Simultaneous Bayesian Sparse Approximation With Structured Sparse Models

Wei Chen, *Member, IEEE*, David Wipf, *Member, IEEE*, Yu Wang, Yang Liu, and Ian J. Wassell

Abstract—**Sparse approximation is key to many signal processing, image processing and machine learning applications. If multiple signals maintain some degree of dependency, for example the support sets are statistically related, then it will generally be advantageous to jointly estimate the sparse representation vectors from the measurements vectors as opposed to solving for each signal individually. In this paper, we propose simultaneous sparse Bayesian learning (SBL) for joint sparse approximation with two structured sparse models (SSMs), where one is row-sparse with embedded element-sparse, and the other one is row-sparse plus element-sparse. While SBL has attracted much attention as a means to deal with a single sparse approximation problem, it is not obvious how to extend SBL to SSMs. By capitalizing on a dual-space view of existing convex methods for SMs, we showcase the precision component model and covariance component model for SSMs, where both models involve a common hyperparameter and an innovation hyperparameter that together control the prior variance for each coefficient. The statistical perspective of precision component vs. covariance component models unfolds the intrinsic mechanism in SSMs, and also leads to our development of SBL-inspired cost functions for SSMs. Centralized algorithms, that include ℓ_1 and ℓ_2 reweighting algorithms, and consensus based decentralized algorithms are developed for simultaneous sparse approximation with SSMs. In addition, theoretical analysis is conducted to provide valuable insights into the proposed approach, which includes global minima analysis of the SBL-inspired nonconvex cost functions and convergence analysis of the proposed ℓ_1 reweighting algorithms for SSMs. Superior performance of the proposed algorithms is demonstrated by numerical experiments.**

I. Introduction

SPARSE approximation, that solves linear inverse problems with the principle of parsimony, is key to many signal processing, image processing and machine learning applications [1]–[3]. For example, compressed sensing (CS) [4], [5], i.e., a new sampling paradigm, enables accurate reconstruction of signals with a reduced number of measurements by exploiting a sparse signal model. Another example is sparse subspace clustering that has been used for motion segmentation and face clustering in computer vision [6], [7]. Sparse subspace clustering exploits the sparsity assumption where signals coming from the low dimensional subspace can be effectively represented as a linear combination of other signals belonging to the same low dimensional subspace, and hence requires solving ill-posed inverse problems under the sparsity assumption.

In many cases, we need to estimate K sparse vectors $\mathbf{x}_k \in \mathbb{R}^m$ $(k = 1, \ldots, K)$ from their measurement vectors $\mathbf{y}_k \in \mathbb{R}^{n_k}$, which is normally formulated as the following optimization problem:

$$\min_{\mathbf{X}} \quad \sum_{k=1}^{K} \|\mathbf{\Phi}_k \mathbf{x}_k - \mathbf{y}_k\|_2^2 + \alpha f(\mathbf{X}),$$

where $\mathbf{\Phi}_k \in \mathbb{R}^{n_k \times m}$ is a sensing matrix that could be different across signals, $\mathbf{X} = [\mathbf{x}_1 \ \ldots \ \mathbf{x}_K]$, $f(\cdot)$ is a regularization term to promote sparsity and $\alpha > 0$. If these measurement vectors and associated coefficients maintain some degree of dependency, for example the locations of zero-valued elements (or support sets) are statistically related, then it will generally be advantageous to jointly estimate the sparse representation vectors from the measurements vectors as opposed to solving for each $\mathbf{x}_k$ individually.

Perhaps the simplest setting in multiple measurement vectors (MMVs) is a row-sparse model where the sparse representation vectors of all tasks share a common support, i.e., the set of indices of the nonzero entries. The row-sparse model has been used to improve the performance of CS to jointly reconstruct multiple signals [8]. It has also been verified as an effective remedy to improve the subspace clustering performance in [7]. Various approaches such as simultaneous orthogonal matching pursuit [9], mixed norm minimization [10], an empirical Bayesian strategy [11] and a hierarchical Bayessian model [12], just to name a few, have been proposed to estimate the common support of the signals together with the amplitudes of the coefficients for each signal.

However, the common support requirement is too ideal and restrictive in many real world applications. For instance, the support of a time-varying sequence of images changes slowly over time as shown in [13]. In this paper, we concentrate on less restrictive models with regard to the common support assumption, which are called structured sparse models (SSMs), as illustrated in Fig. 1 where each column corresponds to a sparse signal representation vector. These SSMs have been exploited in a large number of signal processing and machine learning tasks to jointly estimate multiple sparse vectors, e.g., object recognition [14], functional magnetic resonance imaging (fMRI) analysis [15], and dictionary learning [16],

This work is supported by EPSRC Research Grant EP/K033700/1; the Natural Science Foundation of China (61401018, 61671046, U1334202); the State Key Laboratory of Rail Traffic Control and Safety (RCS2016ZT014) of Beijing Jiaotong University.

Wei Chen is with the State Key Laboratory of Rail Traffic Control and Safety, Beijing Jiaotong University, Beijing, China, and also with the Computer Laboratory, University of Cambridge, Cambridge, UK (e-mail: wc253@cam.ac.uk).

David Wipf is with Microsoft Research, Beijing, China, (e-mail: davidwip@microsoft.com).

Yu Wang, Yang Liu and Ian J. Wassell are with the Computer Laboratory, University of Cambridge, Cambridge, UK (e-mail: yw323,yl504,ijw24@cam.ac.uk).

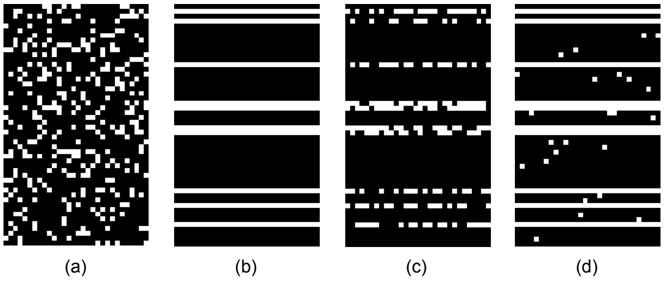


Fig. 1. Various multitask sparse models. From the left to the right: (a) sparse, (b) row-sparse, (c) SSM-1 (row-sparse with embedded element-sparse), and (d) SSM-2 (row-sparse plus element-sparse).

[17]. Convex methods have been developed for simultaneous sparse approximation with SSMs in [14], [15]. The type-2 SSM (SSM-2) is called a dirty sparse model in [14], while the type-1 SSM (SSM-1) is known as a simultaneously structured model. Interestingly, recent work [18] shows that for simultaneously structured models, using optimization with convex-relaxed norms can do no better, orderwise, than exploiting only one of the structures, which reveals a fundamental limitation imposed by using convex relaxation, and gives the motivation to develop nonconvex algorithms for simultaneous sparse approximation with SSMs. Other pre-defined multi-task structural assumptions can be found in [19]–[21].

In addition to the emergence of complex models beyond the element-sparse model and the row-sparse model, decentralized processing has recently attracted increasing attention. It avoids sharing private local data to outsiders, and thus is advantageous compared to centralized processing in privacy-sensitive applications. While most simultaneous sparse approximation algorithms operate in a centralized manner, some decentralized algorithms [22], [23] have been proposed for the case where all signals share a common support. However, those decentralized algorithms cannot be directly extended to SSMs and so benefit from the interaction between the element-sparse model and the row-sparse model.

In this paper, both centralized and decentralized Bayesian algorithms for simultaneous sparse approximation with SSMs are developed, where the cost functions of the optimization problems are nonconvex. While it is not obvious how to model SSMs directly from a statistical perspective, by capitalizing on a dual-space view of existing convex methods for SSMs, we show that the convex penalties for SSM-1 and SSM-2 lead to a precision component model and a covariance component model, respectively, in sparse Bayesian learning (SBL), that deals with a single sparse reconstruction problem from a Bayesian perspective [24], [25]. The intrinsic precision component vs. covariance component models in SSMs inspires our designs that extend SBL to simultaneous sparse approximation with SSMs. With the resultant cost functions corresponding to SSMs, a centralized ℓ_1 reweighting algorithm and a centralized ℓ_2 reweighting algorithm are proposed for both models. Additionally, building on the ℓ_2 reweighting algorithms, decentralized algorithms are developed for both to suit the needs of applications with privacy concerns. In addition, theoretical analyses are conducted to shed further light on the proposed approach, which includes global minima analysis of the SBL-inspired nonconvex cost functions and convergence analysis of the proposed ℓ_1 reweighting algorithms for SSMs. Superior performance of the proposed algorithms is demonstrated by numerical experiments.

The rest of the paper is organized as follows: Section II describes the convex methods for simultaneous sparse approximation with SSMs, and the SBL framework for single sparse reconstruction. In Section III, we investigate convex penalties promoting SSMs from a dual-space view, which provokes our development of SBL-inspired cost functions for promoting SSMs in simultaneous sparse approximation. In Section IV, centralized ℓ_1 reweighting algorithms and centralized ℓ_2 reweighting algorithms are proposed for both models. In the sequel, a consensus-based decentralized approach for SSMs is proposed in Section V. Numerical results are presented in Section VI, followed by experiments on face recognition in Section VII. Conclusions are given in Section VIII.

The following notation is used. For a matrix $\mathbf{X}$, the superscripts $(\mathbf{X})^T$, $(\mathbf{X})^{-1}$, $(\mathbf{X})^{\dagger}$ and $|\mathbf{X}|$ denote the transpose, the inverse, the pseudoinverse and the determinant of $\mathbf{X}$, respectively. The ℓ_0 norm, the ℓ_1 norm, and the ℓ_2 norm of vectors, are denoted by $\|\cdot\|_0$, $\|\cdot\|_1$, and $\|\cdot\|_2$, respectively. The trace of a matrix is denoted by $\texttt{Tr}(\cdot)$. The column i and row i of the matrix $\mathbf{X}$ are denoted by $\mathbf{x}_i$ and $\mathbf{x}_{i,\cdot}$, respectively. $\text{diag}(\mathbf{X})$ denotes a vector with elements composed of the diagonal elements of the matrix $\mathbf{X}$. $\|\mathbf{X}\|_{0,\text{row}}$ denotes the $\ell_{0,\text{row}}$ norm that counts the number of nonzero rows of $\mathbf{X}$. For a set $\mathcal{V}$, $|\mathcal{V}|$ denotes the number of elements in $\mathcal{V}$. $\mathbf{I}$ denotes an identity matrix. $\nabla_x f(x)$ denotes the differential of the function $f(x)$.

II. Background

A. Convex Methods for Simultaneous Sparse Approximation With SSMs

For SSM-1, the matrix $\mathbf{X}$ that is composed of the sparse representation vectors of different signals is row-sparse with embedded element-sparse. Unlike the row-sparse model, SSM-1 does not force different signals to use exactly the same support. This structure is favored in the following optimization problem:

$$\min_{\mathbf{X}} \quad \sum_{k=1}^{K} \|\mathbf{\Phi}_k \mathbf{x}_k - \mathbf{y}_k\|_2^2 + \alpha_1 \sum_{k=1}^{K} \|\mathbf{x}_k\|_0 + \alpha_2 \|\mathbf{X}\|_{0,\text{row}}, \quad (1)$$

where $\alpha_1 > 0$ and $\alpha_2 > 0$ are weights regarding element-sparsity and row-sparsity, respectively. However, the ℓ_0 norm and the $\ell_{0,\text{row}}$ norm in (1) lead to hard combinatorial problems. In [15], the convex ℓ_1 norm and the convex $\ell_{1,2}$ norm are used instead for simultaneous sparse approximation with SSM-1, which leads to solving

$$\min_{\mathbf{X}} \quad \sum_{k=1}^{K} \|\mathbf{\Phi}_k \mathbf{x}_k - \mathbf{y}_k\|_2^2 + \alpha_1 \sum_{k=1}^{K} \|\mathbf{x}_k\|_1 + \alpha_2 \|\mathbf{X}\|_{1,2},$$

where the globally optimal solution can be obtained[1].

[1] With appropriate definition of the contiguous nonzero patterns in fMRI applications, overlapping groups are further considered in [15] to encode structural links between coefficients.

On the other hand, SSM-2 has a structure that is row-sparse plus element-sparse. In order to promote such a structure, a method is proposed in [14] where $\mathbf{X}$ is viewed as the combination of a row-sparse matrix $\mathbf{C} \in \mathbb{R}^{m\times K}$ plus an element-sparse matrix $\mathbf{S} \in \mathbb{R}^{m\times K}$, and the following optimization problem is posed:

$$\min_{\mathbf{C},\mathbf{S}} \quad \sum_{k=1}^{K} \|\mathbf{\Phi}_k(\mathbf{c}_k+\mathbf{s}_k)-\mathbf{y}_k\|_2^2+\beta_1\sum_{k=1}^{K}\|\mathbf{s}_k\|_0+\beta_2\|\mathbf{C}\|_{0,\text{row}}, \tag{2}$$

where $\beta_1 > 0$ and $\beta_2 > 0$ are weights regarding element-sparsity and row-sparsity, respectively. The nonconvex ℓ_0 norm and the nonconvex $\ell_{0,\text{row}}$ norm make the problem in (2) NP-hard. Again, with the use of convex approximation, the problem in (2) is cast as a convex optimization problem[2]

$$\min_{\mathbf{C},\mathbf{S}} \quad \sum_{k=1}^{K} \|\mathbf{\Phi}_k(\mathbf{c}_k+\mathbf{s}_k)-\mathbf{y}_k\|_2^2+\beta_1\sum_{k=1}^{K}\|\mathbf{s}_k\|_1+\beta_2\|\mathbf{C}\|_{1,2}.$$

However, the convex regularizer used for sparsity approximation is known to be too loose to approximate the ℓ_0-type regularizer and so often achieves suboptimal performance.

B. SBL for Single Sparse Approximation

SBL considers the Gaussian likelihood model

$$p(\mathbf{y}_k|\mathbf{x}_k) = \mathcal{N}(\mathbf{y}_k;\mathbf{\Phi}_k\mathbf{x}_k,\nu\mathbf{I})$$

and priors

$$p(\mathbf{x}_k) = \mathcal{N}(\mathbf{x}_k;\mathbf{0},\mathbf{\Gamma}_k),$$

where ν denotes noise variance (which is assumed to be known, although it can be learned), $\mathbf{\Gamma}_k$ is a diagonal matrix with $\text{diag}(\mathbf{\Gamma}_k) = \boldsymbol{\gamma}_k$, and $\boldsymbol{\gamma}_k$ is a vector of hyperparameters governing the prior variance of the elements in signal k. SBL has a cost function favoring a sparse $\boldsymbol{\gamma}_k$, which then leads to a sparse $\mathbf{x}_k$.

