EURASIP Journal on Advances in Signal Processing
a SpringerOpen Journal

RESEARCH Open Access

Enhanced multi-task compressive sensing using Laplace priors and MDL-based task classification

Ying-Gui Wang[1]*, Le Yang[1,2], Liang Tang[3], Zheng Liu[1] and Wen-Li Jiang[1]

Abstract

In multi-task compressive sensing (MCS), the original signals of multiple compressive sensing (CS) tasks are assumed to be correlated. This is explored to recover signals in a joint manner to improve signal reconstruction performance. In this paper, we first develop an improved version of MCS that imposes sparseness over the original signals using Laplace priors. The newly proposed technique, termed as the Laplace prior-based MCS (LMCS), adopts a hierarchical prior model, and the MCS is shown analytically to be a special case of LMCS. This paper next considers the scenario where the CS tasks belong to different groups. In this case, the original signals from different task groups are not well correlated, which would degrade the signal recovery performance of both MCS and LMCS. We propose the use of the minimum description length (MDL) principle to enhance the MCS and LMCS techniques. New algorithms, referred to as MDL-MCS and MDL-LMCS, are developed. They first classify tasks into different groups and then reconstruct signals from each cluster jointly. Simulations demonstrate that the proposed algorithms have better performance over several state-of-art benchmark techniques.

Keywords: Multi-task; Compressive sensing; Laplace priors; Minimum description length; Task classification

1 Introduction

If a signal is compressible in the sense that its representation in a certain linear canonical basis is sparse, it can then be recovered from measurements obtained at a rate much lower than the Nyquist frequency using the technique of compressive sensing (CS) [1-3]. Mathematically, in CS, the signal is measured via

$$\boldsymbol{y} = \boldsymbol{\Phi}_0 \boldsymbol{\Psi} \boldsymbol{\theta} + \boldsymbol{n} = \boldsymbol{\Phi} \boldsymbol{\theta} + \boldsymbol{n} \tag{1}$$

where $\boldsymbol{\theta}$ is the $N \times 1$ original signal vector, $\boldsymbol{\Phi}_0$ denotes the $M \times N$ measurement matrix, $\boldsymbol{\Psi}$ denotes the $N \times N$ linear basis, $\boldsymbol{\Phi} = \boldsymbol{\Phi}_0\boldsymbol{\Psi}$, $\boldsymbol{y}$ is the $M \times 1$ compressive measurement vector, and $\boldsymbol{n}$ is the additive noise. Since M is far smaller than N, the original signal is now compressively represented, but the inverse problem, namely recovering $\boldsymbol{\theta}$ from $\boldsymbol{y}$, is in general ill-posed. If $\boldsymbol{\theta}$ is sparse (i.e., most of its elements are zero), the signal reconstruction problem could become feasible. An approximation to the original signal in this case can be obtained through the technique of basis pursuit that solves

$$\widehat{\boldsymbol{\theta}} = \arg\min_{\boldsymbol{\theta}} \|\boldsymbol{\theta}\|_1, \qquad \text{s.t.} \quad \left\|\boldsymbol{y} - \boldsymbol{\Phi}\boldsymbol{\theta}\right\|_2 \leq \varepsilon \tag{2}$$

where $\|\cdot\|_2$ and $\|\cdot\|_1$ denote the l_2-norm and the l_1-norm, respectively, and the scalar ε is a small constant. Equation 2 has been the starting point for the development of many signal recovery methods in the literature. Among them, the recovery algorithms under the Bayesian framework provide some advantages over other formulations. These include providing probabilistic predictions, automatic estimation of model parameters, and the evaluation of the uncertainty of reconstruction. The existing Bayesian approaches include the Bayesian compressive sensing (BCS) [4] that stems from the relevance vector machine [5] and the Laplace prior-based BCS [6].

In [7], multi-task compressive sensing (MCS) was introduced within the Bayesian framework. In this work, a CS task refers to the union of an original signal vector,

*Correspondence: wyinggui@gmail.com
[1] College of Electronic Science and Engineering, National University of Defense Technology, Deya Road, Changsha 410073, People's Republic of China
Full list of author information is available at the end of the article

the measurement matrix, and the associated compressive measurement vector obtained using Equation 1. In contrast to the CS aim of recovering a single signal from its compressive measurements, MCS exploits the statistical correlation among the original signals of multiple CS tasks and recovers them jointly to improve the signal reconstruction performance. It has been shown in [7] that MCS allows recovering in a robust manner the signals whose compressive measurements are insufficient when they are reconstructed separately. The MCS technique has been investigated extensively in machine learning literature, where it was referred to as simultaneous sparse approximation (SSA) [8-12] as well as distributed compressed sensing [13]. In [14], an empirical Bayesian strategy for SSA was developed.

The contribution of this paper is twofold. We shall first extend the work of [6] on the Laplace prior-based BCS to the MCS scenario. A new MCS algorithm for signal recovery, termed as the Laplace prior-based MCS (LMCS), is developed. We impose Laplace priors on the original signals in a hierarchical manner and show that the MCS is indeed a special case of LMCS. The incorporation of Laplace priors enforces signal sparsity to a higher extent [15] and offers posterior distributions rather than point estimates as in MCS. Another advantage comes from the log-concavity of the Laplace distribution, which leads to unimodal posterior distribution and eliminates the presence of local minima as a result.

The second part of this work comes from the following observation. Specifically, in order to provide satisfactory signal reconstruction performance, the MCS technique from [7], together with the newly proposed LMCS method, requires that the original signals of the multiple CS tasks are well correlated statistically. This assumption may not be fulfilled in many practical applications. For instance, some original signals may be realizations of different signal templates that differ in their supports. In other words, they could belong to different signal groups, and the statistical correlation among them is weak, which would degrade the signal recovery performance. A possible approach to address this problem is to group the CS tasks before the signal reconstruction stage, as in the MCS with Dirichlet process priors (DP-MCS) [16].

The second contribution of this paper is the use of the minimum description length (MDL) principle to augment the MCS and LMCS methods. The obtained techniques are referred to as the MDL-MCS and MDL-LMCS algorithms. The MDL principle has been adopted to solve the model selection problem [17-19] and can also be used in other aspects, such as sparse coding and dictionary learning [20] and radar emitter classification [21-23]. In MDL, the best model for a given data $\boldsymbol{y}$ is the solution to the minimization problem $\widehat{\omega} = \arg\min_{\omega\in\Omega} DL\left(\boldsymbol{y},\omega\right)$. Here, Ω represents the set of possible models and $DL\left(\boldsymbol{y},\omega\right)$ is a codelength assignment function which defines the theoretical codelength required to describe $\boldsymbol{y}$ uniquely, which is the key component in any MDL-based classification technique. Common practice in MDL uses the ideal Shannon codelength assignment [24] to define $DL\left(\boldsymbol{y},\omega\right)$ in terms of a probability assignment $p\left(\boldsymbol{y},\omega\right)$ as $DL\left(\boldsymbol{y},\omega\right) = -\log_2 p\left(\boldsymbol{y},\omega\right)$. Applying $p\left(\boldsymbol{y},\omega\right) = p\left(\boldsymbol{y}\,|\omega\right)p\left(\omega\right)$, we have $\widehat{\omega} = \arg\min_{\omega\in\Omega} -\log_2 p\left(\boldsymbol{y}\,|\omega\right) - \log_2 p\left(\omega\right)$, where $-\log_2 p\left(\omega\right)$ represents the model complexity. Note that the MCS and the new LMCS methods are both under the Bayesian framework, which enables their integration with the statistical MDL technique. Compared with the DP-MCS technique that utilizes variational Bayes (VB) inference and could suffer from local convergence, the newly proposed MDL-MCS and MDL-LMCS methods offer improved correct signal classification rate and better signal reconstruction performance. This is also illustrated via computer simulations in Section 5.

The remainder of this paper is structured as follows. In Section 2, we review the prior sharing concept in MCS and present the prior sharing framework in LMCS. Section 3 develops the proposed LMCS algorithm. We describe in Section 4 the MDL-based MCS and LMCS techniques, namely, the MDL-MCS and MDL-LMCS algorithms. Simulations are given in Section 5 to illustrate the performance of the proposed algorithms. Section 6 concludes the paper.

2 Prior sharing in MCS and LMCS

In the area of machine learning, information sharing among tasks is a well-known technique [25]. Typical approaches, to name a few, include sharing hidden nodes in neural networks [26,27], assigning a common prior in hierarchical Bayesian models [28-30], placing a common structure on the predictor space [31], and the structured regularization in kernel methods [32]. Among them, the use of hierarchical Bayesian models with shared priors is one of the most important methods for multi-task learning [33-37], which is also essential for the development of MCS in [7] and the LMCS algorithm in this paper. For the sake of clarity, in the rest of this section, we shall first review the prior sharing in the MCS algorithm and then proceed to present the hierarchical Bayesian framework of LMCS.

To facilitate the presentation, suppose there are L CS tasks

$$\boldsymbol{y}_i = \boldsymbol{\Phi}_i\boldsymbol{\theta}_i + \boldsymbol{n}_i \tag{3}$$

where $i = 1, 2, \ldots, L$, $\boldsymbol{y}_i$ is the $M_i \times 1$ compressive measurement vector and $\boldsymbol{\Phi}_i$ is the $M_i \times N$ matrix ($M_i \ll N$) whose columns are $\boldsymbol{\Phi}_{i,j}$, $j = 1, 2, \ldots, N$ such that $\boldsymbol{\Phi}_i =$

$\left[\mathbf{\Phi}_{i,1}, \ldots, \mathbf{\Phi}_{i,N}\right]$. Here, $\boldsymbol{\theta}_i = \left[\theta_{i,1}, \ldots, \theta_{i,N}\right]^T$ is the original signal for task i and the measurement noise $\boldsymbol{n}_i$ is assumed to follow an i.i.d. Gaussian distribution with zero mean vector and covariance matrix $\beta^{-1}\mathbf{I}$. The conditional likelihood function of $\boldsymbol{y}_i$ is

$$p\left(\boldsymbol{y}_i|\boldsymbol{\theta}_i, \beta\right) = \mathcal{N}\left(\boldsymbol{y}_i|\mathbf{\Phi}_i\boldsymbol{\theta}_i, \beta^{-1}\mathbf{I}\right) \tag{4}$$

where $\mathcal{N}\left(\boldsymbol{y}_i|\mathbf{\Phi}_i\boldsymbol{\theta}_i, \beta^{-1}\mathbf{I}\right)$ represents a Gaussian distribution with mean vector $\mathbf{\Phi}_i\boldsymbol{\theta}_i$ and covariance matrix $\beta^{-1}\mathbf{I}$. The noise precision β follows a Gamma distribution

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

where a and b are the shape and scale parameters of the Gamma distribution and $\Gamma\left(a\right)$ is the Gamma function.

2.1 Prior sharing in MCS

In MCS [7], the elements in $\boldsymbol{\theta}_i$ are statistically independent, and they follow a joint Gaussian distribution:

$$p\left(\boldsymbol{\theta}_i|\boldsymbol{\alpha}\right) = \prod_{j=1}^{N}\mathcal{N}\left(\theta_{i,j}|0, \alpha_j^{-1}\right). \tag{6}$$

Here, $\boldsymbol{\alpha} = \left[\alpha_1, \alpha_2, \ldots, \alpha_N\right]^T$ is the information vector shared by the original signals $\boldsymbol{\theta}_i$ of all the L tasks. Its distribution function is given by

$$p\left(\boldsymbol{\alpha}|c, d\right) = \prod_{j=1}^{N}\mathrm{Ga}\left(\alpha_j|c, d\right). \tag{7}$$

In [7], the general strategy of setting the hyper-parameters a, c to ones and b, d to zeros in Equations 5 and 7 was adopted so that the prior of $\boldsymbol{\alpha}$ and β are both uniformly distributed. As a result, they can be found via maximizing the following likelihood function:

$$\prod_{i=1}^{L}p\left(\boldsymbol{y}_i|\boldsymbol{\alpha}, \beta\right) = \prod_{i=1}^{L}\int p\left(\boldsymbol{y}_i|\boldsymbol{\theta}_i, \beta\right)p\left(\boldsymbol{\theta}_i|\boldsymbol{\alpha}\right)d\boldsymbol{\theta}_i. \tag{8}$$

This is equivalent to maximizing the posterior distribution of $\boldsymbol{\alpha}$ and β. The original signals $\boldsymbol{\theta}_i$ are then reconstructed using the estimated values of $\boldsymbol{\alpha}$ and β.

2.2 Prior sharing in LMCS

Within the LMCS framework, the original signals are assigned Laplace priors. A possible approach to achieve this is to impose Laplace priors directly on the original signal, or mathematically, let $p\left(\boldsymbol{\theta}_i|\lambda\right) = \frac{\lambda}{2}\exp\left(-\frac{\lambda}{2}\left\|\boldsymbol{\theta}_i\right\|_1\right)$ as in [6]. However, this formulation is not conjugate to the conditional distribution in Equation 4, which would render the Bayesian analysis intractable. Therefore, we adopt the hierarchical prior given by

$$p\left(\boldsymbol{\theta}_i|\boldsymbol{\gamma}\right) = \prod_{j=1}^{N}\mathcal{N}\left(\theta_{i,j}|0, \gamma_j\right) \tag{9}$$

$$\begin{aligned} p\left(\gamma_j|\lambda\right) &= \mathrm{Ga}\left(\gamma_j|1, \lambda/2\right) \\ &= \frac{\lambda}{2}\exp\left(-\frac{\lambda\gamma_j}{2}\right), \gamma_j \geq 0, \lambda \geq 0 \end{aligned} \tag{10}$$

$$p\left(\lambda|\nu\right) = \mathrm{Ga}\left(\lambda|\nu/2, \nu/2\right) \tag{11}$$

where $\boldsymbol{\gamma} = \left[\gamma_1, \ldots, \gamma_N\right]^T$, $p\left(\gamma_j|\lambda\right)$, and $p\left(\lambda|\nu\right)$ are the prior distributions of γ_j and λ, respectively. Compared with the MCS model given in Equation 6, Equation 7 reveals that in LMCS, information sharing is realized via the vector $\boldsymbol{\gamma}$ and the hyper-parameter λ. We have from Equations 9 to 11

$$\begin{aligned} p\left(\boldsymbol{\theta}_i|\lambda\right) &= \int p\left(\boldsymbol{\theta}_i|\boldsymbol{\gamma}\right)p\left(\boldsymbol{\gamma}|\lambda\right)d\boldsymbol{\gamma} \\ &= \prod_{j=1}^{N}\int p\left(\theta_{i,j}|\gamma_j\right)p\left(\gamma_j|\lambda\right)d\gamma_j \\ &= \frac{\lambda^{N/2}}{2^N}\exp\left(-\sqrt{\lambda}\sum_{j=1}^{N}\left|\theta_{i,j}\right|\right). \end{aligned} \tag{12}$$

This verifies that the used hierarchical prior model results in Laplace priors for the original signals $\boldsymbol{\theta}_i$.

