Noname manuscript No.
(will be inserted by the editor)

Stochastic Greedy Algorithms For Multiple Measurement Vectors

**Jing Qin · Shuang Li · Deanna Needell · Anna Ma ·
Rachel Grotheer · Chenxi Huang · Natalie Durgin**

Received: date / Accepted: date

Abstract Sparse representation of a single measurement vector (SMV) has been explored in a variety of compressive sensing applications. Recently, SMV models have been extended to solve multiple measurement vectors (MMV) problems, where the underlying signal is assumed to have joint sparse structures. To circumvent the NP-hardness of the ℓ_0 minimization problem, many deterministic MMV algorithms solve the convex relaxed models with limited efficiency. In this paper, we develop stochastic greedy algorithms for solving the joint sparse MMV reconstruction problem. In particular, we propose the MMV Stochastic Iterative Hard Thresholding (MStoIHT) and MMV Stochastic Gradient Matching Pursuit (MStoGradMP) algorithms, and we also utilize the mini-batching technique to further improve their performance. Convergence analysis indicates that the proposed algorithms are able to converge faster than

Jing Qin was supported by the faculty start-up fund from Montana State University. Shuang Li was supported by NSF CAREER grant CCF-1149225. Deanna Needell was partially supported by the Alfred P. Sloan Foundation, NSF CAREER #1348721, and NSF BIGDATA #1740325.

Jing Qin
Department of Mathematical Sciences
Montana State University
Bozeman, MT 59717.
E-mail: jing.qin@montana.edu

Shuang Li
Department of Electrical Engineering
Colorado School of Mines
Golden, CO 80401.

Deanna Needell
Department of Mathematics
University of California, Los Angeles
Los Angeles, CA 90095.

Anna Ma
Department of Mathematics
Claremont Graduate University
Claremont, CA 91711.

Rachel Grotheer
Center for Data, Mathematical, and Computational Sciences
Goucher College
Baltimore, MD 21204.

Chenxi Huang
Center for Outcomes Research and Evaluation
Yale University
New Haven, CT 06511.

Natalie Durgin
Spiceworks, Austin, TX, 78746.

their SMV counterparts, i.e., concatenated StoIHT and StoGradMP, under certain conditions. Numerical experiments have illustrated the superior effectiveness of the proposed algorithms over their SMV counterparts.

Keywords Multiple measurement vectors (MMV) · stochastic iterative hard thresholding (StoIHT) · stochastic gradient matching pursuit (StoGradMP) · joint sparse signal recovery

PACS 02.70.-c · 02.60.-x · 87.19.le

Mathematics Subject Classification (2000) 65K10 · 94A12 · 94A08

1 Introduction

Reconstruction of sparse signals from limited measurements has been studied extensively with a variety of applications in various imaging sciences, machine learning, computer vision and so on. The major problem is to reconstruct a signal which is sparse by itself or in some transformed domain from a few measurements (or observations) acquired by a certain sensing machine. Let $\mathbf{x} \in \mathbb{R}^n$ be the signal to be reconstructed. Then the sparse signal reconstruction problem can be formulated as an ℓ_0 constrained minimization problem

$$\min_{\mathbf{x}\in\mathbb{R}^n} F(\mathbf{x}), \quad \text{s.t.} \quad \|\mathbf{x}\|_0 \leq k, \tag{1}$$

where the sparsity $\|\mathbf{x}\|_0$ counts nonzeros in $\mathbf{x}$. Here $F(\mathbf{x})$ is a loss function measuring the discrepancy between the acquired measurements $\mathbf{y} \in \mathbb{R}^m$ $(m \ll n)$ and the measurements predicted by the estimated solution. In particular, if the measurements are linearly related to the underlying signal, i.e., there exists a sensing matrix $A \in \mathbb{R}^{m\times n}$ such that $\mathbf{y} = A\mathbf{x} + \mathbf{n}$ where $\mathbf{n}$ is the Gaussian noise, then the least squares loss function is widely used:

$$F(\mathbf{x}) = \frac{1}{2}\|\mathbf{y} - A\mathbf{x}\|_2^2 .$$

In this case, (1) is a single measurement vector (SMV) sparse signal reconstruction problem. The choice of F depends on the generation mechanism of the measurements. Since the measurements are typically generated continuously in most imaging techniques, it becomes significantly important in practice to reconstruct a collection of sparse signals, expressed as a signal matrix, from multiple measurement vectors (MMV). More precisely, the signal matrix $X \in \mathbb{R}^{n\times L}$ with k $(k \leq n)$ nonzero rows can be obtained by solving the following MMV model

$$\min_{X\in\mathbb{R}^{n\times L}} F(X), \quad \text{s.t.} \quad \|X\|_{r,0} \leq k, \tag{2}$$

where $\|X\|_{r,0}$ stands for the row-sparsity of X which counts nonzero rows in X. Note that it is possible that certain columns of X have more zero components than zero rows of X. The MMV sparse reconstruction problem was first introduced in magnetoencephalography (MEG) imaging [1,2], and has been extended to other applications [3,4,5,6,7,8,9].

Many SMV algorithms can be applied to solve MMV problems. The most straightforward way is to use SMV algorithms to reconstruct each signal vector sequentially or simultaneously via parallel computing, and then concatenate all resultant signals to form the estimated signal matrix. We call these types of SMV algorithms, *concatenated SMV* algorithms. On the other hand, the MMV problem can be converted to an SMV one by columnwise stacking the unknown signal matrix X as a vector and introducing a block diagonal matrix as the new sensing matrix A. However, both approaches do not fully take advantage of the joint sparse structure of the underlying signal matrix, and lack computational efficiency as well. In this paper, we develop MMV algorithms without concatenation of the SMV results or vectorization of the unknown signal matrix.

Since the ℓ_0 term in (1) and (2) is non-convex and non-differentiable, many classical convex optimization algorithms fail to produce a satisfactory solution. To handle the NP-hardness of the problem, many convex relaxation methods and their MMV extensions have been developed, e.g., the ℓ_2-regularized

M-FOCUSS [1], and the ℓ_1-regularized MMV extensions of the alternating direction method of multipliers [10,11]. By exploiting the relationship between the measurements and the correct atoms, multiple signal classification (MUSIC) [12] and its improved variants [13,14] have been developed. However, in the rank defective cases when the rank of the measurement matrix Y is much smaller than the desired row-sparsity level, the MUSIC type of methods will mostly fail to identify the correct atoms. The third category of algorithms for solving the ℓ_0 constrained problem is the class of greedy algorithms that seek the sparsest solution by updating the support iteratively, including Orthogonal Matching Pursuit (OMP) [15], simultaneous OMP (S-OMP) [16,17], Compressive Sampling Matching Pursuit (CoSaMP) [18], Regularized OMP (ROMP) [19], Subspace-Pursuit (SP) [20], and Iterative Hard-Thresholding (IHT) [21]. It has been shown that CoSaMP and IHT are more efficient than the convex relaxation methods with strong recovery guarantees. However, most of these algorithms work for compressive sensing applications where F is a least squares loss function. Recently, the Gradient Matching Pursuit (GradMP) [22] has been proposed to extend CoSaMP to handle more general loss functions. To further improve the computational efficiency and consider the non-convex objective function case, Stochastic IHT (StoIHT) and Stochastic GradMP (StoGradMP) have been proposed [23]. Nevertheless, the aforementioned greedy algorithms are designed for solving the SMV problem and the concatenated extension to the MMV versions will result in limited performance especially for large data sets. In this paper, we propose the MMV Stochastic IHT (MStoIHT) and the MMV Stochastic GradMP (MStoGradMP) methods for solving the general MMV joint sparse recovery problem (2). To accelerate convergence, the mini-batching technique is applied to the proposed algorithms. We also show that the proposed algorithms converge faster than their SMV concatenated counterparts under certain conditions. A large variety of numerical experiments on joint sparse matrix recovery and video sequence recovery have demonstrated the superior performance of the proposed algorithms over their SMV counterparts in terms of running time and accuracy.

Organization. The rest of the paper is organized as follows. Preliminary knowledge and notation clarifications are provided in Section 2. Section 3 presents the concatenated SMV algorithms, and the proposed stochastic greedy algorithms, i.e., MStoIHT and MStoGradMP, in detail. Section 4 discusses how to apply the mini-batching technique to accelerate the proposed algorithms. Convergence analysis is provided in Section 5. By choosing the widely used least squares loss function as F, joint sparse signal recovery in distributed compressive sensing is discussed in Section 6. Extensive numerical results are shown in Section 7. Finally, some concluding remarks are made in Section 8.

2 Preliminaries

To make the paper self-contained, we first introduce some useful notation and definitions, and then briefly describe the related algorithms, i.e., StoIHT and StoGradMP. Let $[m] = \{1, 2, \dots, m\}$ and $|\Omega|$ be the number of elements in the set Ω. Consider a finite atom set $\mathcal{D} = \{\mathbf{d}_1, \dots, \mathbf{d}_N\}$ (a.k.a. the dictionary) with each atom $\mathbf{d}_i \in \mathbb{R}^n$.

2.1 Vector Sparsity

Assume that a vector $\mathbf{x} \in \mathbb{R}^n$ can be written as a linear combination of $\mathbf{d}_i$'s, i.e., $\mathbf{x} = \sum_{i=1}^N \alpha_i \mathbf{d}_i = D\alpha$ with

$$D = \begin{bmatrix} \mathbf{d}_1 \cdots \mathbf{d}_N \end{bmatrix}, \quad \alpha = (\alpha_1, \dots, \alpha_N)^T.$$

Then the support of $\mathbf{x}$ with respect to α and $\mathcal{D}$ is defined by

$$\mathrm{supp}_{\alpha,\mathcal{D}}(\mathbf{x}) = \{i \in [N] : \alpha_i \neq 0\} := \mathrm{supp}(\alpha).$$

The ℓ_0-norm of $\mathbf{x}$ with respect to $\mathcal{D}$ is defined as the minimal support size

$$\|\mathbf{x}\|_{0,\mathcal{D}} = \min_{\alpha}\{|T| : \mathbf{x} = \sum_{i\in T} \alpha_i \mathbf{d}_i, T \subseteq [N]\} = \min_{\alpha} |\,\mathrm{supp}_{\alpha,\mathcal{D}}(\mathbf{x})|.$$

Since absolute homogeneity does not hold in general, i.e., $\|\gamma\mathbf{x}\|_{0,\mathcal{D}} = |\gamma| \, \|\mathbf{x}\|_{0,\mathcal{D}}$ holds if and only if $|\gamma| = 1$, this ℓ_0-norm is not a norm. Here the smallest support $\operatorname{supp}_{\alpha,\mathcal{D}}(\mathbf{x})$ is called the support of $\mathbf{x}$ with respect to $\mathcal{D}$, denoted by $\operatorname{supp}_{\mathcal{D}}(\mathbf{x})$. Thus

$$|\operatorname{supp}_{\mathcal{D}}(\mathbf{x})| = \|\mathbf{x}\|_{0,\mathcal{D}}\,.$$

Note that the support may not be unique if $\mathcal{D}$ is over-complete in that there could be multiple representations of $\mathbf{x}$ with respect to the atom set $\mathcal{D}$ due to the linear dependence of the atoms in $\mathcal{D}$. The vector $\mathbf{x}$ is called k-sparse with respect to $\mathcal{D}$ if

$$\|\mathbf{x}\|_{0,\mathcal{D}} \leq k.$$

2.2 Matrix Sparsity

By extending vector sparsity, we define the row sparsity for a matrix $X \in \mathbb{R}^{n\times L}$ as follows

$$\|X\|_{r,0,\mathcal{D}} = \min_{\Omega}\{|\Omega| : \Omega = \bigcup_{i=1}^{L} \operatorname{supp}_{\mathcal{D}}(X_{\cdot,i})\},$$

where $X_{\cdot,i}$ is the i-th column of X. Here the minimal common support Ω is called the (row-wise) *joint support* of X with respect to $\mathcal{D}$, denoted by $\operatorname{supp}^r_{\mathcal{D}}(X)$, which satisfies

$$|\operatorname{supp}^r_{\mathcal{D}}(X)| = \|X\|_{r,0,\mathcal{D}}\,.$$

The matrix X is called k-row sparse with respect to $\mathcal{D}$ if all columns of X share a joint support of size at most k with respect to $\mathcal{D}$, i.e.,

$$\|X\|_{r,0,\mathcal{D}} \leq k.$$

2.3 Functions Defined on A Matrix Space

Given a function $f : \mathbb{R}^{n\times L} \to \mathbb{R}$, the matrix derivative is defined by concatenating gradients [24]

$$\frac{\partial f}{\partial X} = \left[\frac{\partial f}{\partial X_{i,j}}\right]_{n\times L} = \left[\nabla_{X_{\cdot,1}} f \,\cdots\, \nabla_{X_{\cdot,L}} f\right], \tag{3}$$

where $X_{i,j}$ is the (i, j)-th entry of X. Notice that

$$\|X\|_F^2 = \sum_{i=1}^{n} \|X_{i,\cdot}\|_2^2 = \sum_{j=1}^{L} \|X_{\cdot,j}\|_2^2 = \operatorname{Tr}(X^T X),$$

where $X_{i,\cdot}$ is the i-th row vector of X, and $\operatorname{Tr}(\cdot)$ is the trace operator to add up all the diagonal entries of a matrix. The inner product for any two matrices $U, V \in \mathbb{R}^{n\times L}$ is defined as

$$\langle U, V\rangle = \operatorname{Tr}(U^T V).$$

Note that the equality

$$\|U + V\|_F^2 = \|U\|_F^2 + \|V\|_F^2 + 2\langle U, V\rangle \tag{4}$$

and the Cauchy-Schwartz inequality

$$\langle U, V\rangle \leq \|U\|_F \, \|V\|_F \tag{5}$$

still hold. By generalizing the concepts in [23], we define the $\mathcal{D}$-restricted strong convexity property and the strong smoothness property (a.k.a. the Lipschitz condition on the gradient) for the functions defined on a matrix space.