From a Bayesian perspective, there are two different ways to find the sparse representation vectors. The first is to apply maximum a posterior (MAP) estimates of $\mathbf{x}_k$ (referred to *Type I* estimation), which gives

$$\begin{aligned}\mathbf{x}_{k(I)} &= \arg\min_{\mathbf{x}_k,\boldsymbol{\gamma}_k\succeq\mathbf{0}} -\log p(\mathbf{y}_k|\mathbf{x}_k)p(\mathbf{x}_k|\mathbf{\Gamma}_k)\\ &= \arg\min_{\mathbf{x}_k,\boldsymbol{\gamma}_k\succeq\mathbf{0}} \|\mathbf{y}_k-\mathbf{\Phi}_k\mathbf{x}_k\|_2^2+\nu\mathbf{x}_k^T\mathbf{\Gamma}_k^{-1}\mathbf{x}_k.\end{aligned}$$

With appropriate selection of a sparsity-driven hyper-prior, *Type I* estimation also forms the solution in many algorithms including the least absolute shrinkage and selection operator (Lasso) [26], ℓ_p norm approaches [27], FOCUSS [28] and iterative reweighted ℓ_1 methods [29].

Alternatively, instead of minimizing over both $\mathbf{x}_k$ and $\boldsymbol{\gamma}_k$ as in (3), *Type II* estimation treats $\mathbf{x}_k$ as hidden variables, integrates them out, and conducts MAP estimation on $\boldsymbol{\gamma}_k$ as

$$\begin{aligned}\boldsymbol{\gamma}_{k(II)} &= \arg\max_{\boldsymbol{\gamma}_k} p(\boldsymbol{\gamma}_k|\mathbf{y}_k)\\ &= \arg\max_{\boldsymbol{\gamma}_k}\int p(\mathbf{y}_k|\mathbf{x}_k)p(\mathbf{x}_k;\boldsymbol{\gamma}_k)d\mathbf{x}_k \\ &= \arg\min_{\boldsymbol{\gamma}_k}\mathbf{y}_k^T\mathbf{\Sigma}_k^{-1}\mathbf{y}_k+\log|\mathbf{\Sigma}_k|,\end{aligned} \tag{3}$$

where $\mathbf{\Sigma}_k = \nu\mathbf{I}+\mathbf{\Phi}_k\mathbf{\Gamma}_k\mathbf{\Phi}_k^T$. Given the likelihood and prior, the posterior distribution $p(\mathbf{x}_k|\mathbf{y}_k;\mathbf{\Sigma}_k)$ is a Gaussian with mean

$$\mathbf{x}_{k(II)} = \mathbf{\Gamma}_{k(II)}\mathbf{\Phi}_k^T(\nu\mathbf{I}+\mathbf{\Phi}_k\mathbf{\Gamma}_{k(II)}\mathbf{\Phi}_k^T)^{-1}\mathbf{y}_k. \tag{4}$$

Type II estimation is also known as empirical Bayesian and is used in algorithms such as SBL and the relevance vector machine (RVM) [24].

The logarithm term $\log|\mathbf{\Sigma}_k|$ in the cost function in (3) is a concave function with respect to $\boldsymbol{\gamma}_k$ according to Lemma 1 of [25], and thus it favors a sparse $\boldsymbol{\gamma}_k$, which further leads to a sparse solution via (4). The logarithm term in SBL is a non-separable sparse penalty. By "non-separable", it means that the sparse penalty cannot be expressed as a summation over functions of the individual coefficients. Owing to the use of a non-separable sparse penalty, SBL is advantageous, in terms of reconstruction accuracy, to many methods such as ℓ_p norm approaches [27] and FOCUSS [28], which use separable sparse penalties [30]. We refer interested readers to [25], [31] for detailed analysis on the advantages of SBL.

In view of the superiority of SBL in dealing with single sparse approximation, it is desired to extend SBL to the case of SSMs. However, SBL uses independent priors for multiple signals, which fails to consider any inter-signal correlation, and thus is unable to benefit from simultaneous sparse approximation. It is not obvious how to proceed for either SSM-1 or SSM-2 with the current SBL framework.

III. From SBL to Simultaneous Bayesian Sparse Approximation With SSMs

A. A Dual-Space View Of Convex Penalties for SSMs

While *Type I* and *Type II* estimation may seem quite different, comparisons of the two can be made by using a dual-space view of the underlying cost functions [31], [32], i.e., expressing both the Type I and Type II objective in terms of either $\mathbf{x}_k$ or $\boldsymbol{\gamma}_k$. The dual-space view sheds light on the connections between the two approaches, and helps in developing efficient update rules.

1) Precision Component Model For SSM-1: We note that the element-sparse penalty $\|\mathbf{x}_k\|_1$ has a variational representation as

$$\|\mathbf{x}_k\|_1 = \min_{\gamma^a_{k\,j}\geq 0}\frac{1}{2}\sum_j\frac{{x_k}_j^2}{\gamma^a_{k\,j}}+\gamma^a_{k\,j}, \tag{5}$$

while the row-sparse penalty $\|\mathbf{X}\|_{1,2}$ can be viewed as

$$\|\mathbf{X}\|_{1,2} = \min_{\gamma^c_j\geq 0}\frac{1}{2}\sum_j\frac{\sum_k {x_k}_j^2}{\gamma^c_j}+\gamma^c_j, \tag{6}$$

where $\gamma^a_{k\,j}$ and γ^c_j are scalars, $\boldsymbol{\gamma}^c$ is a vector that is common to all signals, and $\boldsymbol{\gamma}^a_k$ is a vector that is uniquely associated with signal k $(k = 1,\ldots,K)$. Therefore, a convex optimization

[2] As a convex approximation, the $\ell_{0,\text{row}}$ norm is replaced by the $\ell_{1,\infty}$ norm in [14].

problem, that favors SSM-1, can be given by

$$\begin{aligned}
&\min_{\mathbf{X}} \sum_{k=1}^{K} \frac{1}{\nu}\|\mathbf{y}_k - \mathbf{\Phi}_k\mathbf{x}_k\|_2^2 + 2\alpha\|\mathbf{x}_k\|_1 + 2\|\mathbf{X}\|_{1,2} \\
&= \min_{\substack{\mathbf{X},\boldsymbol{\gamma}^c\succeq\mathbf{0},\\ \{\boldsymbol{\gamma}_k^a\succeq\mathbf{0}\}}} \sum_{k=1}^{K} \Bigg(\frac{1}{\nu}\|\mathbf{y}_k - \mathbf{\Phi}_k\mathbf{x}_k\|_2^2 + \alpha\mathbf{x}_k^T(\mathbf{\Gamma}_k^a)^{-1}\mathbf{x}_k \\
&\qquad + \mathbf{x}_k^T(\mathbf{\Gamma}^c)^{-1}\mathbf{x}_k + \alpha\mathtt{Tr}(\mathbf{\Gamma}_k^a)\Bigg) + \mathtt{Tr}(\mathbf{\Gamma}^c) \qquad (7) \\
&= \min_{\substack{\mathbf{X},\boldsymbol{\gamma}^c\succeq\mathbf{0},\\ \{\boldsymbol{\gamma}_k^a\succeq\mathbf{0}\}}} \sum_{k=1}^{K} \Bigg(\frac{1}{\nu}\|\mathbf{y}_k - \mathbf{\Phi}_k\mathbf{x}_k\|_2^2 + \mathbf{x}_k^T(\mathbf{\Gamma}_k^a)^{-1}\mathbf{x}_k \\
&\qquad + \mathbf{x}_k^T(\mathbf{\Gamma}^c)^{-1}\mathbf{x}_k + \alpha^2\mathtt{Tr}(\mathbf{\Gamma}_k^a)\Bigg) + \mathtt{Tr}(\mathbf{\Gamma}^c),
\end{aligned}$$

where $\alpha > 0$, and $\mathbf{\Gamma}^c$ and $\mathbf{\Gamma}_k^a$ are diagonal matrices corresponding to $\boldsymbol{\gamma}^c$ and $\boldsymbol{\gamma}_k^a$, respectively.

With the definition of $\mathbf{\Sigma}_k^{ac} = \nu\mathbf{I} + \mathbf{\Phi}_k((\mathbf{\Gamma}^c)^{-1} + (\mathbf{\Gamma}_k^a)^{-1})^{-1}\mathbf{\Phi}_k^T$ and using the relationship

$$\begin{aligned}
&\sum_{k=1}^{K} \mathbf{y}_k^T(\mathbf{\Sigma}_k^{ac})^{-1}\mathbf{y}_k \\
&= \min_{\mathbf{X}} \sum_{k=1}^{K} \frac{1}{\nu}\|\mathbf{y}_k - \mathbf{\Phi}_k\mathbf{x}_k\|_2^2 + \mathbf{x}_k^T(\mathbf{\Gamma}_k^a)^{-1}\mathbf{x}_k + \mathbf{x}_k^T(\mathbf{\Gamma}^c)^{-1}\mathbf{x}_k
\end{aligned}$$

as in [31], a different view of the existing convex cost function (7) can be derived in γ-space, i.e.,

$$L_{(I)}^{\text{pre}}(\gamma^c, \{\gamma_k^a\}) = \sum_{k=1}^{K} \mathbf{y}_k^T(\mathbf{\Sigma}_k^{ac})^{-1}\mathbf{y}_k + \alpha^2\mathtt{Tr}(\mathbf{\Gamma}_k^a) + \mathtt{Tr}(\mathbf{\Gamma}^c). \tag{8}$$

By comparing the data-related term $\mathbf{y}_k^T(\mathbf{\Sigma}_k^{ac})^{-1}\mathbf{y}_k$ for SSM-1 and the data-related term in the cost function of SBL in (3), it is observed that the common component $\boldsymbol{\gamma}^c$ and innovation component $\boldsymbol{\gamma}_k^a$ interact with each other in the manner of a precision component model, i.e.,

$$(\mathbf{\Gamma}_k)^{-1} = (\mathbf{\Gamma}^c)^{-1} + (\mathbf{\Gamma}_k^a)^{-1}. \tag{9}$$

Defining $\mathbf{\Gamma}_{(I)}^c$ and $\{\mathbf{\Gamma}_{k(I)}^a\}$ as the solutions of minimizing (8), and $(\mathbf{\Gamma}_{k(I)})^{-1} = ({\mathbf{\Gamma}^c}_{(I)})^{-1} + (\mathbf{\Gamma}_{k(I)}^a)^{-1}$ according to the precision component model in (9), then the solution obtained from (7) satisfies

$${\mathbf{x}_k}^{\text{pre}} = \mathbf{\Gamma}_{k(I)}\mathbf{\Phi}_k^T(\nu\mathbf{I} + \mathbf{\Phi}_k\mathbf{\Gamma}_{k(I)}\mathbf{\Phi}_k^T)^{-1}\mathbf{y}_k. \tag{10}$$

The precision component model in (9), where the support of the vector $\boldsymbol{\gamma}_k$ is the intersection of $\boldsymbol{\gamma}_k^a$ and $\boldsymbol{\gamma}_k^c$, leads to solutions following SSM-1 via (10).

2) Covariance Component Model For SSM-2: According to the variational representation of the convex element-sparse penalty in (5) and the variational representation of the convex row-sparse penalty in (6), the existing convex method for SSM-2 can be expressed as

$$\begin{aligned}
&\min_{\mathbf{X}} \sum_{k=1}^{K} \frac{1}{\nu}\|\mathbf{y}_k - \mathbf{\Phi}_k(\mathbf{c}_k + \mathbf{s}_k)\|_2^2 + 2\beta\|\mathbf{s}_k\|_1 + 2\|\mathbf{C}\|_{1,2} \\
&= \min_{\substack{\mathbf{C},\mathbf{S},\boldsymbol{\gamma}^c\succeq\mathbf{0},\\ \{\boldsymbol{\gamma}_k^s\succeq\mathbf{0}\}}} \sum_{k=1}^{K} \Bigg(\frac{1}{\nu}\|\mathbf{y}_k - \mathbf{\Phi}_k(\mathbf{c}_k + \mathbf{s}_k)\|_2^2 + \beta\mathbf{s}_k^T(\mathbf{\Gamma}_k^s)^{-1}\mathbf{s}_k \\
&\qquad + \mathbf{c}_k^T(\mathbf{\Gamma}^c)^{-1}\mathbf{c}_k + \beta\mathtt{Tr}(\mathbf{\Gamma}_k^s)\Bigg) + \mathtt{Tr}(\mathbf{\Gamma}^c) \\
&= \min_{\substack{\mathbf{C},\mathbf{S},\boldsymbol{\gamma}^c\succeq\mathbf{0},\\ \{\boldsymbol{\gamma}_k^s\succeq\mathbf{0}\}}} \sum_{k=1}^{K} \Bigg(\frac{1}{\nu}\|\mathbf{y}_k - \mathbf{\Phi}_k(\mathbf{c}_k + \mathbf{s}_k)\|_2^2 + \mathbf{s}_k^T(\mathbf{\Gamma}_k^s)^{-1}\mathbf{s}_k \\
&\qquad + \mathbf{c}_k^T(\mathbf{\Gamma}^c)^{-1}\mathbf{c}_k + \beta^2\mathtt{Tr}(\mathbf{\Gamma}_k^s)\Bigg) + \mathtt{Tr}(\mathbf{\Gamma}^c),
\end{aligned} \tag{11}$$

where $\beta > 0$, $\boldsymbol{\gamma}_k^s = \text{diag}(\mathbf{\Gamma}_k^s)$ is uniquely associated with signal k $(k = 1, \ldots, K)$, and $\boldsymbol{\gamma}^c$ is common to all signals.

