RICE UNIVERSITY

Recovering Data with Group Sparsity by Alternating Direction Methods

by

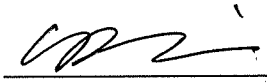
Wei Deng

A Thesis Submitted
in Partial Fulfillment of the
Requirements for the Degree

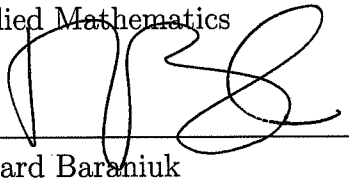
Master of Arts

Approved, Thesis Committee:

Yin Zhang, Chair
Professor of Computational and Applied Mathematics



Wotao Yin, Co-Chair
Assistant Professor of Computational and Applied Mathematics



Richard Baraniuk
Victor E. Cameron Professor of Electrical and Computer Engineering

Houston, Texas

April, 2012

ABSTRACT

Recovering Data with Group Sparsity by Alternating Direction Methods

by

Wei Deng

Group sparsity reveals underlying sparsity patterns and contains rich structural information in data. Hence, exploiting group sparsity will facilitate more efficient techniques for recovering large and complicated data in applications such as compressive sensing, statistics, signal and image processing, machine learning and computer vision. This thesis develops efficient algorithms for solving a class of optimization problems with group sparse solutions, where arbitrary group configurations are allowed and the mixed $\ell_{2,1}$-regularization is used to promote group sparsity. Such optimization problems can be quite challenging to solve due to the mixed-norm structure and possible grouping irregularities. We derive algorithms based on a variable splitting strategy and the alternating direction methodology. Extensive numerical results are presented to demonstrate the efficiency, stability and robustness of these algorithms, in comparison with the previously known state-of-the-art algorithms. We also extend the existing global convergence theory to allow more generality.

Acknowledgements

First and foremost, my utmost gratitude goes to my thesis advisors, Dr. Yin Zhang and Dr. Wotao Yin, who first brought me into the world of research and introduced me to this particular field of optimization theory. I am grateful for their guidance, as it has been a continuous inspiration to me throughout the course of my quest for knowledge. They have offered me a countless number of invaluable directions in the preparation and completion of this study.

I am also highly obliged to Dr. Richard Baraniuk for serving on my thesis committee. I am thankful for his time and scholarly suggestions to bring my work to a fruitful end.

In addition, I would like to thank Dr. Matthias Heinkenschloss, Dr. Jan Hewitt, and Dr. Tim Warburton for helping me improve my writing and presentation skills through the Thesis Writing class. I also wish to thank my fellow colleagues for providing a stimulating and fun environment which allowed me to learn and grow.

Special thanks to King-Pan, for all your encouragement, support and great patience at all times. Thank you for helping me get through the difficult times.

Last but not the least, I wish to thank my parents for their never-ending love and support. They have always encouraged me to do my best in all matters in life. To them I dedicate this thesis.

Contents

Illustrations

Tables

Chapter 1

Introduction

Group sparsity is an emerging terminology that characterizes both the underlying sparsity and structure of the data. Real-world data, such as audio signals, images and videos, are often highly sparse (or compressible) and richly structured. Group sparsity techniques take advantage of the sparsity and structure of the data, thereby facilitating more efficient ways to tackle large and complicated problems in practice.

This thesis focuses on developing efficient algorithms for solving a class of optimization problems with group sparse solutions that arise from a wide range of areas such as compressive sensing, statistics, signal and image processing, machine learning and computer vision. The proposed algorithms are based on a variable splitting strategy and the alternating direction methodology. Extensive numerical results are presented to demonstrate the good efficiency, strong stability and robustness of these algorithms, in comparison with the previously known state-of-the-art algorithms. The global convergence of the proposed algorithms is guaranteed by the existing theory of alternating direction methods under certain parameter restrictions. We also extend the convergence theory to allow more generality.

In this chapter, we will introduce the recent development of sparsity and group sparsity techniques, providing the background and motivation of our work.

1.1 Sparsity

The recent decade has witnessed the fast development of finding sparse solutions to underdetermined linear systems, largely motivated by the emergence of compressive sensing and sparse learning.

Mathematically speaking, an n-dimensional data is called *k-sparse* if it has at most k nonzero components, where k is usually much smaller than n. In practice, most of data are sparse or can be well approximated by sparse data, under some known basis. In the latter case, we also call the data is *compressible*. Sparsity, or compressibility, means that there is usually high redundancy in the data. Exploiting the sparsity of the data leads to more efficient ways for data acquisition, processing, transmission and reconstruction. For example, the idea of sparsity has already been used for image and video compression, such as the JPEG, JPEG2000 and MPEG standards.

Nowadays, sparsity has become a powerful tool to tackle high dimensional data with only a small number of measurements, and has found a wide range of new applications in areas such as compressive sensing, statistics, signal and image processing, machine learning and computer vision. In this section, we will briefly introduce the important application of sparsity in compressive sensing and sparse learning, as well as the commonly used ℓ_1-minimization approach for finding sparse solutions.

1.1.1 Compressive Sensing

In the last few years, compressive sensing has brought a new and most efficient way for signal acquisition (Donoho [8]; Candes, Romberg and Tao [6]). It aims to address the following fundamental question: if an n-dimensional signal has only k ($k \ll n$) nonzero components, why do we spend so much effort acquiring all the n components

and then discard all but k of them? Alternatively, can we directly acquire roughly $O(k)$ or slightly more than $O(k)$ number of "compressed samples" that capture all the information of the signal? Compressive sensing has shown that this is indeed possible. Compared to the conventional signal sampling rate specified by the Shannon-Nyquist theorem, compressive sensing is able to reconstruct the original signal using far fewer measurements, thereby substantially reducing the sampling cost.

Suppose $x \in \mathbb{R}^n$ is an unknown sparse signal. The "compressed samples" are usually acquired by taking random linear measurements of the signal, i.e., $b = Ax \in \mathbb{R}^m$ $(m < n)$, where the sensing matrix $A \in \mathbb{R}^{m \times n}$ is some random matrix, and the number of measurements is fewer than the number of unknowns. The central problem for compressive sensing signal recovery is to find a sparse solution that satisfies this underdetermined linear system. It can be formulated as an optimization problem:

$$\begin{aligned} \min_{x} \quad & \|x\|_0 \\ \text{s.t.} \quad & \|Ax - b\|_2 \leq \delta, \end{aligned} \tag{1.1}$$

where the "ℓ_0-norm" $\|\cdot\|_0$ is a quasi-norm that counts the number of nonzeros, and $\delta \geq 0$ denotes the magnitude of noise. This problem finds the sparsest solution that satisfies the linear measurements, which is most likely to be our true signal.

1.1.2 Sparse Learning

In many statistical regression problems, we are interested in finding the most contributing factors in predicting the responses. Given a data matrix $A \in \mathbb{R}^{m \times n}$ with n factors and an observation vector $b \in \mathbb{R}^m$, we want to find a coefficient vector $x \in \mathbb{R}^n$ that best interprets the linear model:

$$b = Ax + \epsilon, \tag{1.2}$$

where $\epsilon \sim N_n(0, \sigma^2 I)$ is some noise. In practice, it is often that only a small number of factors, among all the n factors, contribute mostly to the responses. It means that the coefficient vector x is likely to be sparse, where the nonzero components correspond to the most important factors. This problem can be formulated as

$$\begin{aligned} \min_{x} \quad & \|Ax - b\|_2^2 \\ \text{s.t.} \quad & \|x\|_0 \leq k, \end{aligned} \tag{1.3}$$

where k is the desired sparsity level.

1.1.3 ℓ_1-regularization

Solving the ℓ_0-problems (1.1) and (1.3) is in general an NP-hard problem. A standard approach is to relax the ℓ_0-regularization to the ℓ_1-regularization, which is also referred to as Lasso [23] in statistics. It is well-known that the ℓ_1-regularization promotes sparsity in the solution, and it is equivalent to the ℓ_0-regularization under certain conditions. For example, by compressive sensing theory, the ℓ_1-regularization can recover a k-sparse signal exactly with high probability, given $m \geq ck \log(n/k)$ i.i.d. random Gaussian measurements, where c is a constant (Donoho [8]; Candes, Romberg and Tao [6]).

The ℓ_1-regularization is much more tractable, because it can be formulated as a linear program and solved by standard linear programming methods, such as the interior point methods. In recent years, a number of more efficient first-order algorithms have been developed, such as a fixed-point continuation (FPC) method [12], a spectral projected gradient method (SPGL1) [25], a fast iterative shrinkage/thresholding algorithm (FISTA) [4], an alternating direction method (YALL1) [30], just to name a few.

In addition, it is worthwhile to mention that there is another commonly used

approach for finding sparse solutions — the greedy algorithms, such as the orthogonal matching pursuit (OMP) [24].

1.2 Group Sparsity

Now we will go beyond sparsity and incorporate the rich structural information in the data, leading to the key concept of this thesis — group sparsity.

1.2.1 Motivation and Concepts

As we know, using sparsity has successfully reduced the number of measurements for effectively dealing with high dimensional data. In order to further reduce the number of measurements, recent studies propose to go beyond sparsity and take into account additional information about the underlying structure of the data. In practice, we always have prior knowledge about the structure of the data, other than just sparsity. Particularly, many practical data naturally have group structures, and the components within a group are likely to be all zeros or all nonzeros, which is referred to as *group sparsity*. In other words, the data is not only sparse, but the zeros/nozeros are clustered into groups.

Group sparse data commonly arises in many applications, such as the group Lasso problem [31], distributed compressive sensing [3], multiple kernel learning [1], microarray data analysis [20], etc.

Intuitively, encoding more structural information in addition to sparsity as priors can reduce the degrees of freedom in the model, thereby leading to less measurement requirement and better performance.

