \tag{4}$$

where $\boldsymbol{e} \triangleq \{e^0, e^1, e^2\}$, $\boldsymbol{f} \triangleq \{f^0, f^1, f^2\}$.

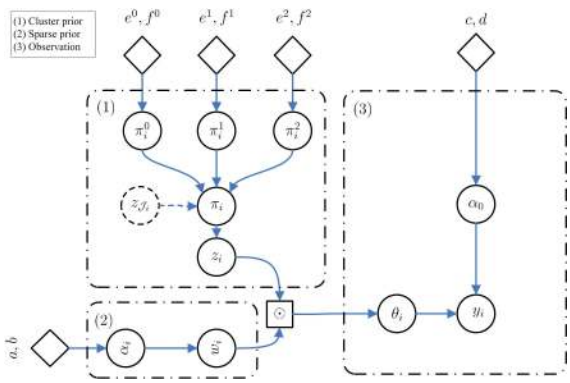


Fig. 2. Graphic depiction for Bayesian clustered sparsity model.

Above all, the overall Bayesian clustered sparsity model is as follows:

$$\begin{aligned}
\boldsymbol{\theta} &= \boldsymbol{w} \circ \boldsymbol{z} \\
\boldsymbol{w} &\sim \mathcal{N}(0, \boldsymbol{\alpha}^{-1}) \\
\alpha_i &\sim Gamma(a, b) \\
z_i &\sim Bernoulli(\pi_i) \\
\boldsymbol{y} &\sim \mathcal{N}(\Phi\boldsymbol{\theta}, \alpha_0^{-1}\boldsymbol{I}) \\
\alpha_0 &\sim Gamma(c, d)
\end{aligned} \tag{5}$$

where π_i is described by (2) and (3). Moreover, it could be depicted by a graphical model in Fig. 2.

III. Variational Bayesian Inference

In this section, we derive a variational Bayesian (VB) algorithm [3] to implement the inference for the Bayesian clustered sparsity model. The main idea of VB is to optimize the lower bound of the log marginal likelihood function, and simultaneously give a maximum for the posterior distribution. Based on the Bayesian clustered sparsity model (5), the complete marginal likelihood can be written as follows:

$$p(\boldsymbol{y}|\mathcal{M}) = \int d\boldsymbol{\theta} d\Theta p(\boldsymbol{\theta}, \boldsymbol{y}, \Theta|\mathcal{M}) \tag{6}$$

with $\mathcal{M} = \{a, b, c, d, \boldsymbol{e}, \boldsymbol{f}\}$ the hyperparameters and $\Theta = \{\boldsymbol{\alpha}, \alpha_0, \boldsymbol{\pi}\}$ the unknown random variables.

In the following, we define $\langle\cdot\rangle_x$ the expectation with respect to random variables[2] x, $\tilde{x}$ the updated estimation for random variable x, and $\boldsymbol{y}_{-k} = \boldsymbol{y} - \sum_{i\neq k} z_i w_i \boldsymbol{\phi}_i$ the contribution of the k-th element of sparse signal on the measurement $\boldsymbol{y}$.

A. The VB-E Step

1) Update for $\boldsymbol{w}$:

$$q(w) \sim \exp\left(\langle\sum_{i=1}^{n} \ln p(w_i|\alpha_i)\rangle_{\alpha_i}\right) \exp\left(\langle \ln p(\boldsymbol{y}|\boldsymbol{w}, \boldsymbol{z}, \alpha_0)\rangle_{\boldsymbol{z},\alpha_0}\right)$$

The posterior of $\boldsymbol{w}$ can be shown to be multivariate Gaussian distribution, with mean $\boldsymbol{\mu}$ and variation Σ, i.e., $\tilde{\boldsymbol{w}} \sim \mathcal{N}(\boldsymbol{\mu}, \Sigma)$, with

$$\begin{aligned} \Sigma &= (\tilde{A} + \tilde{\alpha}_0 \langle Z\Phi^T\Phi Z\rangle)^{-1} \\ \boldsymbol{\mu} &= \tilde{\alpha}_0 \Sigma \tilde{Z} \Phi^T \boldsymbol{y} \end{aligned} \tag{7}$$

where $A = \text{diag}(\boldsymbol{\alpha})$ and $Z = \text{diag}(\boldsymbol{z})$. And given the updated value $\tilde{\boldsymbol{z}}$, we can derive

$$\langle Z\Phi^T\Phi Z\rangle = (\Phi^T\Phi) \circ \left(\tilde{\boldsymbol{z}}\tilde{\boldsymbol{z}}^T + \text{diag}(\tilde{\boldsymbol{z}} \circ (1 - \tilde{\boldsymbol{z}})\right)$$