By defining $\mathbf{\Sigma}_k^{sc} = \nu\mathbf{I} + \mathbf{\Phi}_k(\mathbf{\Gamma}^c + \mathbf{\Gamma}_k^s)^{-1}\mathbf{\Phi}_k^T$ and using the relationship

$$\begin{aligned}
\sum_{k=1}^{K} \mathbf{y}_k^T(\mathbf{\Sigma}_k^{sc})^{-1}\mathbf{y}_k = &\min_{\mathbf{C},\mathbf{S}} \sum_{k=1}^{K} \frac{1}{\nu}\|\mathbf{y}_k - \mathbf{\Phi}_k(\mathbf{c}_k + \mathbf{s}_k)\|_2^2 \\
&+ \mathbf{s}_k^T(\mathbf{\Gamma}_k^s)^{-1}\mathbf{s}_k + \mathbf{c}_k^T(\mathbf{\Gamma}^c)^{-1}\mathbf{c}_k,
\end{aligned} \tag{12}$$

we can express the existing convex cost function in γ-space as

$$L_{(I)}^{\text{cov}}(\gamma^c, \{\gamma_k^s\}) = \sum_{k=1}^{K} \mathbf{y}_k^T(\mathbf{\Sigma}_k^{sc})^{-1}\mathbf{y}_k + \beta^2\mathtt{Tr}(\mathbf{\Gamma}_k^s) + \mathtt{Tr}(\mathbf{\Gamma}^c). \tag{13}$$

Comparing the data-related term $\mathbf{y}_k^T(\mathbf{\Sigma}_k^{sc})^{-1}\mathbf{y}_k$ in (13) and the data-related term in the cost function of the SBL (3), we note that the common component $\boldsymbol{\gamma}^c$ and innovation component $\boldsymbol{\gamma}_k^s$ interact with each other in the manner of a covariance component model, i.e.,

$$\mathbf{\Gamma}_k = \mathbf{\Gamma}^c + \mathbf{\Gamma}_k^s. \tag{14}$$

Assume $\mathbf{\Gamma}_{(I)}^c$ and $\{\mathbf{\Gamma}_{k(I)}^s\}$ are the solutions of minimizing (13), and $\mathbf{\Gamma}_{k(I)} = {\mathbf{\Gamma}^c}_{(I)} + \mathbf{\Gamma}_{k(I)}^s$. Then the solution obtained from (11) satisfies

$${\mathbf{x}_k}^{\text{cov}} = \mathbf{\Gamma}_{k(I)}\mathbf{\Phi}_k^T(\nu\mathbf{I} + \mathbf{\Phi}_k\mathbf{\Gamma}_{k(I)}\mathbf{\Phi}_k^T)^{-1}\mathbf{y}_k. \tag{15}$$

Although all the signals are linked via the common hyperparameters in $\boldsymbol{\gamma}^c$, the interplay between the common component, i.e., $\mathbf{\Gamma}^c$, and innovation components, i.e., $\mathbf{\Gamma}_k^a$ or $\mathbf{\Gamma}_k^s$, are different in the precision component model and the covariance component model. Specifically, the support of $\boldsymbol{\gamma}_k$ is the union of $\boldsymbol{\gamma}_k^a$ and $\boldsymbol{\gamma}_k^c$ in the covariance component model, which promotes SSM-2 via (15).

B. SBL-Inspired Cost Functions for SSMs

Given the dual-space view of the convex penalties for SSMs, a straightforward idea for extending SBL to SSMs is to consider two different parameterizations, one via a precision component model as in (9), and the other one via a covariance component model as in (14). All the signals are linked via the common set of hyperparameters in $\mathbf{\Gamma}^c$. Then following the Bayesian mold as in SBL, the cost functions of the precision component model and the covariance component model have the form

$$\sum_{k=1}^{K} \log\left|\mathbf{\Sigma}_k^{ac}\right| + \mathbf{y}_k^T(\mathbf{\Sigma}_k^{ac})^{-1}\mathbf{y}_k, \tag{16}$$

and

$$\sum_{k=1}^{K} \log\left|\mathbf{\Sigma}_k^{sc}\right| + \mathbf{y}_k^T(\mathbf{\Sigma}_k^{sc})^{-1}\mathbf{y}_k, \tag{17}$$

respectively.

However, the common component $\mathbf{\Gamma}^c$ and the innovation component, i.e., $\mathbf{\Gamma}_k^a$ in SSM-1 and $\mathbf{\Gamma}_k^s$ in SSM-2, are not identifiable in either (16) or (17). Specifically, one can always let γ^c be a vector of all ones and adjust $\{\mathbf{\Gamma}_k^a\}$ accordingly without changing the value of the objective in (16), or let γ^c be a vector of all zeros and adjust $\{\mathbf{\Gamma}_k^s\}$ accordingly without changing the value of the objective in (17).

With regular SBL it is not clear how to make simultaneous Bayesian sparse approximation with SSMs. However, we can replace the convex penalties in the existing models with the SBL counterpoints to reap some of the corresponding benefits, even though we deviate from any formal probabilistic model. By doing so, we put forth the following cost function in the γ-space for SSM-1

$$L^{\text{pre}}(\gamma^c, \{\gamma_k^a\}) = \sum_{k=1}^{K} \alpha \log\left|\mathbf{\Sigma}_k^a\right| + \log\left|\mathbf{\Sigma}_k^c\right| + \mathbf{y}_k^T(\mathbf{\Sigma}_k^{ac})^{-1}\mathbf{y}_k, \tag{18}$$

where $\mathbf{\Sigma}_k^a = \frac{\nu}{2}\mathbf{I} + \mathbf{\Phi}_k\mathbf{\Gamma}_k^a\mathbf{\Phi}_k^T$ and $\mathbf{\Sigma}_k^c = \frac{\nu}{2}\mathbf{I} + \mathbf{\Phi}_k\mathbf{\Gamma}^c\mathbf{\Phi}_k^T$. With the covariance component model, we pose the following cost function in the γ-space for SSM-2

$$L^{\text{cov}}(\gamma^c, \{\gamma_k^s\}) = \sum_{k=1}^{K} \beta \log\left|\mathbf{\Sigma}_k^s\right| + \log\left|\mathbf{\Sigma}_k^c\right| + \mathbf{y}_k^T(\mathbf{\Sigma}_k^{sc})^{-1}\mathbf{y}_k, \tag{19}$$

where $\mathbf{\Sigma}_k^s = \frac{\nu}{2}\mathbf{I} + \mathbf{\Phi}_k\mathbf{\Gamma}_k^s\mathbf{\Phi}_k^T$.

As the log-determinant function is a concave, non-decreasing function, the term $\log|\mathbf{\Sigma}_k^c|$ favors a sparse $\boldsymbol{\gamma}^c$ that is common to all signals, and the term $\log|\mathbf{\Sigma}_k^a|$ and $\log|\mathbf{\Sigma}_k^s|$ promote sparse $\boldsymbol{\gamma}_k^a$ and $\boldsymbol{\gamma}_k^s$ that are unique to each signal. The interaction between $\boldsymbol{\gamma}^c$ and $\mathbf{\Sigma}_k^a$, and the interaction between $\boldsymbol{\gamma}^c$ and $\mathbf{\Sigma}_k^s$, are different in the data related terms in (18) and (19), which promotes different inter-signal structure. Given the estimated hyper-parameters, the estimated sparse representation vectors can be calculated as (4). The weights α and β are used to balance row sparsity and element sparsity in the two cost functions. The value of α and β can be tuned with training data or given by empirical knowledge[3].

C. Some Comments on the Cost Functions

We now provide the rationale why the cost functions, that result from hyperpriors that are distinct from those used in regular SBL have the ability to find exactly the true sparse generating vectors. Ideally, for a signal that has a sparse structure, it is expected that the maximal sparse one should be the solution that minimizes the sparse linear inverse problem (at least in the noiseless case). In the following result, we show that the global minima of the cost functions in (18) and (19) produce the maximally sparse solutions.

Definition 1: The spark, spark$[\mathbf{A}]$, of a given matrix $\mathbf{A}$ is the smallest number of columns of $\mathbf{A}$ that are linearly dependent.

Theorem 1: **(Global Minima)** For $\forall k$, let the maximally sparse solution to $\mathbf{y}_k = \mathbf{\Phi}_k\mathbf{x}_k$ be achieved at $\hat{\mathbf{x}}_k$ with $\|\hat{\mathbf{x}}_k\|_0 < n_k$, and spark$[\mathbf{\Phi}_k] = n_k + 1$. Let $\hat{\gamma}_k$ denote hyper-parameters such that $\hat{\mathbf{x}}_k = \hat{\mathbf{\Gamma}}_k^{1/2}(\mathbf{\Phi}_k\hat{\mathbf{\Gamma}}_k^{1/2})^\dagger\mathbf{y}_k$, Then

- the global minima of the cost function $\lim_{\nu\to 0} L^{\text{pre}}(\gamma^c, \{\gamma_k^a\})$ is achieved at γ^c and $\{\gamma_k^a\}$ such that $\left((\mathbf{\Gamma}^c)^{-1} + (\mathbf{\Gamma}_k^a)^{-1}\right)^{-1} = \hat{\mathbf{\Gamma}}_k$, irrespective of the weight α;
- the global minima of the cost function $\lim_{\nu\to 0} L^{\text{cov}}(\gamma^c, \{\gamma_k^s\})$ is achieved at γ^c and $\{\gamma_k^s\}$ such that $\boldsymbol{\gamma}^c + \boldsymbol{\gamma}_k^s = \hat{\boldsymbol{\gamma}}_k$, irrespective of the weight β.

The proof of this theorem is given in Appendix A. Here, the condition on the spark can be satisfied almost surely by any random matrix with $n_k \leq m$ [34]. This result explains why the proposed cost functions are able to find exactly the true sparse generating vectors. We note that although the global minima of the cost functions in (18) and (19) may be equivalent to the global minima of independently solving a sparsity maximization problem for each signal, the landscape of the entire cost functions are not identical, as the inter-signal structure is considered in the two models via γ^c, which could be advantageous in avoiding distracting local minima.

In addition, owing to the log-determinant terms, the proposed simultaneous sparse approximation problems in (18) and (19) have a non-separable sparse penalty. By "non-separable", it means that the sparse penalty cannot be expressed as a summation over functions of the individual coefficients. The advantages of using a non-separable sparse penalty over a separable sparse penalty are elaborated in detail in [30].

IV. Algorithms for Simultaneous Bayesian Sparse Approximation With SSMs

Both the formulation (18) associated with the precision component model and the formulation (19) associated with the covariance component model are nonconvex and difficult to solve. In this section, we develop two different types of schemes, i.e., ℓ_1 reweighting schemes and ℓ_2 reweighting schemes, to solve the optimization problems. Both schemes are derived by using majorization-minimization that repeatedly minimize and update surrogate functions that majorize the original cost functions.

[3] Owing to the two optimization objectives, i.e., row-sparsity and element-sparsity, in simultaneous sparse approximation with SSMs, the existing approaches [14], [15], [33] also turn the multiobjective optimization problem into a scalar optimization problem with the use of an application-based weight to balance the two objectives.

A. ℓ_1 Reweighting Schemes

Firstly, as the log-determinant term is a concave nondecreasing function, we have the following upper bounds

$$\log|\mathbf{\Sigma}_k^a| = \min_{\mathbf{z}_k^a}(\mathbf{z}_k^a)^T\boldsymbol{\gamma}_k^a - \bar{h}_k^a(\mathbf{z}_k^a),$$

$$\log|\mathbf{\Sigma}_k^s| = \min_{\mathbf{z}_k^s}(\mathbf{z}_k^s)^T\boldsymbol{\gamma}_k^s - \bar{h}_k^s(\mathbf{z}_k^s),$$

$$\sum_{k=1}^{K}\log|\mathbf{\Sigma}_k^c| = \min_{\mathbf{z}^c}(\mathbf{z}^c)^T\boldsymbol{\gamma}^c - \bar{h}^c(\mathbf{z}^c),$$

where $\bar{h}_k^a(\mathbf{z}_k^a) = \min_{\boldsymbol{\gamma}_k^a}(\mathbf{z}_k^a)^T\boldsymbol{\gamma}_k^a - \log|\mathbf{\Sigma}_k^a|$, $\bar{h}_k^s(\mathbf{z}_k^s) = \min_{\boldsymbol{\gamma}_k^s}(\mathbf{z}_k^s)^T\boldsymbol{\gamma}_k^s - \log|\mathbf{\Sigma}_k^s|$, and $\bar{h}^c(\mathbf{z}^c) = \min_{\boldsymbol{\gamma}^c}(\mathbf{z}^c)^T\boldsymbol{\gamma}^c - \sum_{k=1}^{K}\log|\mathbf{\Sigma}_k^c|$ are the concave conjugate functions of $\log|\mathbf{\Sigma}_k^a|$, $\log|\mathbf{\Sigma}_k^s|$ and $\sum_{k=1}^{K}\log|\mathbf{\Sigma}_k^c|$, respectively. This leads to the following upper bounds for the original cost functions of SSM-1 in (18)

$$\begin{aligned}&L^{\text{pre}}(\boldsymbol{\gamma}^c, \{\boldsymbol{\gamma}_k^a\}, \mathbf{z}^c, \{\mathbf{z}_k^a\})\\ =&\mathbf{z}^c\boldsymbol{\gamma}^c - \bar{h}^c(\mathbf{z}^c) + \sum_{k=1}^{K}\alpha(\mathbf{z}_k^a)^T\boldsymbol{\gamma}_k^a - \alpha\bar{h}_k^a(\mathbf{z}_k^a) + \mathbf{y}_k^T(\mathbf{\Sigma}_k^{ac})^{-1}\mathbf{y}_k\\ \geq&L^{\text{pre}}(\boldsymbol{\gamma}^c, \{\boldsymbol{\gamma}_k^a\}).\end{aligned} \tag{20}$$

For the cost functions of SSM-2 in (19), we have the following upper bound

$$\begin{aligned}&L^{\text{cov}}(\boldsymbol{\gamma}^c, \{\boldsymbol{\gamma}_k^s\}, \mathbf{z}^c, \{\mathbf{z}_k^s\})\\ =&\mathbf{z}^c\boldsymbol{\gamma}^c - \bar{h}^c(\mathbf{z}^c) + \sum_{k=1}^{K}\beta(\mathbf{z}_k^s)^T\boldsymbol{\gamma}_k^s - \beta\bar{h}_k^s(\mathbf{z}_k^s) + \mathbf{y}_k^T(\mathbf{\Sigma}_k^{sc})^{-1}\mathbf{y}_k\\ \geq&L^{\text{cov}}(\boldsymbol{\gamma}^c, \{\boldsymbol{\gamma}_k^s\}).\end{aligned}$$

