GROUP SPARSE OPTIMIZATION BY ALTERNATING DIRECTION METHOD

WEI DENG, WOTAO YIN, AND YIN ZHANG*

April 19, 2011

Abstract. This paper proposes efficient algorithms for group sparse optimization with mixed $\ell_{2,1}$-regularization, which arises from the reconstruction of group sparse signals in compressive sensing, and the group Lasso problem in statistics and machine learning. It is known that encoding the group information in addition to sparsity will lead to better signal recovery/feature selection. The $\ell_{2,1}$-regularization promotes group sparsity, but the resulting problem, due to the mixed-norm structure and possible grouping irregularity, is considered more difficult to solve than the conventional ℓ_1-regularized problem. Our approach is based on a variable splitting strategy and the classic alternating direction method (ADM). Two algorithms are presented, one derived from the primal and the other from the dual of the $\ell_{2,1}$-regularized problem. The convergence of the proposed algorithms is guaranteed by the existing ADM theory. General group configurations such as overlapping groups and incomplete covers can be easily handled by our approach. Computational results show that on random problems the proposed ADM algorithms exhibit good efficiency, and strong stability and robustness.

1. Introduction. In the last few years, finding sparse solutions to underdetermined linear systems has become an active research topic, particularly in the area of compressive sensing, statistics and machine learning. Sparsity allows us to reconstruct high dimensional data with only a small number of samples. In order to further enhance the recoverability, recent studies propose to go beyond sparsity and take into account additional information about the underlying structure of the solutions. In practice, a wide class of solutions are known to have certain "group sparsity" structure. Namely, the solution has a natural grouping of its components, and the components within a group are likely to be either all zeros or all nonzeros. Encoding the group sparsity structure can reduce the degrees of freedom in the solution, thereby leading to better recovery performance.

This paper focuses on the reconstruction of group sparse solutions from underdetermined linear measurements, which is closely related with the Group Lasso problem [1] in statistics and machine learning. It leads to various applications such as multiple kernel learning [2], microarray data analysis [3], channel estimation in doubly dispersive multicarrier systems [4], etc. The group sparse reconstruction problem has been well studied recently. A favorable approach in the literature is to use the mixed $\ell_{2,1}$-regularization. Suppose $x \in \mathbb{R}^n$ is an unknown group sparse solution. Let $\{x_{g_i} \in \mathbb{R}^{n_i} : i = 1, ..., s\}$ be the grouping of x, where $g_i \subseteq \{1, 2, \ldots, n\}$ is an index set corresponding to the i-th group, and x_{g_i} denotes the subvector of x indexed by g_i. Generally, g_i's can be any index sets, and they are predefined based on prior knowledge. The $\ell_{2,1}$-norm is defined as follows:

$$\|x\|_{2,1} := \sum_{i=1}^{s} \|x_{g_i}\|_2. \tag{1.1}$$

Just like the use of ℓ_1-regularization for sparse recovery, the $\ell_{2,1}$-regularization is known to facilitate group

*Department of Computational and Applied Mathematics, Rice University, Houston, TX 77005 ({wei.deng, wotao.yin, yzhang}@rice.edu)

sparsity and result in a convex problem. However, the $\ell_{2,1}$-regularized problem is generally considered difficult to solve due to the non-smoothness and the mixed-norm structure. Although the $\ell_{2,1}$-problem can be formulated as a second-order cone programming (SOCP) problem or a semidefinite programming (SDP) problem, solving either SOCP or SDP by standard algorithms is computationally expensive. Several efficient first-order algorithms have been proposed, e.g., a spectral projected gradient method (SPGL1) [5], an accelerated gradient method (SLEP) [6], block-coordinate descent algorithms [7] and SpaRSA [8].

This paper proposes a new approach for solving the $\ell_{2,1}$-problem based on a variable splitting technique and the alternating direction method (ADM). Recently, ADM has been successfully applied to a variety of convex or nonconvex optimization problems, including ℓ_1-regularized problems [9], total variation (TV) regularized problems [10, 11], matrix factorization, completion and separation problems [12, 13, 14, 15]. Witnessing the versatility of ADM approach, we utilized it to tackle the $\ell_{2,1}$-regularized problem. We applied the ADM approach to both the primal and dual forms of the $\ell_{2,1}$-problem and obtained closed form solutions to all the resulting subproblems. Therefore, the derived algorithms have convergence guarantee according to the existing ADM theory. Preliminary numerical results demonstrate that our proposed algorithms are fast, stable and robust, outperforming the previously known state-of-the-art algorithms.

1.1. Notation and Problem Formulation. Throughout the paper, we let matrices be denoted by uppercase letters and vectors by lowercase letters. For a matrix X, we use x^i and x_j to represent its i-th row and j-th column respectively.

To be more general, instead of using the $\ell_{2,1}$-norm (1.1), we consider the weighted $\ell_{2,1}$- (or $\ell_{w,2,1}$-) norm defined by

$$\|x\|_{w,2,1} := \sum_{i=1}^{s} w_i \|x_{g_i}\|_2, \tag{1.2}$$

where $w_i \geq 0\,(i = 1, ..., s)$ are weights associated with each group. Based on prior knowledge, properly chosen weights may result in better recovery performance. For simplicity, we will assume the groups $\{x_{g_i} : i = 1, ..., s\}$ form a partition of x unless otherwise specified. In Section 3, we will show that our approach can be easily extended to general group configurations allowing overlapping and/or incomplete cover. Moreover, adding weights inside groups is also discussed in Section 3.

We consider the following basis pursuit (BP) model:

$$\begin{aligned} \min_{x} \quad & \|x\|_{w,2,1} \\ \text{s.t.} \quad & Ax = b, \end{aligned} \tag{1.3}$$

where $A \in \mathbb{R}^{m\times n}\,(m < n)$ and $b \in \mathbb{R}^m$. Without loss of generality, we assume A has full rank. When the measurement vector b contains noise, the basis pursuit denoising (BPDN) models are commonly used, including the constrained form:

$$\begin{aligned} \min_{x} \quad & \|x\|_{w,2,1} \\ \text{s.t.} \quad & \|Ax - b\|_2 \leq \sigma, \end{aligned} \tag{1.4}$$

and the unconstrained form:

$$\min_x \|x\|_{w,2,1} + \frac{1}{2\mu}\|Ax-b\|_2^2, \tag{1.5}$$

where $\sigma \geq 0$ and $\mu > 0$ are parameters. In this paper, we will stay focused on the basis pursuit model (1.3). The derivation of the ADM algorithms for the basis pursuit denoising models (1.4) and (1.5) follows similarly. Moreover, we emphasize that the basis pursuit model (1.3) is also good for noisy data if the iterations are stopped properly prior to convergence based on the noise level.

1.2. Outline of the Paper. The paper is organized as follows. Section 2 presents two ADM algorithms, one derived from the primal and the other from the dual of the $\ell_{w,2,1}$-problem, and states the convergence results following from the literature. For simplicity, Section 2 assumes the grouping is a partition of the solution. In Section 3, we generalize the group configurations to overlapping groups and incomplete cover, and discuss adding weights inside groups. Section 4 presents the ADM schemes for the jointly sparse recovery problem, also known as the multiple measurement vector (MMV) problem, as a special case of the group sparse recovery problem. In Section 5, we report numerical results on random problems and demonstrate the efficiency of the ADM algorithms in comparison with the state-of-the-art algorithm SPGL1.