Consequently, we can derive the update for $\boldsymbol{w}$ as follows:

$$\tilde{\boldsymbol{w}} = \boldsymbol{\mu} \tag{8}$$

2) Update for $\boldsymbol{z}$: For each element of $\boldsymbol{z}$, the posterior could be given as

$$q(z_i) \sim \exp(\langle \ln p(z_i|\pi_i)\rangle_{\pi_i}) \exp(\langle \ln p(\boldsymbol{y}_{-i}|z_i, w_i, \alpha_0)\rangle_{w_i,\alpha_0})$$

Thus the probability that $z_i = 1$ is proportional to

$$\begin{aligned} &p(z_i = 1) \\ &\sim \exp\left(\langle \ln \pi_i\rangle\right) \exp\left(-\frac{\tilde{\alpha}_0}{2}(\langle w_i^2\rangle \boldsymbol{\phi}_i^T \boldsymbol{\phi}_i - 2\tilde{w}_i \boldsymbol{\phi}_i^T \boldsymbol{y}_{-i})\right) \end{aligned} \tag{9}$$

where $\langle w_i^2\rangle = \tilde{w}_i^2 + \sigma_{ii}$ with σ_{ii} the i-th element of diagonal entries of Σ. The probability that $z_i = 0$ is proportional to

$$p(z_i = 0) \sim \exp(\langle \ln(1 - \pi_i)\rangle) \tag{10}$$

where the update for π_i could be referred in (17) in following subsections.

Thus, the update for $\boldsymbol{z}$ can be easily obtained

$$\tilde{z}_i = \frac{p(z_i = 1)}{p(z_i = 0) + p(z_i = 1)} \tag{11}$$

[2]Denote $\langle\cdot\rangle$ if there is only one random variable inside the triangle bracket.

B. The VB-M Step

1) Update for $\boldsymbol{\alpha}$: For each element of α_i, the posterior could be given as

$$\begin{aligned} q(\alpha_i) &\sim p(\alpha_i|a,b)\exp(\langle \ln p(w_i|\alpha_i)\rangle_{w_i}) \\ &\sim Gamma\,(\alpha_i|a',b') \end{aligned} \tag{12}$$

with

$$\begin{aligned} a' &= a + \frac{1}{2}, \\ b' &= b + \frac{\langle w_i^2\rangle}{2} \end{aligned}$$

where the conjugate property between Gamma prior and Gaussian distribution is used. Thus, the update for α_i is obtained:

$$\tilde{\alpha}_i = a'/b' \tag{13}$$

2) Update for α_0:

$$\begin{aligned} q(\alpha_0) &\sim p(\alpha_0|c,d)\exp\left(\langle \ln p(\boldsymbol{y}|\boldsymbol{w},\boldsymbol{z},\alpha_0)\rangle_{\boldsymbol{w},\boldsymbol{z}}\right) \\ &\sim Gamma(\alpha_0|c',d') \end{aligned} \tag{14}$$

with

$$\begin{aligned} c' &= c + \frac{m}{2}, \\ d' &= d + \frac{\langle \|\boldsymbol{y} - \Phi(\boldsymbol{w}\circ\boldsymbol{z})\|^2\rangle_{\boldsymbol{w},\boldsymbol{z}}}{2} \end{aligned}$$

where the conjugate property between Gamma prior and Gaussian distribution is used. And given the updated value of $\boldsymbol{w},\boldsymbol{z}$, the expectation could be derived:

$$\begin{aligned} &\langle \|\boldsymbol{y} - \Phi(\boldsymbol{w}\circ\boldsymbol{z})\|^2\rangle_{\boldsymbol{w},\boldsymbol{z}} \\ &= \boldsymbol{y}^T\boldsymbol{y} - 2(\tilde{\boldsymbol{w}}\circ\tilde{\boldsymbol{z}})^T\Phi^T\boldsymbol{y} + \boldsymbol{I}^T[\langle \boldsymbol{z}\boldsymbol{z}^T\rangle \circ \langle \boldsymbol{w}\boldsymbol{w}^T\rangle \circ (\Phi^T\Phi)]\boldsymbol{I} \end{aligned} \tag{15}$$

where $\langle \boldsymbol{z}\boldsymbol{z}^T\rangle = \tilde{\boldsymbol{z}}\tilde{\boldsymbol{z}}^T + \mathrm{diag}(\tilde{\boldsymbol{z}}\circ(1-\tilde{\boldsymbol{z}}))$ and $\langle \boldsymbol{w}\boldsymbol{w}^T\rangle = \tilde{\boldsymbol{w}}\tilde{\boldsymbol{w}}^T + \Sigma$.