The previous upper bounds are tight when

$$\mathbf{z}_k^a = \nabla_{\boldsymbol{\gamma}_k^a}\log|\mathbf{\Sigma}_k^a| = \text{diag}\left[\mathbf{\Phi}_k^T\left(\mathbf{\Sigma}_k^a\right)^{-1}\mathbf{\Phi}_k\right], \tag{21}$$

$$\mathbf{z}_k^s = \nabla_{\boldsymbol{\gamma}_k^s}\log|\mathbf{\Sigma}_k^s| = \text{diag}\left[\mathbf{\Phi}_k^T\left(\mathbf{\Sigma}_k^s\right)^{-1}\mathbf{\Phi}_k\right], \tag{22}$$

$$\mathbf{z}^c = \nabla_{\boldsymbol{\gamma}^c}\sum_{k=1}^{K}\log|\mathbf{\Sigma}_k^c| = \sum_{k=1}^{K}\text{diag}\left[\mathbf{\Phi}_k^T\left(\mathbf{\Sigma}_k^c\right)^{-1}\mathbf{\Phi}_k\right]. \tag{23}$$

Given $\{\mathbf{z}_k^a\}$, $\{\mathbf{z}_k^s\}$ and $\mathbf{z}^c$, we obtain surrogate functions which are upper bounds of the original cost functions. Specifically, in order to update the hyper-parameters, one needs to solve the following minimization problem

$$\begin{aligned}&\arg\min_{\{\boldsymbol{\gamma}_k^s\},\boldsymbol{\gamma}^c} L_{\mathbf{z}}^{\text{pre}}(\boldsymbol{\gamma}^c, \{\boldsymbol{\gamma}_k^s\})\\ =&\arg\min_{\{\boldsymbol{\gamma}_k^s\},\boldsymbol{\gamma}^c} (\mathbf{z}^c)^T\boldsymbol{\gamma}^c + \sum_{k=1}^{K}\alpha(\mathbf{z}_k^a)^T\boldsymbol{\gamma}_k^a + \mathbf{y}_k^T(\mathbf{\Sigma}_k^{ac})^{-1}\mathbf{y}_k\end{aligned} \tag{24}$$

for the precision component model, and the following minimization problem

$$\begin{aligned}&\arg\min_{\{\boldsymbol{\gamma}_k^s\},\boldsymbol{\gamma}^c} L_{\mathbf{z}}^{\text{cov}}(\boldsymbol{\gamma}^c, \{\boldsymbol{\gamma}_k^s\})\\ =&\arg\min_{\{\boldsymbol{\gamma}_k^s\},\boldsymbol{\gamma}^c} (\mathbf{z}^c)^T\boldsymbol{\gamma}^c + \sum_{k=1}^{K}\beta(\mathbf{z}_k^s)^T\boldsymbol{\gamma}_k^s + \mathbf{y}_k^T(\mathbf{\Sigma}_k^{sc})^{-1}\mathbf{y}_k\end{aligned} \tag{25}$$

for the covariance component model. The optimization problems in (24) and (25), can be proved to be convex problems using Example 3.4 in [35]. Thus, many standard optimization procedures can be applied. In the following, we show that (24) and (25) can be minimized by solving weighted convex $\ell_1 + \ell_{1,2}$-regularized problems.

Algorithm 1 The ℓ_1 reweighting algorithm with the precision component model

```
Step 1: Initialize $\mathbf{z}^c = \mathbf{1}$ and $\mathbf{z}_k^a = \mathbf{1}$, $\forall k$;
Step 2: Solve the optimization problem (26) to update $\boldsymbol{\gamma}^c$ and $\boldsymbol{\gamma}_k^a$, $\forall k$;
Step 3: Compute the optimal $\mathbf{z}^c$ and $\mathbf{z}_k^a$ $\forall k$ using (23) and (21), respectively;
Step 4: Iterate steps 2 and 3 until convergence;
Step 5: Compute $\mathbf{x}_k = (\mathbf{\Gamma}^{c-1} + \mathbf{\Gamma}_k^{a-1})^{-1}\mathbf{\Phi}_k^T(\mathbf{\Sigma}_k^{ac})^{-1}\mathbf{y}_k$.
```

Lemma 1: Let $\mathbf{Z}_k^a$, $\mathbf{Z}_k^s$ and $\mathbf{Z}^c$ be diagonal matrices corresponding to $\mathbf{z}_k^a$, $\mathbf{z}_k^s$ and $\mathbf{z}^c$, respectively. The objective function (24) associated with the precision component model can be minimized by solving

$$\begin{aligned}\mathbf{X} = \arg\min_{\mathbf{X}} &\sum_{k=1}^{K}\|\mathbf{\Phi}_k\mathbf{x}_k - \mathbf{y}_k\|_2^2 + 2\alpha^{\frac{1}{2}}\nu\sum_{k=1}^{K}\|(\mathbf{Z}_k^a)^{\frac{1}{2}}\mathbf{x}_k\|_1\\ &+ 2\nu\|(\mathbf{Z}^c)^{\frac{1}{2}}\mathbf{X}\|_{1,2},\end{aligned} \tag{26}$$

and then setting ${\gamma^c}_i = ({z^c}_i)^{-1/2}\|\mathbf{x}_{i,\cdot}\|_2$ and $\gamma_{k\,i}^a = (\alpha z_{k\,i}^a)^{-1/2}|x_{ik}|$. The objective function (25) associated with the covariance component model can be minimized by solving

$$\begin{aligned}\{\mathbf{C}, \mathbf{S}\} = \arg\min_{\mathbf{C},\mathbf{S}} &\sum_{k=1}^{K}\|\mathbf{\Phi}_k(\mathbf{c}_k + \mathbf{s}_k) - \mathbf{y}_k\|_2^2\\ &+ 2\beta^{\frac{1}{2}}\nu\sum_{k=1}^{K}\|(\mathbf{Z}_k^s)^{\frac{1}{2}}\mathbf{s}_k\|_1 + 2\nu\|(\mathbf{Z}^c)^{\frac{1}{2}}\mathbf{C}\|_{1,2},\end{aligned} \tag{27}$$

and then setting ${\gamma^c}_i = ({z^c}_i)^{-1/2}\|\mathbf{c}_{i,\cdot}\|_2$ and $\gamma_{k\,i}^s = (\beta z_{k\,i}^s)^{-1/2}|s_{ik}|$.

Given (26) and (27), we derive the ℓ_1 reweighting algorithms with the precision component model and the covariance component model for simultaneous sparse approximation, that are described in Algorithm 1 and Algorithm 2, respectively. In comparison to the algorithms proposed in [14], [15] that solve a convex $\ell_1 + \ell_{1,2}$-regularized problem for promoting SSM-1 and a convex $\ell_1 + \ell_{1,\infty}$-regularized problem for promoting SSM-2, respectively, our ℓ_1 reweighting algorithms are required to solve a convex $\ell_1 + \ell_{1,2}$-regularized problem in each iteration. A more significant difference is that the proposed algorithms operate in the latent variable (hyper-parameter) space, which correspond to Type II estimation, while the algorithms in [14], [15] are Type I estimation, that are equivalent to applying MAP estimation using a sparsity-inducing prior from a Bayesian perspective.

The convergence analysis of the proposed ℓ_1 reweighting algorithms with the precision component model and the covariance component model is provided in the following Theorem,

Algorithm 2 The ℓ_1 reweighting algorithm with the covariance component model

```
Step 1: Initialize $\mathbf{z}^c = \mathbf{1}$ and $\mathbf{z}_k^s = \mathbf{1}$, $\forall k$;
Step 2: Solve the optimization problem (27) to update $\boldsymbol{\gamma}^c$ and
    $\boldsymbol{\gamma}_k^s$, $\forall k$;
Step 3: Compute the optimal $\mathbf{z}^c$ and $\mathbf{z}_k^s$ $\forall k$ using (23) and
    (22), respectively;
Step 4: Iterate steps 2 and 3 until convergence;
Step 5: Compute $\mathbf{x}_k = (\mathbf{\Gamma}_k^s + \mathbf{\Gamma}_k^c)\mathbf{\Phi}_k^T(\mathbf{\Sigma}_k^{sc})^{-1}\mathbf{y}_k$.
```

which demonstrates that the proposed iterative algorithms are guaranteed to converge to a stationary point from all initialization states. Proofs are given in Appendix C.

Theorem 2: Define $\boldsymbol{\theta}^{\text{pre}} = (\{\boldsymbol{\gamma}_k^a\}, \boldsymbol{\gamma}^c)$ and $\boldsymbol{\theta}^{\text{cov}} = (\{\boldsymbol{\gamma}_k^s\}, \boldsymbol{\gamma}^c)$. Let $\{\boldsymbol{\theta}_t^{\text{pre}}\}_{t=0}^{\infty}$ and $\{\boldsymbol{\theta}_t^{\text{cov}}\}_{t=0}^{\infty}$ be sequences of iterates generated by the proposed algorithms for SSM-1 and SSM-2, respectively. Then $\{\boldsymbol{\theta}_t^{\text{pre}}\}_{t=0}^{\infty}$ and $\{\boldsymbol{\theta}_t^{\text{cov}}\}_{t=0}^{\infty}$ are guaranteed to converge to stationary points of (18) and (19), respectively.

B. ℓ_2 Reweighting Schemes

Now, we consider different surrogate functions that majorize the original cost functions of the precision component model and the covariance component model, which leads to the ℓ_2 reweighting schemes. First we consider the following bound

$$\sum_{k=1}^{K} \mathbf{y}_k^T(\mathbf{\Sigma}_k^{ac})^{-1}\mathbf{y}_k \leq \sum_{k=1}^{K} \frac{1}{\nu}\|\mathbf{y}_k - \mathbf{\Phi}_k\mathbf{x}_k\|_2^2 + \sum_{i=1}^{m} \frac{x_{ik}^2}{{\gamma^c}_i} + \frac{x_{ik}^2}{\gamma_{ki}^a}, \tag{28}$$

for the precision component model, and for the covariance component model we have

$$\sum_{k=1}^{K} \mathbf{y}_k^T(\mathbf{\Sigma}_k^{sc})^{-1}\mathbf{y}_k \leq \sum_{k=1}^{K} \frac{1}{\nu}\|\mathbf{y}_k - \mathbf{\Phi}_k(\mathbf{c}_k + \mathbf{s}_k)\|_2^2 + \sum_{i=1}^{m} \frac{c_{ik}^2}{{\gamma^c}_i} + \frac{s_{ik}^2}{\gamma_{ki}^s}. \tag{29}$$

The equality in (28) holds if

$$\mathbf{x}_k = ({\mathbf{\Gamma}^c}^{-1} + {\mathbf{\Gamma}_k^a}^{-1})^{-1}\mathbf{\Phi}_k^T(\mathbf{\Sigma}_k^{ac})^{-1}\mathbf{y}_k \tag{30}$$

for each signal, while equality in (29) holds if

$$\mathbf{s}_k = \mathbf{\Gamma}_k^s\mathbf{\Phi}_k^T(\mathbf{\Sigma}_k^{sc})^{-1}\mathbf{y}_k \tag{31}$$

and

$$\mathbf{c}_k = \mathbf{\Gamma}^c\mathbf{\Phi}_k^T(\mathbf{\Sigma}_k^{sc})^{-1}\mathbf{y}_k \tag{32}$$

for each signal.

Then we consider upper bounds for the log-determinant terms of the cost functions. As the log-determinant terms are concave nondecreasing functions, we define the concave conjugate functions

$$\bar{g}_k^a(\mathbf{z}_k^a) = \min_{\boldsymbol{\gamma}_k^a} \sum_{i=1}^{m} \frac{z_{ki}^a}{\gamma_{ki}^a} - \log\left|{\mathbf{\Gamma}_k^a}^{-1} + \frac{2}{\nu}\mathbf{\Phi}_k^T\mathbf{\Phi}_k\right|,$$

$$\bar{g}_k^s(\mathbf{z}_k^s) = \min_{\boldsymbol{\gamma}_k^s} \sum_{i=1}^{m} \frac{z_{ki}^s}{\gamma_{ki}^s} - \log\left|{\mathbf{\Gamma}_k^s}^{-1} + \frac{2}{\nu}\mathbf{\Phi}_k^T\mathbf{\Phi}_k\right|,$$

$$\bar{g}^c(\mathbf{z}^c) = \min_{\boldsymbol{\gamma}^c} \sum_{i=1}^{m} \frac{{z^c}_i}{{\gamma^c}_i} - \sum_{k=1}^{K}\log\left|{\mathbf{\Gamma}^c}^{-1} + \frac{2}{\nu}\mathbf{\Phi}_k^T\mathbf{\Phi}_k\right|.$$

According to the duality relationship of concave conjugate functions, we have the following upper bounds:

$$\log\left|{\mathbf{\Gamma}_k^a}^{-1} + \frac{2}{\nu}\mathbf{\Phi}_k^T\mathbf{\Phi}_k\right| = \min_{\mathbf{z}_k^a} \sum_{i=1}^{m} \frac{z_{ki}^a}{\gamma_{ki}^a} - \bar{g}_k^a(\mathbf{z}_k^a), \tag{33}$$

$$\log\left|{\mathbf{\Gamma}_k^s}^{-1} + \frac{2}{\nu}\mathbf{\Phi}_k^T\mathbf{\Phi}_k\right| = \min_{\mathbf{z}_k^s} \sum_{i=1}^{m} \frac{z_{ki}^s}{\gamma_{ki}^s} - \bar{g}_k^s(\mathbf{z}_k^s), \tag{34}$$

$$\sum_{k=1}^{K}\log\left|{\mathbf{\Gamma}^c}^{-1} + \frac{2}{\nu}\mathbf{\Phi}_k^T\mathbf{\Phi}_k\right| = \min_{\mathbf{z}^c} \sum_{i=1}^{m} \frac{{z^c}_i}{{\gamma^c}_i} - \bar{g}^c(\mathbf{z}^c), \tag{35}$$

where the bounds are tight when

$$\mathbf{z}_k^a = \operatorname{diag}\left[\left({\mathbf{\Gamma}_k^a}^{-1} + \frac{2}{\nu}\mathbf{\Phi}_k^T\mathbf{\Phi}_k\right)^{-1}\right], \tag{36}$$

$$\mathbf{z}_k^s = \operatorname{diag}\left[\left({\mathbf{\Gamma}_k^s}^{-1} + \frac{2}{\nu}\mathbf{\Phi}_k^T\mathbf{\Phi}_k\right)^{-1}\right], \tag{37}$$

$$\mathbf{z}^c = \sum_{k=1}^{K}\operatorname{diag}\left[\left({\mathbf{\Gamma}^c}^{-1} + \frac{2}{\nu}\mathbf{\Phi}_k^T\mathbf{\Phi}_k\right)^{-1}\right]. \tag{38}$$