1.2.2 Convex Relaxation

Suppose $x \in \mathbb{R}^n$ is an unknown group sparse data. 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. Finding group sparse solutions can be formulated as the following optimization problem:

$$\begin{aligned} \min_{x \in \mathbb{R}^n} \quad & \|x\|_0^{\mathcal{G}} \\ \text{s.t.} \quad & \|Ax - b\|_2 \leq \delta, \end{aligned} \tag{1.4}$$

where $\|x\|_0^{\mathcal{G}}$ counts the number of nonzero groups in x, $A \in \mathbb{R}^{m \times n}$, $b \in \mathbb{R}^m$, and $\delta \geq 0$ is a parameter for the magnitude of noise.

Likewise, this ℓ_0-type problem is in general NP-hard. A standard approach is to find its convex relaxation problem like the ℓ_1-minimization problem. A natural extension of the general ℓ_p-norm to the group setting is the mixed $\ell_{p,q}$-norm, defined by

$$\|x\|_{p,q} := \left(\sum_{i=1}^{s} \|x_{g_i}\|_p^q \right)^{1/q}, \text{ for } p, q > 0, \tag{1.5}$$

where $\|\cdot\|_p$ is the standard ℓ_p-norm. If we let $y := (\|x_{g_1}\|_p, \ldots, \|x_{g_s}\|_p)^T \in \mathbb{R}^s$, then $\|x\|_{p,q} = \|y\|_q$. In other words, the $\ell_{p,q}$-norm takes the ℓ_p-norm over each group and sum them up in the ℓ_q-norm. When $q = 1$, it can be considered as an extension of the ℓ_1-norm for group sparsity. Since minimizing $\|y\|_1$ gives rise to a sparse solution y, it follows that most components of y tend to be zero, meaning that most of the groups are zeros. For convexity, we need $p \geq 1$. However, when $p = 1$ and $q = 1$, it reduces to the standard ℓ_1-norm, thereby losing the group structure. Therefore, $p > 1,\ q = 1$ is of our particular interest. A common choice of p is $p = 2$, since it is relatively easy to compute. This thesis will restrict the attention to the case $p = 2,\ q = 1$, but the

framework can be easily adapted for general $p > 1$.

In addition to the optimization approach, it is worthwhile to mention that the greedy algorithms are another class of commonly used methods for recovering group sparse data, such as the model-based compressive sensing [2], structured orthogonal matching pursuit (StructOMP) [16], etc.

1.2.3 Existing Methods

The mixed $\ell_{2,1}$-minimization problem is a commonly used convex relaxation of the group sparse problem and has proven to be effective in promoting group sparsity. For example, the basis pursuit model in compressive sensing for recovering group sparse signals can be formulated as:

$$\begin{aligned} \min_{x\in\mathbb{R}^n} \quad & \|x\|_{2,1} := \sum_{i=1}^{s} \|x_{g_i}\|_2 \\ \text{s.t.} \quad & \|Ax - b\|_2 \le \delta. \end{aligned} \tag{1.6}$$

The so-called group lasso model [31] in statistics is given by

$$\begin{aligned} \min_{x\in\mathbb{R}^n} \quad & \|Ax - b\|_2^2 \\ \text{s.t.} \quad & \|x\|_{2,1} \le k, \end{aligned} \tag{1.7}$$

or alternatively

$$\min_{x\in\mathbb{R}^n} \quad \|x\|_{2,1} + \frac{1}{2\mu}\|Ax - b\|_2^2, \tag{1.8}$$

where $\mu > 0$ is a parameter.

Due to the mixed-norm structure and possible grouping irregularity, the resulting optimization problems are considered more difficult to solve than the standard ℓ_1-minimization problems. Although the $\ell_{2,1}$-problems can be formulated as a second-order cone programming (SOCP) or a semidefinite programming (SDP), solving either

SOCP or SDP by standard methods (e.g., the interior point methods) is computationally expensive.

Several efficient first-order algorithms have been proposed for solving the $\ell_{2,1}$-problems. In [26], Van den Berg, Schmidt, Friedlander and Murphy proposed a spectral projected gradient method (SPGL1), which uses a linear-time algorithm for the Euclidean projection onto the $\ell_{2,1}$-norm constraints. In [28], Wright, Nowak and Figueiredo proposed an algorithm framework (SpaRSA) for minimizing the sum of a nonsmooth regularization term and a smooth convex function, based on the iterative shrinkage/thresholding methods. Recently, Liu and Ye [19] derived an algorithm (SLEP) using the accelerated gradient method for solving the $\ell_{p,1}$-regularized problem for $p > 1$, where they compute the $\ell_{p,1}$-regularized Euclidean projection by solving two zero finding problems.

This thesis proposes a new approach for solving the $\ell_{2,1}$-problem based on a variable splitting technique and the framework of alternating direction method (ADM). A brief review of the alternating direction method will be given in Chapter 2. Numerical results demonstrate that the proposed ADM algorithms compare favorably to the previously existing state-of-the-art algorithms.

1.3 Notations and Organization

Throughout the thesis, we let matrices be denoted by uppercase letters and vectors by lowercase letters. For simplicity, we use $\|\cdot\|$ to represent the ℓ_2-norm $\|\cdot\|_2$ of a vector or a matrix.

The rest of the thesis is organized as follows. Chapter 2 derives a variety of algorithms for solving several basic group sparse models. Chapter 3 extends the algorithms to enforce nonnegativity in the data and handle arbitrary group configurations,

such as overlapping groups and incomplete cover. In addition, the joint sparsity problems are studied as an important special case of group sparsity problems. Chapter 4 discusses the existing global convergence theory and makes extensions to allow more generality. In Chapter 5, various numerical results are presented to evaluate the efficiency of the proposed algorithms. Chapter 6 gives concluding remarks, and Chapter 7 proposes for future research directions.

Chapter 2

Algorithms

This chapter develops efficient algorithms for solving a variety of $\ell_{2,1}$-regularized group sparse optimization problems, arising from a wide range of areas such as compressive sensing, statistics, signal and image processing, machine learning and computer vision. The derived algorithms are based on a variable splitting strategy and the framework of alternating direction method (ADM).

2.1 Group Sparse Models

Suppose $x \in \mathbb{R}^n$ is an unknown group sparse signal. For simplicity, the groups $\{x_{g_i} : i = 1, ..., s\}$ are assumed to be a partition of x. In Chapter 3, more group configurations will be discussed, such as overlapping groups and incomplete cover. It will be shown that the algorithms developed in this chapter can be easily extended to handle general group configurations.

To be more general, instead of using the $\ell_{2,1}$-norm (1.5), a weighted $\ell_{2,1}$-norm is considered. It is defined by

$$\|x\|_{w,2,1} := \sum_{i=1}^{s} w_i \|x_{g_i}\|_2, \tag{2.1}$$

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. Moreover, adding weights inside groups will also be discussed in Chapter 3.

The following three basic group sparse models are commonly used.

(1) Basis pursuit (BP) model:

$$\begin{aligned} \min_{x} \quad & \|x\|_{w,2,1} \\ \text{s.t.} \quad & Ax = b, \end{aligned} \tag{2.2}$$

(2) Constrained basis pursuit denoising (BP_δ) model:

$$\begin{aligned} \min_{x} \quad & \|x\|_{w,2,1} \\ \text{s.t.} \quad & \|Ax - b\|_2 \leq \sigma, \end{aligned} \tag{2.3}$$

(3) Unconstrained basis pursuit denoising (BP_μ) model:

$$\min_{x} \|x\|_{w,2,1} + \frac{1}{2\mu}\|Ax - b\|_2^2, \tag{2.4}$$

Here $A \in \mathbb{R}^{m\times n}$ $(m < n)$, $b \in \mathbb{R}^m$, $\sigma > 0$ and $\mu > 0$ are parameters.

When the measurement vector b contains noise, the basis pursuit denoising models BP_δ and BP_μ are often used. The parameters σ and μ are chosen to control the noise level. As δ and μ approach to zero, the solutions of BP_δ and BP_μ converge to the solution of (2.2). Moreover, it is worthwhile to point out that the BP model (2.2) is also good for noisy data if the algorithms are stopped properly prior to convergence based on the noise level.

2.2 Review of Alternating Direction Methods

The alternating direction method was first introduced in [9, 11], and has proven to be effective for solving a wide range of optimization problems. We consider the following convex optimization problem with separable objective functions and linear

constraints:

$$\begin{aligned} \min_{x,z} \ & f(x) + g(z) \\ \text{s.t. } & \mathcal{A}x + \mathcal{B}z = c, \\ & x \in \mathcal{X},\ z \in \mathcal{Z}, \end{aligned} \tag{2.5}$$

where $\mathcal{A} \in \mathbb{R}^{p\times n_1}$, $\mathcal{B} \in \mathbb{R}^{p\times n_2}$, $\mathcal{X} \subseteq \mathbb{R}^{n_1}$ and $\mathcal{Z} \subseteq \mathbb{R}^{n_2}$ are closed convex sets, $f : \mathbb{R}^{n_1} \to \mathbb{R}$, and $g : \mathbb{R}^{n_2} \to \mathbb{R}$ are convex functions. Such problems commonly arise in many areas such as compressive sensing, statistics, signal and image processing, machine learning and computer vision. For example, f is $\|\cdot\|_2^2$ for data-fidelity term, and g is a regularization term to enforce certain property in the solution, e.g., ℓ_1-norm $\|\cdot\|_1$ for sparsity, $\ell_{2,1}$-norm $\|\cdot\|_{2,1}$ for group sparsity, and nuclear-norm $\|\cdot\|_*$ for low-rankness of a matrix. In fact, many original problems often do not have the form of (2.5). However, by introducing splitting variables, the problems can be transformed into the above form.