As in MCS, LMCS recovers the original signals in a two-step manner. In particular, it first estimates $\boldsymbol{\gamma}$, λ, β, and ν via maximizing the posterior distribution

$$\begin{aligned} &\prod_{i=1}^{L}p\left(\boldsymbol{\theta}_i, \boldsymbol{\gamma}, \lambda, \beta, \nu|\boldsymbol{y}_i\right) \\ &\quad= \prod_{i=1}^{L}p\left(\boldsymbol{\theta}_i|\boldsymbol{\gamma}, \lambda, \beta, \boldsymbol{y}_i\right)p\left(\boldsymbol{\gamma}, \lambda, \beta, \nu|\boldsymbol{y}_i\right). \end{aligned} \tag{13}$$

Taking the logarithm on both sides of the above equation yields

$$\begin{aligned} &\sum_{i=1}^{L}\ln p\left(\boldsymbol{\theta}_i, \boldsymbol{\gamma}, \lambda, \beta, \nu|\boldsymbol{y}_i\right) \\ &\quad= \sum_{i=1}^{L}\ln p\left(\boldsymbol{\theta}_i|\boldsymbol{\gamma}, \lambda, \beta, \boldsymbol{y}_i\right) + \sum_{i=1}^{L}\ln p\left(\boldsymbol{\gamma}, \lambda, \beta, \nu|\boldsymbol{y}_i\right). \end{aligned} \tag{14}$$

It is straightforward to verify that $p\left(\boldsymbol{\theta}_i|\boldsymbol{\gamma},\lambda,\beta,\boldsymbol{y}_i\right) = \frac{p(\boldsymbol{y}_i|\boldsymbol{\theta}_i,\beta)p(\boldsymbol{\theta}_i|\boldsymbol{\gamma})p(\boldsymbol{\gamma}|\lambda)}{\int p(\boldsymbol{y}_i|\boldsymbol{\theta}_i,\beta)p(\boldsymbol{\theta}_i|\boldsymbol{\gamma})p(\boldsymbol{\gamma}|\lambda)d\boldsymbol{\theta}_i} = \frac{p(\boldsymbol{y}_i|\boldsymbol{\theta}_i,\beta)p(\boldsymbol{\theta}_i|\boldsymbol{\gamma})}{\int p(\boldsymbol{y}_i|\boldsymbol{\theta}_i,\beta)p(\boldsymbol{\theta}_i|\boldsymbol{\gamma})d\boldsymbol{\theta}_i}$. Furthermore, from Equations 4 and 9, we have that it also has a Gaussian distribution $\mathcal{N}\left(\boldsymbol{\theta}_i|\boldsymbol{\mu}'_i,\boldsymbol{\Sigma}'_i\right)$ with the mean vector and covariance matrix equal to

$$\boldsymbol{\mu}'_i = \beta\boldsymbol{\Sigma}'_i\boldsymbol{\Phi}_i^T\boldsymbol{y}_i \tag{15}$$

$$\boldsymbol{\Sigma}'_i = \left[\beta\boldsymbol{\Phi}_i^T\boldsymbol{\Phi}_i + \boldsymbol{\Gamma}_0\right]^{-1} \tag{16}$$

where $\boldsymbol{\Gamma}_0 = \mathrm{diag}\left(1/\gamma_1,\ldots,1/\gamma_N\right)$.

With the estimated $\boldsymbol{\gamma}$, λ, and ν, LMCS then proceeds to reconstruct the original signals from all the L CS tasks.

We illustrate the hierarchical prior model adopted in LMCS in Figure 1. It can be observed that, as in MCS, the distribution of the measurement noise $\boldsymbol{n}_i$ is dependent on the noise precision β while the prior distribution functions of the original signals $\boldsymbol{\theta}_i$ depend on the information sharing vector $\boldsymbol{\gamma}$. The difference here is that LMCS has one more layer of prior information, which is embedded in λ. The introduction of λ makes the prior distribution of the original signal Laplace, which is already shown in Equation 12. As a result, the proposed LMCS would promote the sparsity of the recovered signal, as pointed out in [15].

3 Multi-task compressive sensing using Laplace priors

We shall present the proposed LMCS algorithm in this section. The LMCS method differs from the MCS technique only in the step of identifying the information sharing vector $\boldsymbol{\gamma}$ and the parameters λ and ν while their signal recovery steps are the same. As a result, we shall focus on the estimation of $\boldsymbol{\gamma}$, λ, and ν. Interested readers are directed to [7] for details on the signal recovery process.

As shown in previous works [7,38,39], the signal reconstruction performance would be degraded if the noise precision β is not properly initialized. Therefore, in this work, we consider β as a nuisance parameter and integrate it out to reduce the number of unknowns and improve the robustness of the algorithm. For this purpose, the prior distributions of the original signals $\boldsymbol{\theta}_i$ are rewritten as in [7]:

$$p\left(\boldsymbol{\theta}_i|\boldsymbol{\gamma},\beta\right) = \prod_{j=1}^{N}\mathcal{N}\left(\theta_{i,j}|0,\gamma_j\beta^{-1}\right) \tag{17}$$

where β has a Gamma prior distribution

$$p\left(\beta|a,b\right) = \mathrm{Ga}\left(\beta|a,b\right). \tag{18}$$

Note that in this case, $p\left(\boldsymbol{\theta}_i|\boldsymbol{\gamma},\lambda,\beta,\boldsymbol{y}_i\right)$ given above Equation 15 is still Gaussian with the mean vector and the covariance matrix given in Equation 15 and 16. After taking integration with respect to β, we have

$$\begin{aligned} &p\left(\boldsymbol{\theta}_i|\boldsymbol{\gamma},\lambda,\boldsymbol{y}_i\right)\\ &\quad= \int p\left(\boldsymbol{\theta}_i|\boldsymbol{\gamma},\lambda,\beta,\boldsymbol{y}_i\right)p\left(\beta|a,b\right)d\beta\\ &\quad= \frac{\Gamma\left(a+N/2\right)\left[1+\frac{1}{2b}\left(\boldsymbol{\theta}_i-\boldsymbol{\mu}_i\right)^T\boldsymbol{\Sigma}_i^{-1}\left(\boldsymbol{\theta}_i-\boldsymbol{\mu}_i\right)\right]^{-(a+N/2)}}{\Gamma\left(a\right)\left(2\pi b\right)^{N/2}\left(\det\left(\boldsymbol{\Sigma}_i\right)\right)^{1/2}} \end{aligned} \tag{19}$$

where $\det\left(\cdot\right)$ is the determinant operator and

$$\boldsymbol{\mu}_i = \boldsymbol{\Sigma}_i\boldsymbol{\Phi}_i^T\boldsymbol{y}_i \tag{20}$$

$$\boldsymbol{\Sigma}_i = \left[\boldsymbol{\Phi}_i^T\boldsymbol{\Phi}_i + \boldsymbol{\Gamma}_0\right]^{-1}. \tag{21}$$

Note that $p\left(\boldsymbol{\theta}_i|\boldsymbol{\gamma},\lambda,\boldsymbol{y}_i\right)$ has the functional form of a Student's t distribution, which is heavy tailed and as a result makes the LMCS algorithm more robust to the presence of outliers in the measurement noise in $\boldsymbol{y}_i$ if any, as pointed out in [40].

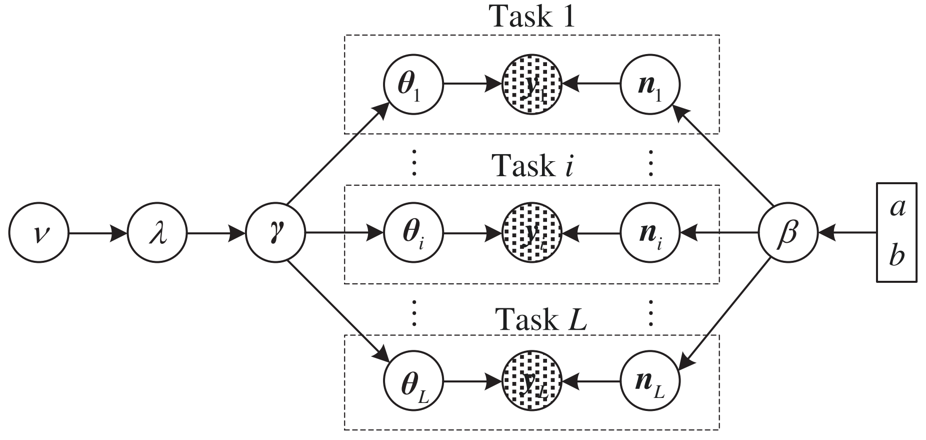

Figure 1 Hierarchical prior model of LMCS.

Taking integration with respect to β on both sides of Equation 13, using Equation 19, and applying the logarithm yields the posterior distribution function $\sum_{i=1}^{L} \ln p\left(\boldsymbol{\theta}_i, \boldsymbol{\gamma}, \lambda, \nu | \boldsymbol{y}_i\right) = \sum_{i=1}^{L} \ln p\left(\boldsymbol{\theta}_i | \boldsymbol{\gamma}, \lambda, \boldsymbol{y}_i\right) + \sum_{i=1}^{L} \ln p\left(\boldsymbol{\gamma}, \lambda, \nu | \boldsymbol{y}_i\right)$. We shall maximize it to estimate the information sharing vector $\boldsymbol{\gamma}$ and the parameter λ. We begin with integrating $\boldsymbol{\theta}_i$ out and applying the relationship $p\left(\boldsymbol{\gamma}, \lambda, \nu | \boldsymbol{y}_i\right) = p\left(\boldsymbol{y}_i, \boldsymbol{\gamma}, \lambda, \nu\right) / p\left(\boldsymbol{y}_i\right) \propto p\left(\boldsymbol{y}_i, \boldsymbol{\gamma}, \lambda, \nu\right)$ to obtain

$$\begin{aligned}
&\mathcal{L}\left(\boldsymbol{\gamma}, \lambda, \nu\right) \\
&\triangleq \sum_{i=1}^{L} \ln p\left(\boldsymbol{y}_i, \boldsymbol{\gamma}, \lambda, \nu\right) \\
&= \sum_{i=1}^{L} \ln \int\int p\left(\boldsymbol{y}_i | \boldsymbol{\theta}_i, \beta\right) p\left(\boldsymbol{\theta}_i | \boldsymbol{\gamma}\right) p\left(\boldsymbol{\gamma} | \lambda\right) p\left(\lambda\right) p\left(\beta\right) d\boldsymbol{\theta}_i d\beta \\
&= -\frac{1}{2} \sum_{i=1}^{L} \Bigg[\left(M_i + 2a\right) \ln \left(\boldsymbol{y}_i^T \mathbf{B}_i^{-1} \boldsymbol{y}_i + 2b\right) + \ln\left(\det\left(\mathbf{B}_i\right)\right) \\
&\quad - 2N \ln \frac{\lambda}{2} + \lambda \sum_{j=1}^{N} \gamma_j - \nu \ln \frac{\nu}{2} + 2 \ln \Gamma\left(\nu/2\right) \\
&\quad - \left(\nu - 2\right) \ln \lambda + \nu\lambda \Bigg] + \frac{1}{2} \sum_{i=1}^{L} \Bigg[2 \ln \frac{2b^a \Gamma\left(M_i/2 + a\right)}{\Gamma\left(a\right)} \\
&\quad - M_i \ln 2\pi \Bigg]
\end{aligned} \tag{22}$$

where $\mathbf{B}_i = \mathbf{I} + \boldsymbol{\Phi}_i \boldsymbol{\Gamma}_0^{-1} \boldsymbol{\Phi}_i^T$, $\mathbf{B}_i^{-1} = \left(\mathbf{I} + \boldsymbol{\Phi}_i \boldsymbol{\Gamma}_0^{-1} \boldsymbol{\Phi}_i^T\right)^{-1} = \mathbf{I} - \boldsymbol{\Phi}_i \boldsymbol{\Sigma}_i \boldsymbol{\Phi}_i^T$, $\det\left(\mathbf{B}_i\right) = \left(\det\left(\boldsymbol{\Gamma}_0\right)\right)^{-1} \left(\det\left(\boldsymbol{\Sigma}_i\right)\right)^{-1}$. The matrices $\boldsymbol{\Gamma}_0$ and $\boldsymbol{\Sigma}_i$ are defined under Equation 16 and in Equation 21, respectively.

In the rest of this section, we shall present two methods for identifying $\boldsymbol{\gamma}$ and λ. The first technique, described in Section 3.1 iteratively maximizes $\mathcal{L}\left(\boldsymbol{\gamma}, \lambda, \nu\right)$ to find the accurate solution. It has high computational complexity, which motivates the development of an alternative method with much lower complexity in Section 3.2.