Thus the update for α_0 could be obtained:

$$\tilde{\alpha}_0 = c'/d' \tag{16}$$

3) Update for $\boldsymbol{\pi}$: For each of the π_i, given the updated value for $\boldsymbol{z}$, then we can easily calculate portion of sparsity pattern effecting on π_i:

$$\begin{cases} p(\text{Pattern 0}) &= (1-\tilde{z}_{i-1})(1-\tilde{z}_{i+1}) \\ p(\text{Pattern 1}) &= \tilde{z}_{i-1}(1-\tilde{z}_{i+1}) + (1-\tilde{z}_{i-1})\tilde{z}_{i+1} \\ p(\text{Pattern 2}) &= \tilde{z}_{i-1}\tilde{z}_{i+1} \end{cases}$$

Thus the posterior of π_i^j, with j denoting the sparsity pattern, could be written as:

$$\begin{aligned} q(\pi_i^j) &\sim p(\pi_i^j|e^j,f^j)\exp(\langle \ln p(z_i|\pi_i^j)\rangle_{z_i}) \\ &\sim Beta(\pi_i^j|e_i'^j, f_i'^j) \end{aligned}$$

where

$$e_i'^j = e^j + p(\text{Pattern } j)\tilde{z}_i$$

$$f_i'^j = f^j + p(\text{Pattern } j)(1 - \tilde{z}_i)$$

thus

$$\langle \ln \pi_i^j \rangle = \psi(e_i'^j) - \psi(e_i'^j + f_i'^j)$$

$$\langle \ln(1 - \pi_i^j) \rangle = \psi(f_i'^j) - \psi(e_i'^j + f_i'^j)$$

where $\psi(x) = \frac{d}{dx} \ln \Gamma(x)$ is a digamma function, and then the update for π_i could be obtained:

$$\begin{aligned} \langle \ln \pi_i \rangle &= \sum_{j=0}^{2} p(\text{Pattern } j) \langle \ln \pi_i^j \rangle \\ \langle \ln(1 - \pi_i) \rangle &= \sum_{j=0}^{2} p(\text{Pattern } j) \langle \ln(1 - \pi_i^j) \rangle \end{aligned} \tag{17}$$

C. Summary of the algorithm and acceleration

Given observation $\boldsymbol{y}$ and sensing matrix Φ, the algorithm could be summarized as Algorithm 1.

Algorithm 1 CS for Cluster Structured Sparse Signals via VB

Initialization The hyperparameters $\mathcal{M} = \{a, b, c, d, \boldsymbol{e}, \boldsymbol{f}\}$, the pre-estimated $\boldsymbol{w}(0) = \Phi^T \boldsymbol{y}$, $\boldsymbol{z}(0) = \frac{\boldsymbol{w}}{\max \boldsymbol{w}}$, a stop criterion $\mathcal{C}$.

1: **repeat**

2: Update unknown parameters via equations (13), (16) and (17):

$$\tilde{\Theta} = \{\tilde{\alpha}, \tilde{\alpha}_0, \tilde{\pi}\}$$

3: Update latent parameters via equations (8) and (11):

$$\tilde{\boldsymbol{x}} = \{\tilde{\boldsymbol{w}}, \tilde{\boldsymbol{z}}\}$$

4: **until** $\mathcal{C}$

At the iteration t, define the residual as

$$\text{Res}(t) = \|\boldsymbol{y} - \Phi \tilde{\boldsymbol{\theta}}(t)\|$$

thus

$$\mathcal{C} \triangleq \text{Res}(t) \leqslant \sqrt{m}\sigma_0$$

where σ_0 is the invariance of noise.

1) Comparison to BCS: In the framework of BCS, the inverse problem (reconstruction) is solved by a Sparse Bayesian Learning (SBL) [24], which has been proven to be capable to find the unique sparse solution (Theorem 2 and 3 in [24]).

On the other hand, by setting $\boldsymbol{z} \equiv \mathbf{1}$, the proposed model would degenerate into BCS model, i.e. without consideration on clusters. Roughly, BCS model could be considered as a special case of Bayesian clustered sparsity model, where the prior on clusters provides a guidance of how to assign the priority of choosing bases: "considering first your neighbors". While, in BCS model, this priority does not exist. Consequently, the performance of the proposed algorithm is at least as good as that of BCS and the worst case is when all significant entries of sparse signal are distributed isolated.