Definition 1 The function $f : \mathbb{R}^{n\times L} \to \mathbb{R}$ satisfies the $\mathcal{D}$-restricted strong convexity ($\mathcal{D}$-RSC) if there exists $\rho_k^- > 0$ such that

$$f(X') - f(X) - \left\langle \frac{\partial f}{\partial X}(X), X' - X \right\rangle \geq \frac{\rho_k^-}{2} \|X' - X\|_F^2 \tag{6}$$

for all matrices $X', X \in \mathbb{R}^{n\times L}$ with $|\operatorname{supp}_{\mathcal{D}}^r(X) \cup \operatorname{supp}_{\mathcal{D}}^r(X')| \leq k$.

Definition 2 The function $f : \mathbb{R}^{n\times L} \to \mathbb{R}$ satisfies the $\mathcal{D}$-restricted strong smoothness ($\mathcal{D}$-RSS) if there exists $\rho_k^+ > 0$ such that

$$\left\| \frac{\partial f}{\partial X}(X) - \frac{\partial f}{\partial X}(X') \right\|_F \leq \rho_k^+ \|X - X'\|_F \tag{7}$$

for all matrices $X', X \in \mathbb{R}^{n\times L}$ with $|\operatorname{supp}_{\mathcal{D}}^r(X) \cup \operatorname{supp}_{\mathcal{D}}^r(X')| \leq k$.

2.4 Related Work

StoIHT (see Algorithm 1) and StoGradMP (see Algorithm 2) have been proposed to solve the ℓ_0 constrained SMV problem [23]

$$\min_{\mathbf{x}\in\mathbb{R}^n} \frac{1}{M}\sum_{i=1}^{M} \tilde{f}_i(\mathbf{x}), \quad \text{subject to} \quad \|\mathbf{x}\|_{0,\mathcal{D}} \leq k. \tag{8}$$

At each iteration of StoIHT, one component function $\tilde{f}_i$ is first randomly selected with probability $p(i)$. Here the input discrete probability distribution $p(i)$'s satisfy

$$\sum_{i=1}^{M} p(i) = 1, \quad \text{and} \quad p(i) \geq 0,\, i = 1, \ldots, M.$$

Next in the "Proxy" step, gradient descent along the selected component is performed. Then the last two steps, i.e., "Identify" and "Estimate", essentially project the gradient descent result to its best k-sparse approximation. Given $\mathbf{w} = (w_1, \ldots, w_n)^T$ and $\eta \geq 1$, the best k-sparse approximation operator acted on $\mathbf{w}$ and η, denoted by $\operatorname{approx}_k(\mathbf{w}, \eta)$, constructs an index set $\mathit{\Gamma}$ with $|\mathit{\Gamma}| = k$ such that

$$\|\mathcal{P}_{\mathit{\Gamma}}\mathbf{w} - \mathbf{w}\|_2 \leq \eta \left\|\mathbf{w} - \mathbf{w}_{(k)}\right\|_2 \quad \text{where} \quad \mathbf{w}_{(k)} = \operatorname*{argmin}_{\substack{\mathbf{y}\in\mathcal{R}(\mathcal{D}_{\mathit{\Gamma}})\\ |\mathit{\Gamma}|\leq k}} \|\mathbf{w} - \mathbf{y}\|_2 .$$

Here $\mathcal{P}_{\mathit{\Gamma}}\mathbf{w}$ is the orthogonal projection of $\mathbf{w}$ onto the subspace $\mathcal{R}(\mathcal{D}_{\mathit{\Gamma}})$ in $\mathbb{R}^n$ spanned by the atoms with indices in $\mathit{\Gamma}$, and $\mathbf{w}_{(k)}$ is the best k-sparse approximation of $\mathbf{w}$ in the subspace $\mathcal{R}(\mathcal{D}_{\mathit{\Gamma}})$. In particular, if $\eta \geq 1$ and $\mathcal{D} = \{\mathbf{e}_i : i = 1, 2, \ldots, n\}$ with $\mathbf{e}_i = [0, \ldots, \underset{(i)}{1}, \ldots, 0]^T$, then $\operatorname{approx}_k(\mathbf{w}, \eta)$ returns the index set of the first k largest entries of $\mathbf{w}$ in absolute value, i.e.,

$$\operatorname{approx}_k(\mathbf{w}, 1) = \{i_1, i_2, \ldots, i_k : |w_{i_1}| \geq \ldots \geq |w_{i_k}| \geq \ldots \geq |w_{i_n}|\} := \widehat{\mathit{\Gamma}}.$$

Then the projection $\mathcal{P}_{\mathit{\Gamma}}\mathbf{w}$ reads as in componentwise form

$$\left(\mathcal{P}_{\widehat{\mathit{\Gamma}}}(\mathbf{w})\right)_j = \begin{cases} w_j & \text{if } j \in \widehat{\mathit{\Gamma}}, \\ 0 & \text{if } j \notin \widehat{\mathit{\Gamma}}. \end{cases}$$

There are two widely used stopping criteria:

$$\frac{\left\|\mathbf{x}^{t+1} - \mathbf{x}^t\right\|_2}{\|\mathbf{x}^t\|_2} < \varepsilon, \quad \text{and} \quad \frac{1}{M}\sum_{i=1}^{M} \tilde{f}_i(\mathbf{x}^t) < \varepsilon,$$

Algorithm 1 Stochastic Iterative Hard Thresholding (StoIHT)

```
Input: $k, \gamma, \eta, p(i), \varepsilon$.
Output: $\widehat{\mathbf{x}} = \mathbf{x}^t$.
Initialize: $\mathbf{x} = \mathbf{0}$.
for $t = 1, 2, \ldots, T$ do
    Randomly select an index $i_t \in \{1, 2, \ldots, M\}$ with probability $p(i_t)$
    Proxy: $\mathbf{b}^t = \mathbf{x}^t - \frac{\gamma}{Mp(i_t)} \nabla \tilde{f}_{i_t}(\mathbf{x}^t)$
    Identify: $\Gamma^t = \text{approx}_k(\mathbf{b}^t, \eta)$.
    Estimate: $\mathbf{x}^{t+1} = \mathcal{P}_{\Gamma^t}(\mathbf{b}^t)$.
    If the stopping criteria are met, exit.
end for
```

where $\varepsilon > 0$ is a small tolerance. It is well known that the first stopping criteria is more robust in practice. Different from StoIHT, StoGradMP involves the gradient matching process, i.e., to find the best k-sparse approximation of the gradient rather than the estimated solution. At the solution estimation step, the original problem is restricted to the components from the estimated support. It has been empirically shown that StoGradMP converges faster than StoIHT due to the more accurate estimation of the support. But StoGradMP requires that the sparsity level k is no more than $n/2$.

Algorithm 2 Stochastic Gradient Matching Pursuit (StoGradMP)

```
Input: $k, \eta_1, \eta_2, p(i), \varepsilon$.
Output: $\widehat{\mathbf{x}} = \mathbf{x}^t$.
Initialize: $\mathbf{x}^0 = \mathbf{0}$, $\Lambda = 0$.
for $t = 1, 2, \ldots, T$ do
    Randomly select an index $i_t \in \{1, 2, \ldots, M\}$ with probability $p(i_t)$
    Calculate the gradient $\mathbf{r}^t = \nabla \tilde{f}_{i_t}(\mathbf{x}^t)$
    $\Gamma = \text{approx}_{2k}(\mathbf{r}^t, \eta_1)$
    $\widehat{\Gamma} = \Gamma \cup \Lambda$
    $\mathbf{b}^t = \text{argmin}_{\mathbf{x}} \frac{1}{M} \sum_{i=1}^M \tilde{f}_i(\mathbf{x}), \quad \mathbf{x} \in \mathcal{R}(\mathcal{D}_{\widehat{\Gamma}})$
    $\Lambda = \text{approx}_k(\mathbf{b}^t, \eta_2)$
    $\mathbf{x}^{t+1} = \mathcal{P}_\Lambda(\mathbf{b}^t)$
    If the stopping criteria are met, exit.
end for
```

3 Proposed Stochastic Greedy Algorithms

In this section, we present concatenated SMV algorithms, and develop stochastic greedy algorithms for MMV problems based on StoIHT and StoGradMP. Suppose that there are M differentiable and convex functions $f_i : \mathbb{R}^{n \times L} \to \mathbb{R}$ that satisfy the $\mathcal{D}$-restricted strong smoothness property (see Definition 2), and their mean

$$F(X) = \frac{1}{M} \sum_{i=1}^{M} f_i(X) \tag{9}$$

satisfies the $\mathcal{D}$-restricted strong convexity property (see Definition 1). These assumptions will be used extensively throughout the entire paper. Consider the following row-sparsity constrained MMV problem

$$\min_{X \in \mathbb{R}^{n \times L}} \frac{1}{M} \sum_{i=1}^{M} f_i(X), \quad \text{subject to} \quad \|X\|_{r,0,\mathcal{D}} \leq k. \tag{10}$$

By vectorizing X, i.e., rewriting X as a vector $\mathbf{x} \in \mathbb{R}^{nL}$ by columnwise stacking, we can *relax* (10) to a sparsity constrained SMV problem of the form (8) where the sparsity level k is replaced by kL. Since $\|\mathbf{x}\|_{0,\mathcal{D}} \leq kL$ does not necessarily guarantee $\|X\|_{r,0,\mathcal{D}} \leq k$, the solution to the relaxed problem may not be the same as the vectorization of the solution to (10). On the other hand, the iterative stochastic algorithms such as StoIHT and StoGradMP, can be developed to the *concatenated* versions for solving (10) under the following assumption on the objective function f_i's.

Assumption 1. The objective function in (10) is *separable*, in the sense that a collection of functions $g_{i,j} : \mathbb{R}^n \to \mathbb{R}$ exist with

$$f_i(X) = \sum_{j=1}^{L} g_{i,j}(X_{\cdot,j}), \quad i = 1, \ldots, M. \tag{11}$$

Under this assumption, the concatenated algorithms, i.e., CStoIHT in Algorithm 3 and CStoGradMP in Algorithm 4, can be applied to solve (10), which essentially reconstruct each column of X by solving the SMV problem (8). Notice that the outer loops of CStoIHT and CStoGradMP can be executed in a parallel manner on a multi-core computer, when the order of the inner loop and the outer loop in Algorithm 3 can be swapped. However, if the sparsity level k is very large, then the support sets of $X_{\cdot,j}$'s are prone to overlap less initially which results in the less accurate estimation of the joint support and larger errors in the initial iterates. In addition, for some nonlinear function $f_i(X)$ which can not be separated as a sum of functions for columns of X, it will be challenging to find an appropriate objective function for the corresponding SMV problem.

Algorithm 3 Concatenated Stochastic Iterative Hard Thresholding (CStoIHT)

```
Input: $k, \gamma, \eta, p(i), \varepsilon$.
Output: $\widehat{X} = X^t$.
Initialize: $X = \mathbf{0} \in \mathbb{R}^{n \times L}$.
for $j = 1, \ldots, L$ do
    for $t = 1, 2, \ldots, T$ do
        Randomly select an index $i_t \in \{1, 2, \ldots, M\}$ with probability $p(i_t)$
        Proxy: $\mathbf{b}^t = X^t_{\cdot,j} - \frac{\gamma}{Mp(i_t)} \nabla g_{i_t,j}(X^t_{\cdot,j})$
        Identify: $\Gamma^t = \text{approx}_k(\mathbf{b}^t, \eta)$.
        Estimate: $X^{t+1}_{\cdot,j} = \mathcal{P}_{\Gamma^t}(\mathbf{b}^t)$.
        If the stopping criteria are met, exit.
    end for
end for
```

Algorithm 4 Concatenated Stochastic Gradient Matching Pursuit (CStoGradMP)

```
Input: $k, \eta_1, \eta_2, p(i), \varepsilon$.
Output: $\widehat{X} = X^t$.
Initialize: $X^0 = \mathbf{0} \in \mathbb{R}^{n \times L}$, $\Lambda = 0$.
for $j = 1, \ldots, L$ do
    for $t = 1, 2, \ldots, T$ do
        Randomly select an index $i_t \in \{1, 2, \ldots, M\}$ with probability $p(i_t)$
        Calculate the gradient $\mathbf{r}^t = \nabla g_{i_t,j}(X^t_{\cdot,j})$
        $\Gamma = \text{approx}_{2k}(\mathbf{r}^t, \eta_1)$
        $\widehat{\Gamma} = \Gamma \cup \Lambda$
        $\mathbf{b}^t = \text{argmin}_{\mathbf{x}} \frac{1}{M} \sum_{i=1}^{M} g_{i,j}(\mathbf{x}), \quad \mathbf{x} \in \mathcal{R}(\mathcal{D}_{\widehat{\Gamma}})$
        $\Lambda = \text{approx}_k(\mathbf{b}^t, \eta_2)$
        $X^{t+1}_{\cdot,j} = \mathcal{P}_{\Lambda}(\mathbf{b}^t)$
        If the stopping criteria are met, exit.
    end for
end for
```