3.1 Iterative solution

Differentiating $\mathcal{L}\left(\boldsymbol{\gamma}, \lambda, \nu\right)$ with respect to γ_j, $j = 1, 2, \ldots, N$ and setting the result to zero yield

$$\begin{aligned}
\frac{d\mathcal{L}\left(\boldsymbol{\gamma}, \lambda, \nu\right)}{d\gamma_j} &= \frac{1}{2} \Bigg[\frac{1}{\gamma_j^2} \sum_{i=1}^{L} \left(\frac{M_i + 2a}{\boldsymbol{y}_i^T \mathbf{B}_i^{-1} \boldsymbol{y}_i + 2b} \boldsymbol{\mu}_{i,j}^2 + \boldsymbol{\Sigma}_{i,jj} \right) \\
&\quad - \frac{L}{\gamma_j} - L\lambda \Bigg] \\
&= 0.
\end{aligned} \tag{23}$$

After some straightforward manipulations, we obtain

$$\gamma_j^{-1} = \frac{L + \sqrt{L^2 + 4L\lambda \sum_{i=1}^{L} \left(\frac{M_i + 2a}{\boldsymbol{y}_i^T \mathbf{B}_i^{-1} \boldsymbol{y}_i + 2b} \boldsymbol{\mu}_{i,j}^2 + \boldsymbol{\Sigma}_{i,jj} \right)}}{2 \sum_{i=1}^{L} \left(\frac{M_i + 2a}{\boldsymbol{y}_i^T \mathbf{B}_i^{-1} \boldsymbol{y}_i + 2b} \boldsymbol{\mu}_{i,j}^2 + \boldsymbol{\Sigma}_{i,jj} \right)} \tag{24}$$

where $\boldsymbol{\mu}_{i,j}$ is the jth element of $\boldsymbol{\mu}_i$ and $\boldsymbol{\Sigma}_{i,jj}$ is the jth diagonal element of $\boldsymbol{\Sigma}_i$. Following a similar approach, λ can be found to be

$$\lambda = \frac{N - 1 + \nu/2}{\sum_{j=1}^{N} \gamma_j/2 + \nu/2}. \tag{25}$$

As in [6], we evaluate ν by solving

$$\ln \frac{\nu}{2} + 1 - \psi\left(\frac{\nu}{2}\right) + \ln \lambda - \lambda = 0 \tag{26}$$

where $\psi\left(\nu/2\right)$ denotes the derivative of $\ln \Gamma\left(\nu/2\right)$ with respect to $\nu/2$.

The iterative algorithm starts with an initial solution guess on $\boldsymbol{\gamma}$, λ and ν. We next update the estimates of γ_i using Equation 24 first, then proceed to evaluate λ and ν using Equations 25 and 26. The above process would be repeated until convergence. The iterative algorithm is based on alternating optimization and is computationally intensive. One of the computational burdens lies in the evaluation of Equations 20 and 21 required in the evaluation of Equation 24, where inverting matrices of size $N \times N$ is needed. This motivates the development of the following alternative algorithm.

3.2 Fast alternative solution

We start with decomposing $\mathbf{B}_i$ defined under Equation 22 as $\mathbf{B}_i = \mathrm{I} + \sum_{k=1(\neq j)}^{N} \gamma_k \boldsymbol{\Phi}_{i,k} \boldsymbol{\Phi}_{i,k}^T + \gamma_j \boldsymbol{\Phi}_{i,j} \boldsymbol{\Phi}_{i,j}^T = \mathbf{B}_{i,-j} + \gamma_j \boldsymbol{\Phi}_{i,j} \boldsymbol{\Phi}_{i,j}^T$, where $\mathbf{B}_{i,-j}$ is $\mathbf{B}_i$ with the contribution of the column $\boldsymbol{\Phi}_{i,j}$ in the matrix $\boldsymbol{\Phi}_i$ removed such that we have $\det\left(\mathbf{B}_i\right) = \det\left(\mathbf{B}_{i,-j}\right) \det\left(1 + \gamma_k \boldsymbol{\Phi}_{i,j}^T \mathbf{B}_{i,-j}^{-1} \boldsymbol{\Phi}_{i,j}\right)$. It can be verified via applying the matrix inversion lemma that the inverse of $\mathbf{B}_i$ is equal to $\mathbf{B}_i^{-1} = \mathbf{B}_{i,-j}^{-1} - \gamma_j \frac{\mathbf{B}_{i,-j}^{-1} \boldsymbol{\Phi}_{i,j} \boldsymbol{\Phi}_{i,j}^T \mathbf{B}_{i,-j}^{-1}}{1 + \gamma_j \boldsymbol{\Phi}_{i,j}^T \mathbf{B}_{i,-j}^{-1} \boldsymbol{\Phi}_{i,j}}$. With the above notations, we are able to introduce $\mathcal{L}_0\left(\boldsymbol{\gamma}\right)$ that collects the terms relating to $\boldsymbol{\gamma}$ in $\mathcal{L}\left(\boldsymbol{\gamma}, \lambda, \nu\right)$ in Equation 22, which is defined as

$$
\begin{aligned}
&\mathcal{L}_0(\boldsymbol{\gamma}) \\
&\triangleq -\frac{1}{2}\sum_{i=1}^{L}\Bigg[(M_i+2a)\ln\left(\mathbf{y}_i^T\mathbf{B}_i^{-1}\mathbf{y}_i+2b\right) \\
&\quad +\ln\left(\det\left(\mathbf{B}_i\right)\right)+\lambda\sum_{j=1}^{N}\gamma_j\Bigg] \\
&= -\frac{1}{2}\sum_{i=1}^{L}\Bigg[(M_i+2a)\ln\left(\mathbf{y}_i^T\mathbf{B}_{i,-j}^{-1}\mathbf{y}_i+2b\right) \\
&\quad +\ln\left(\det\left(\mathbf{B}_{i,-j}\right)\right)+\lambda\sum_{k=1(\neq j)}^{N}\gamma_k\Bigg] \\
&\quad -\frac{1}{2}\sum_{i=1}^{L}\Bigg[(M_i+2a)\ln\left(1-\frac{\gamma_j q_{i,j}^2/g_{i,j}}{1+\gamma_j s_{i,j}}\right) \\
&\quad +\ln\left(1+\gamma_j s_{i,j}\right)+\lambda\gamma_j\Bigg] \\
&= \mathcal{L}_0\left(\gamma_{-j}\right)+l_0\left(\gamma_j\right).
\end{aligned}
\tag{27}
$$

Here, $\boldsymbol{\gamma}_{-j}$ is $\boldsymbol{\gamma}$ with γ_j removed, $s_{i,j} \triangleq \boldsymbol{\Phi}_{i,j}^T\mathbf{B}_{i,-j}^{-1}\boldsymbol{\Phi}_{i,j}$, $q_{i,j} \triangleq \boldsymbol{\Phi}_{i,j}^T\mathbf{B}_{i,-j}^{-1}\mathbf{y}_i$, and $g_{i,j} \triangleq \mathbf{y}_i^T\mathbf{B}_{i,-j}^{-1}\mathbf{y}_i + 2b$.

Differentiating $\mathcal{L}_0(\boldsymbol{\gamma})$ with respect to γ_j and setting the result to zero, we obtain

$$
\begin{aligned}
&\frac{d\mathcal{L}_0(\boldsymbol{\gamma})}{dr_j} \\
&= \frac{dl_0(\gamma_j)}{dr_j} \\
&= -\frac{1}{2}\sum_{i=1}^{L}\Bigg[\frac{s_{i,j}+\lambda-(M_i+2a)\frac{q_{i,j}^2}{g_{i,j}}}{\left[1+\gamma_j\left(s_{i,j}-q_{i.j}^2/g_{i,j}\right)\right]\left(1+\gamma_j s_{i,j}\right)} \\
&\quad +\frac{\gamma_j^2\lambda s_{i,j}\left(s_{i,j}-\frac{q_{i.j}^2}{g_{i,j}}\right)+\gamma_j\left[\lambda s_{i,j}+\left(s_{i,j}+\lambda\right)\left(s_{i,j}-\frac{q_{i.j}^2}{g_{i,j}}\right)\right]}{\left[1+\gamma_j\left(s_{i,j}-q_{i.j}^2/g_{i,j}\right)\right]\left(1+\gamma_j s_{i,j}\right)}\Bigg] \\
&= 0.
\end{aligned}
\tag{28}
$$

Dividing both sides with γ_i^2, we can transform Equation 28 into

$$
\begin{aligned}
&-\frac{1}{2}\sum_{i=1}^{L}\Bigg[\frac{\lambda s_{i,j}\left(s_{i,j}-\frac{q_{i.j}^2}{g_{i,j}}\right)}{\left(\gamma_j^{-1}+s_{i,j}-q_{i.j}^2/g_{i,j}\right)\left(\alpha_j+s_{i,j}\right)} \\
&\quad +\frac{\gamma_j^{-2}\left[s_{i,j}+\lambda-(M_i+2a)\frac{q_{i.j}^2}{g_{i,j}}\right]+\gamma_j^{-1}\left[\lambda s_{i,j}+\left(s_{i,j}+\lambda\right)\left(s_{i,j}-\frac{q_{i,j}^2}{g_{i,j}}\right)\right]}{\left(\gamma_j^{-1}+s_{i,j}-q_{i.j}^2/g_{i,j}\right)\left(\gamma_j^{-1}+s_{i,j}\right)}\Bigg] \\
&= 0.
\end{aligned}
\tag{29}
$$

Applying the approximation $s_{i,j} \gg 1/\gamma_j$, which is generally valid numerically (e.g., typically we have $s_{i,j} > 20/\gamma_j$ [7]), simplifies the denominator of Equation 29 into $\left(s_{i,j}-q_{i.j}^2/g_{i,j}\right)s_{i,j}$. Meanwhile, let $A_0 \triangleq \sum_{i=1}^{L}\frac{s_{i,j}+\lambda-(M_i+2a)q_{i.j}^2/g_{i,j}}{\left(s_{i,j}-q_{i.j}^2/g_{i,j}\right)s_{i,j}}$, $B_0 \triangleq \sum_{i=1}^{L}\frac{\lambda s_{i,j}+\left(s_{i,j}+\lambda\right)\left(s_{i,j}-q_{i.j}^2/g_{i,j}\right)}{\left(s_{i,j}-q_{i.j}^2/g_{i,j}\right)s_{i,j}}$, and $C_0 \triangleq L\lambda$, and as a result, Equation 29 becomes

$$
-1/2\left(\gamma_j^{-2}A_0+\gamma_j^{-1}B_0+C_0\right)=0. \tag{30}
$$

The approximate solution of γ_i^{-1} from Equation 30 has the form

$$
\gamma_j^{-1} \approx \frac{-B_0 \pm \sqrt{\Delta_0}}{2A_0} \tag{31}
$$

where $\Delta_0 = B_0^2 - 4A_0C_0$ and $C_0 \geq 0$.

As shown in Appendix 1, on the basis of the fact that $\gamma_i \geq 0$, the estimate from Equation 31 can only take two possible values, i.e.,

$$
\gamma_j^{-1} \approx \frac{-B_0 - \sqrt{\Delta_0}}{2A_0}, \qquad A_0 < 0 \tag{32}
$$

$$
\gamma_j^{-1} = \infty, \qquad \text{otherwise.} \tag{33}
$$

When $\gamma_j^{-1} = \infty$, it is equivalent to setting $\boldsymbol{\theta}_{i,j}$ to zero (see Equation 17). This indicates that $\boldsymbol{\Phi}_{i,j}$ can be deleted from the matrix $\boldsymbol{\Phi}_i$. As a result, in contrast to the iterative approach for estimating γ_i and λ (see Section 3.1), the alternative algorithm would have a complexity depending on the number of retained columns in the matrix $\boldsymbol{\Phi}_i$. Moreover, the evaluation of Equations 32 and 33 is relatively easy since computing $s_{i,j}$ and $q_{i,j}$ required in A_0 and B_0 can be achieved via [7]:

$$
\begin{aligned}
&s_{i,j} = \frac{S_{i,j}}{1-\gamma_j S_{i,j}}, q_{i,j} = \frac{Q_{i,j}}{1-\gamma_j S_{i,j}}, \\
&g_{i,j} = G_i + \frac{\gamma_j Q_{i,j}^2}{1-\gamma_j S_{i,j}}
\end{aligned}
\tag{34}
$$

where

$$S_{i,j} = \boldsymbol{\Phi}_{i,j}^T\boldsymbol{\Phi}_{i,j} - \boldsymbol{\Phi}_{i,j}^T\boldsymbol{\Phi}_i\boldsymbol{\Sigma}_i\boldsymbol{\Phi}_i^T\boldsymbol{\Phi}_{i,j} \tag{35}$$

$$Q_{i,j} = \boldsymbol{\Phi}_{i,j}^T\boldsymbol{y}_i - \boldsymbol{\Phi}_{i,j}^T\boldsymbol{\Phi}_i\boldsymbol{\Sigma}_i\boldsymbol{\Phi}_i^T\boldsymbol{y}_i \tag{36}$$

$$G_i = \boldsymbol{y}_i^T\boldsymbol{y}_i - \boldsymbol{y}_i^T\boldsymbol{\Phi}_i\boldsymbol{\Sigma}_i\boldsymbol{\Phi}_i^T\boldsymbol{y}_i + 2b. \tag{37}$$

We summarize the procedure of the fast algorithm in Algorithm 1. The convergence criterion there is

$$\frac{\left|\Delta\mathcal{L}_0\left(\boldsymbol{\gamma}^k\right) - \Delta\mathcal{L}_0\left(\boldsymbol{\gamma}^{k-1}\right)\right|}{\left|\max\left(\Delta\mathcal{L}_0\left(\boldsymbol{\gamma}\right)\right) - \Delta\mathcal{L}_0\left(\boldsymbol{\gamma}^k\right)\right|} < \text{thresh} \tag{38}$$

where $\Delta\mathcal{L}_0\left(\boldsymbol{\gamma}^k\right)$ is the increment of $\mathcal{L}_0\left(\boldsymbol{\gamma}\right)$ in the kth iteration and *thresh* denotes a pre-specified threshold value. To improve the convergence speed, in step 5 of Algorithm 1, we select the γ_j^k that leads to the largest increase in $\mathcal{L}_0\left(\boldsymbol{\gamma}\right)$. Other steps in the algorithm, including updating $\boldsymbol{\mu}_i$, $\boldsymbol{\Sigma}_i$, $s_{i,j}$, $q_{i,j}$, and $g_{i,j}$ in steps 10 to 11 and changing the model as in steps 6 to 8, are the same as those detailed in 6 of [7].