2) Acceleration: In the current form, the algorithm requires an inverse problem to update the value of Σ, which is an $n \times n$ matrix, and thus requires an $\mathcal{O}(n^3)$ operations. This can be problematic since in cases of large length of signals, n might be quit large. To alleviate this problem, we compute the Σ as follows:

$$\begin{aligned}\Sigma &= (\tilde{A} + \tilde{\alpha}_0 \langle Z\Phi^T\Phi Z\rangle)^{-1} \\ &= (\tilde{A} + \tilde{\alpha}_0 \tilde{Z}\Phi^T\Phi\tilde{Z} + \text{diag}(\tilde{\boldsymbol{z}} \circ (1-\tilde{\boldsymbol{z}})) \circ (\Phi^T\Phi))^{-1} \\ &= (B + \tilde{\alpha}_0 \tilde{\Phi}^T\tilde{\Phi})^{-1}\end{aligned} \tag{18}$$

where $\tilde{\Phi} = \Phi\tilde{Z}$ and $B = \tilde{A} + \text{diag}(\tilde{\boldsymbol{z}} \circ (1-\tilde{\boldsymbol{z}})) \circ (\Phi^T\Phi)$ which is a diagonal matrix, thus its inverse can be easily computed by directly inverse the elements located at the diagonal, $D = B^{-1}$. Then using the inverse identity property, it has

$$\Sigma = D - D\tilde{\Phi}^T(\alpha_0^{-1}I + \tilde{\Phi}D\tilde{\Phi}^T)^{-1}\tilde{\Phi}D \tag{19}$$

where matrix $\alpha_0^{-1}I + \tilde{\Phi}D\tilde{\Phi}^T$ is with dimension of $m \times m$, which reduces the operations to $\mathcal{O}(m^3)$, with $m \ll n$.

IV. Experiments

To distinguish the algorithm proposed in this paper from the former CluSS algorithm in [27], for simplicity, we denote the proposed algorithm as CluSS-VB. All the hyperparameters are fixed for every experiments as follows: $a = b = c = d = 1e-6$, $(e^0, f^0) = (1/3, 2/3)$, $(e^1, f^1) = (1/3, 1/3)$ and $(e^2, f^2) = (2/3, 1/3)$. The following experiments are organized as follows. A first glance on the performance of CluSS-VB on synthetic cluster structured sparse signals is given. Afterwards, with respect to the oversampling rate, defined as m/s, we compare the recovery accuracy between CluSS-VB and other state-of-the-art CS algorithms, respectively, BP [14], CoSaMP [23], and Baysian Compressive Sensing (BCS) [22]. Then in order to verify the performance for mismatched models, we compare the reconstructions by each of the algorithms with augmenting number of clusters. Meanwhile, the robust to measurement noises is considered and an application on real musical signal is given.

In the following experiments, all sensing matrix are constructed through Gaussian ensemble with normalized row vectors. Moreover, if without clarifying, the measurements are corrupted by a white noise with variance $\sigma_0 = 0.01$. Note that the accuracy of reconstruction is evaluated through SNR between original sparse signal $\boldsymbol{\theta}$ and

its reconstruction $\hat{\boldsymbol{\theta}}$:

$$\text{SNR} = 20\log_{10}\frac{\|\boldsymbol{\theta}\|}{\|\boldsymbol{\theta}-\hat{\boldsymbol{\theta}}\|} \tag{20}$$

A. General view

First, we shall give a glance on the performance of the proposed algorithm and show the basis selection procedure during iterations. The original sparse signal is generated with length $n = 256$, sparsity $s = 30$ and clusters $k = 2$, where cluster information on sizes, locations and numbers is chosen totally randomly and thus is unknown[3]. Moreover, the spikes are randomly generated by Gaussian distribution which coincides with the sparsity model and only $m = 2s = 60$ measurements are obtained by random projections.

In Fig. 3, the inference of $\boldsymbol{z}$ and $\boldsymbol{w}$ are shown along with the iterations. Apparently, the result shows that basis choosing priority is considering the neighbors first. The final result is shown in Fig. 4, where the convergence evolution for $\text{Res}(t)$ is also shown. Moreover, the inference for measurement noise invariance $\hat{\sigma}_0 = 0.0077$, which coincides with the original set $\sigma_0 = 0.01$.

Meanwhile, we also exploit the other state-of-the-art algorithms (BP, CoSaMP, BCS) respectively to reconstruct the same sparse signal, shown in Fig. 5. From the comparison, except for CluSS-VB, all other algorithms cannot be able to reconstruct this sparse signal with only $m = 60$ measurements. On the other hand, like BCS, CluSS-VB can also provide "error bars" for the final reconstruction, which can be used to evaluate the confidence for the estimation.

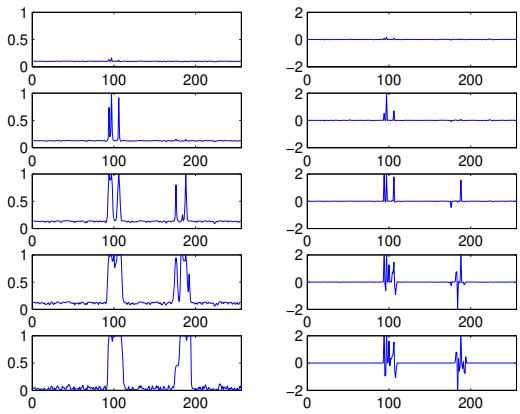


Fig. 3. Inference of $\boldsymbol{z}$ (left column) and $\boldsymbol{w}$(right column) along with iterations: (from top to bottom) 5, 10, 20, 40, 60 iterations.