To circumvent the aforementioned issues, we first propose the MMV Stochastic Iterative Hard Thresholding algorithm (MStoIHT) detailed in Algorithm 5. Compared to StoIHT, MStoIHT replaces the gradient by the matrix derivative (3). The second significant difference lies in the "Identify" and "Estimate" steps, especially the operator $\text{approx}_k(\cdot, \cdot)$. Now we extend the operator $\text{approx}_k(\cdot, \cdot)$ from sparse vectors to row-sparse matrices. Given $X \in \mathbb{R}^{n\times L}$ and $\eta \geq 1$, the best k-row sparse approximation operator acted on X and η, denoted by $\text{approx}_k^r(X,\eta)$, constructs a *row* index set Γ such that

$$\left\|\mathcal{P}_\Gamma X_{\cdot,j} - X_{\cdot,j}\right\|_2 \leq \eta \left\|X_{\cdot,j} - (X_{\cdot,j})_k\right\|_2, \quad j = 1,\ldots,L,$$

where $(X_{\cdot,j})_k$ is the best k-sparse approximation of the column vector $X_{\cdot,j}$ with respect to $\mathcal{D}$. In particular, if $\mathcal{D} = \{\mathbf{e}_i : i = 1,\ldots,n\}$ and $\eta = 1$, $\text{approx}_k^r(X,1)$ returns the row index set of the first k largest ℓ_2 row norms in X, i.e.,

$$\text{approx}_k^r(X,1) = \{i_1, i_2, \ldots, i_k : \|X_{i_1,\cdot}\|_2 \geq \|X_{i_2,\cdot}\|_2 \geq \ldots \geq \|X_{i_n,\cdot}\|_2\} := \widetilde{\Gamma}.$$

By abusing the notation, we define $\mathcal{P}_{\widetilde{\Gamma}}(X)$ to be the projection of X onto the subspace of all row-sparse matrices with row indices restricted to $\widetilde{\Gamma}$. Therefore, we have

$$\mathcal{P}_{\widetilde{\Gamma}} X = \left[\mathcal{P}_{\widetilde{\Gamma}} X_{\cdot,1}\ \mathcal{P}_{\widetilde{\Gamma}} X_{\cdot,2}\ \ldots\ \mathcal{P}_{\widetilde{\Gamma}} X_{\cdot,L}\right].$$

Due to the common support $\widetilde{\Gamma}$, the projection $\mathcal{P}_{\widetilde{\Gamma}}(X)$ can be written as in row-wise form

$$\left(\mathcal{P}_{\widetilde{\Gamma}}(X)\right)_{j,\cdot} = \begin{cases} X_{j,\cdot} & \text{if } j \in \widetilde{\Gamma}, \\ \mathbf{0} & \text{if } j \notin \widetilde{\Gamma}. \end{cases}$$

Here $\mathcal{P}_{\widetilde{\Gamma}}(X)$ returns a k-row sparse matrix, whose nonzero rows correspond to the k rows of X with largest ℓ_2 row norms.

Algorithm 5 MMV Stochastic Iterative Hard Thresholding (MStoIHT)

```
Input: $k, \gamma, \eta, p(i), \varepsilon$.
Output: $\widehat{X} = X^t$.
Initialize: $X = \mathbf{0}$.
for $t = 1, 2, \ldots, T$ do
    Randomly select an index $i_t \in \{1, 2, \ldots, M\}$ with probability $p(i_t)$
    Proxy: $B^t = X^t - \frac{\gamma}{Mp(i_t)} \frac{\partial f_{i_t}(X^t)}{\partial X}$
    Identify: $\Gamma^t = \text{approx}_k^r(B^t, \eta)$.
    Estimate: $X^{t+1} = \mathcal{P}_{\Gamma^t}(B^t)$.
    If the stopping criteria are met, exit.
end for
```

Next, we propose the MMV Stochastic Gradient Matching Pursuit (MStoGradMP) detailed in Algorithm 6, where the two gradient matching steps involve the operator $\text{approx}_k^r(\cdot,\cdot)$. The stopping criteria in all proposed algorithms can be set as the same as those in Algorithm 1 and Algorithm 2.

Algorithm 6 MMV Stochastic Gradient Matching Pursuit (MStoGradMP)

```
Input: $k, \eta_1, \eta_2, p(i), \varepsilon$.
Output: $\widehat{X} = X^t$.
Initialize: $X^0 = \mathbf{0}$, $\Lambda = 0$.
for $t = 1, 2, \ldots, T$ do
    Randomly select an index $i_t \in \{1, 2, \ldots, M\}$ with probability $p(i_t)$
    Calculate the generalized gradient $R^t = \frac{\partial f_{i_t}(X^t)}{\partial X}$
    $\Gamma = \text{approx}_{2k}^r(R^t, \eta_1)$
    $\widehat{\Gamma} = \Gamma \cup \Lambda$
    $B^t = \text{argmin}_X F(X), \quad X \in \mathcal{R}(\mathcal{D}_{\widehat{\Gamma}})$
    $\Lambda = \text{approx}_k^r(B^t, \eta_2)$
    $X^{t+1} = \mathcal{P}_\Lambda(B^t)$
    If the stopping criteria are met, exit.
end for
```

4 Batched Acceleration

To accelerate computations and improve performance, we apply the mini-batching technique to obtain batched variants of Algorithms 5 and 6. We first partition the index set $\{1, 2, \dots, M\}$ into a collection of equal-sized batches $\tau_1, \dots, \tau_d$ where the batch size $|\tau_i| = b$ for all $i = 1, 2, \dots, \lceil M/b \rceil := d$. For simplicity, we assume that d is an integer. Similar to the approach in [25], we reformulate (9) as

$$F(X) = \frac{1}{d} \sum_{i=1}^{d} \left(\frac{1}{b} \sum_{j \in \tau_i} f_j(X) \right). \tag{12}$$

Based on this new formulation, we get the batched version of Algorithm 5, which is termed as BMStoIHT, described in Algorithm 7. Here the input probability $p(i)$ satisfies

$$\frac{1}{d} \sum_{i=1}^{d} p(i) = 1 \quad \text{and} \quad p(i) \geq 0,\, i = 1, \dots, d.$$

Likewise, we get a batched version of MStoGradMP, termed as BMStoGradMP, which is detailed in Algorithm 8. It is empirically shown in Section 7 that the increase of the batch size greatly speeds up the convergence of both algorithms, which is more obvious in BMStoIHT. As a by-product, mini-batching can also improve the recovery accuracy. However, there is a trade-off between the batch size and the performance improvement [25].

Algorithm 7 Batched MMV Stochastic Iterative Hard Thresholding (BMStoIHT)

```
Input: $k, \gamma, \eta$, $b$ and $p(i)$.
Output: $\widehat{X} = X^t$.
Initialize: $X = \mathbf{0}$.
for $t = 1, 2, \dots, T$ do
    Randomly select a batch index $\tau_t \subseteq \{1, 2, \dots, d\}$ of size $b$ with probability $p(\tau_t)$
    Proxy: $B^t = X^t - \frac{\gamma}{dp(\tau_t)} \frac{\partial f_{\tau_t}(X^t)}{\partial X}$
    Identify $\varGamma^t = \text{approx}_k^r(B^t, \eta)$.
    Estimate $X^{t+1} = \mathcal{P}_{\varGamma^t}(B^t)$.
    If the stopping criteria are met, exit.
end for
```

Algorithm 8 Batched MMV Stochastic Gradient Matching Pursuit (BMStoGradMP)

```
Input: $k, \eta_1, \eta_2$, $b$ and $p(i)$.
Output: $\widehat{X} = X^t$.
Initialize: $X^0 = \mathbf{0}$, $\varLambda = 0$.
for $t = 1, 2, \dots, T$ do
    Randomly select a batch index $\tau_t \subseteq \{1, 2, \dots, d\}$ of size $b$ with probability $p(\tau_t)$
    Calculate the generalized gradient $R^t = \frac{\partial f_{\tau_t}(X^t)}{\partial X}$
    $\varGamma = \text{approx}_{2k}^r(R^t, \eta_1)$
    $\widehat{\varGamma} = \varGamma \cup \varLambda$
    $B^t = \text{argmin}_X\, F(X), \quad X \in \text{span}(D_{\widehat{\varGamma}})$
    $\varLambda = \text{approx}_k^r(B^t, \eta_2)$
    $X^{t+1} = \mathcal{P}_{\varLambda}(B^t)$
    If the stopping criteria are met, exit.
end for
```

5 Convergence Analysis

In this section, we provide the convergence guarantees for the proposed MStoIHT and MStoGradMP, together with their SMV counterparts, i.e., CStoIHT and CStoGradMP. To simplify the discussion, the result at the t-th iteration of CStoIHT/CStoGradMP refers to the result obtained after t inner iterations and L outer iterations of Algorithm 3/Algorithm 4, or equivalently the maximum number of inner iterations is set as t. Furthermore, all convergence results can be extended to their batched versions, i.e., BStoIHT and BMStoGradMP. Comparison of contraction coefficients shows that the proposed algorithms have faster convergence under the $\mathcal{D}$-RSC, $\mathcal{D}$-RSS and separability of the objective function (see Assumption 1).

By replacing the ℓ_2-norm and vector inner product with the Frobenius norm and the matrix inner product respectively and using (4) and (5), we get similar convergence results for MStoIHT and MStoGradMP as in [23]:

Theorem 1 *Let X^* be a feasible solution of* (10) *and X^0 be the initial solution. At the $(t+1)$-th iteration of Algorithm 5, the expectation of the recovery error is bounded by*

$$E\left\|X^{t+1}-X^*\right\|_F \le \kappa^{t+1}\left\|X^0-X^*\right\|_F + \frac{\sigma_{X^*}}{1-\kappa}, \tag{13}$$

where

$$\begin{aligned}\kappa &= 2\sqrt{1-\gamma(2-\gamma\alpha_{3k})\rho_{3k}^-} + \sqrt{(\eta^2-1)(1+\gamma^2\alpha_{3k}\overline{\rho}_{3k}^+ - 2\gamma\rho_{3k}^-)},\\ \alpha_k &= \max_{1\le i\le M}\frac{\rho_k^+(i)}{Mp(i)}, \quad \rho_k^+ = \max_{1\le i\le M}\rho_k^+(i), \quad \overline{\rho}_k^+ = \frac{1}{M}\sum_{i=1}^M \rho_k^+(i).\end{aligned} \tag{14}$$

Thus Algorithm 7 converges linearly. In particular, if $\eta=\gamma=1$, then

$$\kappa = 2\sqrt{1-2\rho_{3k}^- + \alpha_{3k}\rho_{3k}^-}. \tag{15}$$

To solve the problem (10), CStoIHT uses StoIHT to restore each column of X separately and then concatenate all column vectors to form a matrix. To analyze the convergence of CStoIHT, we first derive an upper bound for $E\left\|X_{\cdot,j}-X^*_{\cdot,j}\right\|_2^2$ following the proof in [23, Section 8.2]. Note that we look for the contraction coefficient of $E\left\|X_{\cdot,j}-X^*_{\cdot,j}\right\|_2^2$ rather than $E\left\|X_{\cdot,j}-X^*_{\cdot,j}\right\|_2$.

Lemma 1 *Let X^* be a feasible solution of* (10) *and X^0 be the initial solution. Under Assumption 1, there exist $\kappa_j, \sigma_j > 0$ such that the expectation of the recovery error squared at the t-th iteration of Algorithm 3 for estimating the j-th column of X^* is bounded by*

$$E_{I_t}\left\|X^{t+1}_{\cdot,j}-X^*_{\cdot,j}\right\|_2^2 \le \kappa_j^{t+1}\left\|X^0_{\cdot,j}-X^*_{\cdot,j}\right\|_2^2 + \frac{\sigma_j}{1-\kappa_j}, \tag{16}$$

where $X^t_{\cdot,j}$ is the approximation of $X^*_{\cdot,j}$ at the t-th iteration of StoIHT with the initial guess $X^0_{\cdot,j}$, i.e., the result at the t-th inner iteration and j-th outer iteration of Algorithm 3 with the initial guess X^0. Here I_t is the set of all indices $i_1,\ldots,i_t$ randomly selected at or before the t-th step of the algorithm.