2. ADM-based First-Order Primal-Dual Algorithms. In this section, we apply the classic alternating direction method (see, e.g., [16, 17]) to both the primal and dual forms of the $\ell_{w,2,1}$-problem (1.3). The derived algorithms are efficient first-order algorithms and are of primal-dual nature because both primal and dual variables are updated at each iteration. The convergence of the algorithms is established by the existing ADM theory.

2.1. Applying ADM to the Primal Problem. In order to apply ADM to the primal $\ell_{w,2,1}$-problem (1.3), we first introduce an auxiliary variable and transform it into an equivalent problem:

$$\begin{aligned} \min_{x,z} \quad & \|z\|_{w,2,1} = \sum_{i=1}^{s} w_i \|z_{g_i}\|_2 \\ \text{s.t.} \quad & z = x,\ Ax = b. \end{aligned} \tag{2.1}$$

Note that problem (2.1) has two blocks of variables (x and z) and its objective function is separable in the form of $f(x) + g(z)$ since it only involves z, thus ADM is applicable. The augmented Lagrangian problem is of the form

$$\min_{x,z} \|z\|_{w,2,1} - \lambda_1^T(z-x) + \frac{\beta_1}{2}\|z-x\|_2^2 - \lambda_2^T(Ax-b) + \frac{\beta_2}{2}\|Ax-b\|_2^2, \tag{2.2}$$

where $\lambda_1 \in \mathbb{R}^n, \lambda_2 \in \mathbb{R}^m$ are multipliers and $\beta_1, \beta_2 > 0$ are penalty parameters.

Then we apply the ADM approach, i.e., to minimize the augmented Lagrangian problem (2.2) with

respect to x and z alternately. The x-subproblem, namely minimizing (2.2) with respect to x, is given by

$$\begin{aligned} &\min_x \lambda_1^T x + \frac{\beta_1}{2}\|z-x\|_2^2 - \lambda_2^T Ax + \frac{\beta_2}{2}\|Ax-b\|_2^2 \\ \Leftrightarrow &\min_x \frac{1}{2}x^T(\beta_1 I + \beta_2 A^T A)x - (\beta_1 z - \lambda_1 + \beta_2 A^T b + A^T\lambda_2)^T x. \end{aligned} \tag{2.3}$$

Note that it is a convex quadratic problem, hence it reduces to solving the following linear system:

$$(\beta_1 I + \beta_2 A^T A)x = \beta_1 z - \lambda_1 + \beta_2 A^T b + A^T\lambda_2. \tag{2.4}$$

Minimizing (2.2) with respect to z gives the following z-subproblem:

$$\min_z \|z\|_{w,2,1} - \lambda_1^T z + \frac{\beta_1}{2}\|z-x\|_2^2. \tag{2.5}$$

Simple manipulation shows that (2.5) is equivalent to

$$\min_z \sum_{i=1}^{s}\left[w_i\|z_{g_i}\|_2 + \frac{\beta_1}{2}\|z_{g_i} - x_{g_i} - \frac{1}{\beta_1}(\lambda_1)_{g_i}\|_2^2\right], \tag{2.6}$$

which has a closed form solution by the one-dimensional shrinkage (or soft thresholding) formula:

$$z_{g_i} = \max\left\{\|r_i\|_2 - \frac{w_i}{\beta_1}, 0\right\}\frac{r_i}{\|r_i\|_2}, \text{ for } i = 1, ..., s, \tag{2.7}$$

where

$$r_i := x_{g_i} + \frac{1}{\beta_1}(\lambda_1)_{g_i}, \tag{2.8}$$

and the convention $0 \cdot \frac{0}{0} = 0$ is followed. We let the above group-wise shrinkage operation be denoted by $z = Shrink(x + \frac{1}{\beta_1}\lambda_1, \frac{1}{\beta_1}w)$ for short.

Finally, the multipliers λ_1 and λ_2 are updated in the standard way

$$\begin{cases} \lambda_1 \leftarrow \lambda_1 - \gamma_1\beta_1(z-x), \\ \lambda_2 \leftarrow \lambda_2 - \gamma_2\beta_2(Ax-b), \end{cases} \tag{2.9}$$

where γ_1, $\gamma_2 > 0$ are step lengths.

In short, we have derived an ADM iteration scheme for (2.1) as follows:

Algorithm 1: Primal-Based ADM for Group Sparsity

Initialize $z \in \mathbb{R}^n$, $\lambda_1 \in \mathbb{R}^n$, $\lambda_2 \in \mathbb{R}^m$, $\beta_1, \beta_2 > 0$ and $\gamma_1, \gamma_2 > 0$;
while *stopping criterion is not met* **do**
  $x \leftarrow (\beta_1 I + \beta_2 A^T A)^{-1}(\beta_1 z - \lambda_1 + \beta_2 A^T b + A^T \lambda_2)$;
  $z \leftarrow Shrink(x + \frac{1}{\beta_1}\lambda_1, \frac{1}{\beta_1} w)$ (group-wise);
  $\lambda_1 \leftarrow \lambda_1 - \gamma_1 \beta_1 (z - x)$;
  $\lambda_2 \leftarrow \lambda_2 - \gamma_2 \beta_2 (Ax - b)$;

Since the above ADM scheme computes the exact solution for each subproblem, its convergence is guaranteed by the existing ADM theory [16, 17]. We state the convergence result by the following theorem.

THEOREM 2.1. *For β_1, $\beta_2 > 0$ and γ_1, $\gamma_2 \in \left(0, \frac{\sqrt{5}+1}{2}\right)$, the sequence $\{(x^{(k)}, z^{(k)})\}$ generated by Algorithm 1 from any initial point $(x^{(0)}, z^{(0)})$ converges to (x^*, z^*), where (x^*, z^*) is a solution of (2.1).*

2.2. Applying ADM to the Dual Problem. Now we apply the ADM technique to the dual form of the $\ell_{w,2,1}$-problem (1.3) and derive an equally simple yet more efficient algorithm.

The dual of (1.3) is given by

$$\begin{aligned} &\max_y \left\{ \min_x \sum_{i=1}^s w_i \|x_{g_i}\|_2 - y^T(Ax - b) \right\} \\ =&\max_y \left\{ b^T y + \min_x \sum_{i=1}^s \left(w_i \|x_{g_i}\|_2 - y^T A_{g_i} x_{g_i} \right) \right\} \\ =&\max_y \left\{ b^T y : \|A_{g_i}^T y\|_2 \le w_i, \text{ for } i = 1, ..., s \right\}, \end{aligned} \tag{2.10}$$

where $y \in \mathbb{R}^m$, and A_{g_i} represents the submatrix collecting columns of A that corresponds to the i-th group.

Similarly, we introduce a splitting variable to reformulate it as a two-block separable problem:

$$\begin{aligned} \min_{y,z} \quad & -b^T y \\ \text{s.t.} \quad & z = A^T y, \\ & \|z_{g_i}\|_2 \le w_i, \ \text{ for } i = 1, ..., s, \end{aligned} \tag{2.11}$$

whose associated augmented Lagrangian problem is

$$\begin{aligned} \min_{y,z} \quad & -b^T y - x^T(z - A^T y) + \frac{\beta}{2}\|z - A^T y\|_2^2, \\ \text{s.t.} \quad & \|z_{g_i}\|_2 \le w_i, \text{ for } i = 1, ..., s, \end{aligned} \tag{2.12}$$

where $\beta > 0$ is a penalty parameter, $x \in \mathbb{R}^n$ is a multiplier and essentially the primal variable.