Algorithm 1 FAST LMCS

```
1  Inputs: $\boldsymbol{\Phi} = [\boldsymbol{\Phi}_1, \ldots, \boldsymbol{\Phi}_L]$, $\mathbf{Y} = [\boldsymbol{y}_1, \ldots, \boldsymbol{y}_L]$, thresh
2  Outputs: $\boldsymbol{\gamma} = [\gamma_1, \ldots, \gamma_N]^T$, $\lambda$
3  Initialize $\gamma_j = 0, j = 1, \ldots, N$, and $\lambda = 0$. Set $k = 0$
4  while convergence criterion (38) not met Do
5      Select a particular $\gamma_j^k$ out of $\boldsymbol{\gamma}^k = \left[\gamma_1^k, \gamma_2^k, \ldots, \gamma_N^k\right]^T$
6      if $A_0 < 0$ and $\gamma_j^k = 0$, then add $\gamma_j$ to the model
7      else if $A_0 < 0$ and $\gamma_j^k > 0$, then find $\gamma_j^{k+1}$ using (32)
8      else if $A_0 > 0$, then prune $\gamma_j$ and set $\gamma_j^{k+1} = 0$
9      end if
10     Update $\boldsymbol{\mu}_i$ and $\boldsymbol{\Sigma}_i$
11     Update $s_{i,j}$, $q_{i,j}$ and $g_{i,j}$
12     Update $\lambda$ and $\nu$ using (25) and (26)
13     $k = k + 1$
14 end while
```

Before the end of this section, we shall illustrate the relationship between the MCS algorithm and the newly proposed LMCS technique, in order to gain more insights. Within the MCS framework, the elements γ_j in the information sharing vector $\boldsymbol{\gamma}$ are found via [7]:

$$\begin{aligned}\gamma_j^{\text{MCS}} = \arg\max_{\gamma_j} \Bigg\{ -\frac{1}{2}\sum_{i=1}^{L} \Bigg[& \ln\left(1 + \gamma_j s_{i,j}\right) \\ & + (M_i + 2a) \times \ln\left(1 - \frac{\gamma_j q_{i,j}^2/g_{i,j}}{1 + \gamma_j s_{i,j}}\right)\Bigg]\Bigg\}.\end{aligned} \tag{39}$$

On the other hand, from Equation 27, we have LMCS that obtains the estimate of γ_j through

$$\begin{aligned}\gamma_j^{\text{LMCS}} = \arg\max_{\gamma_j} \Bigg\{ -\frac{1}{2}\sum_{i=1}^{L} \Bigg[& \ln\left(1 + \gamma_j s_{i,j}\right) \\ & + (M_i + 2a) \times \ln\left(1 - \frac{\gamma_j q_{i,j}^2/g_{i,j}}{1 + \gamma_j s_{i,j}}\right) + \lambda\gamma_j\Bigg]\Bigg\}.\end{aligned} \tag{40}$$

Clearly, LMCS would reduce to MCS if $\lambda = 0$. This is somewhat expected from the comparison presented at the end of Section 2, where we show that, compared with MCS, LMCS introduces another layer of prior information embedded in the parameter λ. When $\lambda = 0$, we can verify that $A_0 = \sum_{i=1}^{L} \frac{s_{i,j} - (M_i + 2a)q_{i,j}^2/g_{i,j}}{\left(s_{i,j} - q_{i,j}^2/g_{i,j}\right)s_{i,j}}$, $B_0 = L$, and $C_0 = 0$. As a result, Equations 32 and 33 would become

$$\begin{aligned}\gamma_j^{-1} &\approx \frac{L}{\sum_{i=1}^{L} \frac{(M_i+2a)q_{i,j}^2/g_{i,j} - s_{i,j}}{\left(s_{i,j} - q_{i,j}^2/g_{i,j}\right)s_{i,j}}}, \\ &\text{if} \quad \sum_{i=1}^{L} \frac{(M_i + 2a)\, q_{i,j}^2/g_{i,j} - s_{i,j}}{\left(s_{i,j} - q_{i,j}^2/g_{i,j}\right)s_{i,j}} > 0\end{aligned} \tag{41}$$

$$\gamma_j^{-1} = \infty, \quad \text{otherwise} \tag{42}$$

which are identical to the approximate solutions to Equation 39 established in [7] (see Equations 39 and 40 in [7]). This corroborates the validity of the Bayesian derivations that lead to LMCS.

4 MDL-based task classification and signal reconstruction

The MCS algorithm and the newly proposed LMCS method both assume that the original signals of the L CS tasks are statistically correlated. In other words, the original signals belong to the same cluster or group, from the viewpoint of signal classification. When the above assumption is not fulfilled, the signal reconstruction performance of MCS and LMCS would be degraded. We shall develop in this section novel signal classification and recovery algorithms on the basis of the MDL principle. The new methods are referred to as MDL-MCS or MDL-LMCS so as to reflect the fact that we augment the MCS and LMCS techniques with MDL. We start this section with the theoretical derivation of the MDL-based classification for MCS and LMCS.

4.1 MDL-based classification

This subsection presents the basic MDL-based task classification framework. With MDL, the best model out of a family of competing statistical models for a given data is the one that yields the minimum description length for the data. Let $\mathbf{Y}=\left\{\boldsymbol{y}_1,\ldots,\boldsymbol{y}_L\right\}$ be the set collecting the compressive measurements of the L CS tasks in consideration and $\boldsymbol{\iota}=[\iota_1,\ldots,\iota_L]$ be the partition of Y into K clusters, where $\iota_i=k$ means that $\boldsymbol{y}_i$ belongs to the kth cluster, $i=1,\ldots,L$, and $k=1,\ldots,K$. Assuming statistical independence among signals from two different clusters, we can express the likelihood function of Y as

$$p\left(\mathbf{Y}\left|\mathbf{D}\right.,\boldsymbol{\iota}\right)=\prod_{k=1}^{K}p_k\left(\mathbf{Y}_k\left|\boldsymbol{d}_k\right.\right) \tag{43}$$

where $\mathbf{D}=\{\boldsymbol{d}_1,\ldots,\boldsymbol{d}_K\}$ is the set of model parameters, $\boldsymbol{d}_k$ is the model parameter vector of the model for the kth cluster, $\mathbf{Y}_k$ contains the compressive measurements of the CS tasks in the kth cluster, and $p_k\left(\mathbf{Y}_k\left|\boldsymbol{d}_k\right.\right)$ represents the likelihood function of $\mathbf{Y}_k$. The description length of $\mathbf{Y}$ under the model set $\boldsymbol{D}$ is then

$$DL\left(\mathbf{Y},K\right)=DL\left(\mathbf{Y}\left|\mathbf{D},\boldsymbol{\iota}\right.\right)+DL\left(\mathbf{D},\boldsymbol{\iota}\right) \tag{44}$$

where $DL\left(\mathbf{Y}\left|\mathbf{D}\right.,\boldsymbol{\iota}\right)=-\log_2 p\left(\left[\mathbf{Y}\left|\mathbf{D}\right.,\boldsymbol{\iota}\right]_\delta\right)$ measures the goodness of fit between the data and the model. Under the assumption that the model parameter set $\mathbf{D}$ and the CS task partition $\boldsymbol{\iota}$ are statistically independent, we have $DL\left(\mathbf{D},\boldsymbol{\iota}\right)=-\log_2 p\left([\mathbf{D}]_\delta\right)-\log_2 p\left([\boldsymbol{\iota}]_\delta\right)$, and it acts as a penalty function measuring the model complexity. The notation $[\cdot]_\delta$ denotes elementwise quantization with precision δ. With sufficient quantization precision, we have $p\left(\left[\mathbf{Y}\left|\mathbf{D}\right.,\boldsymbol{\iota}\right]_\delta\right)\approx p\left(\mathbf{Y}\left|\mathbf{D}\right.,\boldsymbol{\iota}\right)\delta^{S_{\mathbf{Y}}}$, $p\left([\mathbf{D}]_\delta\right)\approx p\left(\mathbf{D}\right)\delta^{S_{\mathbf{D}}}$, and $p\left([\boldsymbol{\iota}]_\delta\right)\approx p\left(\boldsymbol{\iota}\right)\delta^{S_{\boldsymbol{\iota}}}$ [20]. Here, $p\left(\mathbf{D}\right)$ and $p\left(\boldsymbol{\iota}\right)$ are the priors of $\mathbf{D}$ and $\boldsymbol{\iota}$. $S_{\mathbf{Y}}$, $S_{\mathbf{D}}$, and $S_{\boldsymbol{\iota}}$ denote the numbers of elements in $\mathbf{Y}$, $\mathbf{D}$, and $\boldsymbol{\iota}$. As a result, the description length of $\mathbf{Y}$ can be rewritten as

$$\begin{aligned}DL\left(\mathbf{Y},K\right)=&-\log_2 p\left(\left[\mathbf{Y}\left|\mathbf{D}\right.,\boldsymbol{\iota}\right]_\delta\right)\\&-\log_2 p\left([\mathbf{D}]_\delta\right)-\log_2 p\left([\boldsymbol{\iota}]_\delta\right).\end{aligned} \tag{45}$$

We proceed to evaluate Equation 45 for the cases of LMCS and MCS sequentially. In particular, as shown in Appendix 2, we have that for LMCS,

$$\begin{aligned}&-\log_2 p\left(\left[\mathbf{Y}\left|\mathbf{D}^{\mathrm{LMCS}}\right.,\boldsymbol{\iota}\right]_\delta\right)-\log_2 p\left(\left[\mathbf{D}^{\mathrm{LMCS}}\right]_\delta\right)\\
&\approx\frac{1}{2}\sum_{k=1}^{K}\sum_{i=1}^{L_k}\left[\log_2\left(\det\left(\mathbf{B}_i^{(k)}\right)\right)\right.\\
&\quad\left.+\left(M_i^{(k)}+2a\right)\log_2\left(\left(\boldsymbol{y}_i^{(k)}\right)^T\left(\mathbf{B}_i^{(k)}\right)^{-1}\boldsymbol{y}_i^{(k)}+2b\right)\right]\\
&\quad-\frac{1}{2}\sum_{k=1}^{K}\sum_{i=1}^{L_k}\left[2\log_2\frac{2b^a\Gamma\left(M_i^{(k)}/2+a\right)}{\Gamma\left(a\right)}-M_i^{(k)}\log_2 2\pi\right.\\
&\quad\left.+2M_i^{(k)}\log_2\delta\right]-\frac{1}{2}\sum_{k=1}^{K}\left[2N\log_2\frac{\lambda^{(k)}}{2}\right.\\
&\quad-\left(\lambda^{(k)}\sum_{j=1}^{N}\gamma_j^{(k)}+\nu^{(k)}\lambda^{(k)}\right)\log_2 e+\nu^{(k)}\log_2\frac{\nu^{(k)}}{2}\\
&\quad\left.-2\log_2\Gamma\left(\nu^{(k)}/2\right)+\left(\nu^{(k)}-2\right)\log_2\lambda^{(k)}\right]\\
&\quad-K\left(N+1\right)\log_2\delta\end{aligned} \tag{46}$$

where $\mathbf{D}^{\mathrm{LMCS}}=\left\{\boldsymbol{d}_k^{\mathrm{LMCS}}\right\}$, $k=1,\ldots,K$, is the set of the model parameters in LMCS, $\boldsymbol{d}_k^{\mathrm{LMCS}}=\left\{\boldsymbol{\gamma}^{(k)},\lambda^{(k)}\right\}$ contains the information sharing parameters of the kth cluster, $\mathbf{B}_i^{(k)}=\mathbf{I}^{(k)}+\boldsymbol{\Phi}_i^{(k)}\left(\boldsymbol{\Gamma}_0^{(k)}\right)^{-1}\left(\boldsymbol{\Phi}_i^{(k)}\right)^T$, $\boldsymbol{\Gamma}_0^{(k)}=\mathrm{diag}(1/\gamma_1^{(k)},\ldots,1/\gamma_N^{(k)})$, and L_k is the number of tasks in the kth cluster. Other variables are the same as those in Equation 22.