[3]For Block-CoSaMP proposed in [2], it is impossible to recover this kind of cluster structured signals successfully, and thus we did not make comparison with Block-CoSaMP.

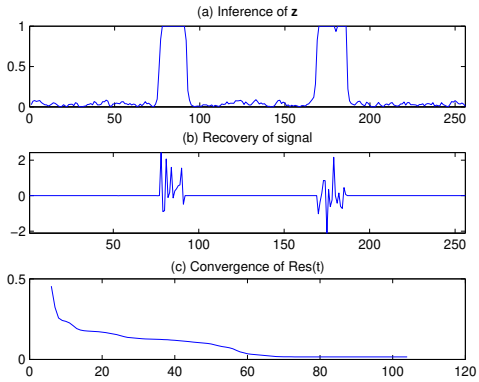


Fig. 4. Performance of CluSS-VB. (a) Inference of latent variable $\boldsymbol{z}$; (b) Recovery of sparse signal; (c) Convergence of Res(t).

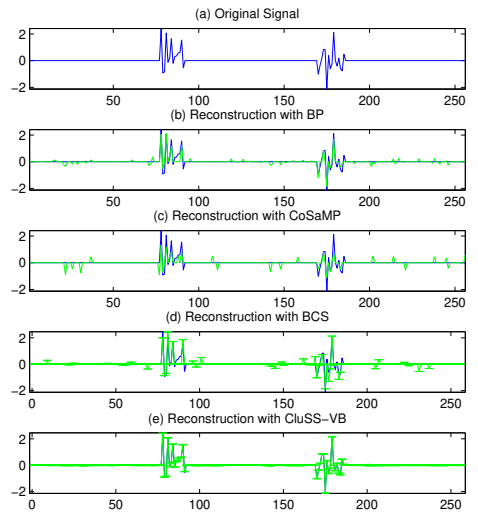
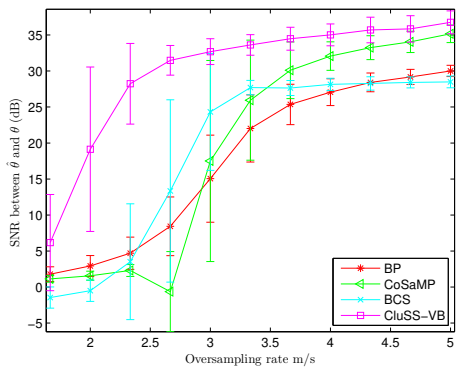


Fig. 5. General view of performance of CluSS-VB. (a) Original Signal, with length $n = 256$, sparsity $s = 30$ and clusters $k = 2$. (b) Recovery via BP and its $SNR = 7.0449dB$; (c) Recovery via CoSaMP and its $SNR = 2.8955dB$; (d) Recovery via BCS and its $SNR = 7.2022dB$; (e) Recovery via CluSS-VB and its $SNR = 30.5702dB$.

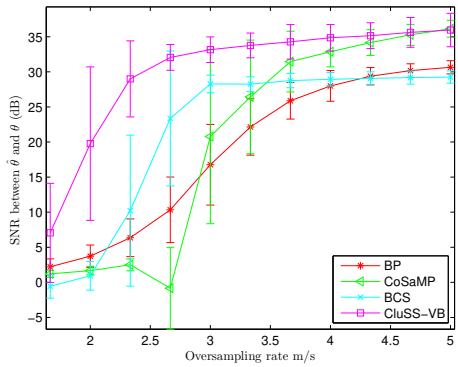
B. Performance with respect to oversampling rate m/s

In order to deeply verify the performance of CluSS-VB, we shall take into account the oversampling rate m/s, which determines how many measurements are enough to reach the successful reconstruction. The cluster structured sparse signals are randomly generated with length $n = 256$ and sparsity $s = 30$, and their nonzero entries drawn from a Gaussian distribution with zero mean and unit variance. Respectively, sparse signals with 1 cluster and 2 clusters are both considered. In each case, for fixed cluster structured sparse signal, the number of measurements is ranging from 50 to 150 with step 10, and for each step, we run the program 100 times with different sensing matrix.

Fig. 6 shows the corresponding results, where (a) is for 1 cluster case and (b) is for 2 clusters case. The results show that CluSS-VB improves both the accuracy and the ability of reconstruction: even with very low oversampling rate, it can obtain desirable reconstructions.