Then we apply the alternating minimization idea to (2.12). The y-subproblem is a convex quadratic

problem:

$$\min_y \ -b^T y + (Ax)^T y + \frac{\beta}{2}\|z - A^T y\|_2^2, \tag{2.13}$$

which can be further reduced to the following linear system:

$$\beta A A^T y = b - Ax + \beta A z. \tag{2.14}$$

The z-subproblem is given by

$$\begin{aligned} \min_z \quad & -x^T z + \frac{\beta}{2}\|z - A^T y\|_2^2, \\ \text{s.t.} \quad & \|z_{g_i}\|_2 \le w_i, \text{ for } i = 1, ..., s. \end{aligned} \tag{2.15}$$

We can equivalently reformulate it as

$$\begin{aligned} \min_z \quad & \sum_{i=1}^{s} \frac{\beta}{2}\|z_{g_i} - A_{g_i}^T y - \frac{1}{\beta}x_{g_i}\|_2^2, \\ \text{s.t.} \quad & \|z_{g_i}\|_2 \le w_i, \text{ for } i = 1, ..., s. \end{aligned} \tag{2.16}$$

It's easy to show that the solution to (2.16) is given by

$$z_{g_i} = \mathcal{P}_{\mathbf{B}_2^i}(A_{g_i}^T y + \frac{1}{\beta}x_{g_i}), \text{ for } i = 1, ..., s. \tag{2.17}$$

Here $\mathcal{P}$ represents a projection (in Euclidean norm) onto a convex set denoted as a subscript and $\mathbf{B}_2^i \triangleq \{z \in \mathbb{R}^{n_i} : \|z\|_2 \le w_i\}$. In short,

$$z = \mathcal{P}_{\mathbf{B}_2}(A^T y + \frac{1}{\beta}x), \tag{2.18}$$

where $\mathbf{B}_2 \triangleq \{z \in \mathbb{R}^n : \|z_{g_i}\|_2 \le w_i, \text{ for } i = 1, ..., s\}$. Finally we update the multiplier (i.e. the primal variable) x by

$$x \leftarrow x - \gamma\beta(z - A^T y), \tag{2.19}$$

where $\gamma > 0$ is a step length.

Therefore, we have derived an ADM iteration scheme for (2.11) as follows:

Algorithm 2: Dual-Based ADM for Group Sparsity

Initialize $x \in \mathbb{R}^n$, $z \in \mathbb{R}^n$, $\beta > 0$ and $\gamma > 0$;
while *stopping criterion is not met* **do**
  $y \leftarrow (\beta AA^T)^{-1}(b - Ax + \beta Az)$;
  $z \leftarrow \mathcal{P}_{\mathbf{B}_2}(A^T y + \frac{1}{\beta}x)$ (group-wise);
  $x \leftarrow x - \gamma\beta(z - A^T y)$;

Note that each subproblem is solved exactly in Algorithm 2. Hence the convergence result follows from [16, 17].

THEOREM 2.2. *For $\beta > 0$ and $\gamma \in \left(0, \frac{\sqrt{5}+1}{2}\right)$, the sequence $\{(x^{(k)}, y^{(k)}, z^{(k)})\}$ generated by Algorithm 2 from any initial point $(x^{(0)}, y^{(0)}, z^{(0)})$ converges to (x^*, y^*, z^*), where x^* is a solution of (1.3), and (y^*, z^*) is a solution of (2.11).*

2.3. Remarks. In the primal-based ADM scheme, the update of x is written in the form of solving an $n \times n$ linear system or inverting an $n \times n$ matrix. In fact, it can be reduced to solving a smaller $m \times m$ linear system or inverting an $m \times m$ matrix by Sherman-Morrison-Woodbury formula:

$$(\beta_1 I + \beta_2 A^T A)^{-1} = \frac{1}{\beta_1} I - \frac{\beta_2}{\beta_1} A^T (\beta_1 I + \beta_2 AA^T)^{-1} A. \tag{2.20}$$

Note that in many compressive sensing applications, A is formed by randomly taking a subset of rows from orthonormal transform matrices, e.g., the discrete cosine transform (DCT) matrix, the discrete Fourier transform (DFT) matrix and the discrete Walsh-Hadamard transform (DWHT) matrix. Then A has orthonormal rows, i.e., $AA^T = I$. In this case, there is no need to solve a linear system for either the primal-based ADM scheme or the dual-based scheme. The main computational cost becomes two matrix-vector multiplications per iteration for both schemes.

For general matrix A, solving the $m \times m$ linear system becomes the most costly part. However, we only need to compute the matrix inverse or do the matrix factorization once. Therefore, the computational cost per iteration is still $O(mn)$. For large problems when solving such an $m \times m$ linear system is no longer affordable, we may just take a steepest descent step instead. In this case, the subproblem is solved inexactly. Hence its convergence remains an issue for further research. However, empirical evidence shows that the algorithms still converge well. By taking a steepest descent step, our ADM algorithms only involve matrix-vector multiplications. Consequently, A can be accepted as two linear operators $A*(\cdot)$ and $A^T*(\cdot)$, and the storage of the matrix A may not be needed.

3. Several Extensions. So far, we have presumed that the grouping $\{x_{g_1}, ..., x_{g_s}\}$ in the problem formulation (1.3) is a non-overlapping cover of x. It is of practical importance to consider more general group configurations such as overlapping groups and incomplete cover. Furthermore, it can be desirable to introduce weights inside each group for better scaling. In this section, we will demonstrate that our approach can be easily extended to these general settings.

3.1. Overlapping Groups. Overlapping group structure commonly arises in many applications. For instance, in microarray data analysis, gene expression data are known to form overlapping groups since each gene may participate in multiple functional groups [18]. The weighted $\ell_{2,1}$-regularization is still applicable yielding the same formulation as in (1.3). However, $\{x_{g_1}, ..., x_{g_s}\}$ now may have overlaps, which makes the problem more challenging to solve. As we will show, our approach can handle this difficulty.

Using the same strategy as before, we first introduce auxiliary variables z_i's and let $z_i = x_{g_i}$ $(i = 1, \ldots, s)$, yielding the following equivalent problem:

$$\begin{aligned} \min_{x,z} \quad & \sum_{i=1}^{s} w_i \|z_i\|_2 \\ \text{s.t.} \quad & z = \tilde{x},\ Ax = b, \end{aligned} \tag{3.1}$$

where $z = [z_1^T, \ldots, z_s^T]^T \in \mathbb{R}^{\tilde{n}}$, $\tilde{x} = [x_{g_1}^T, \ldots, x_{g_s}^T]^T \in \mathbb{R}^{\tilde{n}}$ and $\tilde{n} = \sum_{i=1}^{s} n_i \geq n$. The augmented Lagrangian problem is of the form:

$$\min_{x,z} \sum_{i=1}^{s} w_i \|z_i\|_2 - \lambda_1^T (z - \tilde{x}) + \frac{\beta_1}{2} \|z - \tilde{x}\|_2^2 - \lambda_2^T (Ax - b) + \frac{\beta_2}{2} \|Ax - b\|_2^2, \tag{3.2}$$

where $\lambda_1 \in \mathbb{R}^{\tilde{n}}$, $\lambda_2 \in \mathbb{R}^m$ are multipliers, and β_1, $\beta_2 > 0$ are penalty parameters.