Inserting the upper bounds, (28), (33) and (35), into the cost function (18) and omitting irrelevant terms, we arrive at the following approximation

$$\min_{\boldsymbol{\gamma}^c, \{\boldsymbol{\gamma}_k^a\}} \sum_{i=1}^{m} \frac{x_{ik}^2}{{\gamma^c}_i} + \frac{x_{ik}^2}{\gamma_{ki}^a} + \frac{\alpha z_{ki}^a}{\gamma_{ki}^a} + \frac{{z^c}_i}{{\gamma^c}_i} + K\log|\mathbf{\Gamma}^c| + \alpha\sum_{k=1}^{K}\log|\mathbf{\Gamma}_k^a|,$$

and its solutions are

$$\gamma_{ki}^a = z_{ki}^a + \frac{x_{ik}^2}{\alpha} \tag{39}$$

and

$${\gamma^c}_i = \frac{{z^c}_i + \sum_k x_{ik}^2}{K}. \tag{40}$$

Inserting the upper bounds, (29), (34) and (35), into the cost function (19), we have

$$\min_{\boldsymbol{\gamma}^c, \{\boldsymbol{\gamma}_k^s\}} \sum_{i=1}^{m} \frac{c_{ik}^2}{{\gamma^c}_i} + \frac{s_{ik}^2}{\gamma_{ki}^s} + \frac{\beta z_{ki}^s}{\gamma_{ki}^s} + \frac{{z^c}_i}{{\gamma^c}_i} + K\log|\mathbf{\Gamma}^c| + \beta\sum_{k=1}^{K}\log|\mathbf{\Gamma}_k^s|,$$

where the solutions are

$$\gamma_{ki}^s = z_{ki}^s + \frac{s_{ik}^2}{\beta} \tag{41}$$

and

$${\gamma^c}_i = \frac{{z^c}_i + \sum_k c_{ik}^2}{K}. \tag{42}$$

Algorithm 3 The ℓ_2 reweighting algorithm with the precision component model

```
Step 1: Initialize $\boldsymbol{\gamma}^c = \mathbf{1}$ and $\boldsymbol{\gamma}_k^a = \mathbf{1}$, $\forall k$;
Step 2: Compute the optimal $\mathbf{x}_k$ $\forall k$ using (30);
Step 3: Compute the optimal $\mathbf{z}^c$ and $\mathbf{z}_k^a$ $\forall k$ using (38) and
    (36), respectively;
Step 4: Compute $\boldsymbol{\gamma}^c$ and $\boldsymbol{\gamma}_k^a$ $\forall k$ using (40) and (39), respec-
    tively;
Step 5: Iterate steps 2, 3 and 4 until convergence.
```

Algorithm 4 The ℓ_2 reweighting algorithm with the covariance component model

```
Step 1: Initialize $\boldsymbol{\gamma}^c = \mathbf{1}$ and $\boldsymbol{\gamma}_k^s = \mathbf{1}$, $\forall k$;
Step 2: Compute the optimal $\mathbf{s}_k$ and $\mathbf{c}_k$ $\forall k$ using (31) and
    (32), respectively;
Step 3: Compute the optimal $\mathbf{z}^c$ and $\mathbf{z}_k^s$ $\forall k$ using (38) and
    (37), respectively;
Step 4: Compute $\boldsymbol{\gamma}^c$ and $\boldsymbol{\gamma}_k^s$ $\forall k$ using (42) and (41), respec-
    tively;
Step 5: Iterate steps 2, 3 and 4 until convergence.
```

Therefore, by repeatedly minimizing and updating the majorization functions, we obtain the ℓ_2 reweighting algorithms for SSM-1 and SSM-2, that are described in Algorithm 3 and Algorithm 4, respectively. Although each iteration of the proposed ℓ_2 reweighting algorithms is guaranteed to reduce or leave the cost function (18) and (19) unchanged, it is insufficient to guarantee formal convergence to a stationary point. The convergence analysis for the proposed ℓ_2 reweighting algorithms is very difficult, as it requires, for example, that the additional conditions of the Zangwills Global Convergence Theorem hold [36]. However in practice, we have not encountered any convergence issues. In addition, it should be noted that the ℓ_1 reweighting algorithms developed in the previous subsection, which although have provable convergence, need to iteratively solve $\ell_1 + \ell_{1,2}$-regularized optimization problems in each iteration that do not have close form solutions, while the proposed ℓ_2 reweighting algorithms has a closed form solution to be computed in every step.

V. Decentralized Algorithms for Simultaneous Sparse Approximation With SSMs

In this section, we develop decentralized algorithms for simultaneous sparse approximation with SSMs. A significant advantage of the proposed decentralized algorithms is that the learning process is carried out in a decentralized way without sharing the original data sets of different signals, and thus it is applicable for privacy-sensitive applications. The proposed schemes are derived from the ℓ_2 reweighting algorithms described previously by casting the update steps as a set of decentralized problems with consensus constraints.

For a decentralized scenario, we consider a network with $\tilde{T}$ nodes modeled by a undirectional graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, where $\mathcal{V} = \{1, \ldots, \tilde{T}\}$ is the set of nodes and $\mathcal{E} \subset \mathcal{V} \times \mathcal{V}$ is the set of edges that describe the communication links among the nodes. Each node is able to process locally stored data and exchange messages with its neighbors. We assume there are K_t observation vectors and K_t sensing matrices stored at node t ($t \in \mathcal{V}$), which are denoted by $\mathbf{y}_{k_t} \in \mathbb{R}^{n_{k_t}}$ and $\boldsymbol{\Phi}_{k_t} \in \mathbb{R}^{n_{k_t} \times m}$, respectively, where $k_t \in \mathcal{W}_t$, $\mathcal{W}_t$ denotes the task index set for node t, $|\mathcal{W}_t| = K_t$ and $\mathcal{W}_{t'} \cap \mathcal{W}_{t''} = \emptyset$ if $t' \neq t''$. For all nodes, there are $\sum_{t=1}^{\tilde{T}} K_t = K$ signals in total. The goal is to recover the unknown sparse vectors $\mathbf{x}_{k_t} \in \mathbb{R}^m$ for all nodes. Owning to the correlation between the data sets of different nodes, which is modeled via SSMs in this paper, simultaneous sparse approximation is expected to provide improved performance in comparison to signal reconstruction independently at each node.

Now let us revisit the proposed centralized ℓ_2 reweighting algorithms. Given the common parameters $\mathbf{z}^c$ and $\boldsymbol{\gamma}^c$, nodes can work in parallel to execute (30), (31), (32), (36), (37), (39) and (41). Now, instead of computing $\mathbf{z}^c$ from (38) that requires inter-node communication to exchange information, each node computes

$$\mathbf{z}_k^c = \operatorname{diag}\left[\left(\boldsymbol{\Gamma}^{c-1} + \frac{2}{\nu}\boldsymbol{\Phi}_k^T\boldsymbol{\Phi}_k\right)^{-1}\right]$$

locally, where $\sum_{k=1}^{K} \mathbf{z}_k^c = \mathbf{z}^c$ according to (38). With this revision, the update rules of the common hyper-parameter $\boldsymbol{\gamma}^c$ can be expressed as

$$\boldsymbol{\gamma}^c = \frac{1}{K}\sum_k \mathbf{q}_k, \tag{43}$$

where $q_{k_i} = z_{k_i}^c + x_{ik}^2$ in the precision component model according to (40), and $q_{k_i} = z_{k_i}^c + c_{ik}^2$ in the covariance component model according to (42). Therefore, we only need to decentralize the computation of the common parameters $\boldsymbol{\gamma}^c$ in each iteration of the proposed ℓ_2 reweighting algorithms.

According to the expression in (43), $\boldsymbol{\gamma}^c$ is updated as the average of $\mathbf{q}_k$ ($k = 1, \ldots, K$), which can be obtained by solving the following average consensus problem

$$\min_{\boldsymbol{\gamma}^c} \sum_{t=1}^{\tilde{T}} \sum_{k \in \mathcal{W}_t} \left\|\boldsymbol{\gamma}^c - \mathbf{q}_k\right\|_2^2. \tag{44}$$

This optimization problem can be further reformulated into

$$\begin{aligned} \min_{\substack{(\boldsymbol{\gamma}^c)^1, \ldots, \\ (\boldsymbol{\gamma}^c)^{\tilde{T}}}} \; & \sum_{t=1}^{\tilde{T}} \sum_{k \in \mathcal{W}_t} \left\|(\boldsymbol{\gamma}^c)^t - \mathbf{q}_k\right\|_2^2 \\ \text{s.t.} \; & (\boldsymbol{\gamma}^c)^t = (\boldsymbol{\gamma}^c)^{j_t}, \; \forall j_t \in \mathcal{N}_t, \; \forall t \in \{1, \ldots, \tilde{T}\}, \end{aligned} \tag{45}$$

where $(\boldsymbol{\gamma}^c)^t$ denotes the local estimate of $\boldsymbol{\gamma}^c = \frac{1}{K}\sum_{k=1}^{K} \mathbf{q}_k$ at node t, and $\mathcal{N}_t$ denotes the neighbors of node t. Two nodes are called neighbors if they can communicate with each other to exchange information. Optimization problems (44) and (45) are equivalent if their neighborhood relationship can lead to a connected graph.

We employ the alternating direction method of multipliers (ADMM) [37] to solve (45) in a decentralized manner. According to [38], the simplified ADMM form of (45) consists of the following iterations

$$\begin{aligned}\mathbf{p}_z^{t,\text{new}} &= \mathbf{p}_z^{t,\text{old}} + \rho \sum_{j_k \in \mathcal{N}_t} \left((\boldsymbol{\gamma}^c)^{t,\text{old}} - (\boldsymbol{\gamma}^c)^{j_t,\text{old}} \right), \\ (\boldsymbol{\gamma}^c)^{t,\text{new}} &= \frac{1}{2K_t + 2\rho|\mathcal{N}_t|} \Bigg(2 \sum_{k \in \mathcal{W}_t} \mathbf{q}_k \\ &\quad - \mathbf{p}_z^{t,\text{new}} + \rho \sum_{j_k \in \mathcal{N}_k} \left((\boldsymbol{\gamma}^c)^{t,\text{old}} + (\boldsymbol{\gamma}^c)^{j_t,\text{old}} \right) \Bigg)\end{aligned} \tag{46}$$

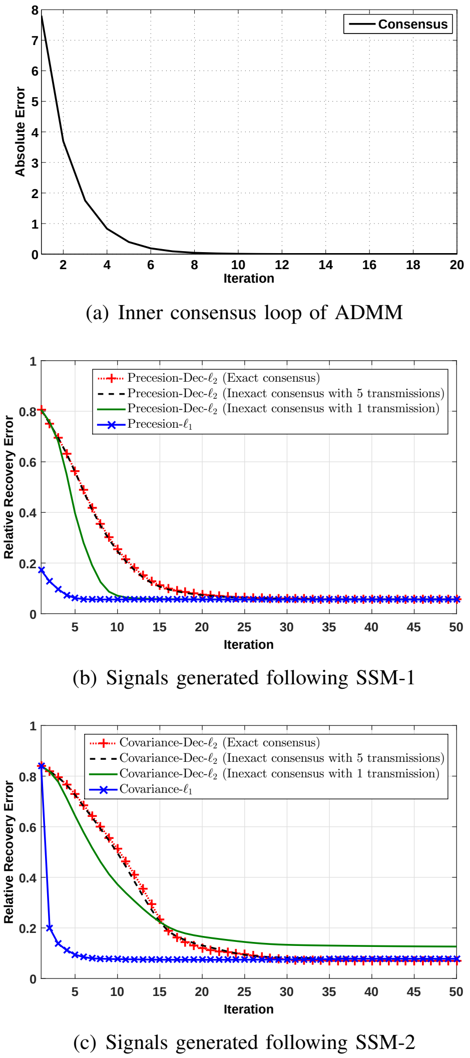
for $\forall t \in \{1, \ldots, \tilde{T}\}$, where $\rho > 0$ is a preselected penalty coefficient. Note that nodes can execute (46) in parallel with the information concerning $(\boldsymbol{\gamma}^c)^{j_t}$ passed from their neighbors. In addition, it has been proved that iteratively executing the steps in (46) will converge to the global solution $\boldsymbol{\gamma}^c$ for any $\rho > 0$ [38].

VI. Numerical Experiments

In this section we present numerical experiment results to compare the recovery performance of various algorithms for simultaneous sparse approximation with SSMs. The following algorithms are considered in our comparison:

- Least absolute shrinkage and selection operator (Lasso): solving a convex optimization problem with an ℓ_1 regularizer to promote element-sparse solutions;
- $\ell_{1,2}$: solving a convex optimization problem with an $\ell_{1,2}$ regularizer to promote row-sparsity;
- $\ell_1/\ell_{1,2}$: solving a convex optimization problem with an ℓ_1 regularizer and an $\ell_{1,2}$ regularizer to promote SSM-1 [15];
- $\ell_1 + \ell_{1,\infty}$: solving a convex optimization problem with an $\ell_{1,2}$ regularizer and an $\ell_{1,\infty}$ regularizer to promote SSM-2 [14];
- SBL: a nonconvex Bayesian learning algorithm for sparsity minimization [25];
- MSBL: a nonconvex Bayesian learning algorithm for row-sparsity minimization [11];
- Proposed algorithms based on the precision model for SSM-1;
- Proposed algorithms based on the covariance model for SSM-2.