The alternating direction method is based on the augmented Lagrangian framework, in which the augmented Lagrangian function of (2.5) is defined by

$$\mathcal{L}_{\mathcal{A}}(x, z, \lambda) := f(x) + g(z) - \lambda^T(\mathcal{A}x + \mathcal{B}z - c) + \frac{\beta}{2}\|\mathcal{A}x + \mathcal{B}z - c\|_2^2, \tag{2.6}$$

where $\lambda \in \mathbb{R}^p$ is the Lagrangian multiplier, $\beta > 0$ is a penalty parameter. In the framework of standard augmented Lagrangian methods, the augmented Lagrangian function is minimized over (x, z) jointly at each iteration. However, such joint minimization is often difficult and costly. Alternatively, the alternating direction method takes advantage of the separability structure of the problems and replace the joint minimization by an alternating minimization. In particular, the augmented Lagrangian function is minimized over x-direction and z-direction alternately at each iteration, after which the multiplier λ is updated. The algorithm framework is outlined as fol-

lows, where $\gamma > 0$ is a step length for updating the multiplier.

Algorithm 1: Alternating Direction Method (ADM)

Initialize $x^0,\ \lambda^0,\ \beta > 0, \gamma > 0$;

for $k = 0,\ 1, \ldots$ **do**

$z^{k+1} = \arg\min_{z \in \mathcal{Z}} \mathcal{L}_{\mathcal{A}}(x^k, z, \lambda^k)$;

$x^{k+1} = \arg\min_{x \in \mathcal{X}} \mathcal{L}_{\mathcal{A}}(x, z^{k+1}, \lambda^k)$;

$\lambda^{k+1} = \lambda^k - \gamma\beta(\mathcal{A}x^{k+1} + \mathcal{B}z^{k+1} - c)$;

The global convergence of the classic alternating direction method has been established in [9, 10, 14]. In practice, it is possible that the resulting subproblems may still be difficult and expensive to solve exactly. Many inexact versions of the alternating direction method have been proposed, in which the subproblems are solved approximately.

2.3 Applying ADM to the Primal Problems

In this section, we apply the alternating direction methodology to the various group sparse problems.

2.3.1 Group-wise Shrinkage

We introduce two lemmas that are useful in the derivation of the algorithms. The first lemma is well-known, and thus we omit the proof in which the solution is given by the so-called shrinkage (or soft thresholding) formula. We extend this lemma for the group $\ell_{2,1}$-norm and obtain the second lemma. Here, we follow the convention that $0 \cdot \frac{0}{0} = 0$.

Lemma 2.1. (Shrinkage) *For any $\alpha, \beta > 0$ and $t \in \mathbb{R}^n$, the minimizer of*

$$\min_{z \in \mathbb{R}^n} \ \alpha \|z\|_2 + \frac{\beta}{2}\|z - t\|_2^2$$

is given by

$$z(t) = \mathit{Shrink}\left(t, \frac{\alpha}{\beta}\right) := \max\left\{\|t\|_2 - \frac{\alpha}{\beta}, 0\right\} \frac{t}{\|t\|_2}. \tag{2.7}$$

Lemma 2.2. (Group-wise Shrinkage) *Suppose the groups $\{g_i : i = 1, ..., s\}$ form a partition of $\{1, 2, \ldots, n\}$. For any $\beta > 0$ and $t \in \mathbb{R}^n$, the minimizer of*

$$\min_{z \in \mathbb{R}^n} \ \|z\|_{w,2,1} + \frac{\beta}{2}\|z - t\|_2^2 \tag{2.8}$$

is given by

$$z_{g_i}(t) = \max\left\{\|t_{g_i}\|_2 - \frac{w_i}{\beta}, 0\right\} \frac{t_{g_i}}{\|t_{g_i}\|_2}, \ \textit{for } i = 1, \ldots, s, \tag{2.9}$$

denoted by $z(t) = \mathit{GShrink}\left(t, \frac{w}{\beta}\right)$ for short.

Proof. Since the groups $\{g_i : i = 1, ..., s\}$ form a partition, we can rewrite (2.8) as

$$\min_{z \in \mathbb{R}^n} \ \sum_{i=1}^{s} \left[w_i \|z_{g_i}\|_2 + \frac{\beta}{2}\|z_{g_i} - t_{g_i}\|_2^2 \right], \tag{2.10}$$

which can be reduced to minimizing the s subproblems individually:

$$\min_{z_{g_i} \in \mathbb{R}^{n_i}} \ w_i \|z_{g_i}\|_2 + \frac{\beta}{2}\|z_{g_i} - t_{g_i}\|_2^2, \ \text{for } i = 1, \ldots, s. \tag{2.11}$$

By the Shrinkage formula (2.7), the closed-form solution of each subproblem is given by (2.9). □

2.3.2 BP model

In order to apply the alternating direction method, we need to first introduce splitting variables and transform the problems in the form of (2.5). For the BP model (2.2),

we introduce $z = x$, and obtain an equivalent problem:

$$\begin{aligned} \min_{x,z} \quad & \|z\|_{w,2,1} = \sum_{i=1}^{s} w_i \|z_{g_i}\|_2 \\ \text{s.t.} \quad & x - z = 0, \\ & Ax = b. \end{aligned} \tag{2.12}$$

The augmented Lagrangian function is given by

$$\mathcal{L}_A := \|z\|_{w,2,1} - \lambda_1^T(x - z) + \frac{\beta_1}{2}\|x - z\|_2^2 - \lambda_2^T(Ax - b) + \frac{\beta_2}{2}\|Ax - b\|_2^2, \tag{2.13}$$

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 alternating minimization approach. That is, we minimize the augmented Lagrangian function (2.13) with respect to x and z alternately.

We first look at the z-subproblem, given by

$$\min_z \ \|z\|_{w,2,1} + \frac{\beta_1}{2}\|z - x^k + \frac{1}{\beta}\lambda_1^k\|_2^2 \tag{2.14}$$

after dropping some constant terms. By Lemma 2.2, we immediately get its closed-form solution by the group-wise shrinkage formula: $z^{k+1} = \text{GShrink}\left(x^k - \frac{1}{\beta_1}\lambda_1^k, \frac{1}{\beta_1}w\right)$. That is,

$$z_{g_i}^{k+1} = \max\left\{\|r_i^k\|_2 - \frac{w_i}{\beta_1}, 0\right\}\frac{r_i^k}{\|r_i^k\|_2}, \text{ for } i = 1, ..., s, \tag{2.15}$$

where $r_i^k := x_{g_i}^k - \frac{1}{\beta_1}(\lambda_1^k)_{g_i}$.

The x-subproblem is clearly a convex quadratic problem. After simple manipulations, it is given by

$$\min_x \ \frac{1}{2}x^T(\beta_1 I + \beta_2 A^T A)x - (\beta_1 z^{k+1} + \lambda_1^k + \beta_2 A^T b + A^T\lambda_2^k)^T x, \tag{2.16}$$

which is equivalent to solving the following linear system:

$$(\beta_1 I + \beta_2 A^T A)x^{k+1} = \beta_1 z^{k+1} + \lambda_1^k + \beta_2 A^T b + A^T\lambda_2^k. \tag{2.17}$$

This $n \times n$ linear system can be reduced to a smaller $m \times m$ system by Sherman-Morrison-Woodbury formula:

$$(\beta_1 I + \beta_2 A^T A)^{-1} = \frac{1}{\beta_1} I - \frac{\beta_2}{\beta_1} A^T (\beta_1 I + \beta_2 A A^T)^{-1} A. \tag{2.18}$$

When m is not very large or $AA^T = I$, we may afford to solve the linear system exactly. Otherwise, we can solve it approximately to reduce the computation cost. For example, we can choose to take just one step of gradient descent, i.e.,

$$x^{k+1} = x^k - \alpha^k g^k, \tag{2.19}$$

where $g^k = (\beta_1 I + \beta_2 A^T A)x^k - \beta_1 z^{k+1} - \lambda_1^k - \beta_2 A^T b - A^T \lambda_2^k$ is the gradient, and $\alpha^k > 0$ is a step length. For exact line search, α^k is given by

$$\alpha^k = \frac{(g^k)^T g^k}{(g^k)^T (\beta_1 I + \beta_2 A^T A) g^k}. \tag{2.20}$$

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

$$\begin{cases} \lambda_1^{k+1} = \lambda_1^k - \gamma\beta_1(x^{k+1} - z^{k+1}), \\ \lambda_2^{k+1} = \lambda_2^k - \gamma\beta_2(Ax^{k+1} - b), \end{cases} \tag{2.21}$$

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

The ADM scheme for (2.12) is outlined below.

Algorithm 2: ADM for Group Sparse BP Model

Initialize x^0, λ_1^0, λ_2^0, $\beta_1, \beta_2 > 0$ and $\gamma_1, \gamma_2 > 0$;

for $k = 0, 1, \ldots$ **do**

$z^{k+1} = \text{GShrink}(x^k - \frac{1}{\beta_1}\lambda_1^k, \frac{1}{\beta_1} w)$;

$x^{k+1} = (\beta_1 I + \beta_2 A^T A)^{-1}(\beta_1 z^{k+1} + \lambda_1^k + \beta_2 A^T b + A^T \lambda_2^k)$;

$\lambda_1^{k+1} = \lambda_1^k - \gamma\beta_1(x^{k+1} - z^{k+1})$;

$\lambda_2^{k+1} = \lambda_2^k - \gamma\beta_2(Ax^{k+1} - b)$;

2.3.3 BP_δ model

For the BP_δ model (2.3), we introduce a different splitting variable. Letting $z = b - Ax$, the problem becomes:

$$\begin{aligned} \min_{x,z} \quad & \|x\|_{w,2,1} \\ \text{s.t.} \quad & Ax + z = b, \\ & \|z\|_2 \leq \delta. \end{aligned} \tag{2.22}$$

Then we solve the augmented Lagrangian problem

$$\begin{aligned} \min_{x,z} \quad & \mathcal{L}_A := \|x\|_{w,2,1} - \lambda^T(Ax + z - b) + \frac{\beta}{2}\|Ax + z - b\|_2^2 \\ \text{s.t.} \quad & \|z\|_2 \leq \delta \end{aligned} \tag{2.23}$$

using the alternating minimization approach.