Then we perform alternating minimization in x and z directions. The benefit from our variable splitting technique is that the weighted $\ell_{2,1}$-regularization term no longer contains overlapping groups of variables x_{g_i}'s. Instead, it only involves z_i's, which do not overlap, thereby allowing us to easily perform exact minimization for the z-subproblem just as the non-overlapping case. The closed form solution of the z-subproblem is given by the shrinkage formula for each group of variables z_i, the same as in (2.7) and (2.8). We note that the x-subproblem is a convex quadratic problem. Thus, the overlapping feature of x does not bring much difficulty. Clearly, $\tilde{x}$ can be represented by

$$\tilde{x} = Gx, \tag{3.3}$$

and each row of $G \in \mathbb{R}^{\tilde{n} \times n}$ has a single 1 and 0's elsewhere. The x-subproblem is given by

$$\min_x \lambda_1^T Gx + \frac{\beta_1}{2} \|z - Gx\|_2^2 - \lambda_2^T Ax + \frac{\beta_2}{2} \|Ax - b\|_2^2, \tag{3.4}$$

which is equivalent to solving the following linear system:

$$(\beta_1 G^T G + \beta_2 A^T A)x = \beta_1 G^T z - G^T \lambda_1 + \beta_2 A^T b + A^T \lambda_2. \tag{3.5}$$

Note that $G^T G \in \mathbb{R}^{n \times n}$ is a diagonal matrix whose i-th diagonal entry is the number of repetitions of x_i in $\tilde{x}$. When the groups form an complete cover of the solution, the diagonal entries of $G^T G$ will be positive, so $G^T G$ is invertible. In the next subsection, we will show that an incomplete cover case can be converted to a complete cover case by introducing an auxiliary group. Therefore, we can generally assume $G^T G$ is invertible. Then Sherman-Morrison-Woodbury formula is applicable, and solving this $n \times n$ linear system

can be can further reduced to solving an $m \times m$ linear system.

We can also formulate the dual problem of (3.1) as follows:

$$\begin{aligned}
&\max_{y,p} \left\{ \min_{x,z} \sum_{i=1}^{s} w_i \|z_{g_i}\|_2 - y^T(Ax-b) - p^T(z-Gx) \right\} \\
=&\max_{y,p} \left\{ b^T y + \min_{z} \sum_{i=1}^{s} \left(w_i \|z_{g_i}\|_2 - p_i^T z_i \right) + \min_{x} \left(-A^T y + G^T p \right)^T x \right\} \\
=&\max_{y,p} \left\{ b^T y : G^T p = A^T y,\ \|p_i\|_2 \le w_i, \text{ for } i = 1, ..., s \right\}, \qquad (3.6)
\end{aligned}$$

where $y \in \mathbb{R}^m$, $p = [p_1^T, \ldots, p_s^T]^T \in \mathbb{R}^{\tilde{n}}$ and $p_i \in \mathbb{R}^{n_i}$ $(i = 1, \ldots, s)$.

We introduce an splitting variable $q \in \mathbb{R}^{\tilde{n}}$ and obtain an equivalent problem:

$$\begin{aligned}
\min_{y,p,q} \quad & -b^T y \qquad (3.7) \\
\text{s.t.} \quad & G^T p = A^T y, \\
& p = q, \\
& \|q_i\|_2 \le w_i, \ \text{ for } i = 1, ..., s.
\end{aligned}$$

Likewise, we minimize its augmented Lagrangian by the alternating direction method. Notice that the (y, p)-subproblem is a convex quadratic problem, and the q-subproblem has a closed form solution by projection onto ℓ_2-norm balls. Therefore, a similar dual-based ADM algorithm can be derived. For the sake of brevity, we omit the derivation here.

3.2. Incomplete Cover. In some applications such as group sparse logistic regression, the groups may be an incomplete cover of the solution because only partial components are sparse. This case can be easily dealt with by introducing a new group containing the uncovered components, i.e., letting $\bar{g} = \{1, \ldots, n\} \setminus \cup_{i=1}^{s} g_i$. Then we can include this group $\bar{g}$ in the $\ell_{w,2,1}$-regularization and associate it with a zero or tiny weight.

3.3. Weights Inside Groups. Although we have considered an weighted version of the $\ell_{2,1}$-norm (1.2), the weights are only added between the groups. In other words, components within a group are associated with the same weight. In applications such as multi-modal sensing/classification, components of each group are likely to have a large dynamic range. Introducing weights inside each group can balance the different scales of the components, thereby improving the accuracy and stability of the reconstruction.

Thus, we consider the weighted ℓ_2-norm in place of the ℓ_2-norm in the definition of $\ell_{w,2,1}$-norm (1.2). For $x \in \mathbb{R}^n$, the weighted ℓ_2-norm is given by

$$\|x\|_{\bar{W},2} := \|\bar{W}x\|_2, \qquad (3.8)$$

where $\bar{W} = diag([\bar{w}_1, \ldots, \bar{w}_n])$ is a diagonal matrix with weights on its diagonal and $\bar{w}_i > 0\,(i = 1, \ldots, n)$.

With weights inside each group, the problem (1.3) becomes

$$\begin{aligned} \min_{x} \quad & \sum_{i=1}^{s} w_i \|W^{(i)} x_{g_i}\|_2 \\ \text{s.t.} \quad & Ax = b, \end{aligned} \tag{3.9}$$

where $W^{(i)} \in \mathbb{R}^{n_i \times n_i}$ is a diagonal weight matrix for the i-th group. After a change of variable by letting $z_i = W^{(i)} x_{g_i}$ $(i = 1, \ldots, s)$, it can be reformulated as

$$\begin{aligned} \min_{z} \quad & \sum_{i=1}^{s} w_i \|z_i\|_2 \\ \text{s.t.} \quad & z = WGx,\ Ax = b, \end{aligned} \tag{3.10}$$

where $z = [z_1^T, \ldots, z_s^T]^T \in \mathbb{R}^{\tilde{n}}$, $Gx = [x_{g_1}^T, \ldots, x_{g_s}^T]^T \in \mathbb{R}^{\tilde{n}}$, $\tilde{n} = \sum_{i=1}^{s} n_i \geq n$ and

$$W := \begin{bmatrix} W^{(1)} & & & \\ & W^{(2)} & & \\ & & \ddots & \\ & & & W^{(s)} \end{bmatrix}.$$

Then the problem can be addressed within our framework.

4. Joint Sparsity. Now we study an interesting special case of the group sparsity structure called joint sparsity. Jointly sparse solutions, namely, a set of sparse solutions that share a common nonzero support, arise in cognitive radio networks [19], distributed compressive sensing [20], direction-of-arrival estimation in radar [21], magnetic resonance imaging with multiple coils [22] and many other applications. The reconstruction of jointly sparse solutions, also known as the multiple measurement vector (MMV) problem, has its origin in sensor array signal processing and recently has received much interest as an extension of the single sparse solution recovery in compressive sensing.

The (weighted) $\ell_{2,1}$-regularization has been popularly used to encode the joint sparsity, given by

$$\begin{aligned} \min_{X} \quad & \|X\|_{w,2,1} := \sum_{i=1}^{n} w_i \|x^i\|_2 \\ \text{s.t.} \quad & AX = B, \end{aligned} \tag{4.1}$$

where $X = [x_1, \ldots, x_l] \in \mathbb{R}^{n \times l}$ denotes a collection of l jointly sparse solutions, $A \in \mathbb{R}^{m \times n}$ $(m < n)$, $B \in \mathbb{R}^{m \times l}$ and $w_i \geq 0$ for $i = 1, ..., n$. Recall that x^i and x_j denote the i-th row and j-th column of X respectively.