In the experiments, we consider random CS measurement vectors $\mathbf{y}_k = \boldsymbol{\Phi}\mathbf{x}_k + \mathbf{e}_k$ $(k = 1, \ldots, K)$, where $\mathbf{e}_k$ is a zero-mean Gaussian noise vector with variance adjusted to have a desired value of the signal to noise ratio (SNR). We use the same sensing matrix $\boldsymbol{\Phi}$ for all signals, where the entries of the sensing matrix are generated independently from $\mathcal{N}(0,1)$ and then normalized for each column. The sparse signal representations $\mathbf{X}$ are generated following SSM-1 or SSM-2. Specifically, for SSM-1, we randomly select d nonzero rows for the sparse signal representation matrix $\mathbf{X}$ with all the nonzero entries drawn independently from $\mathcal{N}(0,1)$. Then r nonzeros of each column are randomly chosen and forced to be zeros. On the other hand, for SSM-2, we generate an element-sparse matrix $\mathbf{S}$ with r nonzero elements for each signal drawn independently from $\mathcal{N}(0,1)$ and a random row-sparse matrix $\mathbf{C}$ with $d - r$ nonzero rows with entries drawn independently from $\mathcal{N}(0,1)$. Then the source matrix is obtained by $\mathbf{X} = \mathbf{C} + \mathbf{S}$. The recovery performance is evaluated via relative recovery error defined by $\frac{\|\hat{\mathbf{X}} - \mathbf{X}\|_F}{\|\mathbf{X}\|_F}$, and averaged over 100 trials.



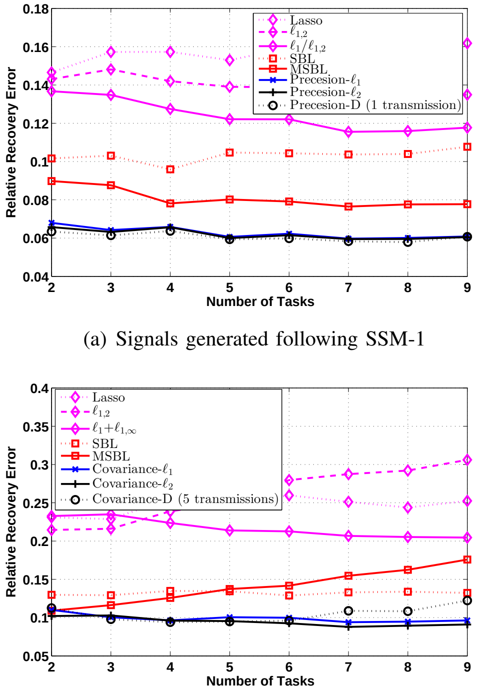
(a) Inner consensus loop of ADMM

(b) Signals generated following SSM-1

(c) Signals generated following SSM-2

Fig. 2. Convergence rates of the proposed algorithms for a single instance.

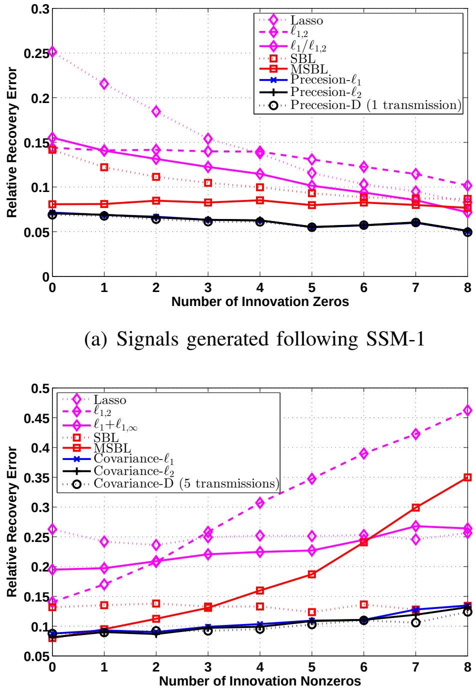
If we do not point out specifically in the experiments, the baseline settings in the simulation are given as: the number of measurements $n = 35$, the ambient dimension $m = 100$, the number of signals $K = 5$, an SNR of 20 dB, the row sparsity $d = 10$, the number of innovation zeros of $\mathbf{x}_k$ in SSM-1 is $r = 3$, and the innovation nonzeros of $\mathbf{s}_k$ in SSM-2 is $r = 3$. We set $\alpha = 1$ for the proposed algorithms with the precision component model, and $\beta = 2$ for the proposed algorithms with the covariance component model. The noise variance ν is given and fixed in all the algorithms, although some learning rules can be used to estimate ν [24], [25]. To evaluate the performance of the proposed decentralized algorithms, we consider a network generated as a $\lceil \frac{K}{2} \rceil$-connected Harary graph with K nodes, where each node is only available to communicate with $\lceil \frac{K}{2} \rceil$ adjacent neighbors to exchange information. The parameter ρ of the ADMM step is set to 0.3 in our simulations.



(a) Signals generated following SSM-1

(b) Signals generated following SSM-2

Fig. 3. Comparison of the reconstruction accuracy with different number of tasks.



(a) Signals generated following SSM-1

(b) Signals generated following SSM-2

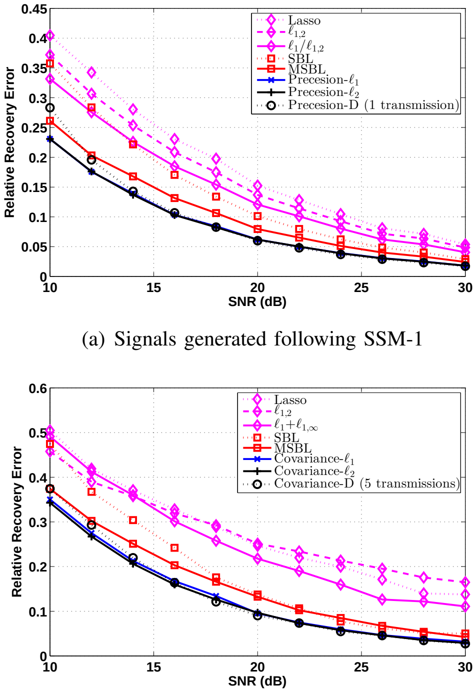
Fig. 4. Comparison of reconstruction accuracy with different innovation levels.

A. Convergence Performance

Firstly, we study the convergence performance of the proposed algorithms, and investigate the possibility to reduce the communication burden of the proposed algorithms by using inexact consensus ADMM, i.e., all the nodes have the same copy of information in the end. Firstly, Fig. 2 (a) shows the convergence rate of the inner consensus loop of the proposed decentralized algorithms, where about 10 rounds of message exchanges are required to achieve convergence in our settings. As shown in Fig. 2 (b), we note that exact consensus is not necessary for the proposed decentralized algorithm for SSM-1, and inexact consensus with a single message exchange per iteration, does not degrade the reconstruction performance. Interestingly, the convergence rate of the proposed decentralized algorithm with the precision component model converges more quickly using inexact consensus than exact consensus in our settings. This phenomenon has been observed in different instances of our simulations although we only show a single instance here. For SSM-2, as shown in Fig. 2 (c), inexact consensus with a single message exchange per iteration is insufficient and degrades the reconstruction accuracy of the proposed algorithm, and more transmissions are required to provide accuracy results (even though exact consensus is still not required). The distinctive convergence characteristics of the proposed algorithms in the two SSMs is caused by the different mechanisms in the component models. Specifically, $\boldsymbol{\gamma}_k^a$ encapsulates all the support information of signal k for the precision component model, but for the covariance component model, $\boldsymbol{\gamma}_k^s$ encapsulates only a part of the support information of signal k and so accurate consensus for $\boldsymbol{\gamma}^c$ is important.

B. Recovery Performance With Different Number of Signals

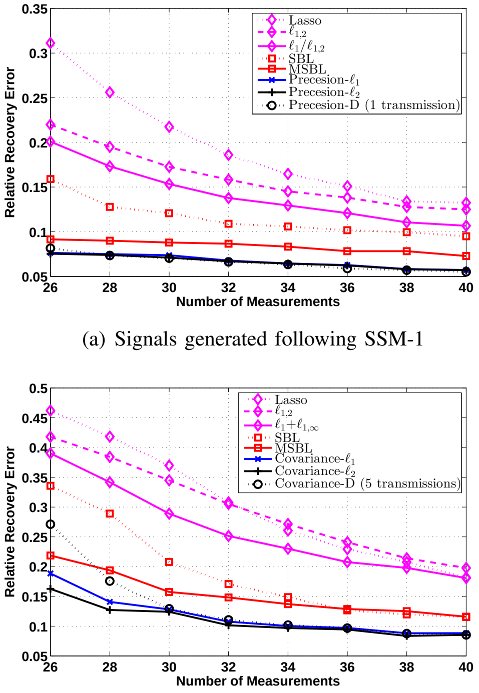
In this experiment we investigate how the proposed algorithms benefit from simultaneous sparse approximation with SSMs for different numbers of signals. As shown in Fig. 3 (a), for SSM-1, algorithms including $\ell_{1,2}$, $\ell_1/\ell_{1,2}$, MSBL and the proposed algorithms with the precision model, that exploit simultaneous sparse approximation, all have improved reconstruction accuracy in comparison to that of the signal-independent reconstruction algorithms, i.e., Lasso and SBL. In addition, $\ell_{1,2}$ and MSBL exploit a general row-sparse model without considering the sparse structure in each nonsparse row, and thus have degraded performance in comparison to the related $\ell_1/\ell_{1,2}$ and the proposed algorithms. On the other hand, for SSM-2, as shown in Fig. 3 (b), the reconstruction accuracy of $\ell_{1,2}$ and MSBL tend to be worse with a growing number of signals owing to the limitation of using the row-sparse model to capture SSM-2, while $\ell_1 + \ell_{1,\infty}$ and the proposed algorithms with the covariance component model are able to benefit from simultaneous sparse approximation. For both SSMs, our proposed algorithms outperform all the others, and proposed ℓ_2 reweighting algorithms have performance close to the ℓ_1 reweighting algorithms. The decentralized algorithms with inexact consensus also achieve superior performance in comparison to other competitors for both SSMs. Here, we consider a single message exchange and five message exchanges per iteration for the precision component model and the covariance component model, respectively, as accurate consensus is more important to the covariance component model, as depicted by Fig. 2.



(a) Signals generated following SSM-1

(b) Signals generated following SSM-2

Fig. 5. Comparison of reconstruction accuracy with different levels of noise.



(a) Signals generated following SSM-1

(b) Signals generated following SSM-2

Fig. 6. Comparison of reconstruction accuracy with different number of measurements.

C. Recovery Performance With Different Innovation Levels

In this experiment we study how the algorithms perform with different innovation levels in each SSM. For SSM-1, we vary the number of innovation zeros in the row-sparse $\mathbf{X}$, and vary the number of nonzeros in each column of $\mathbf{S}$ in SSM-2. It is observed in Fig. 4 (a) for SSM-1 that the reconstruction error of all the algorithms tend to decrease with a growing number of innovation zeros, and that the proposed algorithms achieve the best recovery performance. The performance comparison of various algorithms for SSM-2 is shown in Fig. 4 (b). Here, since $\mathbf{C}$ is generated as a $d-r$ row-sparse matrix in SSM-2, the sparsity of each sparse representation vector $\mathbf{x}_k$ is a fixed value $d = 10$ even as the sparsity of $\mathbf{s}_k$ varies. However, with a growing number of nonzeros in each column of $\mathbf{S}$, the sparse representation matrix $\mathbf{X}$ has more nonzero rows, which leads to a significant performance degradation of $\ell_{1,2}$ and MSBL as shown in Fig. 4 (b). For SSM-2, the proposed algorithms perform as well as MSBL when $r = 0$, which means the sparse representation matrix $\mathbf{X}$ is exactly row-sparse, and performs as well as SBL when r becomes large, which means $\mathbf{X}$ becomes element-sparse. This observation indicates that our proposed algorithms bridge the gap between the element-sparse model and the row-sparse model, and are very advantageous in the case when the sparse representation vectors in simultaneous sparse approximation have a "dirty"-sparse structure.

D. Recovery Performance With Different Levels of Noise and Different Numbers of Measurements

In the previous experiments, we have shown that the proposed algorithms significantly outperform all the algorithms chosen for comparison in a moderate SNR of 20dB and for a fixed number of measurements $n = 35$. Now in Fig. 5, we provide experimental results to show how the proposed algorithms perform in different noise levels and in Fig. 6 for different numbers of measurements. Again, it is observed the the proposed algorithms have the highest reconstruction accuracy among all the algorithms for both SSMs. Both the proposed ℓ_2 reweighting algorithms and the ℓ_1 reweighting algorithms have similar reconstruction accuracy performance. In all the experiments, decentralizing the ℓ_2 reweighting algorithms through carrying out inexact consensus does not degrade performance at high SNR, while performance degrades only slightly at low SNR in Fig. 5.

E. Comparison of Computing Time

We now evaluate the computing time of the proposed methods. Our simulations are performed in a MATLAB R2012b environment on a system with a quad-core 3.4 GHz CPU and 32 GB RAM, running under the Microsoft Windows 7 operating system. As shown in Table I, both the proposed ℓ_2 reweighting algorithms and the ℓ_1 reweighting algorithms take more computing time as the number of tasks increases, and that the proposed ℓ_2 reweighting algorithms take less computing time than the ℓ_1 reweighting algorithms for both SSMs. In comparison to the $\ell_1/\ell_{1,2}$ method and the $\ell_1 + \ell_{1,\infty}$ method that target SSM-1 and SSM-2, respectively, the proposed ℓ_2 reweighting algorithms reduce the computing time by 75% and 99% in the case of 20 tasks for SSM-1 and SSM-2, respectively.