For MCS, according to Equation 30 in [7], we have

$$\begin{aligned}&-\log_2 p\left(\left[\mathbf{Y}\left|\mathbf{D}^{\mathrm{MCS}}\right.,\boldsymbol{\iota}\right]_\delta\right)-\log_2 p\left(\left[\mathbf{D}^{\mathrm{MCS}}\right]_\delta\right)\\
&=-\sum_{k=1}^{K}\sum_{i=1}^{L_k}\log_2 p\left(\left[\boldsymbol{y}_i^{(k)}\left|\boldsymbol{\alpha}^{(k)}\right.\right]_\delta\right)-\log_2 p\left(\left[\mathbf{D}^{\mathrm{MCS}}\right]_\delta\right)\\
&\approx-\sum_{k=1}^{K}\sum_{i=1}^{L_k}\left[\log_2 p\left(\boldsymbol{y}_i^{(k)}\left|\boldsymbol{\alpha}^{(k)}\right.\right)+\log_2\delta^{M_i^{(k)}}\right]\\
&\quad-\sum_{k=1}^{K}\left[\log_2 p\left(\boldsymbol{\alpha}^{(k)}\right)+\log_2\delta^{N}\right]\\
&=\frac{1}{2}\sum_{k=1}^{K}\sum_{i=1}^{L_k}\left[\log_2\left(\det\left(\mathbf{C}_i^{(k)}\right)\right)\right.\\
&\quad\left.+\left(M_i+2a\right)\log_2\left(\left(\boldsymbol{y}_i^{(k)}\right)^T\left(\mathbf{C}_i^{(k)}\right)^{-1}\left(\boldsymbol{y}_i^{(k)}\right)+2b\right)\right]\\
&\quad-\frac{1}{2}\sum_{k=1}^{K}\sum_{i=1}^{L_k}\left[2\log_2\frac{2b^a\Gamma\left(M_i^{(k)}/2+a\right)}{\Gamma\left(a\right)}-M_i^{(k)}\log_2 2\pi\right.\\
&\quad\left.+2M_i^{(k)}\log_2\delta\right]-KN\log_2\delta\end{aligned} \tag{47}$$

where $\mathbf{D}^{\text{MCS}} = \left\{ \boldsymbol{d}_k^{\text{MCS}} \right\}$ is the set of model parameters for MCS, $\boldsymbol{d}_k^{\text{MCS}} = \left\{ \boldsymbol{\alpha}_{\text{MCS}}^{(k)} \right\}$, $\boldsymbol{\alpha}_{\text{MCS}}^{(k)}$ is the information sharing vector of cluster k, $\mathbf{C}_i^{(k)} = \mathbf{I}^{(k)} + \boldsymbol{\Phi}_i^{(k)} \left(\mathbf{A}_{\text{MCS}}^{(k)} \right)^{-1} \left(\boldsymbol{\Phi}_i^{(k)} \right)^T$, and $\mathbf{A}_{\text{MCS}}^{(k)} = \text{diag} \left(\boldsymbol{\alpha}_{\text{MCS}}^{(k)} \right)$. In MCS, $\mathbf{A}_{\text{MCS}}^{(k)}$ is distributed uniformly, so $-\log_2 p \left(\mathbf{D}^{\text{MCS}} \right)$ would be a constant (see Section 2.1).

We now compute $-\log_2 p\,(\boldsymbol{\iota})$ to complete the evaluation of Equation 45 for LMCS and MCS. Let $n\,(L, \boldsymbol{\iota})$ be the number of different ways to partition L tasks into K groups with each group having L_k CS tasks and $\sum_{k=1}^{K} L_k = L$. It can be verified that $n\,(L, \boldsymbol{\iota})$ is equal to

$$n\,(L, \boldsymbol{\iota}) = \frac{C_L^{L_1} C_{L-L_1}^{L_2} \cdots C_{L-L_1-\cdots-L_{K-2}}^{L_{K-1}}}{(K-1)!\,m_1!\,m_2! \ldots m_L!}. \tag{48}$$

The numerator represents the number of different partitions if we generate them by taking sequentially L_k tasks out of the L CS tasks while the denominator removes the partitions produced by simply swapping the tasks within a cluster without changing the clustering structure. Assuming that the $\boldsymbol{\iota}$ has the prior of a uniform distribution, we have

$$\begin{aligned} -\log_2 p\,([\boldsymbol{\iota}]_\delta) &\approx -\log_2 p\,(\boldsymbol{\iota}) - \log_2 \delta^L \\ &= -\log_2 \frac{1}{n\,(L, \boldsymbol{\iota})} - L \log_2 \delta \\ &= \log_2 n\,(L, \boldsymbol{\iota}) - L \log_2 \delta. \end{aligned} \tag{49}$$

Putting Equation 4) together with Equation 46 or 47 back to Equation 45 completes the description length computation for the compressive measurement set $\mathbf{Y}$ of the L CS tasks under LMCS or MCS. Given a quantization precision δ, the MDL criterion finds the optimal number of clusters K via

$$K = \arg \min_{1 \leq K \leq L} DL\,(\mathbf{Y}, K)\,. \tag{50}$$

4.2 MDL-LMCS and MDL-MCS

Solving Equation 50 directly may be computationally prohibitive since it requires calculating the description length of $\mathbf{Y}$ for all possible clustering structures. To address this difficulty, we shall propose the new MDL-LMCS and MDL-MCS algorithms for classifying the CS tasks and recovering all original signals in a joint and iterative manner. The algorithm flow is summarized in Algorithm 2. It takes as its input the sets $\mathbf{Y}$ and $\boldsymbol{\Phi}$ that collect the compressive measurement vectors and the measurement matrices in the L CS tasks, respectively.

Since the tasks have not been classified at the beginning, the algorithm considers that they belong to a single cluster ***clust***$\{1\} = \{\mathbf{Y}, \boldsymbol{\Phi}\}$, and as a result, it sets K, the number of obtained clusters, to be 1, and *num*, the number of unclassified tasks, to be L. The algorithm also initializes $\widehat{\mathbf{Y}}$ and $\widehat{\boldsymbol{\Phi}}$, the sets that collect the compressive measurements and the measurement matrices of the unclassified tasks, as $\widehat{\mathbf{Y}} = \mathbf{Y}$ and $\widehat{\boldsymbol{\Phi}} = \boldsymbol{\Phi}$. Signal reconstruction via LMCS or MCS for MDL-LMCS or MDL-MCS is then performed using $\widehat{\mathbf{Y}}$ and $\widehat{\boldsymbol{\Phi}}$ to obtain the reconstructed signal set $\widehat{\boldsymbol{\Theta}}_1$ and the sharing parameter set $\widehat{\mathbf{D}}_1$. We plug $\widehat{\mathbf{D}}_1$ into Equation 46 or 47 to calculate the total description length (TDL) mdl_1 for all the compressive measurements in $\mathbf{Y}$. This completes the initialization stage of the algorithm.

The proposed algorithm proceeds to classify the L tasks as follows. In the first iteration, it first applies the operation CLASSIFY$(\cdot)$ to form a new cluster $\left\{ \widehat{\mathbf{Y}}_{\min}, \widehat{\boldsymbol{\Phi}}_{\min} \right\}$ from the unclassified task set $\widehat{\mathbf{Y}}$. $\widehat{\mathbf{Y}}_{\min}$ has$L_{\min}$ tasks and their measurement matrices are collected in $\widehat{\boldsymbol{\Phi}}_{\min}$. We remove $\widehat{\mathbf{Y}}_{\min}$ and $\widehat{\boldsymbol{\Phi}}_{\min}$ from $\widehat{\mathbf{Y}}$ and $\widehat{\boldsymbol{\Phi}}$ to update them, while the number of remaining unclassified task becomes num $- L_{\min}$. Now, we have $K = 2$ clusters, ***clust***$\{1\} = \{\hat{\mathbf{Y}}, \hat{\boldsymbol{\Phi}}\}$, and ***clust***$\{2\} = \{\hat{\mathbf{Y}}_{\min}, \hat{\boldsymbol{\Phi}}_{\min}\}$[a]. LMCS or MCS is then applied to both clusters to identify their original signals and sharing parameters. The results are kept in $\widehat{\boldsymbol{\Theta}}_2$ and $\widehat{\mathbf{D}}_2$, the latter of which is substituted into (46) or (47) for MDL-LMCS or MDL-MCS to compute again the TDL of $\mathbf{Y}$, denoted by mdl_2. This completes the processing of iteration 1. We then compare mdl_1 with mdl_2 and if $\text{mdl}_2 < mdl_1$, the algorithm would start its second iteration to continue the task classification, where CLASSIFY$(\cdot)$ will be applied to $\hat{\mathbf{Y}}$ and yield ***clust***$\{3\}$. The above process is repeated until $mdl_K > mdl_{K-1}$ occurs, which implies the appearance of over-fitting. The algorithm finally outputs the clusters available in the $(K-2)$th iteration.

The function CLASSIFY $(\cdot)$ runs as follows. Each time when CLASSIFY $(\cdot)$ is executed, it first selects randomly a task out of the unclassified task set $\widehat{\mathbf{Y}}$. With slight abuse of notation, we denote it as $\boldsymbol{y}_i$. It is paired with every of the remaining tasks in $\widehat{\mathbf{Y}}$, and this yields $num-1$ two-task clusters. In the case of MDL-LMCS, we then apply LMCS to estimate the sharing parameters $\left\{ \boldsymbol{\gamma}^{(t)}, \lambda^{(t)}, \nu^{(t)} \right\}$ of the two-task cluster t, $t = 1, 2, \ldots, num - 1$ and compute the corresponding description length for $\boldsymbol{y}_i$ via

$$
\begin{aligned}
&DL^{(t)}_{\mathrm{LMCS}}\left(\boldsymbol{y}_i\right)\\
&\triangleq -\log_2 p\left(\left[\boldsymbol{y}_i \,\middle|\, \boldsymbol{\gamma}^{(t)}, \lambda^{(t)}\right]_{\delta}\right) - \log_2 p\left(\left[\boldsymbol{\gamma}^{(t)}, \lambda^{(t)}\right]_{\delta}\right)\\
&\approx -\log_2 p\left(\boldsymbol{y}_i \,\middle|\, \boldsymbol{\gamma}^{(t)}, \lambda^{(t)}\right) - \log_2 p\left(\boldsymbol{\gamma}^{(t)}, \lambda^{(t)}\right)\\
&\quad - \log_2 \delta^{M_i^{(t)}} - \log_2 \delta^{N} - \log_2 \delta\\
&= -\log_2 p\left(\boldsymbol{y}_i, \boldsymbol{\gamma}^{(t)}, \lambda^{(t)}\right) - \left(M_i^{(t)} + N + 1\right)\log_2 \delta\\
&= \frac{1}{2}\left[\left(M_i^{(t)} + 2a\right)\log_2\left(\boldsymbol{y}_i^T\left(\mathbf{B}_i^{(t)}\right)^{-1}\boldsymbol{y}_i + 2b\right)\right.\\
&\quad + \log_2\left(\det\left(\mathbf{B}_i^{(t)}\right)\right) - 2N\log_2\frac{\lambda^{(t)}}{2} - \nu^{(t)}\log_2\frac{\nu^{(t)}}{2}\\
&\quad + \left(\lambda^{(t)}\sum_{j=1}^{N}\gamma_j^{(t)} + \nu^{(t)}\lambda^{(t)}\right)\log_2 \mathrm{e} + 2\log_2\Gamma\left(\nu^{(t)}/2\right)\\
&\quad \left. - \left(\nu^{(t)} - 2\right)\log_2\lambda^{(t)}\right] - \log_2\frac{2b^a\Gamma\left(M_i^{(k)}/2 + a\right)}{\Gamma\left(a\right)}\\
&\quad + \frac{M_i^{(k)}}{2}\log_2 2\pi - \left(M_i^{(t)} + N + 1\right)\log_2\delta.
\end{aligned}
\tag{51}
$$

We next perform a grouping operation on the obtained num − 1 description length $DL^{(t)}_{\mathrm{LMCS}}\left(\boldsymbol{y}_i\right)$ to identify those tasks in $\hat{\mathbf{Y}}$ that are likely to correlate well with $\boldsymbol{y}_i$ and should be grouped with $\boldsymbol{y}_i$ in a new cluster $\hat{\mathbf{Y}}_{\min}$ (see Algorithm 2). Recall that each description length indeed corresponds to a task in $\hat{\mathbf{Y}}$ other than $\boldsymbol{y}_i$. The grouping procedure is based on the well-known K-mean technique. The difference here is that before the application of the K-mean, we first compute the algorithmic mean of $DL^{(t)}_{\mathrm{LMCS}}\left(\boldsymbol{y}_i\right)$ and set those above the mean value to be equal to the mean. This is equivalent to excluding the tasks that lead to large value of $DL^{(t)}_{\mathrm{LMCS}}\left(\boldsymbol{y}_i\right)$ when being paired with $\boldsymbol{y}_i$ because they are unlikely to be well correlated with $\boldsymbol{y}_i$. We next apply K-mean to the remaining description length to obtain two groups. The mean description length for both groups are found. The tasks belonging to the group with a smaller mean description length are combined with $\boldsymbol{y}_i$ to produce the output of CLASSIFY(·), $\hat{\mathbf{Y}}_{\min}$.