The z-subproblem given by

$$\begin{aligned} \min_{z} \quad & \|z - (\frac{1}{\beta}\lambda^k - Ax^k + b)\|_2^2 \\ \text{s.t.} \quad & \|z\|_2 \leq \delta \end{aligned} \tag{2.24}$$

is easy to solve. In fact, it has a closed-form solution by projection:

$$z^{k+1} = \mathcal{P}_{B_\delta}\left(\frac{1}{\beta}\lambda^k - Ax^k + b\right), \tag{2.25}$$

where $\mathcal{P}_{B_\delta}$ is the projection onto the 2-norm ball $B_\delta := \{z \in \mathcal{R}^m : \ \|z\|_2 \leq \delta\}$, i.e.,

$$\mathcal{P}_{B_\delta}(r) = \begin{cases} r & \text{if } \|r\|_2 \leq \delta; \\ \delta r/\|r\|_2 & \text{otherwise.} \end{cases} \tag{2.26}$$

However, the x-subproblem:

$$\min_{x} \ \|x\|_{w,2,1} + \frac{\beta}{2}\|Ax + z^{k+1} - b - \frac{1}{\beta}\lambda^k\|_2^2 \tag{2.27}$$

is no easier than solving a BP_μ problem (2.4). Instead, it is solved approximately based on the linear proximal method, which linearizes the quadratic penalty term and adds a proximal term. Specifically, the following approximate problem is solved:

$$\min_x \ \|x\|_{w,2,1} + \beta\left((g^k)^T(x-x^k) + \frac{1}{2\tau}\|x-x^k\|_2^2\right), \tag{2.28}$$

where $g^k := A^T(Ax^k + z^{k+1} - b - \frac{1}{\beta}\lambda^k)$ is the gradient of the quadratic penalty term, and $\tau > 0$ is a proximal parameter. It can be viewed as using an identity matrix $\frac{1}{\tau}$ to approximate A^TA, the Hessian of the quadratic penalty term. Then, this approximate problem (2.28) becomes easy to solve, whose solution is given by the group-wise shrinkage formula:

$$x^{k+1} = \mathrm{GShrink}\left(x^k - \tau g^k, \frac{\tau}{\beta}w\right). \tag{2.29}$$

Finally, the multiplier λ is updated by

$$\lambda^{k+1} = \lambda^k - \gamma\beta(Ax^{k+1} + z^{k+1} - b), \tag{2.30}$$

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

Hence, we obtain an ADM scheme for (2.3).

Algorithm 3: ADM for Group Sparse BP_δ Model

1 Initialize x^0, λ_1^0, λ_2^0, $\beta > 0$ and $\gamma > 0$;
2 **for** $k = 0, 1, \ldots$ **do**
3 $\quad z^{k+1} = \mathcal{P}_{B_\delta}\left(\frac{1}{\beta}\lambda^k - Ax^k + b\right)$;
4 $\quad x^{k+1} = \mathrm{GShrink}\left(x^k - \tau g^k, \frac{\tau}{\beta}w\right)$;
5 $\quad \lambda^{k+1} = \lambda^k - \gamma\beta(Ax^{k+1} + z^{k+1} - b)$;

2.3.4 BP$_\mu$ model

Similarly, an ADM scheme can be derived for the BP$_\mu$ model (2.4). We first introduce splitting variable $z = Ax - b$ and formulate the following equivalent problem:

$$\begin{aligned} \min_{x,z} \quad & \|x\|_{w,2,1} + \frac{1}{2\mu}\|z\|_2^2 \\ \text{s.t.} \quad & Ax + z = b. \end{aligned} \tag{2.31}$$

The augmented Lagrangian function is given by

$$\mathcal{L}_A(x, z, \lambda) := \|x\|_{w,2,1} + \frac{1}{2\mu}\|z\|^2 - \lambda^T(Ax + z - b) + \frac{\beta}{2}\|Ax + z - b\|_2^2. \tag{2.32}$$

It is easy to see that the z-subproblem is a convex quadratic problem, which is equivalent to solve

$$\nabla_z \mathcal{L}_A(x^k, z, \lambda^k) = \frac{1}{\mu}z - \lambda^k + \beta(Ax^k + z - b) = 0, \tag{2.33}$$

i.e.,

$$z^{k+1} = \frac{\mu\beta}{1 + \mu\beta}(\lambda^k/\beta - Ax^k + b). \tag{2.34}$$

The x-subproblem is the same with (2.27). As our previous discussion, it is solved approximately using the linear proximal method, and an approximate solution can be obtained by the group-wise shrinkage formula. Finally, the multiplier λ is updated using the same formula as (2.30). We summarize the ADM scheme for (2.4) as follows.

Algorithm 4: ADM for Group Sparse BP$_\mu$ Model

1 Initialize x^0, λ_1^0, λ_2^0, $\beta > 0$ and $\gamma > 0$;

2 **for** $k = 0, 1, \ldots$ **do**

3 $\quad z^{k+1} = \frac{\mu\beta}{1+\mu\beta}(\lambda^k/\beta - Ax^k + b)$;

4 $\quad x^{k+1} = \text{GShrink}\left(x^k - \tau g^k, \frac{\tau}{\beta}w\right)$;

5 $\quad \lambda^{k+1} = \lambda^k - \gamma\beta(Ax^{k+1} + z^{k+1} - b)$;

2.4 Applying ADM to the Dual Problems

Alternatively, we can apply the alternating direction method to the dual of the problems (2.2), (2.3) and (2.4).

2.4.1 Dual of BP model

The dual of the BP model (2.2) can be derived by

$$\begin{aligned}
&\max_y \left\{ \min_x \sum_{i=1}^{s} w_i \|x_{g_i}\|_2 - y^T(Ax-b) \right\} \\
=&\max_y \left\{ b^T y + \min_x \sum_{i=1}^{s} \left(w_i \|x_{g_i}\|_2 - y^T A_{g_i} x_{g_i} \right) \right\} \\
=&\max_y \left\{ b^T y : \|A_{g_i}^T y\|_2 \le w_i, \text{ for } i = 1, ..., s \right\}, \qquad (2.35)
\end{aligned}$$

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

By introducing a splitting variable $z = A^T y$, it is reformulated as

$$\begin{aligned}
\min_{y,z} \quad & - b^T y \\
\text{s.t.} \quad & z = A^T y, \qquad (2.36) \\
& \|z_{g_i}\|_2 \le w_i, \text{ for } i = 1, ..., s.
\end{aligned}$$

The augmented Lagrangian problem is given by

$$\begin{aligned}
\min_{y,z} \quad & - b^T y - x^T(z - A^T y) + \frac{\beta}{2}\|z - A^T y\|_2^2 \\
\text{s.t.} \quad & \|z_{g_i}\|_2 \le w_i, \text{ for } i = 1, ..., s,
\end{aligned} \qquad (2.37)$$

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

In the alternating minimization iterations, the z-subproblem is given by

$$\begin{aligned} \min_{z} \quad & -(x^k)^T z + \frac{\beta}{2}\|z - A^T y^k\|_2^2 \\ \text{s.t.} \quad & \|z_{g_i}\|_2 \le w_i, \text{ for } i = 1, ..., s, \end{aligned} \tag{2.38}$$

which can be written as

$$\begin{aligned} \min_{z} \quad & \sum_{i=1}^{s} \frac{\beta}{2}\|z_{g_i} - A_{g_i}^T y^k - \frac{1}{\beta} x_{g_i}^k\|_2^2 \\ \text{s.t.} \quad & \|z_{g_i}\|_2 \le w_i, \text{ for } i = 1, ..., s. \end{aligned} \tag{2.39}$$

It's easy to show that the solution to (2.39) is given by

$$z_{g_i}^{k+1} = \mathcal{P}_{\mathbf{B}_2^i}(A_{g_i}^T y^k + \frac{1}{\beta} x_{g_i}^k), \text{ for } i = 1, ..., s. \tag{2.40}$$

Here $\mathcal{P}_{\mathbf{B}_2^i}$ represents a projection onto the ball $\mathbf{B}_2^i := \{z \in \mathbb{R}^{n_i} : \|z\|_2 \le w_i\}$. In short,

$$z^{k+1} = \mathcal{P}_{\mathbf{B}_2^{\mathcal{G}}}(A^T y^k + \frac{1}{\beta} x^k), \tag{2.41}$$

where $\mathbf{B}_2^{\mathcal{G}} := \{z \in \mathbb{R}^n : \|z_{g_i}\|_2 \le w_i, \text{ for } i = 1, ..., s\}$.