TABLE I
COMPARISON OF COMPUTING TIME

(a) Computing Time for SSM-1 (in Seconds)

Number of tasks	Lasso	$\ell_{1,2}$	$\ell_1/\ell_{1,2}$	SBL	MSBL	Precesion-ℓ_1	Precesion-ℓ_2
2	0.45	0.41	1.24	0.40	0.21	50.17	0.12
5	0.87	0.88	2.94	1.02	0.19	51.55	0.40
20	2.68	3.04	5.41	3.71	0.19	69.40	1.35

(b) Computing Time for SSM-2 (in Seconds)

Number of tasks	Lasso	$\ell_{1,2}$	$\ell_1+\ell_{1,\infty}$	SBL	MSBL	Covariance-ℓ_1	Covariance-ℓ_2
2	1.15	0.11	2.10	0.39	0.21	9.28	0.13
5	1.55	0.28	40.94	0.92	0.19	18.62	0.34
20	3.40	0.42	142.49	3.54	0.19	92.12	1.11

VII. Experiments: Using SSMs for Face Recognition

We now illustrate the benefit of using SSMs for face recognition. Here, we use the AR database that contains more than 4000 images of 126 people. There are 26 facial images for each subject, which involve different illumination scenarios, different expressions and different facial 'disguise' (sunglasses and scarves). The size of each image is 154×120 pixels. Following the standard evaluation procedure, we use a subset of the database consisting of 2600 images from 50 male and female subjects respectively. For each subject, we randomly select 20 facial images for training and the other 6 for testing. In the following experiment, each facial image is projected onto a 540 dimension feature vector with a randomly generated matrix from a zero-mean normal distribution. We consider the standard dictionary learning approaches including the sparse representation-based classification (SRC) [39] and the incoherent class-specific dictionary (ICSD) [40], that are widely used for face recognition, to learn a dictionaries with 500 atoms. We also use the class-specific residue for face recognition as in [39]. The regularisation parameters, i.e., ν, α and β, are determined by cross validation on the training dataset. The value used in this experiment are ($\nu = 10^{-3}$, $\alpha = 0.1$) for the precision component model, and ($\nu = 10^{-3}$, $\beta = 10$) for the covariance component model.

While the sparse coding step in LASSO is conducted in parallel for different testing images, our proposed approaches and the $\ell_{1,2}$ norm minimization approach consider the images of the same subject as a group for the testing phase. The prior knowledge of grouping information is available in some scenarios, e.g., a sequence of facial images of the same subject is extracted from a video for face recognition. For classification using the SSM-2, we only use those representation supports shared across tasks in order to remove the innovations in various images of the same subject. The experimental results are summarized in Table II. We implement four convex formulations, i.e., LASSO, $\ell_{1,2}$ norm minimization, and $\ell_1 + \ell_{1,\infty}$ norm minimization and $\ell_1/\ell_{1,2}$ norm minimization. However, $\ell_1 + \ell_{1,\infty}$ does not have better performance than the other three convex methods. Our approaches outperform Lasso and the $\ell_{1,2}$ norm minimization approach by at least 1.5 and 1 percentage points in terms of recognition error, respectively.

VIII. Conclusion

While SBL is successful for single sparse approximation problems, how to extend it to estimate multiple sparse approximations that follow SSMs is not obvious. The dual-space view of the convex methods for SSMs allow us to understand x-space dirty structures from the perspective of γ-space approaches, which unfolds the intrinsic precision component vs. covariance component models in simultaneous sparse approximation with SSMs. Superior performance of the proposed approaches including centralized methods and decentralized methods, have been demonstrated by simulation results. We envisage that the fundamental mechanism in the precision component vs. covariance component models could be suitable for a broad range of data models involving either simultaneous structures or additive/dirty structures, although doing so is out of the scope this paper.

Appendix A
Proof of the Theorem 1

The following proofs are based on the results of Theorem 4 in [31], which considers a single sparse approximation problem with SBL. However, for simultaneous sparse approximation with SSMs, some modifications are required.

According to the formulations of the cost functions in (18) and (19), the minimum occurs when

$$\exists k, \left|\frac{\nu}{2}\mathbf{I} + \mathbf{\Phi}_k \mathbf{\Gamma}^a_k \mathbf{\Phi}_k^T\right| = 0 \text{ or } \left|\frac{\nu}{2}\mathbf{I} + \mathbf{\Phi}_k \mathbf{\Gamma}^c \mathbf{\Phi}_k^T\right| = 0,$$
$$\sum_{k=1}^{K} \mathbf{y}_k^T (\nu \mathbf{I} + \mathbf{\Phi}_k \mathbf{\Gamma}_k \mathbf{\Phi}_k^T)^{-1} \mathbf{y}_k \leq \rho_1,$$

for SSM-1, and for SSM-2

$$\exists k, \left|\frac{\nu}{2}\mathbf{I} + \mathbf{\Phi}_k \mathbf{\Gamma}^s_k \mathbf{\Phi}_k^T\right| = 0 \text{ or } \left|\frac{\nu}{2}\mathbf{I} + \mathbf{\Phi}_k \mathbf{\Gamma}^c \mathbf{\Phi}_k^T\right| = 0,$$
$$\sum_{k=1}^{K} \mathbf{y}_k^T (\nu \mathbf{I} + \mathbf{\Phi}_k \mathbf{\Gamma}_k \mathbf{\Phi}_k^T)^{-1} \mathbf{y}_k \leq \rho_2,$$

where $\rho_1 > 0$ and $\rho_2 > 0$ denote some finite bounds. Now, all that is required is to prove that the solutions, which lead to accurate reconstruction, satisfy these conditions.

According to (4), the support of $\hat{\mathbf{x}}_\mathbf{k}$ is the same as the support associated with $\hat{\gamma}_k$ when $\nu = 0$. For the precision component model in (9), we let $\hat{\gamma}^c$ be a vector with all elements being a unit value, then $(\hat{\mathbf{\Gamma}}^a_k)^{-1} = (\hat{\mathbf{\Gamma}}_k)^{-1} - (\hat{\mathbf{\Gamma}}^c)^{-1}$, which suggests that the support of $\hat{\mathbf{x}}_\mathbf{k}$ is the same as the support of $\hat{\gamma}^a_k$. Since $\|\hat{\mathbf{x}}_k\|_0 < n_k$, we have $\left|\mathbf{\Phi}_k \mathbf{\Gamma}^a_k \mathbf{\Phi}_k^T\right| = 0$. For the covariance component model in (14), it is known that the support of $\hat{\mathbf{x}}_\mathbf{k}$ is the union of the support of $\hat{\gamma}^a_k$ and the support of $\hat{\gamma}^c$. Therefore, both $\left|\mathbf{\Phi}_k \mathbf{\Gamma}^c \mathbf{\Phi}_k^T\right|$ and $\left|\mathbf{\Phi}_k \mathbf{\Gamma}^s_k \mathbf{\Phi}_k^T\right|$ are equal to zero.

TABLE II
FACE RECOGNITION ERROR

Method	Precision-ℓ_1	Covariance-ℓ_1	SBL	MSBL	Lasso	$\ell_{1,2}$	$\ell_1/\ell_{1,2}$
SRC	**7.67%**	**8.33%**	15.33%	9.33%	15.83%	10.17%	15.67%
ICSD	**4.5%**	**5.03%**	13.17%	5.4%	13.50%	6.17%	13.5%

In addition, we have

$$\begin{aligned}&\lim_{\nu\to 0}\mathbf{y}_k^T(\nu\mathbf{I}+\mathbf{\Phi}_k\hat{\mathbf{\Gamma}}_k\mathbf{\Phi}_k^T)^{-1}\mathbf{y}_k\\ =&\lim_{\nu\to 0}\hat{\mathbf{x}}_k^T\hat{\mathbf{\Gamma}}_k^{-1/2}\hat{\mathbf{\Gamma}}_k^{1/2}\mathbf{\Phi}^T(\nu\mathbf{I}+\mathbf{\Phi}_k\hat{\mathbf{\Gamma}}_k\mathbf{\Phi}_k^T)^{-1}\mathbf{\Phi}\hat{\mathbf{\Gamma}}_k^{1/2}\hat{\mathbf{\Gamma}}_k^{-1/2}\hat{\mathbf{x}}_k\\ =&\hat{\mathbf{x}}_k^T\hat{\mathbf{\Gamma}}_k^{-1}\hat{\mathbf{z}}_k\le\frac{1}{\delta_k}\|\hat{\mathbf{x}}_k\|_2^2,\end{aligned}$$

where $\delta_k > 0$ is the minimum nonzero entry of $\hat{\mathbf{\Gamma}}_k$. Now we complete the proof.

APPENDIX B
PROOF FOR LEMMA 1

First we consider the following bound

$$\begin{aligned}&\sum_{k=1}^{K}\mathbf{y}_k^T(\mathbf{\Sigma}_k^{ac})^{-1}\mathbf{y}_k\\ &=\arg\min_{\mathbf{X}}\sum_{k=1}^{K}\frac{1}{\nu}\|\mathbf{y}_k-\mathbf{\Phi}_k\mathbf{x}_k\|_2^2+\sum_{i=1}^{m}\frac{x_{ik}^2}{{\gamma^c}_i}+\frac{x_{ik}^2}{\gamma^a_{k\,i}},\end{aligned}\tag{47}$$

which leads to an upper-bounding surrogate function of (24) as

$$\begin{aligned}L_{\mathbf{z}}^{\text{pre}}(\boldsymbol{\gamma}^c,\{\boldsymbol{\gamma}_k^a\})\le&(\mathbf{z}^c)^T\boldsymbol{\gamma}^c+\sum_{k=1}^{K}\alpha(\mathbf{z}_k^a)^T\boldsymbol{\gamma}_k^a+\frac{1}{\nu}\|\mathbf{y}_k-\mathbf{\Phi}_k\mathbf{x}_k\|_2^2\\ &+\sum_{i=1}^{m}\frac{x_{ik}^2}{{\gamma^c}_i}+\frac{x_{ik}^2}{\gamma^a_{k\,i}}\\ =&\tilde{L}_{\mathbf{z}}^{\text{pre}}(\boldsymbol{\gamma}^c,\{\boldsymbol{\gamma}_k^a\},\mathbf{X}),\end{aligned}\tag{48}$$

where the equality holds when $\mathbf{X}$ is the solution of (47). The function $\tilde{L}_{\mathbf{z}}^{\text{pre}}(\boldsymbol{\gamma}^c,\{\boldsymbol{\gamma}_k^a\},\mathbf{X})$ in (48) is jointly convex in $\boldsymbol{\gamma}^c$, $\{\boldsymbol{\gamma}_k^a\}$, and $\mathbf{X}$, and thus by checking the first-order optimality condition we have the optimal solutions ${\gamma^c}_i={z^c}_i^{-1/2}\|\mathbf{x}_{i,\cdot}\|_2$ and $\gamma^a_{k\,i}=(\alpha z^a_{k\,i})^{-1/2}|x_{ik}|$. Then substituting the solutions into (48), we have the optimization problem in (26).

For the covariance component model we have the following expression

$$\begin{aligned}&\sum_{k=1}^{K}\mathbf{y}_k^T(\mathbf{\Sigma}_k^{sc})^{-1}\mathbf{y}_k\\ &=\arg\min_{\mathbf{S},\mathbf{C}}\sum_{k=1}^{K}\frac{1}{\nu}\|\mathbf{y}_k-\mathbf{\Phi}_k(\mathbf{c}_k+\mathbf{s}_k)\|_2^2+\sum_{i=1}^{m}\frac{c_{ik}^2}{{\gamma^c}_i}+\frac{s_{ik}^2}{\gamma^s_{k\,i}},\end{aligned}\tag{49}$$

which leads to an upper-bounding surrogate function of (25) as

$$\begin{aligned}L_{\mathbf{z}}^{\text{cov}}(\boldsymbol{\gamma}^c,\{\boldsymbol{\gamma}_k^s\})\le&(\mathbf{z}^c)^T\boldsymbol{\gamma}^c+\sum_{k=1}^{K}\beta(\mathbf{z}_k^s)^T\boldsymbol{\gamma}_k^s+\\ &\frac{1}{\nu}\|\mathbf{y}_k-\mathbf{\Phi}_k(\mathbf{c}_k+\mathbf{s}_k)\|_2^2+\sum_{i=1}^{m}\frac{c_{ik}^2}{{\gamma^c}_i}+\frac{s_{ik}^2}{\gamma^s_{k\,i}}\\ =&\tilde{L}_{\mathbf{z}}^{\text{cov}}(\boldsymbol{\gamma}^c,\{\boldsymbol{\gamma}_k^s\},\mathbf{C},\mathbf{S}),\end{aligned}\tag{50}$$

where the equality holds when $\mathbf{C}$ and $\mathbf{S}$ are the solutions of (49). The function $\tilde{L}_{\mathbf{z}}^{\text{pre}}(\boldsymbol{\gamma}^c,\{\boldsymbol{\gamma}_k^s\},\mathbf{C},\mathbf{S})$ in (50) is jointly convex in $\boldsymbol{\gamma}^c$, $\{\boldsymbol{\gamma}_k^s\}$, $\mathbf{C}$ and $\mathbf{S}$, and thus by checking first-order optimality condition we have the optimal solutions ${\gamma^c}_i={z^c}_i^{-1/2}\|\mathbf{c}_{i,\cdot}\|_2$ and $\gamma^s_{k\,i}=(\beta z^s_{k\,i})^{-1/2}|s_{ik}|$. Then substituting the solutions into (50), we have the optimization problem in (27).

APPENDIX C
PROOF FOR THEOREM 2

The idea behind the proof Theorem 2 is to show that the proposed algorithms satisfy all the conditions of Zangwill's global convergence theorem [36]. Let $\mathbf{\Theta}$ be a set of all possible solutions, $\boldsymbol{\theta}\in\mathbf{\Theta}$ be a point in the set, and $\mathcal{A}(\boldsymbol{\theta})$ be a mapping of $\boldsymbol{\theta}$ to every point in $\mathbf{\Theta}$ that satisfies the updating steps of an algorithm. Let $\{\boldsymbol{\theta}_t\}_{t=0}^{\infty}$ be a sequence of points such that $\boldsymbol{\theta}_{t+1}\in\mathcal{A}(\boldsymbol{\theta}_t)$. Zangwill's global convergence theorem requires

1) all points $\boldsymbol{\theta}_t$ are contained in a compact set;
2) there is a continuous function $L(\cdot)$, for every non-stationary point $\boldsymbol{\theta}_t\in\mathbf{\Theta}$, $L(\boldsymbol{\theta}_{t+1})<L(\boldsymbol{\theta}_t)$, while for every stationary point $\boldsymbol{\theta}_t\in\mathbf{\Theta}$, $L(\boldsymbol{\theta}_{t+1})\le L(\boldsymbol{\theta}_t)$;
3) $\mathcal{A}(\boldsymbol{\theta}_t)$ is closed at all non-stationary point $\boldsymbol{\theta}_t$.

Here, we provide the proof for the convergence of the proposed algorithm for SSM-1, and the convergence of the proposed algorithm for SSM-2 can be proved following related arguments.