Proof Due to the separable form of f_i in Assumption 1, we consider L problems of the form

$$\min_{\mathbf{w}} \frac{1}{M}\sum_{i=1}^M g_{i,j}(\mathbf{w}), \quad \|\mathbf{w}\|_{0,\mathcal{D}} \le k, \quad j=1,\ldots,L, \tag{17}$$

where $g_{i,j}$ are given in (11). This relaxation is also valid since $X^*_{\cdot,j}$ is also a feasible solution of (17) if X^* is a feasible solution of (10). Let $\mathbf{w}^t = X^t_{\cdot,j}$, $\mathbf{w}^* = X^*_{\cdot,j}$, $x = \left\|\mathbf{w}^{t+1}-\mathbf{w}^*\right\|_2$,

$$u = \left\|\mathbf{w}^t - \mathbf{w}^* - \frac{\gamma}{Mp(i_t)}\mathcal{P}_\Omega(\nabla g_{i_t,j}(\mathbf{w}^t) - \nabla g_{i_t,j}(\mathbf{w}^*))\right\|_2 + \left\|\frac{\gamma}{Mp(i_t)}\mathcal{P}_\Omega\nabla g_{i_t,j}(\mathbf{w}^*)\right\|_2 := u_1 + u_2,$$

and

$$v = (\eta^2 - 1) \left\| \mathbf{w}^t - \mathbf{w}^* - \frac{\gamma}{Mp(i_t)} \nabla g_{i_t,j}(\mathbf{w}^t) \right\|_2^2 .$$

Let

$$v_1 = (\eta^2 - 1) \left\| \mathbf{w}^t - \mathbf{w}^* - \frac{\gamma}{Mp(i_t)} (\nabla g_{i_t,j}(\mathbf{w}^t) - \nabla g_{i_t,j}(\mathbf{w}^*)) \right\|_2^2 ,$$

$$v_2 = (\eta^2 - 1) \left\| \frac{\gamma}{Mp(i_t)} \nabla g_{i_t,j}(\mathbf{w}^*) \right\|_2^2 .$$

The inequality $(a+b)^2 \leq 2a^2 + 2b^2$ yields that

$$v \leq 2v_1 + 2v_2.$$

By the direct computation, we can show that

$$x^2 - 2ux - v \leq 0,$$

which implies that

$$x \leq u + \sqrt{u^2 + v}.$$

Thus the inequality $a + b \leq \sqrt{2a^2 + 2b^2}$ yields

$$x^2 \leq 2u^2 + 2(u^2 + v) = 4u^2 + 2v \leq 8(u_1^2 + u_2^2) + 4v_1 + 4v_2.$$

By taking the conditional expectation $E_{i_t|I_{t-1}}$ on both sides, we obtain

$$\begin{aligned}
E_{i_t|I_{t-1}} x^2 &\leq 8E_{i_t|I_{t-1}} u_1^2 + 8E_{i_t|I_{t-1}} u_2^2 + 4E_{i_t|I_{t-1}} v_1 + 4E_{i_t|I_{t-1}} v_2 \\
&\leq 8(1 - (2\gamma - \gamma^2 \alpha_{3k,j})\rho_{3k,j}^-) \left\| \mathbf{w}^t - \mathbf{w}^* \right\|_2^2 + \frac{8\gamma^2}{\min_{i_t} M^2(p(i_t))^2} E_{i_t|I_{t-1}} \left\| \mathcal{P}_\Omega \nabla g_{i_t,j}(\mathbf{w}^*) \right\|_2^2 \\
&+ 4(\eta^2 - 1)(1 + \gamma^2 \alpha_{3k,j} \bar{\rho}_{3k,j}^+ - 2\gamma\rho_{3k,j}^-) \left\| \mathbf{w}^t - \mathbf{w}^* \right\|_2^2 + \frac{4\gamma^2(\eta^2 - 1)}{\min_{i_t} M^2(p(i_t))^2} E_{i_t} \left\| \nabla g_{i_t,j}(\mathbf{w}^*) \right\|_2^2 \\
&= \Big(8(1 - (2\gamma - \gamma^2 \alpha_{3k,j})\rho_{3k,j}^-) + 4(\eta^2 - 1)(1 + \gamma^2 \alpha_{3k,j} \bar{\rho}_{3k,j}^+ - 2\gamma\rho_{3k,j}^-) \Big) \left\| \mathbf{w}^t - \mathbf{w}^* \right\|_2^2 \\
&+ \frac{4\gamma^2}{\min_{i_t} M^2(p(i_t))^2} \Big(2E_{i_t} \left\| \mathcal{P}_\Omega \nabla g_{i_t,j}(\mathbf{w}^*) \right\|_2^2 + (\eta^2 - 1) E_{i_t} \left\| \nabla g_{i_t,j}(\mathbf{w}^*) \right\|_2^2 \Big) \\
&:= \kappa_j \left\| \mathbf{w}^t - \mathbf{w}^* \right\|_2^2 + \sigma_j .
\end{aligned}$$

Note that all coefficients $\alpha_{3k,j}$, $\rho_{3k,j}^-$ and $\bar{\rho}_{3k,j}^+$ depend on the function g_{ij} in (11). Therefore

$$E_{I_t} \left\| \mathbf{w}^{t+1} - \mathbf{w}^* \right\|_2^2 \leq \kappa_j E_{I_{t-1}} \left\| \mathbf{w}^t - \mathbf{w}^* \right\|_2^2 + \sigma_j .$$

which implies that

$$E \left\| X_{\cdot,j}^{t+1} - X_{\cdot,j}^* \right\|_2^2 \leq \kappa_j^{t+1} \left\| X_{\cdot,j}^0 - X_{\cdot,j}^* \right\|_2^2 + \frac{\sigma_j}{1 - \kappa_j}, \quad j = 1, \dots, L.$$

Here the contraction coefficient is

$$\kappa_j = 8(1 - (2\gamma - \gamma^2 \alpha_{3k,j})\rho_{3k,j}^-) + 4(\eta^2 - 1)(1 + \gamma^2 \alpha_{3k,j} \bar{\rho}_{3k,j}^+ - 2\gamma\rho_{3k,j}^-), \tag{18}$$

and the tolerance parameter is

$$\sigma_j = \frac{4\gamma^2}{\min\limits_{1 \leq i \leq M} M^2(p(i_t))^2} \Big(2E_{i_t} \left\| \mathcal{P}_\Omega \nabla g_{i_t,j}(\mathbf{w}^*) \right\|_2^2 + (\eta^2 - 1) E_{i_t} \left\| \nabla g_{i_t,j}(\mathbf{w}^*) \right\|_2^2 \Big) . \tag{19}$$

In particular, if $\gamma = \eta = 1$ and $p(i) = 1/M$, then

$$\kappa_j = 8(1 - 2\rho_{3k,j}^- + \alpha_{3k,j}\rho_{3k,j}^-), \quad \sigma_j = \frac{8}{M} \sum_{i=1}^{M} \left\| \mathcal{P}_\Omega \nabla g_{i,j}(X_{\cdot,j}^*) \right\|_2^2 .$$

Theorem 2 *Let X^* be a feasible solution of* (10) *and X^0 be the initial solution. Under Assumption 1, at the $(t+1)$-th iteration of Algorithm 3, the expectation of the recovery error is bounded by*

$$E\left\|X^{t+1}-X^*\right\|_F \le \widehat{\kappa}^{t+1}\left\|X^0-X^*\right\|_F+\delta, \tag{20}$$

where X^t is the approximation of X^* at the t-th iteration of Algorithm 3 with the initial guess $X^0_{\cdot,j}$. Here the contraction coefficient $\widehat{\kappa}$ and the tolerance parameter δ are defined as

$$\widehat{\kappa}=\sqrt{\max_{1\le j\le L}\kappa_j}, \quad \delta=\sqrt{\frac{\sum_{j=1}^L \sigma_j}{1-\max\limits_{1\le j\le L}\kappa_j}},$$

where κ_j is the contraction coefficient for each StoIHT defined in (18).

Proof For each $j=1,2,\ldots,L$, StoIHT with the initial guess $X^0_{\cdot,j}$ generates $X^{t+1}_{\cdot,j}$ after t iterations, i.e., the result at the $(t+1)$-th inner iteration and j-th outer iteration of Algorithm 3. Then the expectation of the recovery error squared is bounded by

$$E\left\|X^{t+1}_{\cdot,j}-X^*_{\cdot,j}\right\|_2^2 \le \kappa_j^{t+1}\left\|X^0_{\cdot,j}-X^*_{\cdot,j}\right\|_2^2+\frac{\sigma_j}{1-\kappa_j},$$

where κ_j and σ_j are defined in Lemma 1. Note that κ_j depends only on the constants in the $\mathcal{D}$-RSC and $\mathcal{D}$-RSS properties of the objective function or its component function $g_{i,j}$ while σ_j depends on the feasible solution $X^*_{\cdot,j}$. By combining all L components of X^{t+1}, we get

$$\begin{aligned}
E\left\|X^{t+1}-X^*\right\|_F &\le \sqrt{E\left\|X^{t+1}-X^*\right\|_F^2}\\
&= \sqrt{\sum_{j=1}^L E\left\|X^{t+1}_{\cdot,j}-X^*_{\cdot,j}\right\|_2^2}\\
&\le \sqrt{\sum_{j=1}^L\left(\kappa_j^{t+1}\left\|X^0_{\cdot,j}-X^*_{\cdot,j}\right\|_2^2+\frac{\sigma_j}{1-\kappa_j}\right)}\\
&\le \sqrt{(\max_{1\le j\le L}\kappa_j)^{t+1}\left\|X^0-X^*\right\|_F^2+\frac{\sum_{j=1}^L\sigma_j}{1-\max_{1\le j\le L}\kappa_j}}\\
&\le \left(\sqrt{\max_{1\le j\le L}\kappa_j}\right)^{t+1}\left\|X^0-X^*\right\|_F+\sqrt{\frac{\sum_{j=1}^L\sigma_j}{1-\max\limits_{1\le j\le L}\kappa_j}}\\
&:= \widehat{\kappa}^{t+1}\left\|X^0-X^*\right\|_F+\delta.
\end{aligned}$$

In particular, if $\gamma=\eta=1$, then

$$\widehat{\kappa}=\sqrt{\max_{1\le j\le L}\kappa_j}=\sqrt{\max_{1\le j\le L}8(1-2\rho^-_{3k,j}+\alpha_{3k,j}\rho^-_{3k,j})}=2\sqrt{2}\sqrt{\max_{1\le j\le L}(1-2\rho^-_{3k,j}+\alpha_{3k,j}\rho^-_{3k,j})}.$$

In the case that $\rho^-_{3k,j}=\rho_{3k}$, $\alpha_{3k,j}=\alpha_{3k}$ and $\rho^+_{3k,j}=\rho_{3k}$, we have

$$\widehat{\kappa}=\sqrt{2}\kappa,$$

where κ is defined Theorem 1. This shows that the CStoIHT converges slower than the proposed MStoIHT provided the same coefficients in the strong smoothness and convexity of the objective function, which we show holds for the distributed compressive sensing setting in Section 6.

By using the same proof techniques as in Theorem 1, we can get the following convergence result for MStoGradMP.

Theorem 3 *Let X^* be a feasible solution of* (10) *and X^0 be the initial solution. At the $(t+1)$-th iteration of Algorithm 6, the expectation of the recovery error is bounded by*

$$E\left\|X^{t+1}-X^*\right\|_F \leq \kappa^{t+1}\left\|X^0-X^*\right\|_F+\frac{\sigma_{X^*}}{1-\kappa}, \tag{21}$$

where

$$\kappa=(1+\eta_2)\sqrt{\frac{\alpha_{4k}}{\rho_{4k}^-}}\left(\max_{1\leq i\leq M}\sqrt{Mp(i)}\sqrt{\frac{\rho_{4k}^+(2\eta_1^2-1)}{\rho_{4k}^-\eta_2^2}-1}+\frac{\sqrt{\eta_1^2-1}}{\eta_1}\right),$$

$$\sigma_{X^*}=\frac{(1+\eta_2)}{\rho_{4k}^-\min\limits_{1\leq i\leq M}Mp(i)}\left(2\max_{1\leq i\leq M}Mp(i)\sqrt{\frac{\alpha_{4k}}{\rho_{4k}^-}}+3\right)\max_{\substack{|\Omega|\leq 4k\\ 1\leq i\leq M}}\left\|\mathcal{P}_\Omega\frac{\partial f_i}{\partial X}(X^*)\right\|_F.$$

Thus Algorithm 6 converges linearly. In particular, if $\eta_1=\eta_2=1$ and $p(i)=1/M$, then

$$\kappa=\frac{2\sqrt{\alpha_{4k}(\rho_{4k}^+-\rho_{4k}^-)}}{\rho_{4k}^-}. \tag{22}$$

Similar to CStoIHT, we start the convergence analysis for CStoGradMP by finding the contraction coefficient for the expectation of recovery error squared at each iteration of CStoGradMP.

Lemma 2 *Let X^* be a feasible solution of* (12) *and X^0 be the initial solution. Under Assumption 1, the expectation of the recovery error squared at the t-th iteration of Algorithm 4 for estimating the j-th column of X^* is bounded by*

$$E_{I_t}\left\|\mathbf{b}^t-X_{\cdot,j}^*\right\|_2^2\leq\beta_1E_{I_t}\left\|\mathcal{P}_{\widehat{\Gamma}}(\mathbf{b}^t-X_{\cdot,j}^*)\right\|_2^2+\delta_1, \tag{23}$$

where

$$\beta_1=\frac{\alpha_{4k}}{2\rho_{4k}^--\alpha_{4k}},\quad \delta_1=\frac{2}{\alpha_{4k}(2\rho_{4k}^--\alpha_{4k})\min\limits_{1\leq i\leq M}M^2(p(i))^2}E_{I_t}E_i\left\|\mathcal{P}_{\widehat{\Gamma}}\nabla g_{i,j}(X_{\cdot,j})\right\|_2^2. \tag{24}$$

Here I_t is the set of all indices $i_1,\ldots,i_t$ randomly selected at or before the t-th step of the algorithm.