The y-subproblem is a convex quadratic problem:

$$\min_{y} (Ax^k - b)^T y + \frac{\beta}{2}\|A^T y - z^{k+1}\|_2^2, \tag{2.42}$$

which can be reduced to solving the following linear system:

$$\beta A A^T y^{k+1} = b - Ax^k + \beta A z^{k+1}. \tag{2.43}$$

A special case is that A has orthonormal rows, i.e., $AA^T = I$. Then we get the exact solution y^{k+1} immediately without solving a linear system. In general, we need to solve a $m \times m$ linear system, which is costly when the dimension m becomes large. As we discussed in Section 2.3.2, we can solve it approximately by taking one-step gradient descent, i.e.,

$$y^{k+1} = y^k - \alpha^k g^k. \tag{2.44}$$

Here, $g^k = \beta AA^T y^k + Ax^k - \beta Az^{k+1} - b$ is the gradient of the quadratic function, and $\alpha^k > 0$ is a step length. For exact line search, α^k is given by

$$\alpha^k = \frac{\|g^k\|^2}{\beta\|Ag^k\|^2}. \tag{2.45}$$

At the end of each iteration, the multiplier (i.e. the primal variable) x is updated by

$$x^{k+1} = x^k - \gamma\beta(z^{k+1} - A^T y^{k+1}), \tag{2.46}$$

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

Therefore, the ADM iteration scheme for (2.36) is as follows.

Algorithm 5: ADM for Dual Group Sparse BP Model

1 Initialize $x^0,\ y^0$, $\beta > 0$ and $\gamma > 0$;

2 **for** $k = 0, 1, \ldots$ **do**

3 $\quad z^{k+1} = \mathcal{P}_{\mathbf{B}_2^{\mathcal{G}}}(A^T y^k + \frac{1}{\beta}x^k)$;

4 $\quad y^{k+1} = (\beta AA^T)^{-1}(b - Ax^k + \beta Az^{k+1})$;

5 $\quad x^{k+1} = x^k - \gamma\beta(z^{k+1} - A^T y^{k+1})$;

2.4.2 Dual of BP_δ model

The dual of the BP_δ model (2.24) can be derived by

$$\begin{aligned}
&\max_y \left\{ \min_{x,z} \sum_{i=1}^{s} w_i \|x_{g_i}\|_2 - y^T(Ax + z - b) :\ \|z\| \le \delta \right\} \\
=&\max_y \left\{ b^T y + \min_x \sum_{i=1}^{s} \left(w_i \|x_{g_i}\|_2 - y^T A_{g_i} x_{g_i} \right) + \min_{\|z\| \le \delta} (-y^T z) \right\} \\
=&\max_y \left\{ b^T y - \delta\|y\|_2 : \|A_{g_i}^T y\|_2 \le w_i, \text{ for } i = 1, ..., s \right\}.
\end{aligned} \tag{2.47}$$

By introducing a splitting variable $z = A^T y$, we have

$$\begin{aligned} \min_{y,z} \quad & -b^T y + \delta \|y\|_2 \\ \text{s.t.} \quad & z = A^T y, \\ & \|z_{g_i}\|_2 \leq w_i, \ \text{ for } i = 1, ..., s. \end{aligned} \tag{2.48}$$

The augmented Lagrangian problem is given by

$$\begin{aligned} \min_{y,z} \quad & -b^T y + \delta \|y\|_2 - x^T (z - A^T y) + \frac{\beta}{2} \|z - A^T y\|_2^2 \\ \text{s.t.} \quad & \|z_{g_i}\|_2 \leq w_i, \text{ for } i = 1, ..., s, \end{aligned} \tag{2.49}$$

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

We have the same z-subproblem as (2.39), and it is easy to compute its solution by (2.41). However, the y-subproblem:

$$\min_y \ \delta \|y\|_2 + (Ax^k - b)^T y + \frac{\beta}{2} \|A^T y - z^{k+1}\|^2, \tag{2.50}$$

is a little different from (2.42) with an additional term $\delta \|y\|_2$, making it more difficult to solve. We will discuss two cases when $AA^T = I$ and $AA^T \neq I$.

(i) For a special case that $AA^T = I$, it is easy to solve the y-subproblem exactly. In this case, (2.50) can be rewritten as

$$\min_y \ \delta \|y\|_2 + \frac{\beta}{2} \|y - Az^{k+1} + (Ax^k - b)/\beta\|_2^2. \tag{2.51}$$

By Lemma 2.1, we have a closed-form solution by the shrinkage formula:

$$y^{k+1} = \text{Shrink}\left(Az^{k+1} - (Ax^k - b)/\beta, \ \frac{\delta}{\beta} \right). \tag{2.52}$$

(ii) When $AA^T \neq I$, we use the linear proximal approach to solve the y-subproblem approximately. That is, we linearize the quadratic term and add a proximal term, giving the following problem:

$$\min_y \delta\|y\|_2 + \beta\left((g^k)^T(y-y^k) + \frac{1}{2\tau}\|y-y^k\|_2^2\right), \tag{2.53}$$

where $g^k := A(A^Ty^k - z^{k+1}) + (Ax^k - b)/\beta$ is the gradient of the quadratic terms, and $\tau > 0$ is a proximal parameter. By Lemma 2.1, its solution is given by

$$y^{k+1} = \text{Shrink}\left(y^k - \tau g^k,\ \frac{\tau\delta}{\beta}\right), \tag{2.54}$$

which is an approximate solution of the original y-subproblem.

We summarize the ADM iteration scheme for (2.48) as follows.

Algorithm 6: ADM for Dual Group Sparse BP_δ Model

1 Initialize $x^0,\ y^0$, $\beta > 0$ and $\gamma > 0$;

2 **for** $k = 0, 1, \ldots$ **do**

3 $\quad z^{k+1} = \mathcal{P}_{\mathbf{B}_2^g}(A^Ty^k + \frac{1}{\beta}x^k)$;

4 $\quad y^{k+1} = \text{Shrink}\left(y^k - \tau g^k,\ \frac{\tau\delta}{\beta}\right)$;

5 $\quad x^{k+1} = x^k - \gamma\beta(z^{k+1} - A^Ty^{k+1})$;

2.4.3 Dual of BP_μ model

The dual of the BP_δ model (2.24) can be derived by

$$\begin{aligned}
&\max_y \left\{\min_{x,z} \sum_{i=1}^s w_i\|x_{g_i}\|_2 + \frac{1}{2\mu}\|z\|^2 - y^T(Ax+z-b)\right\} \\
=&\max_y \left\{b^Ty - \frac{\mu}{2}\|y\|^2 + \min_x \sum_{i=1}^s \left(w_i\|x_{g_i}\|_2 - y^TA_{g_i}x_{g_i}\right) + \min_z \frac{1}{2\mu}\|z-\mu y\|^2\right\} \\
=&\max_y \left\{b^Ty - \frac{\mu}{2}\|y\|^2 : \|A_{g_i}^Ty\|_2 \leq w_i, \text{ for } i = 1, ..., s\right\}.
\end{aligned} \tag{2.55}$$

By introducing a splitting variable $z = A^T y$, we have

$$\begin{aligned} \min_{y,z} \quad & -b^T y + \frac{\mu}{2}\|y\|^2 \\ \text{s.t.} \quad & z = A^T y, \\ & \|z_{g_i}\|_2 \le w_i, \ \text{ for } i = 1, ..., s. \end{aligned} \tag{2.56}$$

The alternating minimization scheme is very similar to that of the dual BP model (2.39). The only difference is the y-subproblem due to the additional term $\frac{\mu}{2}\|y\|^2$. The y-subproblem is still a convex quadratic problem, which can be reduced to solving the following linear system:

$$(\mu I + \beta A A^T) y^{k+1} = b - A x^k + \beta A z^{k+1}. \tag{2.57}$$

Similarly, when $AA^T = I$, it is easy to compute the exact solution. In general, we need to solve an $m \times m$ linear system. When m is large, we can solve it approximately by taking one-step gradient descent.

The ADM iteration scheme for (2.56) is summarized as follows.

Algorithm 7: ADM for Dual Group Sparse BP_μ Model

Initialize $x^0,\ y^0$, $\beta > 0$ and $\gamma > 0$;

for $k = 0, 1, \ldots$ **do**

$z^{k+1} = \mathcal{P}_{\mathbf{B}_2^{\mathcal{G}}}(A^T y^k + \frac{1}{\beta} x^k)$;

$y^{k+1} = (\mu I + \beta A A^T)^{-1}(b - A x^k + \beta A z^{k+1})$;

$x^{k+1} = x^k - \gamma\beta(z^{k+1} - A^T y^{k+1})$;

2.5 Remarks

- In many of the ADM schemes, one subproblem often involves solving linear systems. Although some updating formula is written in the form of inverting

a matrix in the algorithm schemes, we emphasize that we never actually invert a matrix at each iteration. Since the coefficient matrices of the linear systems remain the same over iterations, we only need to compute the matrix inverse or do matrix factorization once. For large problems when inverting a matrix or do matrix factorization is no longer affordable, we propose to just take one gradient descent step to solve the linear system approximately.

- The computation cost of these ADM algorithms is reasonably low. Both the shrinkage and projection operations are very cheap to compute. Although solving $m \times m$ linear systems appears to be costly, we emphasize that we can either apply the precomputed matrix inverse or just take one gradient descent step, thereby reducing the cost to several matrix-vector multiplications. Therefore, the main computation cost of these ADM algorithms is only matrix-vector multiplications. In practice, many matrix-vector multiplications can be performed by fast transforms such as fast Fourier transform (FFT), fast cosine transform (FCT) and fast Walsh-Hadamard transform (FWHT), making the algorithms more efficient.

- The one-step gradient descent is a very practical approach for solving quadratic subproblems in the ADM algorithms. Using this approach, simple manipulation shows that all the previous ADM schemes require only two matrix-vector multiplications per iteration: one with A and the other with A^T. Consequently, the storage of the matrix A is not needed, and A can be accepted as two linear operators for the multiplications with A and A^T. It makes the algorithms capable of dealing with large-scale problems.

- As we can see, $AA^T = I$ is a favorable case for our algorithms. It not only

makes both of the subproblems possible to be solved exactly, but also reduces each iteration's computation cost to only two matrix-vector multiplications. Therefore, it is beneficial to use A with orthonormal rows. In many applications, for example in compressive sensing, A is often formed by randomly taking a subset of rows from orthonormal transform matrices such as 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, satisfying $AA^T = I$.