Indeed, joint sparsity can be viewed as a special non-overlapping group sparsity structure with each group containing one row of the solution matrix. Clearly, the joint $\ell_{w,2,1}$-norm given in (4.1) is consistent with the definition of group $\ell_{w,2,1}$-norm (1.3). Further, we can cast problem (4.1) in the form of a group

sparsity problem. Let us define

$$\tilde{A} := I_l \otimes A = \begin{bmatrix} A & & & \\ & A & & \\ & & \ddots & \\ & & & A \end{bmatrix}, \quad \tilde{x} := vec(X) = \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_l \end{bmatrix} \quad \text{and} \quad \tilde{b} := vec(B) = \begin{bmatrix} b_1 \\ b_2 \\ \vdots \\ b_l \end{bmatrix}, \tag{4.2}$$

where $I_l \in \mathbb{R}^{l\times l}$ is the identity matrix, $vec(\cdot)$ and $\otimes$ are standard notations for the vectorization of a matrix and the Kronecker product respectively. We partition $\tilde{x}$ into n groups $\{x_{g_1}, ..., x_{g_n}\}$ where $x_{g_i} \in \mathbb{R}^l\, (i = 1, \ldots, n)$ corresponds to the i-th row of the matrix X. Then it is easy to see that problem (4.1) is equivalent to the following group $\ell_{w,2,1}$-problem:

$$\begin{aligned} \min_{\tilde{x}} \quad & \|\tilde{x}\|_{w,2,1} := \sum_{i=1}^{n} w_i \|\tilde{x}_{g_i}\|_2 \\ \text{s.t.} \quad & \tilde{A}\tilde{x} = \tilde{b}. \end{aligned} \tag{4.3}$$

Moreover, under the joint sparsity setting, our primal-based ADM scheme for (4.3) has the following form:

$$\begin{cases} X \leftarrow (\beta_1 I + \beta_2 A^T A)^{-1}(\beta_1 Z - \Lambda_1 + \beta_2 A^T B + A^T \Lambda_2), \\ Z \leftarrow Shrink(X + \frac{1}{\beta_1}\Lambda_1, \frac{1}{\beta_1} w)\,(\text{row-wise}), \\ \Lambda_1 \leftarrow \Lambda_1 - \gamma_1\beta_1(Z - X), \\ \Lambda_2 \leftarrow \Lambda_2 - \gamma_2\beta_2(AX - B). \end{cases} \tag{4.4}$$

Here $\Lambda_1 \in \mathbb{R}^{n\times l}$, $\Lambda_2 \in \mathbb{R}^{m\times l}$ are multipliers, β_1, $\beta_2 > 0$ are penalty parameters, γ_1, $\gamma_2 > 0$ are step lengths and the updating of Z by row-wise shrinkage represents

$$z^i = \max\left\{\|r^i\|_2 - \frac{w_i}{\beta_1}, 0\right\} \frac{r^i}{\|r^i\|_2}, \ \text{ for } i = 1, ..., n, \tag{4.5}$$

where

$$r^i := x^i + \frac{1}{\beta_1}\lambda_1^i. \tag{4.6}$$

Correspondingly, the dual of (4.1) is given by

$$\begin{aligned} \max_{Y} \quad & B \bullet Y \\ \text{s.t.} \quad & \|A_i^T Y\|_2 \le w_i, \ \text{ for } i = 1, ..., n, \end{aligned} \tag{4.7}$$

where "$\bullet$" denotes the sum of component-wise products. And the dual-based ADM scheme for the joint

sparsity problem is of the following form:

$$\begin{cases} Y \leftarrow (\beta AA^T)^{-1}(B - AX + \beta AZ), \\ Z \leftarrow \mathcal{P}_{\mathbf{B}_2'}(A^T Y + \frac{1}{\beta} X) \ \text{(row-wise)}, \\ X \leftarrow X - \gamma\beta(Z - A^T Y). \end{cases} \tag{4.8}$$

Here $\beta > 0$ and $\gamma > 0$ are the penalty parameter and the step length respectively as before, X is the primal variable and $\mathbf{B}_2' := \{Z \in \mathbb{R}^{n\times l} : \|z^i\|_2 \le w_i, \ \text{for } i = 1, ..., n\}$.

In addition, we can consider a more generalized joint sparsity scenario where each column of the solution matrix corresponds to a different measurement matrix. Specifically, we consider the following problem:

$$\begin{aligned} \min_X \quad & \|X\|_{w,2,1} := \sum_{i=1}^{n} w_i \|x^i\|_2 \\ \text{s.t.} \quad & A_j x_j = b_j, \ j = 1, \ldots, l, \end{aligned} \tag{4.9}$$

where $X \in \mathbb{R}^{n\times l}$, $A_j \in \mathbb{R}^{m_j \times n}$ $(m_j < n)$, $b_j \in \mathbb{R}^{m_j \times l}$ and $w_i \ge 0$ for $i = 1, ..., n$. Likewise, we can reformulate it in the form of (4.3), just replacing $\tilde{A}$ in (4.2) by

$$\tilde{A} := \begin{bmatrix} A_1 & & & \\ & A_2 & & \\ & & \ddots & \\ & & & A_l \end{bmatrix}, \tag{4.10}$$

and deal with it as a group sparsity problem.

5. Numerical Experiments. In this section, we present numerical results to evaluate the performance of our proposed ADM algorithms in comparison with the state-of-the-art algorithm SPGL1 (version 1.7) [5]. We tested them on two sets of synthetic data with group sparse solutions and jointly sparse solutions, respectively. Both speed and solution quality are compared. The numerical experiments were run in MATLAB 7.10.0 on a Dell desktop with an Intel Core 2 Duo 2.80GHz CPU and 2GB of memory.

Several other existing algorithms such as SpaRSA [8], SLEP [6] and block-coordinate descent algorithms (BCD) [7] have not been included in these experiments for the following reasons. Unlike ADMs and SPGL1 that directly solve the constrained models (1.3) and (1.4), SpaRSA, SLEP and BCD are all designed to solve the unconstrained problem (1.5). For the unconstrained problem, the choice of the penalty parameter μ is a critical issue, affecting both the reconstruction speed and accuracy of these algorithms. In order to conduct fair comparison, it is important to make a good choice of the penalty parameter. However, it is usually difficult to choose and may need heuristic techniques such as continuation. In addition, we notice that current versions of both SLEP and BCD cannot accept the measurement matrix A as an operator. Therefore, we mainly compare the ADM algorithms with SPGL1 in the experiments, which will provide good insight on the behavior of the ADM algorithms. More comprehensive numerical experiments will be conducted in the future.

5.1. Group Sparsity Experiment. In this experiment, we generate group sparse solutions as follows: we first randomly partition an n-vector x into s groups and then randomly pick k of them as active groups whose entries are *iid* random Gaussian while the remaining groups are all zeros. We use randomized partial Walsh-Hadamard transform matrices as measurement matrices $A \in \mathbb{R}^{m\times n}$. These transform matrices are suitable for large-scale computation and have the property $AA^T = I$. Fast matrix-vector multiplications with partial Walsh-Hadamard matrix A and its transpose A^T are implemented in C with a MATLAB mex-interface available to all codes compared. We emphasize that on matrices other than Walsh-Hadamard, similar comparison results are obtained. The problem size is set to $n = 8192$, $m = 2048$ and $s = 1024$. We test on both noiseless measurement data $b = Ax$ and noisy measurement data with 0.5% additive Gaussian noise.