In the case of MDL-MCS, CLASSIFY(·) is realized in the same manner as described above, except that the description length for $\boldsymbol{y}_i$ is evaluated over every two-task cluster using

$$
\begin{aligned}
&DL^{(t)}_{\mathrm{MCS}}\left(\boldsymbol{y}_i\right)\\
&\triangleq -\log_2 p\left(\left[\boldsymbol{y}_i^{(t)} \,\middle|\, \boldsymbol{\alpha}^{(t)}_{\mathrm{MCS}}\right]_{\delta}\right) - \log_2 p\left(\left[\boldsymbol{\alpha}^{(t)}_{\mathrm{MCS}}\right]_{\delta}\right)\\
&\approx -\log_2 p\left(\boldsymbol{y}_i^{(t)} \,\middle|\, \boldsymbol{\alpha}^{(t)}_{\mathrm{MCS}}\right) - \log_2 p\left(\boldsymbol{\alpha}^{(t)}_{\mathrm{MCS}}\right)\\
&\quad - \log_2 \delta^{M_i^{(t)}} - \log_2 \delta^{N}\\
&= -\log_2 p\left(\boldsymbol{y}_i^{(t)}, \boldsymbol{\alpha}^{(t)}_{\mathrm{MCS}}\right) - \left(M_i^{(t)} + N\right)\log_2\delta\\
&= \frac{1}{2}\left[\left(M_i^{(t)} + 2a\right)\log_2\left(\left(\boldsymbol{y}_i^{(t)}\right)^T\left(\mathbf{C}_i^{(t)}\right)^{-1}\boldsymbol{y}_i^{(t)} + 2b\right)\right.\\
&\quad \left. + \log_2\left(\det\left(\mathbf{C}_i^{(t)}\right)\right)\right] - \log_2\frac{2b^a\Gamma\left(M_i^{(k)}/2 + a\right)}{\Gamma\left(a\right)}\\
&\quad + \frac{M_i^{(k)}}{2}\log_2 2\pi - \left(M_i^{(t)} + N\right)\log_2\delta.
\end{aligned}
\tag{52}
$$

Algorithm 2 MDL-LMCS (or MDL-MCS)

```
Inputs: $\mathbf{Y}, \boldsymbol{\Phi}, L$
Outputs: $K$, $\boldsymbol{clust}$, $\widehat{\boldsymbol{\Theta}}$
Initialize $\widehat{\mathbf{Y}} \leftarrow \{\mathbf{Y}\}$, $\widehat{\boldsymbol{\Phi}} \leftarrow \{\boldsymbol{\Phi}\}$, $num \leftarrow L$, $K \leftarrow 1$, $\boldsymbol{clust}\{1\} \leftarrow \left\{\widehat{\mathbf{Y}}, \widehat{\boldsymbol{\Phi}}\right\}$, $\left\{\widehat{\boldsymbol{\Theta}}_1, \widehat{\mathbf{D}}_1\right\} \leftarrow \mathrm{LMCS}\left(\boldsymbol{clust}\right)$
  (or $\left\{\widehat{\boldsymbol{\Theta}}_1, \widehat{\mathbf{D}_1}\right\} \leftarrow \mathrm{MCS}\left(\boldsymbol{clust}\right)$), $mdl_1 \leftarrow \mathrm{TDL}\left(\boldsymbol{clust}, \widehat{\mathbf{D}}_1\right)$, $\boldsymbol{mdl} = [mdl_1]$
while 1
    $\left\{\widehat{\mathbf{Y}}_{\min}, \widehat{\boldsymbol{\Phi}}_{\min}, L_{\min}\right\} \leftarrow \mathrm{CLASSIFY}\left(\widehat{\mathbf{Y}, num}\right)$
    $\widehat{\mathbf{Y}} \leftarrow \mathbf{Y} - \widehat{\mathbf{Y}}_{\min}$, $\widehat{\boldsymbol{\Phi}} \leftarrow \widehat{\boldsymbol{\Phi}} - \widehat{\boldsymbol{\Phi}}_{\min}$, $num \leftarrow num - L_{\min}$, $K \leftarrow K + 1$, $\boldsymbol{clust}\{K\} \leftarrow \left\{\widehat{\mathbf{Y}}_{\min}, \widehat{\boldsymbol{\Phi}}_{\min}\right\}$,
    $\boldsymbol{clust}\{1\} \leftarrow \left\{\widehat{\mathbf{Y}}, \widehat{\boldsymbol{\Phi}}\right\}$
    $\left\{\widehat{\boldsymbol{\Theta}}_K, \widehat{\mathbf{D}}_K\right\} \leftarrow \mathrm{LMCS}\left(\boldsymbol{clust}\right)$ (or $\left\{\widehat{\boldsymbol{\Theta}}_{\boldsymbol{K}}, \widehat{\mathbf{D}}_{\boldsymbol{K}}\right\} \leftarrow \mathrm{MCS}\left(\boldsymbol{clust}\right)$), $mdl_K \leftarrow \mathrm{TDL}\left(\boldsymbol{clust}, \widehat{\mathbf{D}}_K\right)$
    if $mdl_K > mdl_{K-1}$, or $mdl_K < mdl_{K-1}$ and $mdl_{K-1} - mdl_K < \epsilon\left(mdl_{K-2} - mdl_{K-1}\right)$, where $K > 2$
      and $\epsilon$ is a small constant, for instance, $\epsilon = 0.2$, then $\boldsymbol{clust}\{K-1\} \leftarrow \left\{\widehat{\mathbf{Y}} + \widehat{\mathbf{Y}}_{\min}, \widehat{\boldsymbol{\Phi}} + \widehat{\boldsymbol{\Phi}}_{\min}\right\}$,
      $\boldsymbol{clust}\{K\} \leftarrow []$, $\widehat{\boldsymbol{\Theta}} \leftarrow \widehat{\boldsymbol{\Theta}}_{K-1}$, $K \leftarrow K - 1$, and terminate the algorithm
    else $\boldsymbol{mdl} = [\boldsymbol{mdl}, mdl_K]$
    end if
end while
```

4.3 Implementation aspect

The development of MDL-LMCS and MDL-MCS presented in the previous subsection implicitly assumes that the quantization precision δ is known *a priori*. Nevertheless, in an ideal case, δ should be determined jointly with the optimal number of clusters K through minimizing the right-hand side of Equation 50 with respect to δ and K.

We shall follow the approach similar to the one adopted in [20] to determine the quantization precision. First, it can be verified that the value of δ would have no impact on locating the unclassified tasks that are correlated with a randomly selected one if the compressive measurement vectors of all the tasks have the same dimension. This is because, in this case, the term depending on δ in Equations 51 and 52 would be the same for any value of t. As a result, δ will affect the task classification performance via Equations 46 and 47 only, from which it can be seen that a very fine quantization would lead to a smaller number of clusters. This may degrade the signal reconstruction performance as weakly correlated signals may be recovered jointly. A large value of δ would not necessarily improve performance, as in this case, the original signals may tend to be recovered separately. Our experiments suggest that δ be within the range of 0.01 to 0.1, depending on the type of data to be processed. Throughout the experiments in Section 5, we shall fix δ to be 0.1, instead of attempting to optimize it for different experiments.

5 Simulations

Monte Carlo (MC) simulations using synthetic data and images are performed to illustrate the performance of the LMCS algorithm developed in Section 3 and the MDL-augmented MCS algorithms, namely, the MDL-LMCS and MDL-MCS techniques presented in Section 4.

5.1 Synthetic signals

In each simulation of this subsection, the length of the original signals of all the CS tasks is fixed at $N = 512$, and we generate two sets of results. One set of results is produced when the non-zero elements of the original signals take binary values ± 1 in a random manner. The other set is generated with the non-zero elements of the original signals being independently drawn from zero-mean Gaussian distribution with unit variance. The elements of the measurement matrix of any CS task, on the other hand, can only be drawn from a Gaussian distribution with zero mean and variance one. Each column of any measurement matrix is normalized to have a unit norm.

For the purpose of comparison, we implement the BCS and MCS techniques developed in [4] and [7]. We shall denote them as ST-BCS and MCS in the figures. Here, ST stands for single task, and it is introduced to highlight that ST-BCS and MCS recover the original signals separately and jointly. We also implement the Laplace prior-based BCS proposed in [6] and denote it as LST-BCS. When implementing the three benchmark algorithms (ST-BCS, MCS, and LST-BCS) and the three proposed methods (LMCS, MDL-LMCS, and MDL-MCS), we always initialize $a = 10^3$ and $b = 1$ so that the noise precision β has the same prior distribution for all the algorithms considered (see Equation 5).

We shall follow the previous works [4,6,7] that proposed the three benchmark methods and use the average normalized signal reconstruction error as the primary performance metric. It is defined as $\frac{1}{L}\sum_{i=1}^{L}\left\|\boldsymbol{\theta}_i - \hat{\boldsymbol{\theta}}_i\right\|_2/\|\boldsymbol{\theta}_i\|_2$, where $\boldsymbol{\theta}_i$ and $\hat{\boldsymbol{\theta}}_i$ are the true and the estimated original signal vectors of the ith CS task. Note that the average normalized signal reconstruction error measures the Euclidean distance between the waveforms of the recovered and the original signals. It is not very informative regarding the quality of the recovered signal supports. Therefore, we shall also include in some experiments performance results of different algorithms in recovering the signal supports, which are quantified by the average incorrect support recovery ratio $\frac{1}{L}\sum_{i=1}^{L}||S(\boldsymbol{\theta}_i) - S(\hat{\boldsymbol{\theta}}_i)||_0/N$. Here, $||\cdot||_0$ denotes the l_0-norm and $S(\mathbf{x})$ sets all the non-zero elements in $\mathbf{x}$ to be 1.

5.1.1 LMCS

We consider the case of $L = 2$ CS tasks as in [7], in order to illustrate the performance of the proposed LMCS technique and the existing methods under a simulation setup already used in the literature. The original signal of each task contains 64 non-zero elements at random locations. Zero-mean Gaussian noise with a standard deviation of 0.01 is added to the two obtained compressive measurement vectors[b].

In the first simulation, we illustrate in Figure 2 the impact of different choices of the parameters λ and ν on the performance of LMCS. The two signals are assumed to have 75% of their non-zero elements overlapped. We realize LMCS with $\lambda = 0$, $\lambda = 1$, $\lambda = 2$, and λ estimated using Equation 25. The results shown are averaged over 200 runs. In particular, Figure 2a,b plots the average signal reconstruction error as a function of the number of compressive measurements for the two cases where the original signals are random binary numbers ± 1 and zero-mean Gaussian random variables with unit variance. The results show that in both cases, the reconstruction error of LMCS gradually improves as the number of compressive measurements increases, and the best performance is obtained when λ is estimated using Equation 25. Moreover, we can see that the LMCS with $\nu = 0$ and ν estimated using Equation 26 yields similar signal reconstruction performance. The underlying reason is that the value of λ estimated jointly with ν is nearly identical to that obtained with $\nu = 0$. This can be better explained

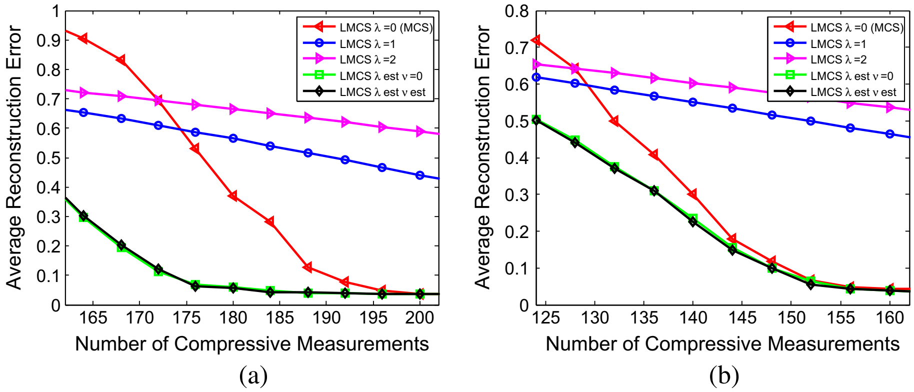

Figure 2 Performance comparison of LMCS with different choices of λ and ν. (a) Binary original signals and **(b)** Gaussian original signals.

as follows. The value of ν, when it is identified together with λ, is generally non-zero but less than one in this simulation. Careful examination of Equation 25 that gives the estimate of λ reveals that the impact of a small non-zero ν on λ is negligible, when the original signal length N is large (in this section, $N = 512$) and the measurement noise level is low, which implies a large value of the noise precision β, and as a result, large values of the hyper-parameters γ_j for original signals having a unit variance (see Equation 17). Therefore, in the remaining simulations, we fix ν at zero when realizing LMCS and MDL-LMCS.

It is worthwhile to point out that rigorously, $\nu = 0$ is a boundary value for the Gamma distribution. As ν approaches 0, the prior distribution of λ would provide vague information on λ as $p(\lambda) \propto 1/\lambda$ (also see Equation 19 in [6]). However, this would not change the fact that Laplace prior is imposed on the original signals, as shown in Equation 12. In other words, LMCS would still outperform MCS because it enhances the sparsity constraints on the non-zero elements of the original signals. This is also supported by the following simulation results (see Figures 3 and 4).

Figure 3 demonstrates the impact of the correlation between the two original signals on the performance of LMCS. It considers the cases when the two original signals have binary non-zero elements, and they have 75% and 50% of their non-zero elements overlapped. Figure 3a,b plots the average signal reconstruction error and the incorrect support recovery ratio of LMCS as a function of the number of compressive measurements. The results shown are averaged over 50 runs. For comparison, we also include in the figures the results from ST-BCS, LST-BCS, and MCS. We can observe from Figure 3a that LMCS and MCS outperform greatly over ST-BCS and LST-BCS due to the utilization of the prior sharing mechanism (see Section 2). The performance of LMCS and MCS improves as the number of the overlapping non-zero elements in the two original signals increases, as expected. More importantly, LMCS exhibits superior performance in terms of a much lower signal reconstruction error over MCS for the two cases where the two original signals have 75% and 50%

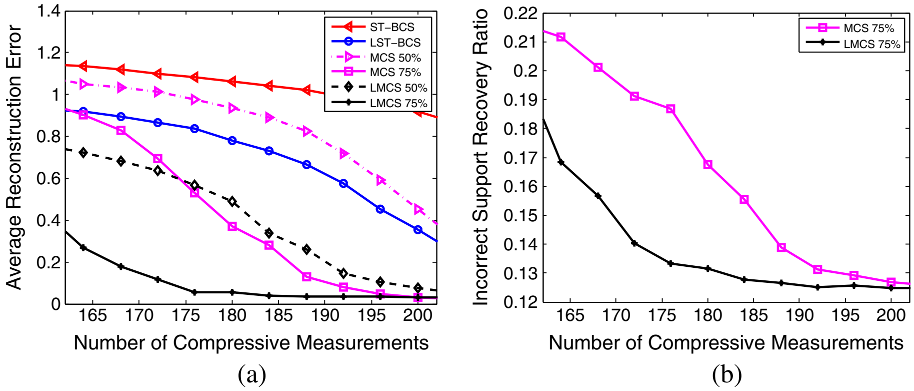

Figure 3 Comparison of ST-BCS, LST-BCS, MCS, and LMCS in reconstructing signals with binary non-zero elements. (a) Average reconstruction error and **(b)** incorrect support recovery ratio.