Chapter 3

A Special Case and Several Extensions

3.1 Joint Sparsity

An interesting special case of group sparsity is the so-called joint sparsity. A set of signals is called jointly sparse if they are not only sparse, but also share a common nonzero support. Such signals arise in cognitive radio networks [21], distributed compressive sensing [3], direction-of-arrival estimation in radar [18], 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.

Let $X = [x_1, \ldots, x_l] \in \mathbb{R}^{n\times l}$ be a collection of l signals that are jointly sparse. That is, only a few rows of X contain nonzero components, while most of rows are all zeros. Indeed, joint sparsity is a special non-overlapping group sparsity structure, where each group corresponds to one row of the matrix X. Likewise, the (weighted) $\ell_{2,1}$-regularization has been commonly used to enforce joint sparsity. We consider the joint sparse basis pursuit model:

$$\begin{aligned} \min_X \quad & \|X\|_{w,2,1} := \sum_{i=1}^{n} w_i \|x^i\|_2 \\ \text{s.t.} \quad & AX = B, \end{aligned} \tag{3.1}$$

where $A \in \mathbb{R}^{m\times n}\,(m < n)$, $B \in \mathbb{R}^{m\times l}$ and $w_i \geq 0$ for $i = 1, ..., n$. We use x^i and x_j to

denote the i-th row and j-th column of X respectively.

The primal-based ADM scheme (Algorithm 2) reduces to the following form:

$$\begin{cases} Z^{k+1} = \text{GShrink}(X^k + \frac{1}{\beta_1}\Lambda_1^k, \frac{1}{\beta_1}w); \\ X^{k+1} = (\beta_1 I + \beta_2 A^T A)^{-1}(\beta_1 Z^{k+1} + \Lambda_1^k + \beta_2 A^T B + A^T \Lambda_2^k), \\ \Lambda_1^{k+1} = \Lambda_1^k - \gamma\beta_1(X^{k+1} - Z^{k+1}), \\ \Lambda_2^{k+1} = \Lambda_2^k - \gamma\beta_2(AX^{k+1} - B). \end{cases} \tag{3.2}$$

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, $\gamma > 0$ is a step-length. The group-wise shrinkage operator "GShrink" represents

$$z^i = \max\left\{\|r^i\|_2 - \frac{w_i}{\beta_1}, 0\right\}\frac{r^i}{\|r^i\|_2}, \text{ for } i = 1, ..., n, \tag{3.3}$$

where

$$r^i := x^i + \frac{1}{\beta_1}\lambda_1^i. \tag{3.4}$$

Likewise, the dual of (3.1) is given by

$$\begin{aligned} \max_Y \quad & B \bullet Y \\ \text{s.t.} \quad & \|A_i^T Y\|_2 \le w_i, \text{ for } i = 1, ..., n, \end{aligned} \tag{3.5}$$

where "$\bullet$" denotes the sum of component-wise products. The dual-based ADM scheme (Algorithm 5) for the joint sparsity problem is of the following form:

$$\begin{cases} Z^{k+1} = \mathcal{P}_{\mathbf{B}_2'}(A^T Y^k + \frac{1}{\beta}X^k); \\ Y^{k+1} = (\beta AA^T)^{-1}(B - AX^k + \beta AZ^{k+1}), \\ X^{k+1} = X^k - \gamma\beta(Z^{k+1} - A^T Y^{k+1}). \end{cases} \tag{3.6}$$

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

Algorithms for the basis pursuit denoising models BP_δ and BP_μ can be similarly derived, and thus are omitted.

3.2 Nonnegativity

We consider an extension of the group sparse models to enforce nonnegativity in the solution. In many applications, the signals naturally have nonnegative components. For example, the pixel values of images are nonnegative. Our ADM algorithms can be easily extended to include the nonnegativity constraint $x \geq 0$ (or equivalently $z \geq 0$ when we do splitting $z = x$).

In fact, for the primal models, we only need to modify the group-wise shrinkage formula to take into account the nonnegativity, which is given by Lemma 3.2.

Lemma 3.1. (Shrinkage with Nonnegativity) *For any $\alpha, \beta > 0$ and $t \in \mathbb{R}^n$, the minimizer of*

$$\begin{aligned} \min_{z \in \mathbb{R}^n} \quad & \alpha \|z\|_2 + \frac{\beta}{2} \|z - t\|_2^2 \\ s.t. \quad & z \geq 0 \end{aligned} \tag{3.7}$$

is given by

$$z(t) = Shrink\left(t_+, \frac{\alpha}{\beta}\right) := \max\left\{\|t_+\|_2 - \frac{\alpha}{\beta}, 0\right\} \frac{t_+}{\|t_+\|_2}, \tag{3.8}$$

where $t_+ = \max\{t, 0\}$ denotes the nonnegative part of t.

Proof. Suppose z^* is the minimizer of (3.7). Obviously, if $t_i \leq 0$, then $z_i^* = 0$. Therefore, it is easy to show that (3.7) is equivalent to

$$\begin{aligned} \min_{z \in \mathbb{R}^n} \quad & \alpha \|z\|_2 + \frac{\beta}{2} \|z - t_+\|_2^2 \\ \text{s.t.} \quad & z \geq 0, \end{aligned} \tag{3.9}$$

when t is replaced by t_+. Since $t_+ \geq 0$, the minimizer of

$$\min_{z \in \mathbb{R}^n} \quad \alpha \|z\|_2 + \frac{\beta}{2} \|z - t_+\|_2^2 \tag{3.10}$$

must be nonnegative. That is to say, (3.9) is equivalent to the above problem without the nonnegativity constraint. Therefore, the original problem (3.7) is equivalent to

(3.10). By Lemma 2.1, the minimizer of (3.10) is given by the shrinkage formula (3.8). □

Lemma 3.2. (Group-wise Shrinkage with Nonnegativity) *Suppose the groups* $\{g_i : i = 1, ..., s\}$ *form a partition of* $\{1, 2, \ldots, n\}$. *For any* $\beta > 0$ *and* $t \in \mathbb{R}^n$, *the minimizer of*

$$\begin{aligned} \min_{z \in \mathbb{R}^n} \quad & \|z\|_{w,2,1} + \frac{\beta}{2}\|z - t\|_2^2 \\ s.t. \quad & z \geq 0 \end{aligned} \tag{3.11}$$

is given by

$$z(t) = GShrink\left(t_+, \frac{w}{\beta}\right), \tag{3.12}$$

i.e.,

$$z_{g_i}(t) = \max\left\{\|(t_{g_i})_+\|_2 - \frac{w_i}{\beta}, 0\right\} \frac{(t_{g_i})_+}{\|(t_{g_i})_+\|_2}, \ for\ i = 1, \ldots, s. \tag{3.13}$$

Proof. Since the groups $\{g_i : i = 1, ..., s\}$ form a partition, it is easy to see that (3.11) is equivalent to minimizing the following s subproblems individually:

$$\begin{aligned} \min_{z_{g_i} \in \mathbb{R}^{n_i}} \quad & w_i\|z_{g_i}\|_2 + \frac{\beta}{2}\|z_{g_i} - t_{g_i}\|_2^2, \\ \text{s.t.} \quad & z_{g_i} \geq 0, \end{aligned} \tag{3.14}$$

for $i = 1, \ldots, s$. By Lemma 3.1, the closed-form solution of each subproblem is given by (3.13). □

Lemma 3.2 tells us how to modify the group-wise shrinkage formula in order to have nonnegativity. Essentially, the group-wise shrinkage is applied to the nonnegative part of an vector, rather than the vector itself. This simple change will make our algorithms capable of solving the primal group sparse models with nonnegativity constraints.

Adding the nonnegativity constraint makes the dual problems slightly different. For example, we derive the dual of the BP model (2.2) with $x \geq 0$. The dual is given by

$$\begin{aligned} &\max_y \left\{ \min_{x \geq 0} \sum_{i=1}^{s} w_i \|x_{g_i}\|_2 - y^T(Ax - b) \right\} \\ =&\max_y \left\{ b^T y + \min_{x \geq 0} \sum_{i=1}^{s} \left(w_i \|x_{g_i}\|_2 - y^T A_{g_i} x_{g_i} \right) \right\}. \end{aligned} \tag{3.15}$$

Since

$$\min_{x_{g_i} \geq 0} \ \left(w_i \|x_{g_i}\|_2 - y^T A_{g_i} x_{g_i} \right) = \begin{cases} 0 & \text{if } \|(A_{g_i}^T y)_+\| \leq w_i \\ -\infty & \text{otherwise} \end{cases}, \tag{3.16}$$

it follows that the dual is given by

$$\max_y \left\{ b^T y : \|(A_{g_i}^T y)_+\|_2 \leq w_i, \text{ for } i = 1, ..., s \right\}, \tag{3.17}$$

or equivalently,

$$\begin{aligned} \min_{y,z} \quad & -b^T y \\ \text{s.t.} \quad & z = A^T y, \\ & z \in \mathbf{B}_2^{\mathcal{G}+}, \end{aligned} \tag{3.18}$$

where $\mathbf{B}_2^{\mathcal{G}+} := \{z \in \mathbb{R}^n : \|(z_{g_i})_+\|_2 \leq w_i, \text{ for } i = 1, ..., s\}$. The dual derivation is similar for BP_δ and BP_μ models. By adding the nonnegativity constraint, the only difference resulted in the dual problems is that the constraint $z \in \mathbf{B}_2^{\mathcal{G}}$ is now replaced by $z \in \mathbf{B}_2^{\mathcal{G}+}$. Consequently, the only change we need to make in the dual ADM algorithms is to replace the projection $\mathcal{P}_{\mathbf{B}_2^{\mathcal{G}}}$ by $\mathcal{P}_{\mathbf{B}_2^{\mathcal{G}+}}$ for the z-subproblem.