We set the parameters for the primal-based ADM algorithm (PADM) as follows: $\beta_1 = 0.3/mean(abs(b))$, $\beta_2 = 3/mean(abs(b))$ and $\gamma_1 = \gamma_2 = 1.618$. Here we use the MATLAB-type notation $mean(abs(b))$ to denote the arithmetic average of the absolute value of b. For the dual-base ADM algorithm (DADM), we set $\beta = 2 * mean(abs(b))$ and $\gamma = 1.618$. Recall that the step length being $1.618 \approx (\sqrt{5}+1)/2$ is the upper bound for theoretical convergence guarantee. We use the default parameter setting for SPGL1 except setting proper tolerance values for different tests. Notice that SPGL1 is designed to solve the constrained denoising model (1.4), we set its input argument *sigma* ideally to the true noise magnitude. All the algorithms use zero as the starting point. The weights in the $\ell_{w,2,1}$-norm are set to one.

We present two comparison results. Figure 5.1 shows the decreasing behavior of relative error as each algorithm proceeds for 1000 iterations. Here the number of active groups is fixed at $k = 100$. Figure 5.2 illustrates the performance of each algorithm in terms of relative error, running time and number of iterations as k varies from 70 to 110. The ADM algorithms are terminated when $\|x^{(k+1)} - x^{(k)}\| < tol \cdot \|x^{(k)}\|$, i.e., the relative change of two consecutive iterates becomes smaller than the tolerance. The tolerance value *tol* is set to 10^{-6} for noiseless data and 5×10^{-4} for noisy data with 5% Gaussian noise. SPGL1 has more sophisticated stopping criteria. In order to make a fair comparison, we let SPGL1 reach roughly similar accuracy as PADM and DADM. We empirically tuned the tolerance parameters of SPGL1, namely *bpTol*, *decTol* and *optTol*, for different sparsity levels, since we found it difficult to use a consistent way of setting the tolerance parameters. Specifically, we chose a decreasing sequence of tolerance values as the sparsity level increases. Therefore, all the algorithms achieved comparable relative errors.

5.2. Joint Sparsity Experiment. Jointly sparse solutions $X \in \mathbb{R}^{n\times l}$ are generated by randomly selecting k rows to have *iid* random Gaussian entries and letting the other rows to be zero. Randomized partial Walsh-Hadamard transform matrices are utilized as measurement matrices $A \in \mathbb{R}^{m\times n}$. Here, we set $n = 1024$, $m = 256$ and $l = 16$. The parameter setting for each algorithm is the same as described in the previous section 5.1.

Similarly, we perform two classes of numerical tests. In one test, we fix the number of nonzero rows $k = 115$ and study the decreasing rate of relative error for each algorithm, as is shown in Figure 5.3. In the other test, we set proper stopping tolerance values as described in section 5.1, thus all the algorithms will reach comparable relative error. Then we compare the CPU time and number of iterations consumed by each algorithm for different sparsity levels from $k = 100$ to 120. The result is presented in Figure 5.4.

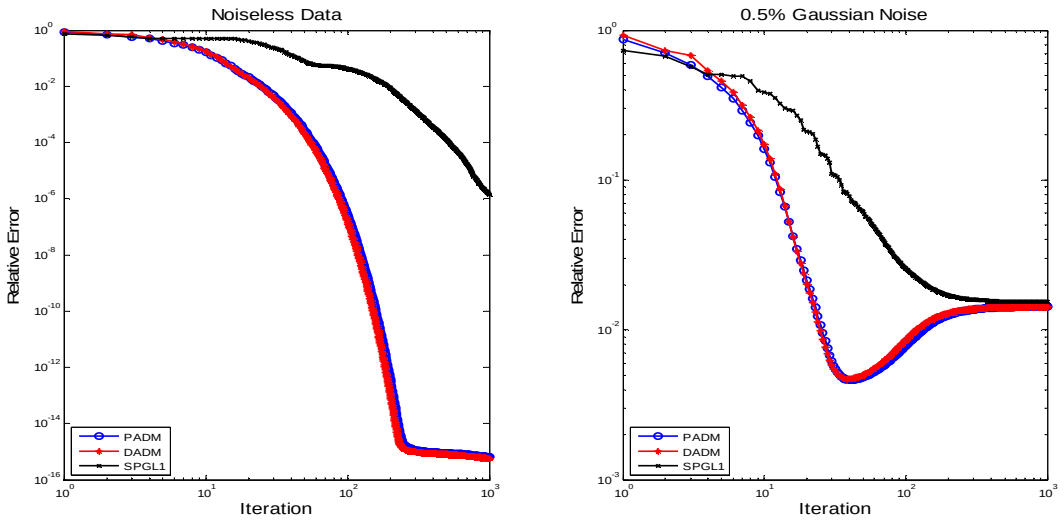

FIG. 5.1. *Convergence rate results of PADM, DADM and SPGL1 on $\ell_{2,1}$-regularized group sparsity problem ($n = 8192$, $m = 2048$, $s = 1024$ and $k = 100$). The x-axes represent number of iterations, and the y-axes represent relative error. The left plot corresponds to noiseless data and the right plot corresponds to data with 0.5% Gaussian noise. The results are average of 50 runs.*

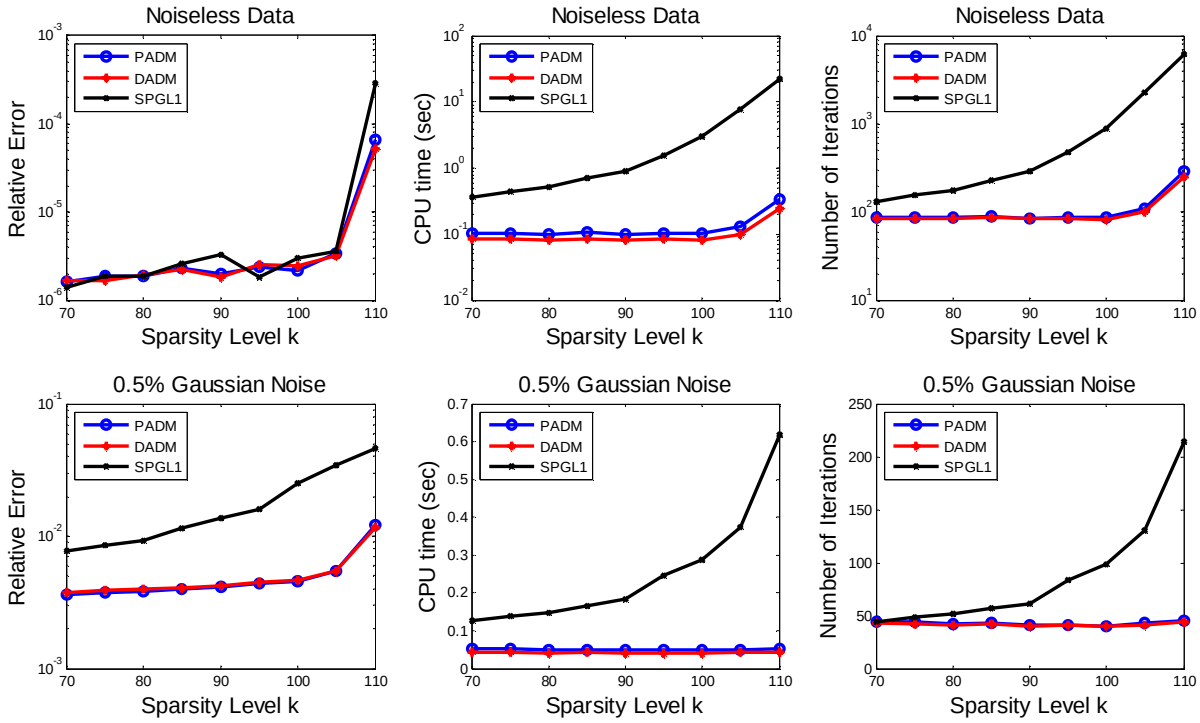

FIG. 5.2. *Comparison results of PADM, DADM and SPGL1 on $\ell_{2,1}$-regularized group sparsity problem ($n = 8192$, $m = 2048$ and $s = 1024$). The x-axes represent number of nonzero groups, and the y-axes represent relative error, CPU time and number of iterations (from left to right). The top row corresponds to noiseless data and the bottom row corresponds to data with 0.5% Gaussian noise. The results are average of 50 runs.*