(a) Sparse signal with length $n = 256$, sparsity $s = 30$, and $k = 1$ clusters.



(b) Sparse signal with length $n = 256$, sparsity $s = 30$, and $k = 2$ clusters.

Fig. 6. Performance comparison with respect to oversampling rate.

C. Robustness to noise

This experiment is to testify the robustness of the performance of CluSS-VB to noise (perturbations). Similarly, considering the sparse signals with length $n = 256$, sparsity $s = 30$ and clusters $k = 2$ and with nonzero elements drawn from Gaussian with zero mean and unit variance, we obtain only $m = 100$ measurements, corrupted by a white noise. Let the variance of noise range from $\sigma_0 = 0.01$ to $\sigma_0 = 0.09$, and for each noise level, repeat the experiments 100 times with same sparse signal and sensing matrix, then record the recovery SNR. The results are shown in Fig. 7, and show that the SNR of CluSS-VB is proportional to the noise bound. Meanwhile, it is shown that with consideration of cluster structures, CluSS-VB exhaustively improves the recovery accuracy comparing to other algorithms.

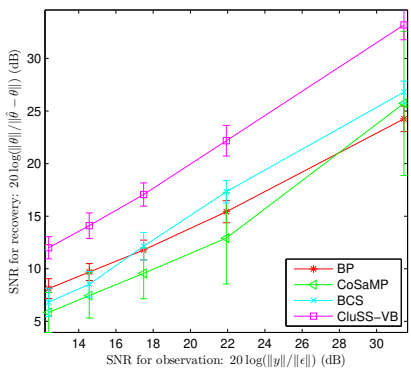


Fig. 7. Robustness to different level of measurement noise.

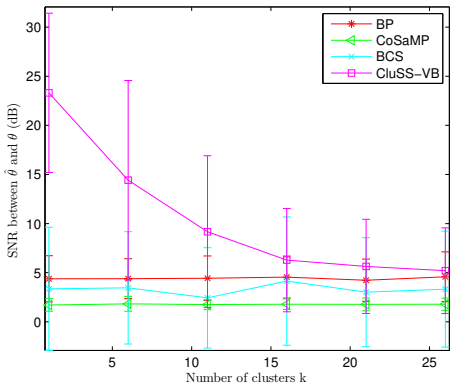
D. Effects of clusters: mismatch models

In this section, we will investigate the effects of clusters on the performance of CluSS-VB. Consider the sparse signals with length $n = 256$ and sparsity $s = 30$ and with nonzero elements drawn from Gaussian. Let the number of clusters k range from 1 to sparsity s, then for different oversampling rate, namely, measurements $m \in \{60, 80, 100\}$, we repeat the experiments 100 times for each number of clusters and each of oversampling rates. The results are shown in Fig. 8, where it shows that: with the number of clusters ascending, (1) the SNR of recovery by CluSS-VB is decreasing, (2) the variance of recovery by CluSS-VB is increasing, (3) while the SNR of recovery by other algorithms almost does not change. Even though the performance of CluSS-VB is decreasing, it still outperforms the other algorithms. This result also implies the robustness to mismatch models for CluSS-VB. And more interestingly, when the number of clusters k goes to the sparsity s, the performance of CluSS-VB tends to converge to the performance of BCS, which coincides with the discussion in section III.

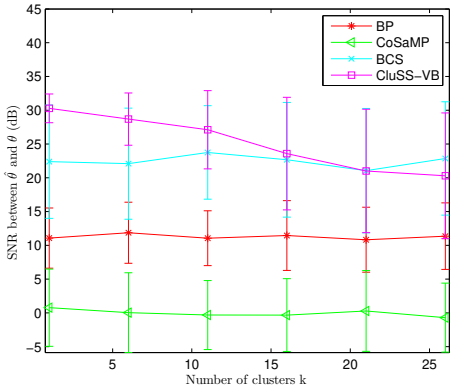
E. Experiments on real musical signals

In the last experiment, we apply the proposed algorithm on real musical signals, which have the property of cluster structured sparsity if considering signals in the frequency domain[4], as shown in Fig. 9, where the significant spectrums are almost clustered together. We choose a clip of music of Mozart played by flute as the test example. The CS procedure is carried out piecewise with length $n = 256$ for each of the pieces. Then varying the number of measurements obtained by random projections, from $m = \lfloor n/2 \rfloor$ to $\lfloor n/5 \rfloor$, corrupted by white noises with variance $\sigma_0 = 0.01$, respectively, we use the different algorithms to recover the original musical signals piecewise from the compressed measurements and then concatenate the recovered pieces. As shown in Fig. 10, with only $m = \lfloor n/5 \rfloor$ measurements, the spectrograms of recovered signals are given for each algorithm. It is shown that CluSS-VB can desirably preserve the clusters of the original sparsity while suppress the isolated spikes, and hence can give better