Proof Consider the problem (17). By the proof in [23, Lemma 2], we get

$$\left\|\mathcal{P}_{\widehat{\Gamma}}(\mathbf{b}^t-X_{\cdot,j}^*)\right\|_2^2\leq 2(1-(2\gamma-\gamma^2\alpha_{4k})\rho_{4k}^-\left\|\mathbf{b}^t-X_{\cdot,j}^*\right\|_2^2+\frac{2\gamma^2}{\min\limits_{1\leq i\leq M}M^2(p(i))^2}E_i\left\|\mathcal{P}_{\widehat{\Gamma}}\nabla g_{i,j}(X_{\cdot,j})\right\|_2^2,$$

where we use the inequality $(a+b)^2\leq 2a^2+2b^2$ for $a,b\in\mathbb{R}$ and the expectation inequality $(EX)^2\leq E(X^2)$. Then we have

$$\begin{aligned}&\left\|\mathbf{b}^t-X_{\cdot,j}^*\right\|_2^2=\left\|\mathcal{P}_{\widehat{\Gamma}}(\mathbf{b}^t-X_{\cdot,j}^*)\right\|_2^2+\left\|\mathcal{P}_{\widehat{\Gamma}^c}(\mathbf{b}^t-X_{\cdot,j}^*)\right\|_2^2\\&\leq 2(1-(2\gamma-\gamma^2\alpha_{4k})\rho_{4k}^-)\left\|\mathbf{b}^t-X_{\cdot,j}^*\right\|_2^2+\frac{2\gamma^2}{\min\limits_{1\leq i\leq M}M^2(p(i))^2}E_i\left\|\mathcal{P}_{\widehat{\Gamma}}\nabla g_{i,j}(X_{\cdot,j}^*)\right\|_2^2+\left\|\mathcal{P}_{\widehat{\Gamma}^c}(\mathbf{b}^t-X_{\cdot,j}^*)\right\|_2^2.\end{aligned}$$

Moving the first term on the right hand side to the left hand side leads to

$$\left\|\mathbf{b}^t-X_{\cdot,j}^*\right\|_2^2\leq\frac{2\gamma^2}{\phi\min\limits_{1\leq i\leq M}M^2(p(i))^2}E_i\left\|\mathcal{P}_{\widehat{\Gamma}}\nabla g_{i,j}(X_{\cdot,j})\right\|_2^2+\frac{1}{\phi}\left\|\mathcal{P}_{\widehat{\Gamma}^c}(\mathbf{b}^t-X_{\cdot,j}^*)\right\|_2^2,$$

where $\phi=2\rho_{4k}^-(2\gamma-\gamma^2\alpha_{4k})-1$. Maximizing ϕ with respect to γ yields $\gamma=1/\alpha_{4k}$ and $\phi_{\max}=(2\rho_{4k}^--\alpha_{4k})/\alpha_{4k}$. By choosing the optimal value of γ and taking the expectation with respect to I_t on the both sides of the above inequality, we get (23).

Similarly, using the inequality $EX\leq\sqrt{E(X^2)}$ and the fact that $a\leq b+c$ yields $a^2\leq 2b^2+2c^2$, we are able to get the following result, which is different from Lemma 3 in [23] in that we consider the expectation for the ℓ_2-norm squared here rather than that for the ℓ_2-norm.

Lemma 3 *Let X^* be a feasible solution of* (12) *and X^0 be the initial solution. Under Assumption 1, the expectation of the recovery error squared at the t-th iteration of Algorithm 4 for estimating the j-th column of X^* is bounded by*

$$E_{i_t} \left\| \mathcal{P}_{\widehat{\Gamma}^c}(\mathbf{b}^t - X^*_{\cdot,j}) \right\|_2^2 \leq \beta_2 \left\| X^t_{\cdot,j} - X^*_{\cdot,j} \right\|_2^2 + \delta_2, \tag{25}$$

where i_t is the index randomly selected at the t-th iteration of the CStoGradMP and

$$\begin{aligned} \beta_2 &= 4 \max_{1\leq i\leq M} Mp(i) \frac{(2\eta_1^2 - 1)\rho^+_{4k} - \eta_1^2 \rho^-_{4k}}{\eta_1^2 \rho^-_{4k}} + \frac{2(\eta_1^2 - 1)}{\eta_1^2}, \\ \delta_2 &= 8 \left(\frac{\max\limits_{1\leq i\leq M} p(i)}{\rho^-_{4k} \min\limits_{1\leq i\leq M} p(i)} \right)^2 \max_{\substack{|\Omega|\leq 4k \\ 1\leq i\leq M}} \left\| \mathcal{P}_\Omega \nabla g_{i,j}(X^*_{\cdot,j}) \right\|_2^2 . \end{aligned} \tag{26}$$

Theorem 4 *Let X^* be a feasible solution of* (12) *and X^0 be the initial solution. Under Assumption 1, at the $(t+1)$-th iteration of Algorithm 4, there exist $\widetilde{\kappa}, \eta > 0$ such that the expectation of the recovery error is bounded by*

$$E \left\| X^{t+1} - X^* \right\|_F \leq \widetilde{\kappa}^{t+1} \left\| X^0 - X^* \right\|_F + \eta, \tag{27}$$

where $X^t_{\cdot,j}$ is the approximation of $X^*_{\cdot,j}$ at the t-th iteration of CStoGradMP with the initial guess $X^0_{\cdot,j}$.

Proof At the t-th iteration of CStoGradMP, i.e., Algorithm 4, we have

$$\left\| X^{t+1}_{\cdot,j} - \mathbf{b}^t \right\|_2^2 \leq \eta_2^2 \left\| \mathbf{b}^t_{(k)} - \mathbf{b}^t \right\|_2^2 \leq \eta_2^2 \left\| X^*_{\cdot,j} - \mathbf{b}^t \right\|_2^2 ,$$

where $\mathbf{b}^t_{(k)}$ is the best k-sparse approximation of $\mathbf{b}^t$ with respect to the atom set $\mathcal{D}$. Therefore, we get

$$\begin{aligned} \left\| X^{t+1}_{\cdot,j} - X^*_{\cdot,j} \right\|_2^2 &\leq \left\| X^{t+1}_{\cdot,j} - \mathbf{b}^t + \mathbf{b}^t - X^*_{\cdot,j} \right\|_2^2 \\ &\leq 2 \left\| X^{t+1}_{\cdot,j} - \mathbf{b}^t \right\|_2^2 + 2 \left\| \mathbf{b}^t - X^*_{\cdot,j} \right\|_2^2 \\ &\leq (2 + 2\eta_2^2) \left\| \mathbf{b}^t - X^*_{\cdot,j} \right\|_2^2 . \end{aligned}$$

Next we establish the relationships among various expectations

$$\begin{aligned} E_{I_t} \left\| X^{t+1}_{\cdot,j} - X_{\cdot,j} \right\|_2^2 &\leq (2 + 2\eta_2^2) E_{I_t} \left\| \mathbf{b}^t - X^*_{\cdot,j} \right\|_2^2 \\ &\leq (2 + 2\eta_2^2)(\beta_1 E_{I_t} \left\| \mathcal{P}_{\hat{\Gamma}}(\mathbf{b}^t - X^*_{\cdot,j}) \right\|_2^2 + \delta_1) \\ &\leq (2 + 2\eta_2^2)\beta_1(\beta_2 E_{I_t} \left\| X^t_{\cdot,j} - X^*_{\cdot,j} \right\|_2^2 + \delta_2) + (2 + 2\eta_2^2)\delta_1 \\ &:= \kappa_j \left\| X^t_{\cdot,j} - X^*_{\cdot,j} \right\|_2^2 + \sigma_j , \end{aligned}$$

where the first inequality is guaranteed by Lemma 2 and the second inequality is guaranteed by Lemma 3. Here the contraction coefficient κ_j and the tolerance parameter σ_j are defined by

$$\kappa_j = (2 + 2\eta_2^2)\beta_1\beta_2, \quad \sigma_j = (2 + 2\eta_2^2)\beta_1\delta_2 + (2 + 2\eta_2^2)\delta_1,$$

where β_1, δ_1 are defined in (24) and β_2, β_2 are defined in (26). Then similar to the proof of Theorem 2, we can derive that

$$E \left\| X^{t+1} - X^* \right\|_F \leq \widetilde{\kappa}^{t+1} \left\| X^0 - X^* \right\|_F + \eta$$

where

$$\widetilde{\kappa} = \sqrt{\max_{1\leq j\leq L} \kappa_j}, \quad \text{and} \quad \eta = \sqrt{\frac{\sum_{j=1}^L \sigma_j}{1 - \max\limits_{1\leq j\leq L} \kappa_j}}.$$

In particular, if $\eta_1 = \eta_2 = 1$ and $p(i) = 1/M$, then

$$\widetilde{\kappa} = 4\sqrt{\frac{\alpha_{4k}(\rho_{4k}^{+} - \rho_{4k}^{-})}{\rho_{4k}^{-}(2\rho_{4k}^{-} - \alpha_{4k})}}.$$

If, in addition, $\alpha_{4k} = \rho_{4k}^{-}$, then we have

$$\widetilde{\kappa} = 2\kappa,$$

where the contraction coefficient κ for MStoGradMP is given in (22), which implies that MStoGradMP converges faster than CStoGradMP in this case due to the smaller contraction coefficient. Compared with MStoIHT, MStoGradMP has even larger convergence improvement in terms of recovery accuracy and running time.

6 Distributed Compressive Sensing Application

In this section, we show that the objective function commonly used in the distributed compressive sensing problem satisfies the $\mathcal{D}$-RSC and $\mathcal{D}$-RSS properties, which paves the theoretical foundation for using the proposed algorithms in this application. Suppose that there are L underlying signals $\mathbf{x}_j \in \mathbb{R}^n$ for $j = 1, 2, \ldots, L$, and their measurements are generated by

$$\mathbf{y}_j = A^{(j)}\mathbf{x}_j, \quad j = 1, 2, \ldots, L$$

where $A^{(j)} \in \mathbb{R}^{m\times n}$ $(m \ll n)$ is the measurement matrix (a.k.a. the sensing matrix). For discussion simplicity, we assume all measurement matrices are the same, i.e., $A^{(j)} = A = [A_{\cdot,1}, \ldots, A_{\cdot,n}]$. By concatenating all vectors as a matrix, we rewrite the above equation as follows

$$Y = AX, \quad Y = [\mathbf{y}_1, \ldots, \mathbf{y}_L] \in \mathbb{R}^{m\times L}, \quad X = [\mathbf{x}_1, \ldots, \mathbf{x}_L] \in \mathbb{R}^{n\times L}.$$

Now assume that the atom set is finite and denote $\mathcal{D} = \{\mathbf{d}_1, \ldots, \mathbf{d}_N\}$ with the corresponding dictionary $D = [\mathbf{d}_1, \ldots, \mathbf{d}_N]$. Consider the following distributed compressive sensing model with common sparse supports [26]

$$\min_X \frac{1}{2m}\sum_{j=1}^{L} \|\mathbf{y}_j - A\mathbf{x}_j\|_2^2 \quad \text{s.t.} \quad \mathbf{x}_j = D\theta_j \quad \operatorname{supp}(\theta_j) = \varOmega \subseteq \{1, 2, \ldots, N\}. \tag{28}$$

Here the objective function has the form

$$F(X) = \frac{1}{2m}\|Y - AX\|_F^2. \tag{29}$$

Then $F(X)$ can be written as

$$F(X) = \frac{1}{M}\sum_{i=1}^{M} f_i(X),$$

where $M = m/b$ and

$$f_i(X) = \frac{1}{2b}\sum_{j=1}^{L}(Y_{i,j} - \sum_{k=1}^{n} A_{i,k}X_{k,j})^2 = \frac{1}{2b}\|Y_{i,\cdot} - A_{i,\cdot}X\|_2^2. \tag{30}$$

The above expression shows that f_i's satisfy the Assumption 1 and thereby the concatenated algorithms in Section 3 can be applied. We first compute the partial derivative. For $s = 1, 2, \dots, n$ and $t = 1, 2, \dots, L$, we have

$$\begin{aligned}\frac{\partial f_i(X)}{\partial X_{s,t}} &= \frac{1}{2b}\sum_{j=1}^{L} 2\left(\sum_{k=1}^{n} A_{i,k}X_{k,j} - Y_{i,j}\right)\sum_{k=1}^{n} A_{i,k}\frac{\partial X_{k,j}}{\partial X_{s,t}}\\ &= \frac{1}{b}\sum_{j=1}^{L}\left(\sum_{k=1}^{n} A_{i,k}X_{k,j} - Y_{i,j}\right)\sum_{k=1}^{n} A_{i,k}\delta_{k,s}\delta_{j,t}\\ &= \frac{1}{b}\left(\sum_{k=1}^{n} A_{i,k}X_{k,t} - Y_{i,t}\right)A_{i,s}.\end{aligned}$$

Here $\delta_{i,j} = 1$ if $i = j$ and zero otherwise. Thus the generalized gradient of $f_i(X)$ with respect to X has the form

$$\frac{\partial f_i(X)}{\partial X} = \frac{1}{b}A_{i,\cdot}^T(A_{i,\cdot}X - Y_{i,\cdot}).$$

Lemma 4 *If the sensing matrix $A \in \mathbb{R}^{m\times n}$ satisfies the Restricted Isometry Property (RIP), i.e., there exists $\delta_k > 0$ such that*

$$(1-\delta_k)\left\|\mathbf{x}\right\|_2^2 \le \left\|A\mathbf{x}\right\|_2^2 \le (1+\delta_k)\left\|\mathbf{x}\right\|_2^2$$

for any k-sparse vector $\mathbf{x} \in \mathbb{R}^n$, then the function $F(X)$ defined in (29) satisfies the $\mathcal{D}$-restricted strong convexity property.

Proof Let $X \in \mathbb{R}^{n\times L}$ with k nonzero rows, which implies that each column of X has at most k nonzero components. By the RIP of A, we have

$$(1-\delta_k)\left\|X_{\cdot,j}\right\|_2^2 \le \left\|AX_{\cdot,j}\right\|_2^2 \le (1+\delta_k)\left\|X_{\cdot,j}\right\|_2^2, \quad j = 1, \dots, L.$$

Note that $\left\|X\right\|_F^2 = \sum_{j=1}^{L}\left\|X_{\cdot,j}\right\|_2^2$. Thus we get

$$(1-\delta_k)\left\|X\right\|_F^2 \le \left\|AX\right\|_F^2 \le (1+\delta_k)\left\|X\right\|_F^2.$$

For any two $X, X' \in \mathbb{R}^{n\times L}$ with $|\operatorname{supp}_{\mathcal{D}}^r(X) \cup \operatorname{supp}_{\mathcal{D}}^r(X')| \le k$, we have

$$\begin{aligned}&F(X') - F(X) - \left\langle \frac{\partial F(X)}{\partial X}, X' - X\right\rangle\\ &= \frac{1}{2m}\left(\left\|Y - AX'\right\|_F^2 - \left\|Y - AX\right\|_F^2\right) - \left\langle \frac{1}{m}A^T(AX - Y), X' - X\right\rangle\\ &= \frac{1}{2m}\left\|A(X' - X)\right\|_F^2\\ &\ge \frac{1-\delta_k}{2m}\left\|X' - X\right\|_F^2.\end{aligned}$$

Thus $F(X)$ satisfies the $\mathcal{D}$-restricted strong smoothness property with $\rho_k^- = \frac{1-\delta_k}{2m}$.