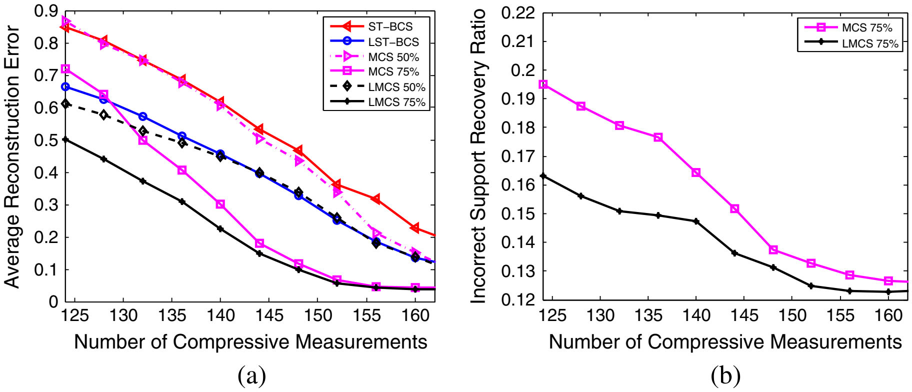

Figure 4 Comparison of ST-BCS, LST-BCS, MCS, and LMCS in reconstructing signals with Gaussian non-zero elements. (a) Average reconstruction error and **(b)** incorrect support recovery ratio.

of their non-zeros overlapped. The performance enhancement mainly comes from the use of Laplace priors on the original signals in LMCS. Compared with MCS, LMCS imposes another layer of prior information on the hyperparameters of the original signals, which makes MCS a special case of LMCS as shown in Equations 39 and 40 at the end of Section 4. As a result, LMCS offers more flexibility in modeling the sparsity of the original signals. This is also corroborated by Figure 3b, where it shows that in the case where the two original signals have 75% of their non-zero elements colocated, LMCS can provide a lower incorrect support recovery ratio and can better recover the sparse signal support.

Figure 4 repeats the simulation experiment in Figure 3, but it considers the case where the two original signals have the non-zero elements drawn from zero-mean Gaussian distribution with unit variance. The obtained observations are similar to those in Figure 3.

5.1.2 MDL-based task classification and signal reconstruction

In this subsection, we present simulation results to illustrate the performance of MDL-MCS and MDL-LMCS developed in Section 4. For the purpose of comparison, we also show the results of the ST-BCS, LST-BCS, MCS, and LMCS methods as well as the DP-MCS technique.

The simulated algorithms are used to recover the original signals of $L = 40$ CS tasks that belong to eight clusters with five tasks each. Every cluster has its own signal template that differs in the signal supports. All the original signals have 64 non-zero components, and their locations are initially chosen so that the correlation between any two original signals from different clusters is zero. Later, we perform the following perturbation to induce slight correlation among clusters. Specifically, in each ensemble run, six non-zero elements in each signal template are selected randomly and set to zero elements, while at the same time, six elements that are zeros in the original template are reset to be non-zeros. In this way, the five signals within the same cluster are highly correlated, but the signals from different clusters have distinct sparsity structures. The simulation results are obtained via averaging over 50 ensemble runs.

In Figure 5a,b, we plot as a function of the number of compressive measurements the binary signal reconstruction error and the correct task classification ratio of the simulated seven algorithms. As we can see from Figure 5a, pretending that the 40 CS tasks belong to the same group and recovering the original signals using LMCS or MCS would lead to a signal reconstruction error even higher than reconstructing the original signals separately via LST-BCS. This clearly demonstrates the impact of incorrect task classification on the signal recovery performance. On the other hand, the proposed MDL-LMCS and MDL-MCS algorithms outperform the DP-MCS technique in terms of lower signal reconstruction error. The performance improvement can be better explained by examining Figure 5b. We can see that the application of the MDL principle to augment LMCS and MCS leads to a greatly improved correct task classification ratio, compared with the DP-MCS technique. With the CS task correctly grouped, MDL-LMCS and MDL-MCS can better recover the original signals of every group.

We repeat the simulation used to generate Figure 5 with the original signals being zero-mean Gaussian random variables with unit variance. The obtained results are summarized in Figure 6. The observations obtained are similar to those in Figure 5.

5.2 Images

In this subsection, we compare the performance of MDL-MCS and MDL-LMCS with that of DP-MCS in recovering 2-D images of random bars. In this experiment, the elements of the measurement matrices of the three

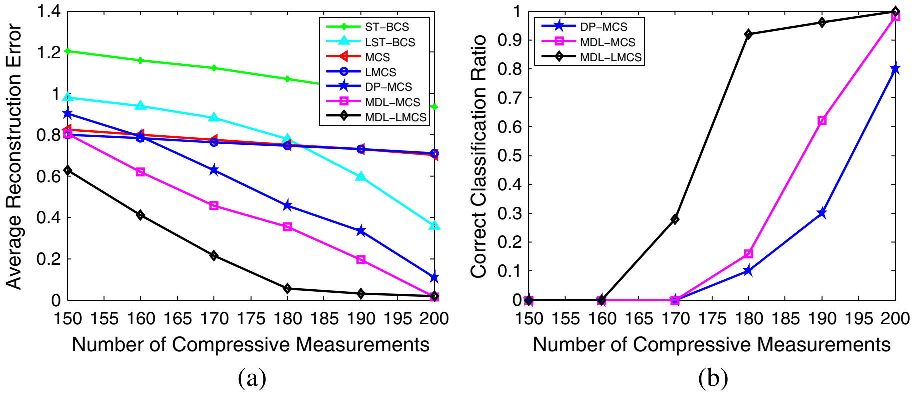

Figure 5 Comparison of signal reconstruction and classification performance for binary signals. (a) Binary signal reconstruction error and **(b)** correct classification ratio.

algorithms in consideration are drawn from a uniform spherical distribution.

Figure 7 summarizes the reconstruction results from a particular run. The first three images in Figure 7a, labeled as tasks 1 to 3, are taken from [7], and they belong to the same cluster. The remaining six images in Figure 7a forms another two clusters, where one cluster consists of tasks 4 to 6 and the other is composed of tasks 7 to 9. These six images are modified from the first three images via permuting randomly the intensities of the rectangles and shifting their positions by distance randomly sampled from a uniform distribution.

All original images have the dimension of 1,024×1,024. Here, we utilize the Haar wavelet expansion with a coarsest scale of 3 and a finest scale of 6. Figure 7a gives the result of the inverse wavelet transform with 4,096 samples, denoted as linear in the figure. This is the best performance achievable by all the CS algorithms considered here. The reconstruction result from DP-MCS is shown in Figure 7b, where we adopted the hybrid CS scheme that compresses the fine-scale coefficients only as in [7] into $M_i = 680$ $(i = 1, \ldots, 9)$ measurements for each task. Figure 7c,d gives the recovery results of MDL-MCS and MDL-LMCS, respectively.

We fix the original images and repeat the above experiment 20 times, each time with independently generated measurement matrices for all the three algorithms. In every run, the image reconstruction error for each task is evaluated and averaged to obtain the normalized image reconstruction error, which is again averaged over 20 runs to yield the average image reconstruction error summarized in Table 1. We also include in Table 1 the correct classification ratio.

The results in Figure 7 and Table 1 show that MDL-LMCS has the best image reconstruction and classification performance, while MDL-MCS yields a better performance than DP-MCS. This is consistent with the observations obtained from Figures 5 and 6.

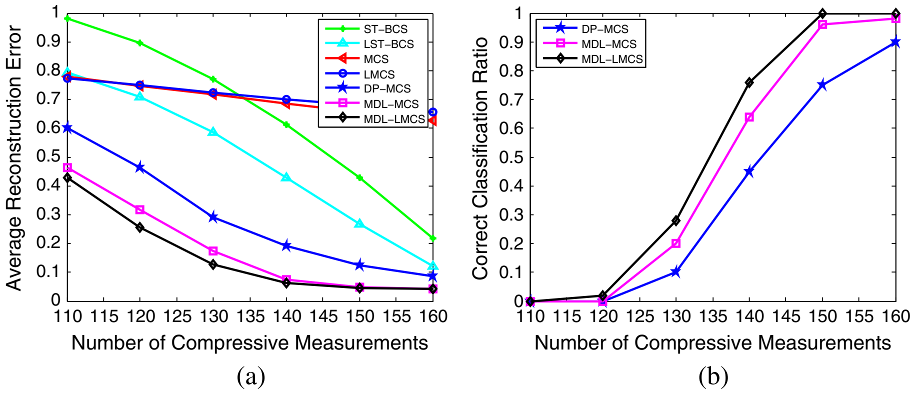

Figure 6 Comparison of signal reconstruction and classification performance for Gaussian signals. (a) Gaussian signal reconstruction error and **(b)** Correct classification ratio.

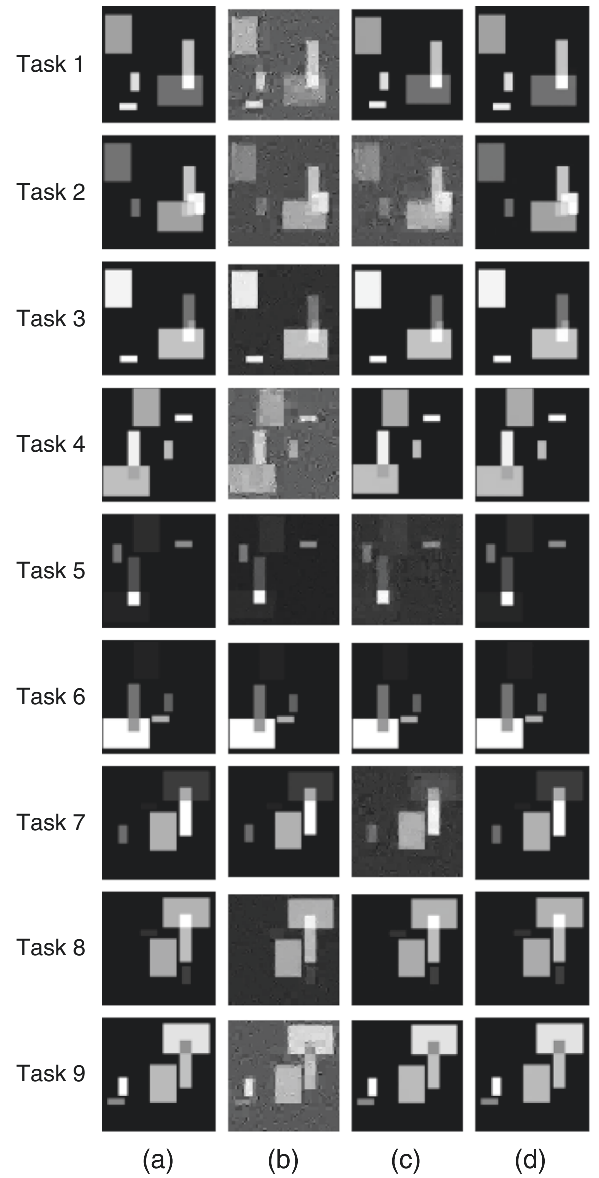

Figure 7 Comparison of DP-MCS, MDL-MCS, and MDL-LMCS in image reconstruction. (a) Linear, **(b)** DP-MCS, **(c)** MDL-MCS, and **(d)** MDL-LMCS.

6 Conclusions

In this paper, we first extended previous works on the Laplace prior-based Bayesian CS to the scenario of multiple CS tasks and developed the LMCS technique. The hierarchical prior model was adopted to impose the Laplace priors, and it was shown that the MCS algorithm is indeed a special case of LMCS. Next, this paper considered the scenario where the multiple CS tasks are from different groups, under which the performance of both MCS and LMCS would be degraded, since they attempt to recover the uncorrelated signals jointly. We proposed the MDL-based MCS techniques, namely, MDL-MCS and MDL-LMCS, which first classify tasks into different groups using the MDL principle and then reconstruct signals of every cluster. Simulations verified the enhanced performance of MDL-MCS and MDL-LMCS in terms of lower signal reconstruction error over the benchmark MCS and DP-MCS techniques as well as single-task CS algorithms.

Table 1 Image reconstruction and classification performance of DP-MCS, MDL-MCS, and MDL-LMCS

	Average reconstruction error	Correct classification ratio
Linear	0.22623	-
DP-MCS	0.27647	0.35
MDL-MCS	0.24511	0.60
MDL-LMCS	0.22642	1.00

Endnotes

[a] It can be easily verified that in our algorithm, K is equal to the iteration index plus one. Besides, ***clust***$\{1\}$ always contains all the unclassified tasks and ***clust***$\{K\}$ is the newest cluster formed in the current iteration.

[b] Our choice of the noise standard deviation of 0.01 is on the same order of the values adopted in the literature. For example, in [6] and [7], the noise standard deviation is set to be 0.03 and 0.005.

Appendix 1

Derivation and analysis of Equations 32 and 33

In this appendix, we shall present the derivation that leads to Equations 32 and 33 and show that it is only a suboptimal solution to the maximization of Equation 27.

Our derivation applies the approximation that $s_{i,j} \gg 1/\gamma_j$, which has been found to be valid numerically [7]. This results in the estimate of γ_j having the functional form given in Equation 30. When $A_0 > 0$, both solutions in Equation 30 would be negative, which violates the requirement that γ_j must be positive. If $A_0 < 0$, only the solution $\gamma_j^{-1} = \left(-B_0 - \sqrt{\Delta_0}\right)/(2A_0)$ is valid. For the case $A_0 = 0$, from Equation 27, γ_j will have the accurate solution $\gamma_j = 0$. This completes the derivation of Equations 32 and 33.