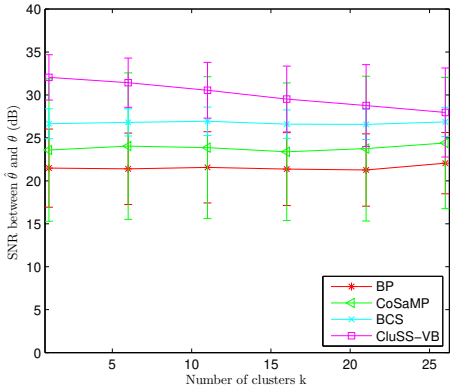
[4]Other transforms are not considered since it is not the important issue in this experiment.



(a) with 60 measurements.



(b) with 80 measurements.



(c) with 100 measurements.

Fig. 8. Effects of clusters on the performance.

reconstructions with high recovery SNR. Meanwhile, we record the SNR for each of the algorithms with different level of oversampling rate, shown in Tab. II. The results show that CluSS-VB gives better reconstructions, especially with lower oversampling rate.

TABLE II

RECOVERY SNR FOR DIFFERENT NUMBER OF MEASUREMENTS. (DB)

Measurements	$\lfloor n/2 \rfloor$	$\lfloor n/3 \rfloor$	$\lfloor n/4 \rfloor$	$\lfloor n/5 \rfloor$
BP	12.6384	9.8339	7.3692	5.8631
CoSaMP	10.5388	7.9345	4.8592	4.4899
BCS	**13.1023**	9.688	7.0389	5.4191
CluSS-VB	12.7616	**11.1035**	**9.2251**	**7.5384**

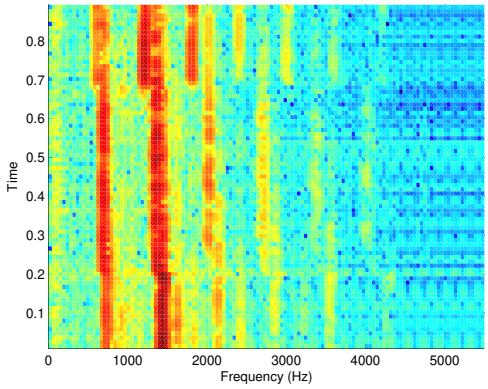


Fig. 9. Spectrogram of a musical signal.

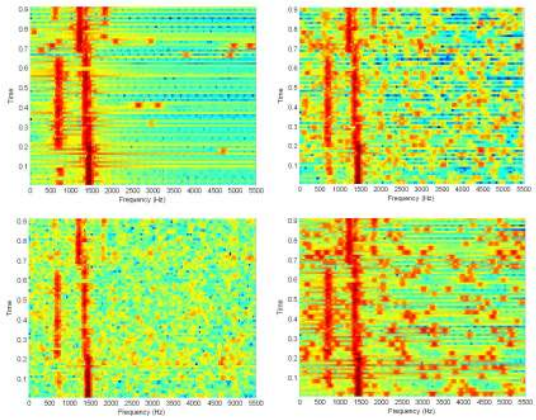


Fig. 10. Spectrogram of reconstructions of musical signals via CluSS-VB (left-top), BCS (right-top), BP (left-bottom) and CoSaMP (right-bottom).

V. CONCLUSION AND PERSPECTIVES

In this paper, we proposed an algorithm, namely, CluSS-VB, to solve the cluster structured sparse signals from compressed measurements. Besides sparse prior, cluster prior on sparsity patterns are considered. Using a statistical

Bayesian graphical model, both priors are injected into a systematical Bayesian framework, where conjugate priors are exploited and results in an analytical solution from the variational Bayesian approach. Unlike the MCMC re-sampling inference, the global convergence (means always with energy descending for cost function) of CluSS-VB is guaranteed. Moreover, roughly, the cluster sparsity model could be considered as a generalization of BCS, and hence CluSS-VB will converge to BCS with mismatched models, i.e. spikes of sparse signals are randomly distributed (without cluster prior).

On the other hand, the theoretical guarantee for the sparse solution with CluSS-VB is not considered in this paper and thus still an open problem. Moreover, even though the algorithm is accelerated by reducing the dimension of the matrix involved into the inverse, an $m \times m$ inverse problem still needs $\mathcal{O}(m^3)$ operations. Consequently, the acceleration of the algorithm needs to be considered in the future works.

REFERENCES

[1] 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.

[2] R. G. Baraniuk, V. Cevher, M. F. Duarte, and C. Hegde, "Model-based compressive sensing," *IEEE Trans. Inf. Theory*, vol. 56, no. 4, pp. 1982–2001, 2010.