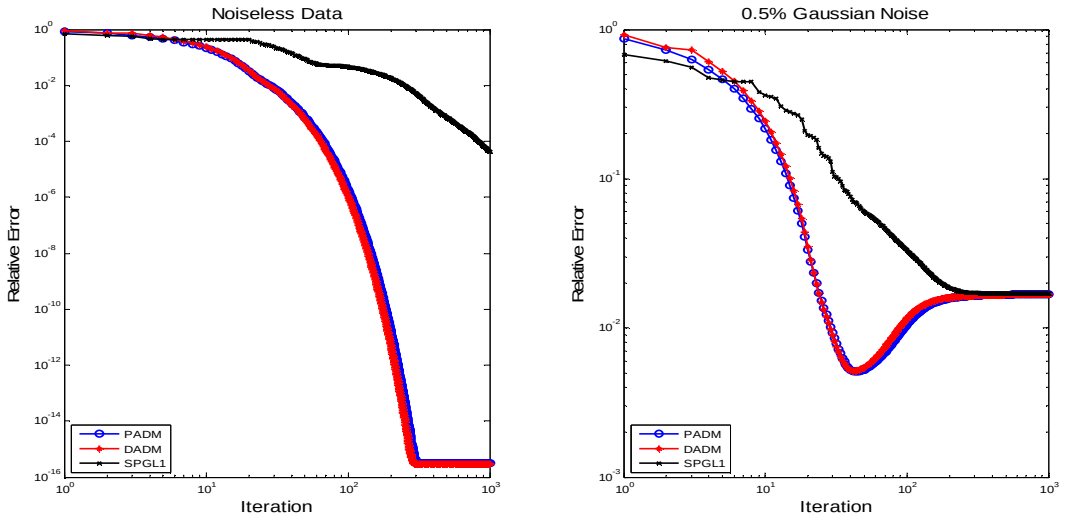

FIG. 5.3. *Convergence rate results of PADM, DADM and SPGL1 on $\ell_{2,1}$-regularized joint sparsity problem ($n = 1024$, $m = 256$, $l = 16$ and $k = 115$). The x-axes represent number of iterations and the y-axes represent relative error. The left plot corresponds to noiseless data and the right plot corresponds to data with 0.5% Gaussian noise. The results are average of 50 runs.*

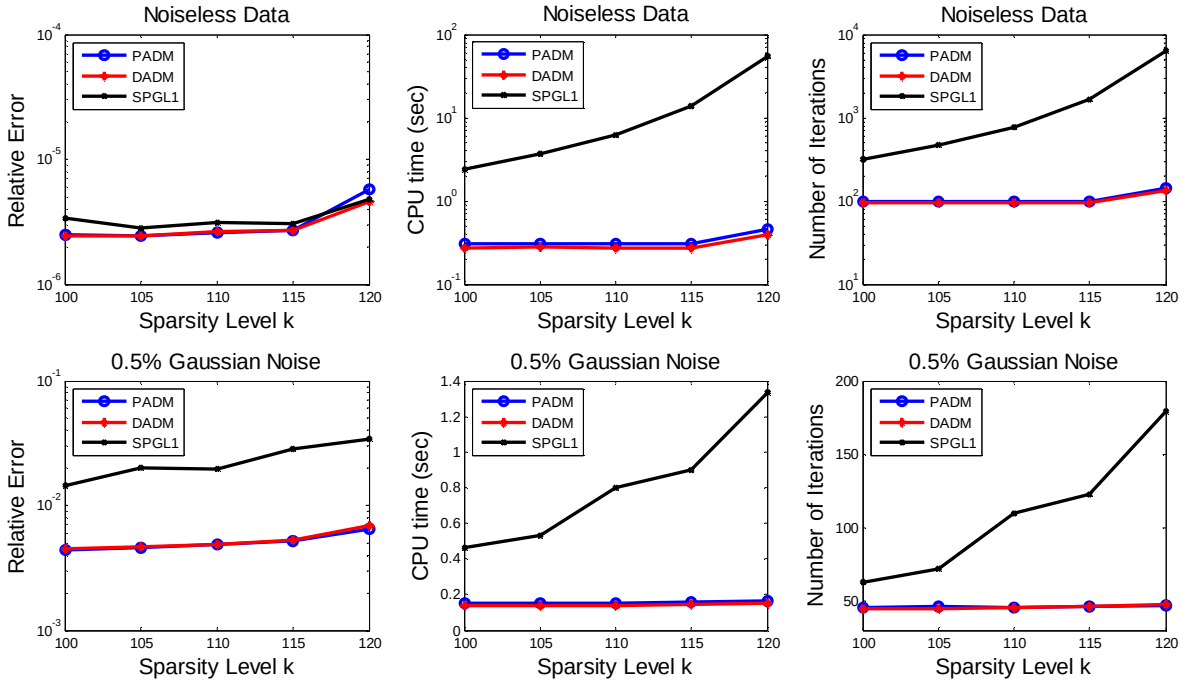

FIG. 5.4. *Comparison results of PADM, DADM and SPGL1 on $\ell_{2,1}$-regularized joint sparsity problem ($n = 1024$, $m = 256$ and $l = 16$). The x-axes represent number of nonzero rows, and the y-axes represent relative error, CPU time and number of iterations (from left to right). The top row corresponds to noiseless data and the bottom row corresponds to data with 0.5% Gaussian noise. The results are average of 50 runs.*

5.3. Discussions. As we can see from both Figure 5.1 and Figure 5.3, the relative error curves produced by PADM and DADM almost coincide and fall quickly below the one by SPGL1 in either noiseless or noisy case. In other words, the ADM algorithms decrease the relative error much faster than SPGL1. With noiseless data, the ADM algorithms reach machine precision 10^{-16} after $200 \sim 300$ iterations, whereas SPGL1 attains only $10^{-5} \sim 10^{-6}$ accuracy after 1000 iterations. Although SPGL1 will reach machine precision eventually, it needs far more iterations.

When the data contains noise, high accuracy is generally not achievable. With 0.5% additive Gaussian noise, all the algorithms converge to the same relative error level around 10^{-2}. However, we can observe that the ADM algorithms and SPGL1 have different solution paths. While SPGL1 decreases the relative error almost monotonically, the relative error curves of the ADM algorithms have a "down-then-up" behavior. Specifically, their relative error curves first go down quickly and reach the lowest level around 5×10^{-3}, but then they start to go up a bit until convergence. This "down-then-up" phenomenon is because the optima of the $\ell_{2,1}$-problem with erroneous data may not necessarily yield the best solution quality. In fact, the ADM algorithms still keep decreasing the objective values even though the relative errors start to increase. In other words, the ADM algorithms may give a better solution if it is stopped properly prior to convergence. We can see that SPGL1 takes approximately 200 iterations to decrease the relative error to 10^{-2}, while the ADM algorithms need only 30 iterations to reach even higher accuracy.