Lemma 5 *If the sensing matrix $A \in \mathbb{R}^{m\times n}$ satisfies the following property: for any k-sparse vector $\mathbf{x} \in \mathbb{R}^n$, there exists $\delta_k > 0$ such that*

$$\frac{1}{b}\left\|A_{\tau_i,\cdot}^T A_{\tau_i,\cdot}\mathbf{x}\right\|_2 \le (1+\delta_k)\left\|\mathbf{x}\right\|_2$$

where $A_{\tau_i,\cdot}$ is formed by extracting rows of A with row indices in τ_i. Then the function $f_i(X)$ defined in (30) satisfies the $\mathcal{D}$-restricted strong convexity property.

Proof Let $X \in \mathbb{R}^{n\times L}$ have k nonzero rows. Then

$$\frac{1}{b}\left\|A_{\tau_i,\cdot}^T A_{\tau_i,\cdot} X_{\cdot,j}\right\|_2 \leq (1+\delta_k)\left\|X_{\cdot,j}\right\|_2, \quad j=1,\ldots,L,$$

which implies that

$$\frac{1}{b}\left\|A_{\tau_i,\cdot}^T A_{\tau_i,\cdot} X\right\|_F \leq (1+\delta_k)\left\|X\right\|_F.$$

For any two $X, X' \in \mathbb{R}^{n\times L}$ with $|\operatorname{supp}_{\mathcal{D}}^r(X) \cup \operatorname{supp}_{\mathcal{D}}^r(X')| \leq k$, we have

$$\begin{aligned}\left\|\frac{\partial f_i(X)}{\partial X} - \frac{\partial f_i(X')}{\partial X}\right\|_F &= \frac{1}{b}\left\|A_{i,\cdot}^T(A_{i,\cdot}X - Y_{i,\cdot}) - A_{i,\cdot}^T(A_{i,\cdot}X' - Y_{i,\cdot})\right\|_F \\ &= \frac{1}{b}\left\|A_{i,\cdot}^T A_{i,\cdot}(X - X')\right\|_F \\ &\leq (1+\delta_k)\left\|X - X'\right\|_F.\end{aligned}$$

Therefore $f_i(X)$ satisfies the $\mathcal{D}$-restricted strong convexity with $\rho_k^+(i) = 1+\delta_k$.

By Lemma 4, Lemma 5 and the convergence analysis in Section 5, the contraction coefficient in the proposed algorithms depends on the coefficient in the RIP condition, whose infimum for some special type of matrices are available [27].

7 Numerical Experiments

In this section, we conduct a variety of numerical experiments to validate the effectiveness of the proposed algorithms. More specifically, our tests include reconstruction of row sparse signals from a linear system and joint sparse video sequence recovery. To compare different results quantitatively, we use the relative error defined as follows

$$\text{ReErr} = \frac{\|X^t - X^*\|_F}{\|X^*\|_F},$$

where X^* is the ground truth and X^t is the estimation of X^* at the t-th iteration. Regarding the computational efficiency, we also record the running time which counts all the computation time over a specified number of iterations excluding data loading or generation. Here we use the commands `tic` and `toc` in Matlab. To assess the concatenated SMV algorithms, we apply the SMV algorithm sequentially to the same sensing matrix and all columns of the measurement matrix Y, and save all intermediate approximations of each column of X for further computation of the relative error. In all tests, we use the discrete uniform distribution, i.e., $p(i) = 1/M$ for $i = 1, 2, \ldots, M$ in the non-batched version and $p(i) = 1/d$ for $i = 1, 2, \ldots, d$ in the batched version. The parameter η is fixed as 1. By default, each algorithm is stopped when either the relative error between two subsequent approximations of X^* reaches the tolerance or the maximum number of iterations is achieved.

All our experiments are performed in a desktop with an Intel® Xeon® CPU E5-2650 v4 @ 2.2GHz and 64GB RAM in double precision. The algorithms are implemented in Matlab 2016a running on Windows 10.

7.1 Joint Sparse Matrix Recovery

In the first set of experiments, we compare the proposed algorithms and their concatenated SMV counterparts in terms of reconstruction error and running time. In particular, we investigate the impact of the sparsity level k, and the number of underlying signals L to be reconstructed on the performance of the BStoIHT and BStoGradMP in both concatenated SMV and MMV versions, in terms of relative error and the running time. To reduce randomness in the results, we run 50 trials for each test with fixed parameters and then take the average over the number of trials.

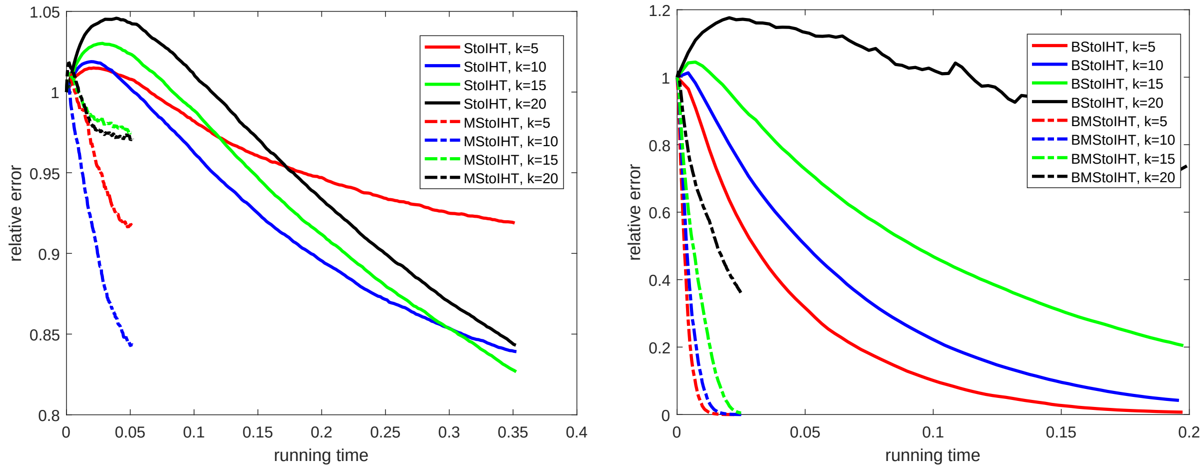


Fig. 1 Comparison of StoIHT and MStoIHT in both non-batched and batched versions for various sparsity levels k of the signal matrix. From left to right: batch sizes are 1 and 10.

First, we compare StoIHT and MStoIHT in both non-batched and batched versions, and fix the maximum number of iterations as 1000 and $\gamma = 1$ in both algorithms. To start with, we create a sensing matrix $A \in \mathbb{R}^{100\times 200}$ where each entry follows the normal distribution with zero mean and variance of $1/100$ and each column of A is normalized by dividing its ℓ_2-norm. In this way, it can be shown that the spark of A, i.e., the smallest number of linearly dependent columns of A, is 100 with probability one. To create a signal matrix $X^* \in \mathbb{R}^{200\times 40}$, we first generate a Gaussian distributed random matrix of size 200×40, and then randomly zero out $(200-k)$ rows where k is the row sparsity of X^*. The measurement matrix Y is created by AX for the noise-free cases. By choosing the sparsity level $k \in \{5, 10, 15, 20\}$ and the batch size $b \in \{1, 10\}$, we obtain the results shown in Figure 1. Since the initial guess for the signal matrix is set a zero matrix, all the error curves start with the point $(0, 1)$. Notice that to show the computational efficiency, we use the running time in seconds as the horizonal axis rather than the number of iterations. It can be seen that as the sparsity level grows, i.e., the signal matrix is less joint sparse, more running time (or iterations) is required to achieve the provided tolerance in terms of the relative reconstruction error. Meanwhile, as the batch size increases, BMStoIHT performs better than the sequential BStoIHT. With large sparsity levels, the inaccurate joint support obtained in the concatenated SMV algorithms cause large relative errors in the first few iterations (see Figure 1).

Next, we fix the sparsity level k as 5 and choose the number of signals as $L \in \{20, 40, 60, 80\}$. Figure 2 compares the results obtained by BStoIHT and BMStoIHT when the batch size is 1 and 10. In general, BMStoIHT takes less running time than its sequential concatenated SMV counterpart. We can see that mini-batching significantly improves the reconstruction accuracy and reduces the running time of BMStoIHT. After a large number of tests, we also find that the computational speedup of BMStoIHT is almost linear with respect to the number of signals to be reconstructed. Lastly, to test the robustness to noise, we add the Gaussian noise with zero mean and standard deviation (a.k.a. noise level) $\sigma \in \{0.02, 0.04, 0.06, 0.08\}$ to the measurement matrix Y. The relative errors for all BStoIHT and BMStoIHT results versus running time are shown in Figure 3. It is worth noting that the change of sparsity and noise levels have insignificant impact on the running time, which explains that the curve corresponding to the same algorithm stops almost at the same horizontal coordinate in Figure 1 and Figure 3. By contrast, the running time grows as the number of signals to be recovered increases which suggests that the endpoint of each curve has different horizontal coordinates in Figure 2.

In the second set of tests, we compare StoGradMP and MStoGradMP in non-batched and batched versions. It is known that StoGradMP usually converges much faster than StoIHT. We fix the maximum number of iterations as 30, $\gamma = 1$, and the batch size as 1 (non-batched version). Similar to the previous tests, we create a 100×200 random matrix whose columns are normalized, and fix the number of signals $L = 40$. Figure 4 shows the results obtained by StoGradMP and MStoGradMP with sparsity level

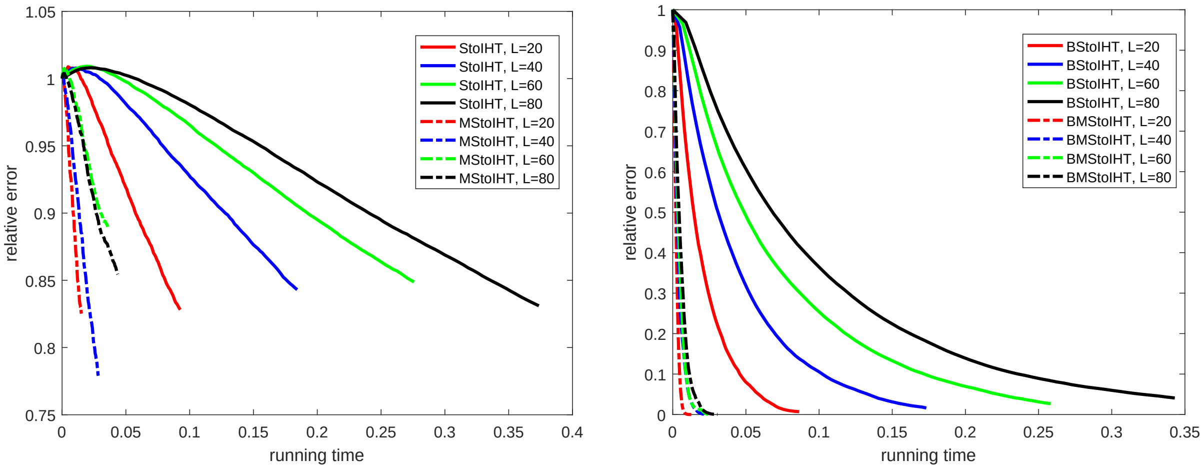


Fig. 2 Comparison of StoIHT and MStoIHT in both non-batched and batched versions for various numbers of signals L to be reconstructed. From left to right: batch sizes are 1 and 10.

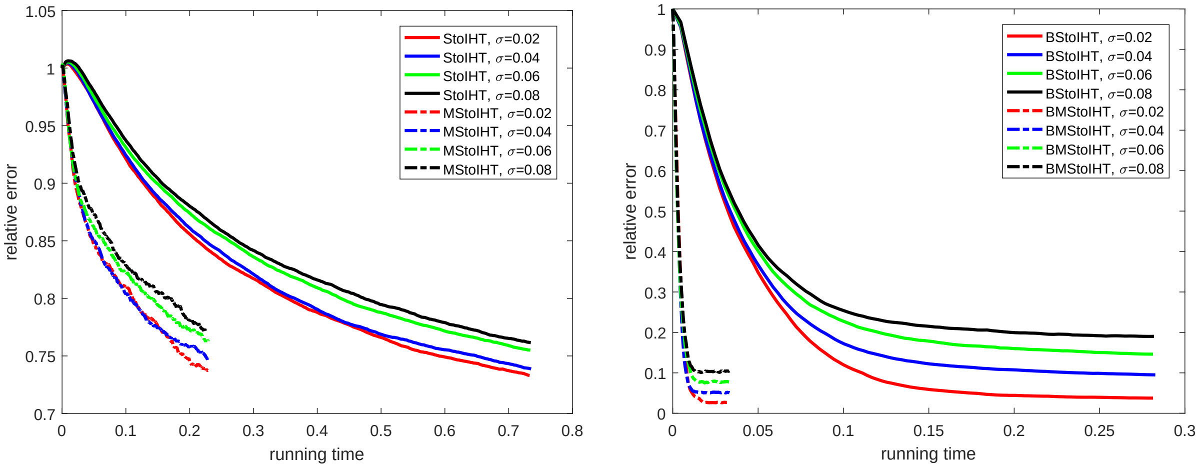


Fig. 3 Comparison of StoIHT and MStoIHT in both non-batched and batched versions for various noise levels σ to the measurement matrix. From left to right: batch sizes are 1 and 10.