Firstly, for any non-stationary point $\boldsymbol{\theta}_t\in\mathbf{\Theta}$, the actual cost function $L^{\text{pre}}(\boldsymbol{\theta}_t)$ in (18) is strictly a tangent to the auxiliary cost function $L_z^{\text{pre}}(\boldsymbol{\theta}_t)$ in (24) where $\{\mathbf{z}_k^a\}$ and $\mathbf{z}^c$ are given by (21) and (23), respectively. As $\boldsymbol{\theta}_t$ is a non-stationary point, the slope of $L_z^{\text{pre}}(\boldsymbol{\theta}_t)$ is nonzero. Then the proposed algorithm will find another point $\boldsymbol{\theta}_{t+1}$ satisfying $L_z^{\text{pre}}(\boldsymbol{\theta}_{t+1})<L_z^{\text{pre}}(\boldsymbol{\theta}_t)$, which further leads to $L^{\text{pre}}(\boldsymbol{\theta}_{t+1},\{\mathbf{z}_k^a\},\mathbf{z}^c)<L^{\text{pre}}(\boldsymbol{\theta}_t,\{\mathbf{z}_k^a\},\mathbf{z}^c)$. According to (20), we have

$$\begin{aligned}L^{\text{pre}}(\boldsymbol{\theta}_{t+1})&\le L^{\text{pre}}(\boldsymbol{\theta}_{t+1},\{\mathbf{z}_k^a\},\mathbf{z}^c)\\ &<L^{\text{pre}}(\boldsymbol{\theta}_t,\{\mathbf{z}_k^a\},\mathbf{z}^c)\\ &=L^{\text{pre}}(\boldsymbol{\theta}_t).\end{aligned}$$

Secondly, if $\boldsymbol{\theta}_t\in\mathbf{\Theta}$ is a stationary point of the actual cost function $L^{\text{pre}}(\boldsymbol{\theta})$, then it must be a stationary point of

$L_z^{\text{pre}}(\boldsymbol{\theta}_t)$ in (24), where $\{\mathbf{z}_k^a\}$ and $\mathbf{z}^c$ are given by (21) and (23), respectively. Therefore, the proposed algorithm will returns $\boldsymbol{\theta}_{t+1} = \boldsymbol{\theta}_t$ with $L_z^{\text{pre}}(\boldsymbol{\theta}_t) \leq L_z^{\text{pre}}(\boldsymbol{\theta}_{t+1})$, which leads to $L(\boldsymbol{\theta}_{t+1}) \leq L^{\text{pre}}(\boldsymbol{\theta}_t)$.

Finally, if any element of $\boldsymbol{\theta}_t$ is unbounded, $L^{\text{pre}}(\boldsymbol{\theta}_t)$ diverges to infinity. Therefore, given an initial point $\boldsymbol{\theta}_0$, there exists a closure for $\{\boldsymbol{\theta}_t\}$, and thus $\{\boldsymbol{\theta}_t\}$ belongs to a compact set. In addition, as the cost function of the precision model is a real-valued continuous function on $\boldsymbol{\theta}_t \succeq \mathbf{0}$, by the Weierstrass theorem [41], it follows that $\mathcal{A}(\boldsymbol{\theta}_t)$ is nonempty for every $\boldsymbol{\theta}_t \succeq \mathbf{0}$ and therefore it is also closed by the Lemma 1 in [42].

References

[1] M. Zibulevsky and M. Elad, "L1-L2 optimization in signal and image processing," *IEEE Signal Processing Magazine*, vol. 27, no. 3, pp. 76–88, 2010.

[2] R. M. Willett, M. F. Duarte, M. A. Davenport, and R. G. Baraniuk, "Sparsity and structure in hyperspectral imaging: Sensing, reconstruction, and target detection," *IEEE Signal Processing Magazine*, vol. 31, no. 1, pp. 116–126, 2014.

[3] R. Rubinstein, M. Zibulevsky, and M. Elad, "Double sparsity: Learning sparse dictionaries for sparse signal approximation," *IEEE Transactions on Signal Processing*, vol. 58, no. 3, pp. 1553–1564, 2010.

[4] E. Candès, J. Romberg, and T. Tao, "Robust uncertainty principles: exact signal reconstruction from highly incomplete frequency information," *IEEE Transactions on Information Theory*, vol. 52, no. 2, pp. 489–509, 2006.

[5] D. Donoho, "Compressed sensing," *IEEE Transactions on Information Theory*, vol. 52, no. 4, pp. 1289–1306, 2006.

[6] R. Vidal, "Subspace clustering," *IEEE Signal Processing Magazine*, vol. 28, no. 2, pp. 52–68, 2011.

[7] E. Elhamifar and R. Vidal, "Sparse subspace clustering: Algorithm, theory, and applications," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 35, no. 11, pp. 2765–2781, 2013.

[8] M. Duarte and Y. Eldar, "Structured compressed sensing: From theory to applications," *IEEE Transactions on Signal Processing*, vol. 59, no. 9, pp. 4053–4085, Sept 2011.

[9] J. A. Tropp, A. C. Gilbert, and M. J. Strauss, "Algorithms for simultaneous sparse approximation. part i: Greedy pursuit," *Signal Processing*, vol. 86, no. 3, pp. 572 – 588, 2006.

[10] J. A. Tropp, "Algorithms for simultaneous sparse approximation. part ii: Convex relaxation," *Signal Processing*, vol. 86, no. 3, pp. 589 – 602, 2006.

[11] D. Wipf and B. Rao, "An empirical bayesian strategy for solving the simultaneous sparse approximation problem," *IEEE Transactions on Signal Processing*, vol. 55, no. 7, pp. 3704–3716, July 2007.

[12] S. Ji, D. Dunson, and L. Carin, "Multitask compressive sensing," *IEEE Transactions on Signal Processing*, vol. 57, no. 1, pp. 92–106, Jan 2009.

[13] N. Vaswani and W. Lu, "Modified-cs: Modifying compressive sensing for problems with partially known support," *IEEE Transactions on Signal Processing*, vol. 58, no. 9, pp. 4595–4607, 2010.

[14] A. Jalali, P. Ravikumar, and S. Sanghavi, "A dirty model for multiple sparse regression," *IEEE Transactions on Information Theory*, vol. 59, no. 12, pp. 7947–7968, Dec 2013.

[15] N. Rao, C. Cox, R. Nowak, and T. T. Rogers, "Sparse overlapping sets lasso for multitask learning and its application to fmri analysis," in *Advances in neural information processing systems*, 2013, pp. 2202–2210.

[16] Y. Yan, Y. Yang, D. Meng, G. Liu, W. Tong, A. G. Hauptmann, and N. Sebe, "Event oriented dictionary learning for complex event detection," *IEEE Transactions on Image Processing*, vol. 24, no. 6, pp. 1867–1878, 2015.

[17] Y. Suo, M. Dao, T. Tran, H. Mousavi, U. Srinivas, and V. Monga, "Group structured dirty dictionary learning for classification," in *2014 IEEE International Conference on Image Processing (ICIP)*, 2014, pp. 150–154.

[18] S. Oymak, A. Jalali, M. Fazel, Y. Eldar, and B. Hassibi, "Simultaneously structured models with application to sparse and low-rank matrices," *IEEE Transactions on Information Theory*, vol. 61, no. 5, pp. 2886–2908, May 2015.

[19] Y. Yan, E. Ricci, R. Subramanian, G. Liu, O. Lanz, and N. Sebe, "A multi-task learning framework for head pose estimation under target motion," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 38, no. 6, pp. 1070–1083, June 2016.

[20] Y. Zhang and D.-Y. Yeung, "A convex formulation for learning task relationships in multi-task learning," in *Proceedings of the 26th Conference on Uncertainty in Artificial Intelligence*, 2010.

[21] Y. Yan, E. Ricci, G. Liu, and N. Sebe, "Egocentric daily activity recognition via multitask clustering," *IEEE Transactions on Image Processing*, vol. 24, no. 10, pp. 2984–2995, 2015.

[22] Q. Ling, Z. Wen, and W. Yin, "Decentralized jointly sparse optimization by reweighted ℓ_q minimization," *IEEE Transactions on Signal Processing*, vol. 61, no. 5, pp. 1165–1170, March 2013.

[23] Q. Ling and Z. Tian, "Decentralized support detection of multiple measurement vectors with joint sparsity," in *IEEE International Conference on Acoustics, Speech and Signal Processing*. IEEE, 2011, pp. 2996–2999.

[24] M. E. Tipping, "Sparse Bayesian learning and the relevance vector machine," *The journal of machine learning research*, vol. 1, pp. 211–244, 2001.

[25] D. Wipf and B. Rao, "Sparse bayesian learning for basis selection," *IEEE Transactions on Signal Processing*, vol. 52, no. 8, pp. 2153–2164, Aug 2004.

[26] R. Tibshirani, "Regression shrinkage and selection via the lasso," *Journal of the Royal Statistical Society. Series B (Methodological)*, pp. 267–288, 1996.

[27] R. Chartrand and W. Yin, "Iteratively reweighted algorithms for compressive sensing," in *IEEE international conference on Acoustics, speech and signal processing*. IEEE, 2008, pp. 3869–3872.

[28] I. F. Gorodnitsky and B. D. Rao, "Sparse signal reconstruction from limited data using focuss: A re-weighted minimum norm algorithm," *IEEE Transactions on Signal Processing*, vol. 45, no. 3, pp. 600–616, 1997.

[29] E. J. Candès, M. B. Wakin, and S. P. Boyd, "Enhancing sparsity by reweighted ℓ_1 minimization," *Journal of Fourier analysis and applications*, vol. 14, no. 5, pp. 877–905, 2008.

[30] D. Wipf and S. Nagarajan, "Iterative reweighted ℓ_1 and ℓ_2 methods for finding sparse solutions," *IEEE Journal of Selected Topics in Signal Processing*, vol. 4, no. 2, pp. 317–329, April 2010.

[31] D. Wipf, B. Rao, and S. Nagarajan, "Latent variable bayesian models for promoting sparsity," *IEEE Transactions on Information Theory*, vol. 57, no. 9, pp. 6236–6255, Sept 2011.

[32] Y. Wu and D. P. Wipf, "Dual-space analysis of the sparse linear model," in *Advances in Neural Information Processing Systems*, 2012, pp. 1745–1753.

[33] P. Gong, J. Ye, and C. Zhang, "Multi-stage multi-task feature learning," *The Journal of Machine Learning Research*, vol. 14, no. 1, pp. 2979–3010, 2013.

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

[35] S. Boyd and L. Vandenberghe, *Convex optimization*. Cambridge university press, 2004.

[36] W. I. Zangwill, *Nonlinear programming: a unified approach*. Prentice-Hall Englewood Cliffs, NJ, 1969, vol. 196, no. 9.

[37] S. Boyd, N. Parikh, E. Chu, B. Peleato, and J. Eckstein, "Distributed optimization and statistical learning via the alternating direction method of multipliers," *Foundations and Trends in Machine Learning*, vol. 3, no. 1, pp. 1–122, 2011.

[38] G. Mateos, J. Bazerque, and G. Giannakis, "Distributed sparse linear regression," *IEEE Transactions on Signal Processing*, vol. 58, no. 10, pp. 5262–5276, Oct 2010.

[39] J. Wright, A. Y. Yang, A. Ganesh, S. S. Sastry, and Y. Ma, "Robust face recognition via sparse representation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 31, no. 2, pp. 210–227, 2009.

[40] I. Ramirez, P. Sprechmann, and G. Sapiro, "Classification and clustering via dictionary learning with structured incoherence and shared features," pp. 3501–3508, 2010.

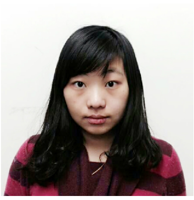
[41] M. Minoux, *Mathematical programming: theory and algorithms*. John Wiley & Sons, 1986.

[42] B. K. Sriperumbudur and G. R. Lanckriet, "A proof of convergence of the concave-convex procedure using zangwill's theory," *Neural computation*, vol. 24, no. 6, pp. 1391–1407, 2012.

Wei Chen (M'13) received the B.Eng. degree and M.Eng. degree in Communications Engineering from Beijing University of Posts and Telecommunications, China, in 2006 and 2009, respectively, and the Ph.D. degree in Computer Science from the University of Cambridge, UK, in 2012. He is currently an Associate Professor with Beijing Jiaotong University, Beijing, China, and also an Research Associate with the Computer Laboratory, University of Cambridge. Dr. Chen is the recipient of the 2013 IET Wireless Sensor Systems Premium Award. His current research interests include sparse representation, Bayesian inference, wireless sensor networks and image processing.

David Wipf (M05) received the B.S. degree with highest honors from the University of Virginia, and the Ph.D. degree from UC San Diego, where he was an NSF IGERT Fellow. Later he was an NIH Postdoctoral Fellow at UC San Francisco. Since 2011 he has been with Microsoft Research in Beijing. His research interests include Bayesian learning techniques applied to signal/image processing and computer vision. He is the recipient of several awards including the 2012 Signal Processing Society Best Paper Award, the Biomag 2008 Young Investigator Award, and the 2006 NIPS Outstanding Paper Award.

Yu Wang received the B.S. degree in Communication Engineering from Beijing Jiaotong University, Beijing, China, in 2010, the M.S degree with distinction in Wireless Communication from University of Southampton, UK, in 2011, and the Ph.D. degree in Computer Science from University of Cambridge, UK in 2016. She is now a postdoctoral research associate at the Department of Pure Mathematics and Mathematical Statistics, University of Cambridge, Cambridge, UK. Her research interests include machine learning, Bayesian inference, deep learning and sparse inference.



Yang Liu received the B.Sc. degrees in Telecommunication Engineering from Beijing University of Posts and Telecommunications and Queen Mary, University of London in 2013, respectively, and the MPhil degree from the Computer Laboratory, University of Cambridge in 2014. She is currently working towards her Ph.D. degree in Computer Science at the Computer Laboratory, University of Cambridge, UK. Her current research interests include pattern recognition, computer vision, and applied machine learning.

Ian Wassell received his B.Sc. and B.Eng. degrees from the University of Loughborough in 1983, and his Ph.D. degree from the University of Southampton in 1990. He is a Senior Lecturer at the Computer Laboratory, University of Cambridge and has experience in excess of 25 years in the simulation and design of radio communication systems gained via a number of positions in industry and higher education. He has published more than 190 papers and his current research interests include: wireless sensor networks, cooperative wireless networks, propagation modelling, sparse representation and compressive sensing. He is a member of the IET and a Chartered Engineer.