3.3 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 [17].

We consider the BP model (2.2), where the groups $\{x_{g_1}, ..., x_{g_s}\}$ now have overlaps making 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:

$$\begin{aligned} \min_{x,z} \quad & \sum_{i=1}^{s} w_i \|z_i\|_2 \\ \text{s.t.} \quad & z = \tilde{x} \\ & Ax = b, \end{aligned} \tag{3.19}$$

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:

$$\min_{x,z} \sum_{i=1}^{s} w_i \|z_i\|_2 - \lambda_1^T (z - \tilde{x}) + \frac{\beta_1}{2} \|z - \tilde{x}\|_2^2 - \lambda_2^T (Ax - b) + \frac{\beta_2}{2} \|Ax - b\|_2^2, \tag{3.20}$$

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 group-wise shrinkage formula for each group of variables z_i. 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

$$\tilde{x} = Gx, \tag{3.21}$$

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

$$\min_x \ \lambda_1^T Gx + \frac{\beta_1}{2}\|z - Gx\|_2^2 - \lambda_2^T Ax + \frac{\beta_2}{2}\|Ax - b\|_2^2, \tag{3.22}$$

which is equivalent to solving the following linear system:

$$(\beta_1 G^T G + \beta_2 A^T A)x = \beta_1 G^T z - G^T \lambda_1 + \beta_2 A^T b + A^T \lambda_2. \tag{3.23}$$

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 (3.19) as follows:

$$\begin{aligned}
& \max_{y,p} \quad \left\{ \min_{x,z} \sum_{i=1}^{s} w_i \|z_{g_i}\|_2 - y^T(Ax - b) - p^T(z - Gx) \right\} \\
= \ & \max_{y,p} \quad \left\{ b^T y + \min_z \sum_{i=1}^{s} \left(w_i \|z_{g_i}\|_2 - p_i^T z_i \right) + \min_x \left(-A^T y + G^T p \right)^T x \right\} \\
= \ & \max_{y,p} \quad \left\{ b^T y : G^T p = A^T y, \ \|p_i\|_2 \le w_i, \text{ for } i = 1, ..., s \right\},
\end{aligned} \tag{3.24}$$

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:

$$\begin{aligned}
\min_{y,p,q} \quad & -b^T y \\
\text{s.t.} \quad & G^T p = A^T y, \\
& p = q, \\
& \|q_i\|_2 \le w_i, \ \text{ for } i = 1, ..., s.
\end{aligned} \tag{3.25}$$

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.

3.4 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.

3.5 Weights Inside Groups

Although we have considered an weighted version of the $\ell_{2,1}$-norm (2.1), 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 (2.1). For $x \in \mathbb{R}^n$, the weighted ℓ_2-norm is given by

$$\|x\|_{\bar{W},2} := \|\bar{W}x\|_2, \tag{3.26}$$

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 (2.2) becomes

$$\begin{aligned} \min_x \quad & \sum_{i=1}^{s} w_i \|W^{(i)} x_{g_i}\|_2 \\ \text{s.t.} \quad & Ax = b, \end{aligned} \tag{3.27}$$

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

$$\begin{aligned} \min_z \quad & \sum_{i=1}^{s} w_i \|z_i\|_2 \\ \text{s.t.} \quad & z = WGx,\ Ax = b, \end{aligned} \tag{3.28}$$

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.

Chapter 4

Convergence of Alternating Direction Methods

This chapter reviews the existing ADM theory that guarantees the global convergence of all the previously derived algorithms under certain parameter restrictions. In addition, we also extend the convergence theory to allow more generality.

4.1 General Framework

The group sparse problems, either in the primal or the dual form, can be categorized as the following optimization problem:

$$\begin{aligned} \min_{x,z} \ & f(x) + g(z) \\ \text{s.t. } & \mathcal{A}x + \mathcal{B}z = c, \\ & x \in \mathcal{X}, \ z \in \mathcal{Z}, \end{aligned} \tag{4.1}$$

where $f : \mathbb{R}^{n_1} \to \mathbb{R}$, $g : \mathbb{R}^{n_2} \to \mathbb{R}$ are convex functions, $\mathcal{A} \in \mathbb{R}^{p \times n_1}$, $\mathcal{B} \in \mathbb{R}^{p \times n_2}$, $\mathcal{X} \subseteq \mathbb{R}^{n_1}$ and $\mathcal{Z} \subseteq \mathbb{R}^{n_2}$ are closed convex sets. The representation of these notations corresponding to the different group sparse models is summarized in Table 4.1. Note that the matrix $\mathcal{A}$ here is denoted by calligraphic letter to distinguish from the "sensing matrix" A in the group sparse models.

4.2 Existing Convergence Result for Exact ADM

The convergence has been established for the standard alternating direction method (Algorithm 1) in which both subproblems are solved exactly in every iteration. The

Table 4.1 : Representation of the notations in (4.1) with respect to each group sparse model.

	BP	BP_δ	BP_μ	Dual BP	Dual BP_δ	Dual BP_μ
f	0	$\|x\|_{w,2,1}$	$\|x\|_{w,2,1}$	$-b^T x$	$-b^T x + \delta\|x\|$	$-b^T x + \frac{\mu}{2}\|x\|^2$
g	$\|z\|_{w,2,1}$	0	$\frac{1}{2\mu}\|z\|^2$	0	0	0
$\mathcal{A}$	$\begin{bmatrix} I \\ A \end{bmatrix}$	A	A	A^T	A^T	A^T
$\mathcal{B}$	$\begin{bmatrix} -I \\ 0 \end{bmatrix}$	I	I	$-I$	$-I$	$-I$
c	$\begin{bmatrix} 0 \\ b \end{bmatrix}$	b	b	0	0	0
$\mathcal{X}$	$\mathbb{R}^n$	$\mathbb{R}^n$	$\mathbb{R}^n$	$\mathbb{R}^m$	$\mathbb{R}^m$	$\mathbb{R}^m$
$\mathcal{Z}$	$\mathbb{R}^n$	$\mathbf{B}_\delta$	$\mathbb{R}^m$	$\mathbf{B}_2^{\mathcal{G}}$	$\mathbf{B}_2^{\mathcal{G}}$	$\mathbf{B}_2^{\mathcal{G}}$

convergence result is stated by the following theorem. The proof can be found in [9, 10] (for $\mathcal{A} = I$ and without $x \in \mathcal{X}, z \in \mathcal{Z}$) and [14] (for the general case).

Theorem 4.1. *Let $\{(x^k, z^k, \lambda^k)\}$ be the sequence generated by the alternating direction method (Algorithm 1) from any initial point (x^0, z^0, λ^0) with $\beta > 0$ and $\gamma \in \left(0, \frac{\sqrt{5}+1}{2}\right)$. If both matrices $\mathcal{A}$ and $\mathcal{B}$ have full column rank, then the sequence $\{(x^k, z^k, \lambda^k)\}$ converges to an optimal primal-dual solution of* (4.1).

Therefore, the convergence of the previously derived algorithms for the group sparse optimization problems follows directly, provided that both subproblems are to be solved exactly.

Corollary 4.1. *Let $\{(x^k, z^k)\}$ be the sequence generated by Algorithm 2 (BP), in which both subproblems are solved exactly, and the parameters $\beta_1, \beta_2 > 0$, $\gamma \in \left(0, \frac{\sqrt{5}+1}{2}\right)$. Then, from any initial point, the sequence $\{(x^k, z^k)\}$ converges to an*

optimal solution of the BP model (2.12).

Corollary 4.2. *Suppose the matrix A (in the group sparse models) has full row rank. Let $\{(x^k, y^k, z^k)\}$ be the sequence generated by Algorithms 5-7 (dual of BP, BP_δ and BP_μ, respectively), in which both subproblems are solved exactly, and the parameters $\beta > 0$, $\gamma \in \left(0, \frac{\sqrt{5}+1}{2}\right)$. Then, from any initial point, the sequence $\{(x^k, y^k, z^k)\}$ converges to an optimal primal-dual solution of the dual BP* (2.36)*, dual BP_δ* (2.48) *and dual BP_μ* (2.56) *models, respectively. Therefore, $\{x^k\}$ converges to an optimal solution of the primal BP* (2.2)*, BP_δ* (2.3) *and BP_μ* (2.4) *models.*

Note that the "sensing matrix" A in compressive sensing is usually chosen to be some random matrix with fewer rows than columns, so it almost always satisfies the full row rank condition.

For Algorithm 3 (BP_δ) and Algorithm 4 (BP_μ), however, the x-subproblem is generally no easier to solve than the original problems, so it is always solved approximately by the linear proximal approach. Even though exact minimization for both subproblems is possible for the other algorithms, it is not practical when the problem becomes large and the "sensing matrix" A does not satisfy $AA^T = I$. Therefore, it is of practical importance to consider the convergence for those inexact ADM schemes that solve subproblems approximately using the linear proximal method or the one-step gradient descent method. In [13], the convergence of such inexact ADM schemes has been established for $\gamma = 1$. In the following sections, we will extend the convergence result to allow $\gamma > 1$. This is a meaningful extension, since empirical evidence shows that the algorithms often convergence faster when $\gamma > 1$.

4.3 Inexact Alternating Direction Methods

In the previously derived ADM schemes for the group sparse problems, recall that the z-subproblem is always easy to solve. For example, the closed-form solution is obtained by group-wise shrinkage or projection, and can be easily computed. However, the x-subproblem is much difficult to solve. It is either no easier than solving the original problem, or is a quadratic problem which is expensive to minimize exactly. Therefore, the linear proximal method and the one-step gradient descent method have been used to solve such subproblems approximately.