[3] M. Beal, "Variational algorithms for approximate bayesian inference," Ph.D. dissertation, Gatsby Computational Neuroscience Unit, University College London, 2003.

[4] C. M. Bishop, *Pattern Recognition and Machine Learning (Information Science and Statistics)*. Secaucus, NJ, USA: Springer-Verlag New York, Inc., 2006.

[5] D. M. Blei, A. Y. Ng, and M. I. Jordan, "Latent dirichlet allocation," *J. Mach. Learn. Res.*, vol. 3, pp. 993–1022, March 2003. [Online]. Available: http://dx.doi.org/10.1162/jmlr.2003.3.4-5.993

[6] T. Blumensath and M. E. Davies, "Sampling theorems for signals from the union of finite-dimensional linear subspaces," *IEEE Trans. Inf. Theory*, vol. 55, no. 4, pp. 1872–1882, 2009.

[7] T. Blumensath, M. Yaghoobi, and M. E. Davies, "Iterative hard thresholding and l0 regularisation," in *Proc. IEEE International Conference on Acoustics, Speech and Signal Processing ICASSP 2007*, vol. 3, 15–20 April 2007, pp. III–877–III–880.

[8] E. J. Candès and T. Tao, "Decoding by linear programming," *IEEE Trans. Inform. Theory*, vol. 51, no. 12, pp. 4203–4215, Dec. 2005.

[9] ——, "Near-optimal signal recovery from random projections: Universal encoding strategies?" *IEEE Trans. Inf. Theory*, vol. 52, no. 12, pp. 5406–5425, Dec. 2006.

[10] E. J. Candès and M. B. Wakin, "An introduction to compressive sampling," *IEEE Signal Process. Mag.*, vol. 25, no. 2, pp. 21–30, March 2008.

[11] V. Cevher, P. Indyk, L. Carin, and R. G. Baraniuk, "Sparse signal recovery and acquisition with graphical models," *IEEE Signal Proc Mag*, vol. 27, no. 6, pp. 92–103, 2010.

[12] V. Cevher, C. Hegde, M. F. Duarte, and R. G. Baraniuk, "Sparse signal recovery using markov random fields," in *Proc. Workshop on Neural Info. Proc. Sys. (NIPS)*, 2008.

[13] V. Cevher, P. Indyk, C. Hegde, and R. G. Baraniuk, "Recovery of clustered sparse signals from compressive measurements," in *Int. Conf. on Sampling Theory and Applications (SAMPTA)*, 2009.

[14] 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.

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

[16] Y. C. Eldar, P. Kuppinger, and H. Bolcskei, "Block-sparse signals: Uncertainty relations and efficient recovery," *IEEE Trans. Signal Process.*, vol. 58, no. 6, pp. 3042–3054, 2010.

[17] Y. C. Eldar and M. Mishali, "Robust recovery of signals from a structured union of subspaces," *IEEE Trans. Inf. Theory*, vol. 55, no. 11, pp. 5302–5316, 2009.

[18] L. He and L. Carin, "Exploiting structure in wavelet-based bayesian compressive sensing," *IEEE Trans. Signal Process.*, vol. 57, no. 9, pp. 3488–3497, 2009.

[19] L. He, H. Chen, and L. Carin, "Tree-structured compressive sensing with variational bayesian analysis," *IEEE Signal Process. Lett.*, vol. 17, no. 3, pp. 233–236, 2010.

[20] J. Huang, T. Zhang, and D. Metaxas, "Learning with structured sparsity," in *The 26th International Conference on Machine Learning, ICML09*, Montreal, Quebec, Canada, June 2009.

[21] S. Ji, D. Dunson, and L. Carin, "Multitask compressive sensing," *IEEE Trans. Signal Process.*, vol. 57, no. 1, pp. 92–106, 2009.

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

[23] D. Needell and J. Tropp, "Cosamp: Iterative signal recovery from incomplete and inaccurate samples," *Applied and Computational Harmonic Analysis*, vol. 26, no. 3, pp. 301 – 321, 2009. [Online]. Available: http://www.sciencedirect.com/science/article/B6WB3-4T1Y404-1/2/a3a764ae1efc1bd0569dcde301f0c6f1

[24] D. P. Wipf and D. D. Rao, "Sparse bayesian learning for basis selection," *IEEE Trans. Signal Process.*, vol. 52, no. 8, pp. 2153–2164, August 2004.

[25] M. E. Tipping, "Sparse bayesian learning and the relevance vector machine," *J. Mach. Learn. Res.*, vol. 1, pp. 211–244, 2001.

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

[27] L. Yu, H. Sun, J. P. Barbot, and G. Zheng, "Bayesian compressive sensing for cluster structured sparse signals," *Signal Processing (submitted)*, 2010.