$k \in \{60, 70, 80, 90\}$. Note that for better visualization, we skip the starting point (0,1) for all relative error plots, and use the base 10 logarithmic scale for the horizontal axis of running time since the MStoGradMP takes much less running time than StoGradMP after the same number of iterations. Unlike StoIHT and MStoIHT, both StoGradMP and MStoGradMP require that the sparsity level is no more than $n/2$, i.e., 100 in our case. As the sparsity k increases, the operator `pinv` for computing the pseudo-inverse matrix becomes more computationally expensive for matrices with more columns than their rank, which results in the significant growth of running time. For sparse signal matrices, StoGradMP performs better than StoIHT in terms of convergence. Next, we set the number of signals as $20, 40, 60, 80$, and get the results shown in Figure 5. It can be seen that MStoGradMP always takes less running time with even higher accuracy than the sequential StoGradMP. We also discovered that the computation speedup of MStoIHT is almost constant with respect to the number of signals to be reconstructed. The robustness comparison is shown in Figure 6, where the noise level ranges in $\{0.02, 0.04, 0.06, 0.08\}$. Furthermore, it is empirically shown that the BMStoGradMP performs much better than BMStoIHT considering their respective convergence behavior and robustness.

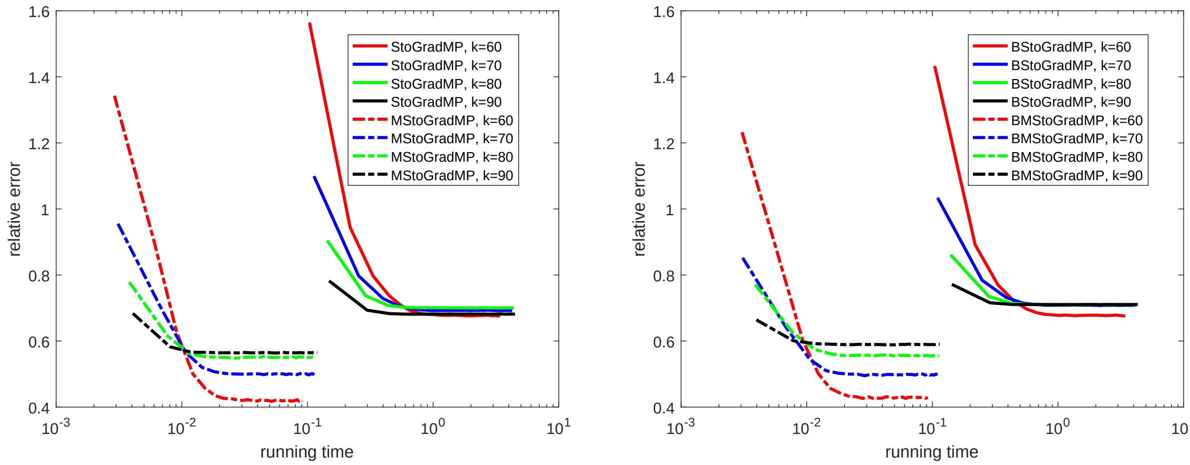


Fig. 4 Comparison of BStoGradMP and BMStoGradMP with various sparsity levels k. From left to right: batch sizes are 1 and 10.

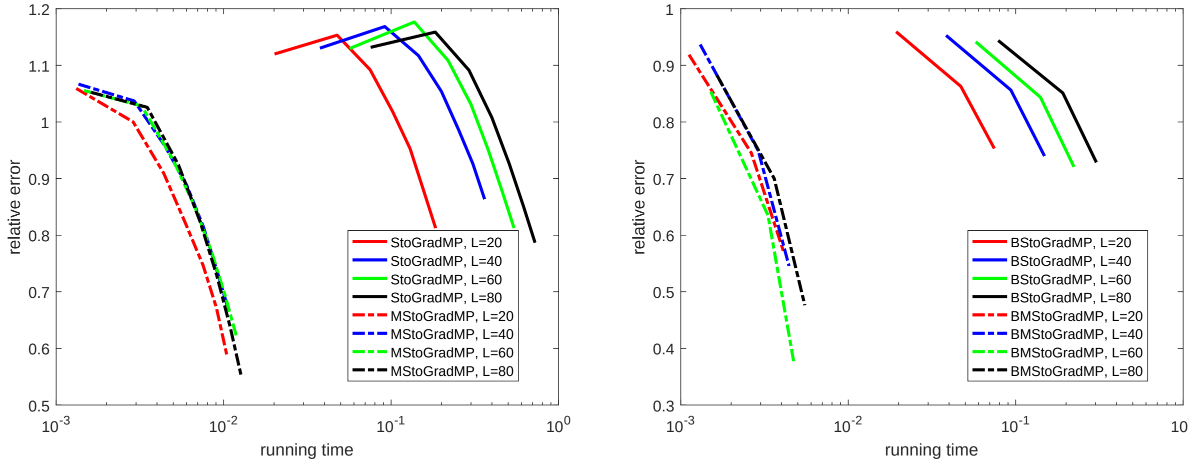


Fig. 5 Comparison of BStoGradMP and BMStoGradMP with various numbers of signals L. From left to right: batch sizes are 1 and 10.

7.2 Joint Sparse Video Sequence Recovery

In this set of experiments, we compare the proposed Algorithm 8 and the split Bregman algorithm for constrained MMV problem (SBC) [11, Algorithm 2], on joint sparse video sequence reconstruction. We first download a candle video consisting of 75 frames from the Dynamic Texture Toolbox in `http://www.vision.jhu.edu/code/`. In order to make the test video sequence possess a joint sparse structure, we extract 11 frames of the original data, i.e., frames 1 to 7, 29, 37, 69 and 70, each of which is of size 80 × 30. Then we create a data matrix $X \in \mathbb{R}^{2400\times 11}$, whose columns are a vectorization of all video frames. To further obtain a sparse representation of X, the K-SVD algorithm [28] is applied to obtain a dictionary $\Psi \in \mathbb{R}^{2400\times 50}$ for X. The K-SVD dictionary Ψ and the support of the corresponding coefficient matrix Θ for the extracted 11 frames are shown in Figure 7 (a) and (b). Some selected columns of the dictionary Ψ, namely *atoms*, are reshaped as an image of size 80 × 30 illustrated in Figure 7 (c) and (d). It can be seen that these 11 frames are nearly joint sparse under the learned dictionary. The relative error of using this K-SVD dictionary Ψ to represent the data matrix X is $\frac{\|X-\Psi\Theta\|_F}{\|X\|_F} = 0.0870$. A Gaussian random matrix $\Phi \in \mathbb{R}^{60\times 2400}$ with zero mean and unit variance is set as a sensing matrix,

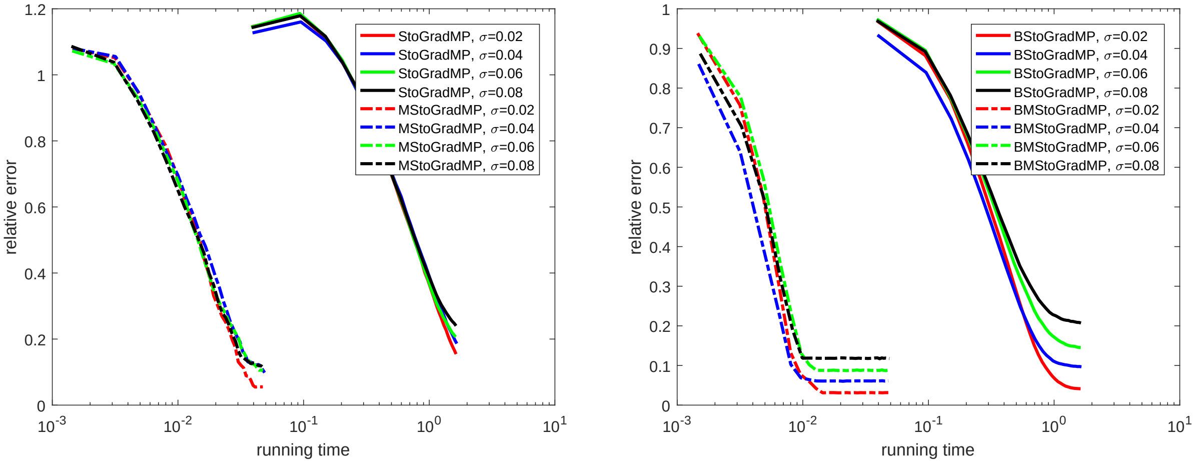


Fig. 6 Comparison of BStoGradMP and BMStoGradMP with various noise levels σ. From left to right: batch sizes are 1 and 10.

which is used to measure this data matrix. In other words, the measurements $Y \in \mathbb{R}^{60\times 11}$ are generated via $Y = \Phi X$. Given the measurements Y and the new sparse representation dictionary $A = \Phi\Psi$, we then apply Algorithm 8 and SBC to recover the joint sparse coefficient matrix $\hat{\Theta}$. In Algorithm 8, the sparsity level k is set as 10, and the block size b is set as 3, which implies that there are $d = 20$ blocks. In addition, we set $\eta_1 = \eta_2 = 1$. Both algorithms stop when the residual error reaches a tolerance threshold $\tau = 10^{-6}$, i.e., $\|Y - A\hat{\Theta}\|_F \leq \tau$. The first four recovered frames are shown in Figure 8. Regarding the computation time, the proposed Algorithm 8 is about ten times faster than SBC according to our experiments. The proposed algorithm takes less running time than SBC since it stops at the 10-th iteration while SBC stops at the 1691-th iteration. Even surprisingly, although the same error tolerance is set for both algorithms, the proposed algorithm yields a smaller relative error $(\frac{\|X_{:,i}-\Psi\hat{\Theta}_{:,i}\|_2}{\|X_{:,i}\|_2})$ than SBC. Figure 9 compares the relative discrepancy $\|Y - A\hat{\Theta}\|_F / \|Y\|_F$ versus running time for both algorithms, where we can see the zigzag pattern in the SBC result.

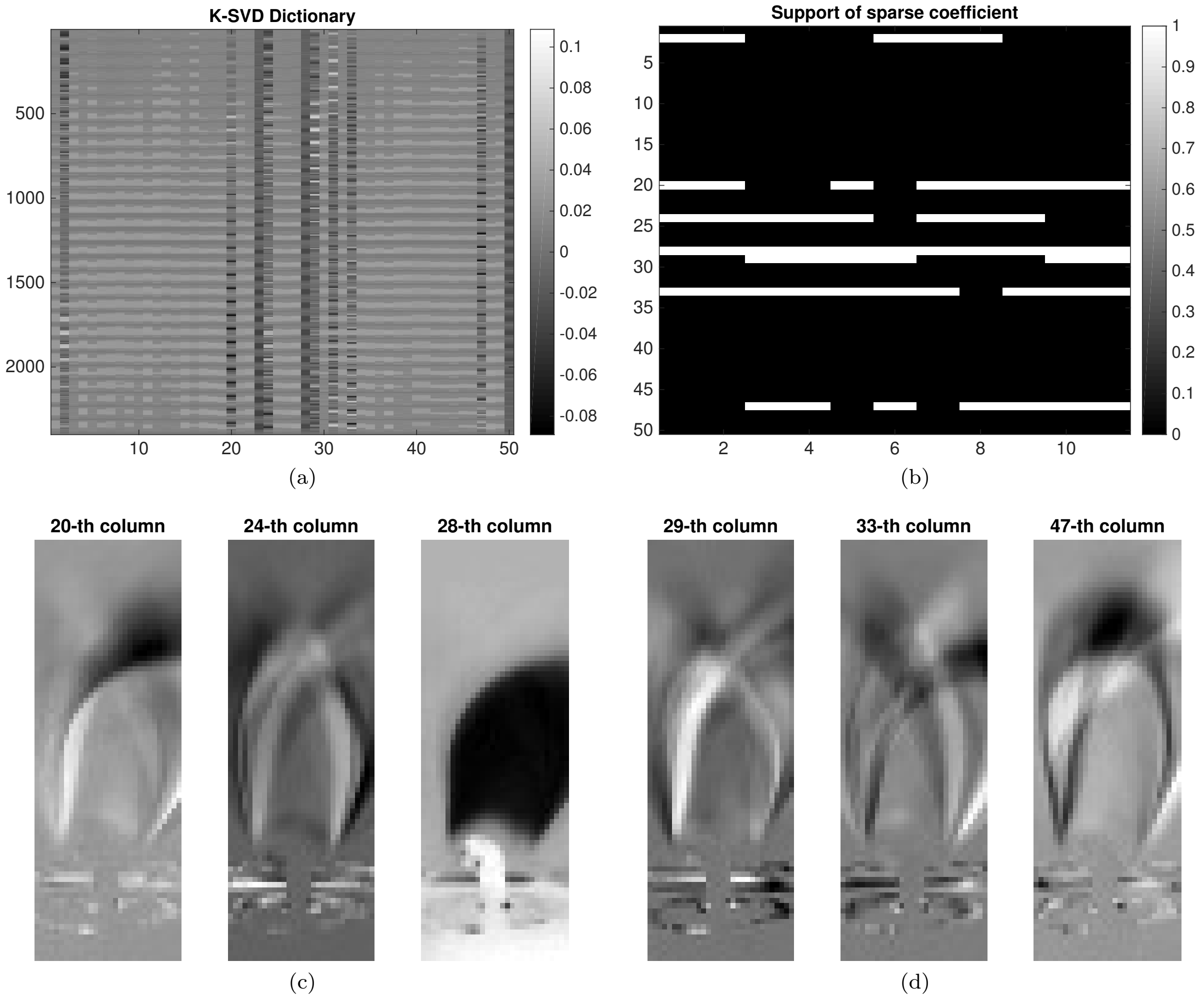
8 Conclusions

In this paper, we study the multiple measurement vector sparse reconstruction problem, which is of great importance in a large amount of signal processing applications. We propose two stochastic greedy algorithms, MStoIHT and MStoGradMP, together with their respective accelerated versions by applying the mini-batching technique. Our convergence analysis has shown theoretically that the proposed algorithms converge faster than their concatenated SMV counterparts. A variety of numerical experiments on linear systems and video frame processing have shown that the proposed algorithms outperform the concatenated SMV algorithms in terms of efficiency, accuracy and robustness to the noise.