We next show that the solution in Equation 32 and 33 is suboptimal. For this purpose, utilizing the approximation $s_{i,j} \gg 1/\gamma_j$ transforms Equation 28 into

$$\frac{d\mathcal{L}_0(\boldsymbol{\gamma})}{dr_j} = \frac{dl_0(\gamma_j)}{dr_j} \approx -\frac{1}{2}\left(\gamma_j^{-2}A_0 + \gamma_j^{-1}B_0 + C_0\right). \tag{53}$$

We can also obtain easily

$$\begin{aligned}\frac{d^2\mathcal{L}_0(\boldsymbol{\gamma})}{dr_j^2} &\approx \frac{1}{2}\left(2\gamma_j^{-3}A_0 + \gamma_j^{-2}B_0\right)\\ &= \frac{1}{2}\gamma_j^{-2}\left(2\gamma_j^{-1}A_0 + B_0\right).\end{aligned} \tag{54}$$

Substituting Equation 32 into Equation 54 yields

$$\frac{d^2\mathcal{L}_0(\boldsymbol{\gamma})}{dr_j^2} \approx -\frac{1}{2}\gamma_j^{-2}\sqrt{\Delta_0} < 0 \tag{55}$$

This indicates that the solution in Equation 32 is the unique maximizer of the approximated version of Equation 27. However, solving Equation 29 accurately, which is equal to finding all the candidate maximizers for Equation 27, may yield two or more positive estimates of γ_j. Among them, one would be relatively close to the approximate solution in Equation 32. In other words, the approximate solution is within the vicinity of a stationary point of Equation 27, which may only correspond to a local maxima.

Appendix 2

Derivation of Equation 46

To avoid confusion, we use superscript (k) to denote the kth cluster in the following derivation. For mathematical tractability, besides the independence among signals from two different clusters, we also assume the independence among signals within the same cluster. As a result, we have

$$\begin{aligned}&-\log_2 p\left(\left[\mathbf{Y}\,|\mathbf{D}^{\mathrm{LMCS}}, \boldsymbol{\iota}\right]_\delta\right)\\ &= -\sum_{k=1}^{K}\sum_{i=1}^{L_k}\log_2 p\left(\left[\boldsymbol{y}_i^{(k)}\left|\boldsymbol{\gamma}^{(k)}, \lambda^{(k)}\right.\right]_\delta\right)\end{aligned} \tag{56}$$

where L_k is the number of tasks in the kth cluster such that $\sum_{k=1}^{K} L_k = L$, $\mathbf{D}^{\mathrm{LMCS}} = \left\{\boldsymbol{d}_k^{\mathrm{LMCS}}\right\}$, $k = 1, \ldots, K$, and $\boldsymbol{d}_k^{\mathrm{LMCS}} = \left\{\boldsymbol{\gamma}^{(k)}, \lambda^{(k)}\right\}$ contain the information sharing parameters of the kth cluster. Similarly, assuming statistical independence among $\boldsymbol{d}_k^{\mathrm{LMCS}}$, we obtain

$$\begin{aligned}&-\log_2 p\left(\left[\mathbf{D}^{\mathrm{LMCS}}\right]_\delta\right)\\ s &= -\sum_{k=1}^{K}\log_2 p\left(\left[\boldsymbol{\gamma}^{(k)}, \lambda^{(k)}\right]_\delta\right)\\ &= -\sum_{k=1}^{K}\left[\log_2 p\left(\left[\boldsymbol{\gamma}^{(k)}\left|\lambda^{(k)}\right.\right]_\delta\right) + \log_2 p\left(\left[\lambda^{(k)}\right]_\delta\right)\right].\end{aligned} \tag{57}$$

Combining Equations 56 and 57 yields

$$\begin{aligned}&-\log_2 p\left(\left[\mathbf{Y}\,|\mathbf{D}^{\mathrm{LMCS}}, \boldsymbol{\iota}\right]_\delta\right) - \log_2 p\left(\left[\mathbf{D}^{\mathrm{LMCS}}\right]_\delta\right)\\ &\approx -\sum_{k=1}^{K}\sum_{i=1}^{L_k}\left[\log_2\frac{p\left(\boldsymbol{y}_i^{(k)}, \boldsymbol{\gamma}^{(k)}, \lambda^{(k)}\right)}{p\left(\boldsymbol{\gamma}^{(k)}, \lambda^{(k)}\right)} + M_i^{(k)}\log_2\delta\right]\\ &\quad -\sum_{k=1}^{K}\left[\log_2 p\left(\boldsymbol{\gamma}^{(k)}\left|\lambda^{(k)}\right.\right) + \log_2 p\left(\lambda^{(k)}\right) + (N+1)\log_2\delta\right].\end{aligned} \tag{58}$$

From Equation 22, Equation 58 can be rewritten as

$$\begin{aligned}&-\log_2 p\left(\left[\mathbf{Y}\,|\mathbf{D}^{\mathrm{LMCS}}, \boldsymbol{\iota}\right]_\delta\right) - \log_2 p\left(\left[\mathbf{D}^{\mathrm{LMCS}}\right]_\delta\right)\\ &\approx -\sum_{k=1}^{K}\sum_{i=1}^{L_k}\left[\log_2\iint p\left(\boldsymbol{y}_i^{(k)}|\boldsymbol{\theta}_i^{(k)}, \beta\right)p\left(\boldsymbol{\theta}_i^{(k)}|\boldsymbol{\gamma}^{(k)}\right)p(\beta)\,d\boldsymbol{\theta}_i^{(k)}d\beta\right.\\ &\quad\left. + M_i^{(k)}\log_2\delta\right] - \sum_{k=1}^{K}\left[\log_2 p\left(\boldsymbol{\gamma}^{(k)}\left|\lambda^{(k)}\right.\right) + \log_2 p\left(\lambda^{(k)}\right)\right.\\ &\quad\left. + (N+1)\log_2\delta\right].\end{aligned} \tag{59}$$

Carrying out the integration, simplifying and applying some straightforward manipulations give Equation 46.

Competing interests
The authors declare that they have no competing interests.

Acknowledgements
The authors wish to thank the editor and the anonymous reviewers for their constructive suggestions. The authors thank S. Derin Babacan, Shihao Ji, and David Dunson for sharing codes of their algorithms. This work was supported in part by Hunan Provincial Innovation Foundation for Postgraduates under Grant CX2012B019, Fund of Innovation, Graduate School of National University of Defense Technology under grant B120404, and National Natural Science Foundation of China (no. 61304264).

Author details
[1]College of Electronic Science and Engineering, National University of Defense Technology, Deya Road, Changsha 410073, People's Republic of China. [2]School of Internet of Things (IoT) Engineering, Jiangnan University, Lihu Road, Wuxi 214122, People's Republic of China. [3]Key Laboratory of Universal Wireless Communications, Ministry of Education, Beijing University of Posts and Telecommunications, Xitucheng Road, Beijing 100876, People's Republic of China.

Received: 28 April 2013 Accepted: 1 October 2013
Published: 17 October 2013

References

1. Baraniuk R, A lecture on compressive sensing. IEEE Mag. Signal Process. **24**(4), 118–121 (2007)
2. E Candés, J Romberg, T Tao, Robust uncertainty principles: exact signal reconstruction from highly incomplete frequency information. IEEE Trans. Inf. Theory. **52**(2), 489–509 (2006)
3. DL Donoho, Compressed sensing. IEEE Trans. Inf. Theory. **52**(4), 1289–1306 (2006)
4. S Ji, Y Xue, L Carin, Bayesian compressive sensing. IEEE Trans. Signal Process. **56**(6), 2346–2356 (2008)
5. ME Tipping, Sparse Bayesian learning and the relevance vector machine. J. Mach. Learn. Res. **1**, 211–244 (2001)
6. S Babacan, Molina R, A Katsaggelos, Bayesian compressive sensing using Laplace priors. IEEE Trans. Image Process. **19**(1), 53–63 (2010)

7. S Ji, D Dunson, L Carin, Multi-task compressive sensing. IEEE Trans. Signal Process. **57**(1), 92–106 (2009)
8. D Leviatan, VN Temlyakov, Simultaneous approximation by greedy algorithms, Technical report, University of South Carolina (2003)
9. SF Cotter, BD Rao, K Engan, K Kreutz-Delgado, Sparse solutions to linear inverse problems with multiple measurement vectors. IEEE Trans. Signal Process. **53**(7), 2477–2488 (2005)
10. JA Tropp, AC Gilbert, MJ Strauss, Algorithms for simultaneous sparse approximation. Part I: Greedy pursuit. Signal Process. **86**, 572–588 (2006)
11. JA Tropp, Algorithms for simultaneous sparse approximation. Part II: convex relaxation. Signal Process. **86**, 589–602 (2006)
12. D Escoda, L Granai, P Vandergheynst, On the use of a priori information for sparse signal approximations. IEEE Trans. Signal Process. **54**(9), 3468–3482 (2006)
13. D Baron, MF Duarte, Sarvotham S, Wakin M B, Baraniuk R G, An information-theoretic approach to distributed compressed sensing, in *Proceedings of the 43rd Allerton Conference on Communication, Control, and Computing,* Monticello, IL, Sept 2005
14. DP Wipf, BD Rao, An empirical Bayesian strategy for solving the simultaneous sparse approximation problem. IEEE Trans. Signal Process. **55**(7), 3704–3716 (2007)
15. MW Seeger, H Nickisch, Compressed sensing and Bayesian experimental design, in *Proceedings of the 25th International Conference on Machine Learning,* Helsinki, July 2008
16. Y Qi, D Liu, L Carin, D Dunson, Multi-task compressive sensing with Dirichlet process priors, in *Proceedings of the 25th International Conference on Machine Learning,* Helsinki, July 2008
17. J Rissanen, Modeling by shortest data description. Automatica. **14**, 465—471 (1978)
18. J Rissanen, Universal coding, prediction information, estimation. IEEE Trans. Inf. Theory. **30**(4), 629—636 (1984)
19. A Barron, BYu J Rissanen, The minimum description length principle in coding and modeling. IEEE Trans. Inf. Theory. **44**(6), 2743—2760 (998)
20. I Ramirez, G Sapiro, An MDL framework for sparse coding and dictionary learning. IEEE Trans. Signal Process. **60**(6), 2913—2927 (2012)
21. J Liu, SW Gao, ZQ Luo, TN Davidson, JPY Lee, The minimum description length criterion applied to emitter number detection and pulse classification, in *Proceedings of the Ninth IEEE Workshop on Statistical Signal and Array Processing,* Portland, OR, Sept 1998
22. KM Wong, ZQ Luo, J Liu, JPY Lee, Gao S W, Radar emitter classification using intrapulse data. Int. J. Electron. Comm. **12**, 324–332 (1999)
23. J Liu, JPY Lee, L Li, Z Luo, KM Wong, Online clustering algorithms for radar emitter classification. IEEE Trans. Pattern Anal. Mach. Intell. **27**(8), 1185–1196 (2005)
24. T Cover, J Thomas, *Elements of Information Theory*, 2nd edn (Wiley, New York, 2006)
25. R Caruana, Multi-task learning. Mach. Learn. **28**(1), 41–75 (1997)
26. J Baxter, Learning internal representations, in *Proceedings of the Eighth Annual Conference on Computational Learning Theory,* Santa Cruz, CA, July 1995
27. J Baxter, A model of inductive bias learning. J. Artif. Intell. Res. **12**, 149–198 (2000)
28. ND Lawrence, JC Platt, Learning to learn with the informative vector machine, in *Proceedings of the 21st International Conference on Machine Learning,* Banff, Alberta, July 2004, 65
29. K Yu, V Tresp, Schwaighofer A, Learning Gaussian processes from multiple tasks, in *Proc. 22nd Int. Conf. Mach. Learn.* (ICML 22), 2005
30. J Zhang, Z Ghahramani, Y Yang, Learning multiple related tasks using latent independent component analysis, in *Advances in Neural Information Processing Systems (NIPS),* Vancouver, British Columbia, Dec 2006
31. RK Ando, T Zhang, A framework for learning predictive structures from multiple tasks and unlabeled data. J. Mach. Learn. Res. **6**, 1817–1853 (2005)
32. T Evgeniou, CA Micchelli, Pontil M, Learning multiple tasks with kernel methods. J. Mach. Learn. Res. **6**, 615–637 (2005)
33. D Burr, H Doss, A Bayesian semiparametric model for random-effects meta-analysis. J. Amer. Stat. Assoc. **100**, 242–251 (2005)
34. F Dominici, G Parmigiani, R Wolpert, K Reckhow, Combining information from related regressions. J. Agric. Biolog. Environ. Stat. **2**(3), 294–312 (1997)
35. PD Hoff, Nonparametric modeling of hierarchically exchangeable data, Technical report, University of Washington (2003)
36. P Muller, F Quintana, G Rosner, A method for combining inference across related nonparametric Bayesian models. J. R. Stat. Soc. Ser. B. **66**(3), 735–749 (2004)
37. BK Mallick, SG Walker, Combining information from several experiments with nonparametric priors. Biometrika. **84**(3), 697–706 (1997)
38. L Tang, Z Zhou, L Shi, H Yao, ZhangJ, Y Ye, Laplace prior based distributed compressive sensing, in *Proceeding of the 5th International ICST Conference on Communications and Networking in China,* Beijing, Aug 2010
39. KE Themelis, AA Rontogiannis, KD Koutroumbas, A Novel Hierarchical Bayesian Approach for Sparse Semisupervised Hyperspectral Unmixing. IEEE Trans. Signal Process. **60**(2), 585–599 (2012)
40. CM Bishop, *Pattern Recognition and Machine Learning* (Springer-Verlag, New York, 2006)

doi:10.1186/1687-6180-2013-160
Cite this article as: Wang *et al.*: **Enhanced multi-task compressive sensing using Laplace priors and MDL-based task classification.** *EURASIP Journal on Advances in Signal Processing* 2013 **2013**:160.