To further assess the efficiency of the ADM algorithms, we study the comparison results on relative error, CPU time and number of iterations for different sparsity levels. As can be seen in Figure 5.2 and Figure 5.4, PADM and DADM have very similar performance though DADM is often slightly faster. They both exhibit good stability, attaining the desired accuracy with roughly the same number of iterations over different sparsity levels. Although SPGL1 can also stably reach comparable accuracy, it consumes substantially more iterations as the sparsity level increases. For noisy data, the ADM algorithms obtain a bit higher accuracy than SPGL1, which is due to the different solution paths as shown in Figure 5.1 and Figure 5.3. For the ADM algorithms, recall that the stopping tolerance for relative change is set to 5×10^{-4}. Using similar tolerance values that are consistent with the noise level, the ADM algorithms are often terminated near the point with the lowest relative error. However, SPGL1 could hardly further lower its relative error by using different tolerance values.

Moreover, we can observe that the speed advantage of the ADM algorithms over SPGL1 is significant. Notice that the dominating computational load for all the algorithms are matrix-vector multiplications. For both PADM and DADM, the number of matrix-vector multiplications are two per iteration. The number used by SPGL1 may vary in each iteration, usually more than two per iteration on average. Compared to SPGL1, the ADM algorithms not only consume fewer iterations to obtain the same or even higher accuracy, but are also less computationally expensive at each iteration. Therefore, the ADM algorithms are much faster in terms of CPU time, especially as sparsity level increases. For noiseless data, we observe that the ADM algorithms are $2 \sim 3$ orders of magnitude faster than SPGL1. From Figure 5.1 and Figure 5.3, it is clear to see that the speed advantage will be even more significant for higher accuracy. For noisy data, the ADM algorithms gain $3 \sim 8$ times speed up over SPGL1.

6. Conclusion. We have proposed efficient alternating direction methods for group sparse optimization using $\ell_{2,1}$-regularization. General group configurations such as overlapping groups and incomplete cover are allowed. The convergence of these ADM algorithms are guaranteed by the existing theory if one minimizes a convex quadratic function exactly at each iteration. When the measurement matrix A is a partial transform matrix that has orthonormal rows, the main computational cost is only two matrix-vector multiplications per iteration. In addition, such a matrix A can be treated as a linear operator without explicit storage, which is particularly desirable for large-scale computation. For a general matrix A, solving a linear system is additionally needed. Alternatively, we may choose to minimize the quadratics approximately, e.g, by taking a steepest descent step. Empirical evidence has led us to believe that for this latter case, convergence guarantee should still hold under certain conditions on the step lengths γ_1, γ_2 (or γ). Our numerical results have demonstrated the effectiveness of the ADM algorithms for group and joint sparse solution reconstructions. In particular, our implementations of the ADM algorithms exhibit a clear and significant speed advantage over the state-of-the-art solver SPGL1. Moreover, it has been observed that at least on random problems ADM algorithms are capable of achieving a higher solution quality than SPGL1 can when data contains noise.

Acknowledgments. We would like to thank Dr. Ewout van den Berg for clarifying the parameters of SPGL1. The work of Wei Deng was supported in part by NSF Grant DMS-0811188 and ONR Grant N00014-08-1-1101. The work of Wotao Yin was supported in part by NSF career award DMS-07-48839, NSF ECCS-1028790, ONR Grant N00014-08-1-1101, and an Alfred P. Sloan Research Fellowship. The work of Yin Zhang was supported in part by NSF Grant DMS-0811188 and ONR Grant N00014-08-1-1101.

REFERENCES

[1] M. Yuan and Y. Lin, "Model selection and estimation in regression with grouped variables," *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, vol. 68, no. 1, pp. 49–67, 2006.

[2] F. Bach, "Consistency of the group Lasso and multiple kernel learning," *The Journal of Machine Learning Research*, vol. 9, pp. 1179–1225, 2008.

[3] S. Ma, X. Song, and J. Huang, "Supervised group Lasso with applications to microarray data analysis," *BMC bioinformatics*, vol. 8, no. 1, p. 60, 2007.

[4] D. Eiwen, G. Taubock, F. Hlawatsch, and H. Feichtinger, "Group Sparsity Methods For Compressive Channel Estimation In Doubly Dispersive Multicarrier Systems," 2010.

[5] E. van den Berg, M. Schmidt, M. Friedlander, and K. Murphy, "Group sparsity via linear-time projection," *Dept. Comput. Sci., Univ. British Columbia, Vancouver, BC, Canada*, 2008.

[6] J. Liu, S. Ji, and J. Ye, "SLEP: Sparse Learning with Efficient Projections," *Arizona State University*, 2009.

[7] Z. Qin, K. Scheinberg, and D. Goldfarb, "Efficient Block-coordinate Descent Algorithms for the Group Lasso," 2010.

[8] S. Wright, R. Nowak, and M. Figueiredo, "Sparse reconstruction by separable approximation," *Signal Processing, IEEE Transactions on*, vol. 57, no. 7, pp. 2479–2493, 2009.

[9] J. Yang and Y. Zhang, "Alternating Direction Algorithms for ℓ_1-Problems in Compressive Sensing," *Arxiv preprint arXiv:0912.1185*, 2009.

[10] Y. Wang, J. Yang, W. Yin, and Y. Zhang, "A new alternating minimization algorithm for total variation image reconstruction," *SIAM Journal on Imaging Sciences*, vol. 1, no. 3, pp. 248–272, 2008.

[11] J. Yang, W. Yin, Y. Zhang, and Y. Wang, "A fast algorithm for edge-preserving variational multichannel image restoration," *SIAM Journal on Imaging Sciences*, vol. 2, no. 2, pp. 569–592, 2009.

[12] Y. Zhang, "An Alternating Direction Algorithm for Nonnegative Matrix Factorization," *TR10-03, Rice University*, 2010.

[13] Z. Wen, W. Yin, and Y. Zhang, "Solving a low-rank factorization model for matrix completion by a nonlinear successive over-relaxation algorithm," *TR10-07, Rice University*, 2010.

[14] Y. Shen, Z. Wen, and Y. Zhang, "Augmented Lagrangian alternating direction method for matrix separation based on low-rank factorization," *TR11-02, Rice University*, 2011.

[15] Y. Xu, W. Yin, Z. Wen, and Y. Zhang, "An Alternating Direction Algorithm for Matrix Completion with Nonnegative Factors," *Arxiv preprint arXiv:1103.1168*, 2011.

[16] R. Glowinski and P. Le Tallec, *Augmented Lagrangian and operator-splitting methods in nonlinear mechanics*. Society for Industrial Mathematics, 1989.

[17] R. Glowinski, *Numerical methods for nonlinear variational problems*. Springer Verlag, 2008.

[18] L. Jacob, G. Obozinski, and J. Vert, "Group Lasso with overlap and graph Lasso," in *Proceedings of the 26th Annual International Conference on Machine Learning*, pp. 433–440, ACM, 2009.

[19] J. Meng, W. Yin, H. Li, E. Hossain, and Z. Han, "Collaborative Spectrum Sensing from Sparse Observations in Cognitive Radio Networks," 2010.

[20] D. Baron, M. Wakin, M. Duarte, S. Sarvotham, and R. Baraniuk, "Distributed compressed sensing," *preprint*, 2005.

[21] H. Krim and M. Viberg, "Two decades of array signal processing research: the parametric approach," *Signal Processing Magazine, IEEE*, vol. 13, no. 4, pp. 67–94, 2002.

[22] K. Pruessmann, M. Weiger, M. Scheidegger, and P. Boesiger, "SENSE: sensitivity encoding for fast MRI," *Magnetic Resonance in Medicine*, vol. 42, no. 5, pp. 952–962, 1999.