Acknowledgements The initial research for this effort was conducted at the Research Collaboration Workshop for Women in Data Science and Mathematics, July 17-21 held at ICERM. Funding for the workshop was provided by ICERM, AWM and DIMACS (NSF grant CCF-1144502).

References

1. S. F. Cotter, B. D. Rao, K. Engan, and K. Kreutz-Delgado, “Sparse solutions to linear inverse problems with multiple measurement vectors,” *IEEE Transactions on Signal Processing*, vol. 53, no. 7, pp. 2477–2488, 2005.

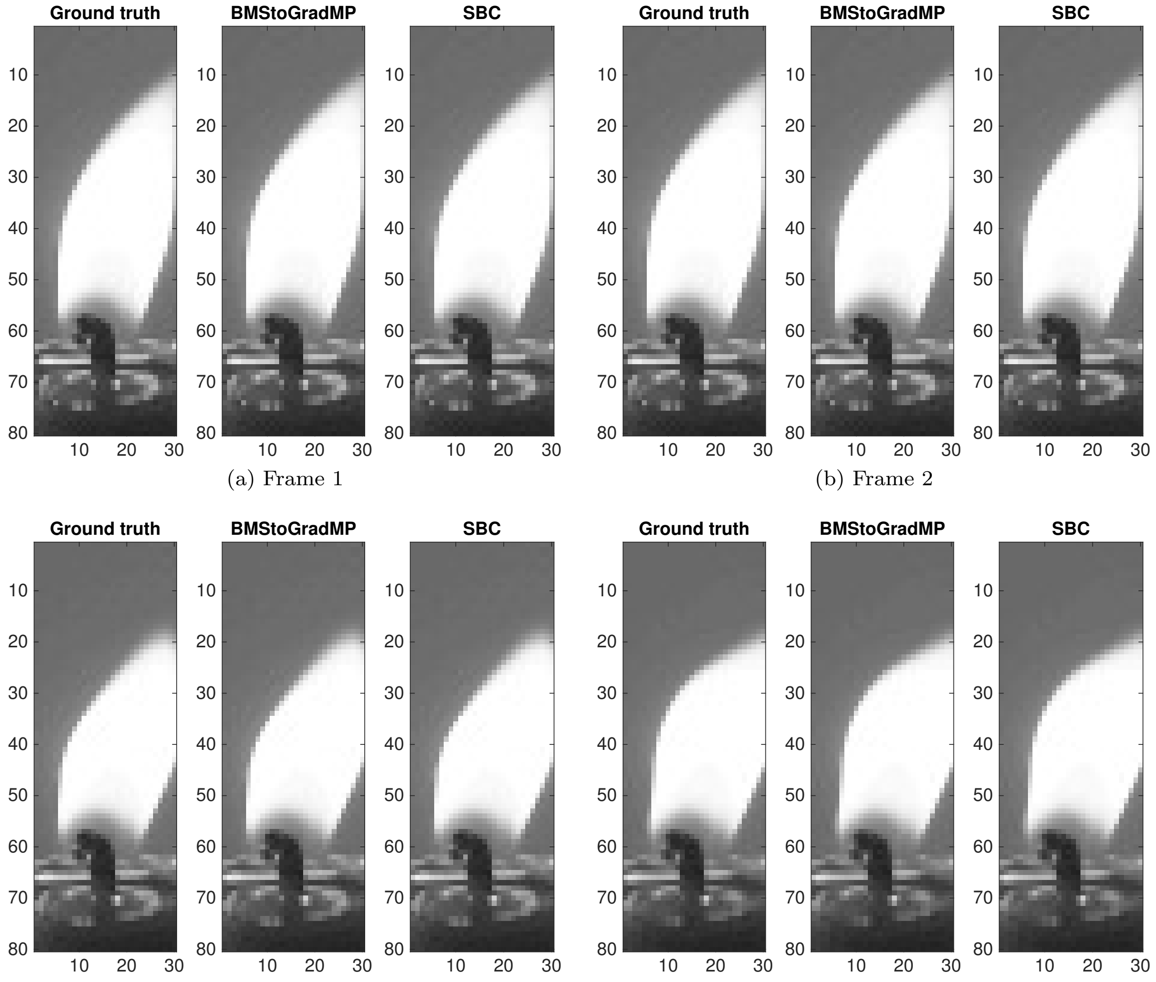


(a) (b)

(c) (d)

Fig. 7 (a) K-SVD dictionary learned from the total 75 frames from the candle video. (b) Support of sparse coefficient matrix for extracted 11 frames. The sparse coefficient matrix has non-zero entries on white area. (c-d) Some columns of the learned K-SVD dictionary.

2. B. D. Rao, K. Engan, and S. Cotter, “Diversity measure minimization based method for computing sparse solutions to linear inverse problems with multiple measurement vectors,” in *Acoustics, Speech, and Signal Processing, 2004. Proceedings.(ICASSP’04). IEEE International Conference on*, vol. 2, pp. ii–369, IEEE, 2004.
3. Z. He, A. Cichocki, R. Zdunek, and J. Cao, “CG-M-FOCUSS and its application to distributed compressed sensing,” in *International Symposium on Neural Networks*, pp. 237–245, Springer, 2008.
4. J. A. Bazerque and G. B. Giannakis, “Distributed spectrum sensing for cognitive radio networks by exploiting sparsity,” *IEEE Transactions on Signal Processing*, vol. 58, no. 3, pp. 1847–1862, 2010.
5. A. Majumdar and R. K. Ward, “Joint reconstruction of multiecho mr images using correlated sparsity,” *Magnetic resonance imaging*, vol. 29, no. 7, pp. 899–906, 2011.
6. A. Majumdar and R. K. Ward, “Face recognition from video: An MMV recovery approach,” in *Acoustics, Speech and Signal Processing (ICASSP), 2012 IEEE International Conference on*, pp. 2221–2224, IEEE, 2012.
7. A. Majumdar and R. Ward, “Rank awareness in group-sparse recovery of multi-echo mr images,” *Sensors*, vol. 13, no. 3, pp. 3902–3921, 2013.
8. M. E. Davies and Y. C. Eldar, “Rank awareness in joint sparse recovery,” *IEEE Transactions on Information Theory*, vol. 58, no. 2, pp. 1135–1146, 2012.
9. S. Li, D. Yang, G. Tang, and M. B. Wakin, “Atomic norm minimization for modal analysis from random and compressed samples,” *arXiv preprint arXiv:1703.00938*, 2017.
10. H. Lu, X. Long, and J. Lv, “A fast algorithm for recovery of jointly sparse vectors based on the alternating direction methods,” in *Proceedings of the Fourteenth International Conference on Artificial Intelligence and Statistics*, pp. 461–469, 2011.
11. Z. Jian, F. Yuli, Z. Qiheng, and L. Haifeng, “Split bregman algorithms for multiple measurement vector problem,” *Multidim Syst Sign Process*, 2015.



(a) Frame 1

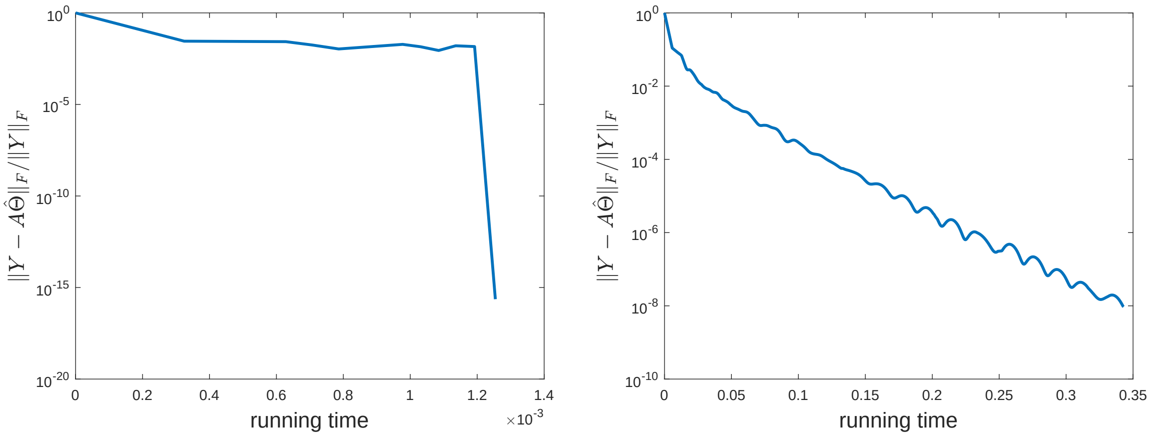
(b) Frame 2

(c) Frame 3

(d) Frame 4

Fig. 8 The first four frames reconstructed by BMStoGradMP and SBC. The relative reconstruction errors of all BMStoGradMP results, i.e., $\|X_{:,i} - \Psi\hat{\Theta}_{:,i}\|_2/\|X_{:,i}\|_2$, are: (a) 1.3819×10^{-15}, (b) 1.3819×10^{-15}, (c) 1.1547×10^{-15}, (d) 4.8385×10^{-15}. By contrast, the relative reconstruction errors of all SBC results are: (a) 5.0906×10^{-15}, (b) 5.0906×10^{-15}, (c) 6.8482×10^{-10}, and (d) 3.5012×10^{-8}.

12. P. Feng and Y. Bresler, “Spectrum-blind minimum-rate sampling and reconstruction of multiband signals,” in *Acoustics, Speech and Signal Processing (ICASSP), 1996 IEEE International Conference on*, vol. 3, pp. 1688–1691, IEEE, 1996.
13. J. M. Kim, O. K. Lee, and J. C. Ye, “Compressive MUSIC: Revisiting the link between compressive sensing and array signal processing,” *IEEE Transactions on Information Theory*, vol. 58, no. 1, pp. 278–301, 2012.
14. K. Lee, Y. Bresler, and M. Junge, “Subspace methods for joint sparse recovery,” *IEEE Transactions on Information Theory*, vol. 58, no. 6, pp. 3613–3641, 2012.
15. J. A. Tropp, “Greed is good: Algorithmic results for sparse approximation,” *IEEE Transactions on Information theory*, vol. 50, no. 10, pp. 2231–2242, 2004.
16. J. Chen and X. Huo, “Theoretical results on sparse representations of multiple-measurement vectors,” *IEEE Transactions on Signal Processing*, vol. 54, no. 12, pp. 4634–4643, 2006.
17. 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.
18. D. Needell and J. A. Tropp, “CoSaMP: Iterative signal recovery from incomplete and inaccurate samples,” *Applied and Computational Harmonic Analysis*, vol. 26, no. 3, pp. 301–321, 2009.
19. D. Needell and R. Vershynin, “Signal recovery from incomplete and inaccurate measurements via regularized orthogonal matching pursuit,” *IEEE Journal of selected topics in signal processing*, vol. 4, no. 2, pp. 310–316, 2010.
20. W. Dai and O. Milenkovic, “Subspace pursuit for compressive sensing signal reconstruction,” *IEEE Transactions on Information Theory*, vol. 55, no. 5, pp. 2230–2249, 2009.
21. T. Blumensath and M. E. Davies, “Iterative hard thresholding for compressed sensing,” *Applied and computational harmonic analysis*, vol. 27, no. 3, pp. 265–274, 2009.



(a) BMStoGradMP

(b) SBC

Fig. 9 Plot of the relative discrepancy $\|Y - A\hat{\Theta}\|_F / \|Y\|_F$ versus the running time for BMStoGradMP and SBC.

22. N. Nguyen, S. Chin, and T. Tran, "A unified iterative greedy algorithm for sparsityconstrainted optimization. 2013."
23. N. Nguyen, D. Needell, and T. Woolf, "Linear convergence of stochastic iterative greedy algorithms with sparse constraints," *IEEE Transactions on Information Theory*, 2017.
24. J. R. Magnus, "On the concept of matrix derivative," *Journal of Multivariate Analysis*, vol. 101, no. 9, pp. 2200–2206, 2010.
25. D. Needell and R. Ward, "Batched stochastic gradient descent with weighted sampling," in *Approximation Theory XV: San Antonio 2016*, pp. 279–306, Springer, 2016.
26. D. Baron, M. B. Wakin, M. F. Duarte, S. Sarvotham, and R. G. Baraniuk, "Distributed compressed sensing," *IEEE Transactions on Information Theory*, vol. 52, no. 12, pp. 5406–5425, 2006.
27. S. Foucart and H. Rauhut, *A mathematical introduction to compressive sensing*, vol. 1. Birkhäuser Basel, 2013.
28. M. Aharon, M. Elad, and A. Bruckstein, "K-svd: An algorithm for designing overcomplete dictionaries for sparse representation," *IEEE Transactions on signal processing*, vol. 54, no. 11, pp. 4311–4322, 2006.