4.3.1 Linear Proximal Method

Consider the x-subproblem:

$$\min_{x\in\mathcal{X}} f(x) + \frac{\beta}{2}\|\mathcal{A}x + \mathcal{B}z^{k+1} - c - \lambda^k/\beta\|_2^2. \tag{4.2}$$

The linear proximal method linearizes the quadratic penalty term and adds a proximal term, so it solves the following problem

$$\min_{x\in\mathcal{X}} f(x) + \beta\left((g^k)^T(x - x^k) + \frac{1}{2\tau}\|x - x^k\|_2^2\right), \tag{4.3}$$

where $\tau > 0$ is a proximal parameter, and

$$g^k := \mathcal{A}^T(\mathcal{A}x^k + \mathcal{B}z^{k+1} - c - \lambda^k/\beta) \tag{4.4}$$

is the gradient of the quadratic penalty term. Essentially, this method can be viewed as using a quadratic approximation with an identity Hessian matrix $\frac{\beta}{\tau}I$ to approximate the original quadratic penalty term whose Hessian is $\beta\mathcal{A}^T\mathcal{A}$.

For example, when f is the $\ell_{w,2,1}$-norm or ℓ_2-norm, this problem (4.3) has a close-form solution by the shrinkage formula, which is much easier than solving the original x-subproblem.

4.3.2 One-step Projected Gradient Descent

When f is quadratic, both (4.2) and (4.3) need to minimize a quadratic function, which could be expensive for large-scale problems. One simple way is to just take one step of gradient descent and then project onto the set $\mathcal{X}$:

$$x^{k+1} = \mathcal{P}_{\mathcal{X}}\left(x^k - \alpha \bar{g}^k\right) \tag{4.5}$$

where

$$\bar{g}^k := \nabla f(x^k) + \beta \mathcal{A}^T(\mathcal{A}x^k + \mathcal{B}z^{k+1} - c - \lambda^k/\beta) \tag{4.6}$$

is the gradient of the objective function at the current point, and $\alpha > 0$ is a step length. From another point of view, it is equivalent to the following minimization problem:

$$\min_{x \in \mathcal{X}} \ (\bar{g}^k)^T(x - x^k) + \frac{1}{2\alpha}\|x - x^k\|^2. \tag{4.7}$$

Thus, it can be regarded as applying the linear proximal approach to the objective function, instead of just the quadratic penalty term.

While this approach can be applied to general functions f, we emphasize that taking just one step of projected gradient descent may not be good enough. In general, more steps may be taken to get better approximation. However, we are often faced with minimizing large-scale quadratic functions or equivalently solving large linear systems in many applications. As will be shown, at least for quadratic functions this simple step is sufficient for the algorithm to converge to an optimal solution, while significantly reducing the computational cost at each iteration.

4.3.3 Generalized Inexact Minimization Approach

In fact, both the linear proximal method and the one-step projected gradient descent method (for quadratic function f) can be generalized as follows:

$$\min_{x\in\mathcal{X}} \mathcal{L}_{\mathcal{A}}(x, z^{k+1}, \lambda^k) + \frac{1}{2}\|x - x^k\|^2_{\hat{P}}, \tag{4.8}$$

where $\hat{P}$ is some positive definite matrix, $\mathcal{L}_{\mathcal{A}}$ is the augmented Lagrangian function, and $\|x\|_M := \sqrt{x^T M x}$. That is, a proximal term is added to the original x-subproblem.

By (4.3), it is easy to see that the linear proximal method corresponds to the case that $\hat{P} = \frac{\beta}{\tau} I - \beta \mathcal{A}^T\mathcal{A}$. For quadratic function f, let $H_f := \nabla^2 f(x) \succeq 0$ be the Hessian matrix. Then, (4.7) indicates that the one-step projected gradient descent is given by $\hat{P} = \frac{1}{\alpha} I - H_f - \beta\mathcal{A}^T\mathcal{A}$. In general, different positive definite matrices $\hat{P}$ will give rise to many other inexact minimization schemes.

Therefore, we are interested in the following inexact alternating direction method (Algorithm 8) which solves one subproblem approximately.

Algorithm 8: Inexact Alternating Direction Method

1 Initialize $x^0,\ \lambda^0,\ \beta > 0, \gamma > 0$;

2 **for** $k = 0,\ 1, \ldots$ **do**

3 $\quad z^{k+1} = \arg\min_{z\in\mathcal{Z}} \mathcal{L}_{\mathcal{A}}(x^k, z, \lambda^k)$;

4 $\quad x^{k+1} = \arg\min_{x\in\mathcal{X}} \mathcal{L}_{\mathcal{A}}(x, z^{k+1}, \lambda^k) + \frac{1}{2}\|x - x^k\|^2_{\hat{P}}$;

5 $\quad \lambda^{k+1} = \lambda^k - \gamma\beta(\mathcal{A}x^{k+1} + \mathcal{B}z^{k+1} - c)$;

4.4 Global Convergence

The global convergence of Algorithm 8 has been established in [13] for the case that $\gamma = 1$. In this sections, we will extend the convergence result to allow more generality on γ. This is a meaningful extension since empirical evidence shows that the algorithms often convergence faster when $\gamma > 1$.

4.4.1 Optimality Conditions

For convenience, we assume both functions f and g are continuously differentiable. Let (x^*, z^*, λ^*) be an optimal solution. From optimization theory, the optimality conditions for (4.1) are

$$\langle z - z^*, \nabla g(z^*) - \mathcal{B}^T\lambda^* \rangle \geq 0, \quad \forall z \in \mathcal{Z}, \tag{4.9}$$

$$\langle x - x^*, \nabla f(x^*) - \mathcal{A}^T\lambda^* \rangle \geq 0, \quad \forall x \in \mathcal{X}, \tag{4.10}$$

$$\mathcal{A}x^* + \mathcal{B}z^* - c = 0. \tag{4.11}$$

Since the z-subproblem is assumed to be solved exactly, its optimality condition is

$$\langle z - z^{k+1}, \nabla g(z^{k+1}) - \mathcal{B}^T\lambda^k + \beta\mathcal{B}^T(\mathcal{A}x^k + \mathcal{B}z^{k+1} - c) \rangle \geq 0, \ \forall z \in \mathcal{Z}. \tag{4.12}$$

To simplify the notations in our analysis, we introduce $\hat{x} = x^{k+1}$ and

$$\hat{\lambda} := \lambda^k - \beta(\mathcal{A}x^{k+1} + \mathcal{B}z^{k+1} - c). \tag{4.13}$$

Note that $\hat{\lambda} \neq \lambda^{k+1}$ unless $\gamma = 1$. Then (4.12) can be written as

$$\langle z - z^{k+1}, \nabla g(z^{k+1}) - \mathcal{B}^T\hat{\lambda} + \beta\mathcal{B}^T\mathcal{A}(x^k - x^{k+1}) \rangle \geq 0, \ \forall z \in \mathcal{Z}. \tag{4.14}$$

The optimality for the inexact x-subproblem (4.8) is given by

$$\langle x - x^{k+1}, \nabla f(x^{k+1}) - \mathcal{A}^T\hat{\lambda} - \hat{P}(x^k - x^{k+1}) \rangle \geq 0, \ \forall x \in \mathcal{X}. \tag{4.15}$$

4.4.2 Convergence Analysis

First, some vectors and matrices are to be introduced to simplify the notations in the analysis. Let $P = \hat{P} + \beta \mathcal{A}^T \mathcal{A}$, and

$$u := \begin{pmatrix} x \\ \lambda \end{pmatrix} \in \mathbb{R}^{n+p},\ G_0 := \begin{pmatrix} I_n & \\ & \gamma I_p \end{pmatrix},\ G_1 := \begin{pmatrix} P & \\ & \frac{1}{\beta} I_p \end{pmatrix},\ G := G_0^{-1} G_1. \quad (4.16)$$

Lemma 4.1.

$$(u^k - u^*)^T G_1 (u^k - \hat{u}) \geq \|u^k - \hat{u}\|_{G_1}^2 + \langle A(x^k - \hat{x}), \lambda^k - \hat{\lambda} \rangle. \quad (4.17)$$

Proof. On z-subproblem, letting $z = z^{k+1}$ in (4.9) and $z = z^*$ in (4.14), we have

$$\begin{cases} \langle z^{k+1} - z^*, \nabla g(z^*) - \mathcal{B}^T \lambda^* \rangle \geq 0, \\ \langle z^* - z^{k+1}, \nabla g(z^{k+1}) - \mathcal{B}^T \hat{\lambda} + \beta \mathcal{B}^T \mathcal{A}(x^k - x^{k+1}) \rangle \geq 0. \end{cases} \quad (4.18)$$

By adding them together and using the convexity of g, we have

$$\langle \mathcal{B}(z^{k+1} - z^*), \hat{\lambda} - \lambda^* - \beta \mathcal{A}(x^k - x^{k+1}) \rangle \geq 0. \quad (4.19)$$

Similarly, on x-subproblem, by combining (4.10) and (4.15), and using the convexity of f, we have

$$\langle \mathcal{A}(x^{k+1} - x^*),\ \hat{\lambda} - \lambda^* - \beta \mathcal{A}(x^k - x^{k+1}) \rangle + \langle x^{k+1} - x^*, P(x^k - x^{k+1}) \rangle \geq 0. \quad (4.20)$$

From (4.11) and (4.13), it follows that

$$\mathcal{A}(x^{k+1} - x^*) + \mathcal{B}(z^{k+1} - z^*) = \frac{1}{\beta}(\lambda^k - \hat{\lambda}). \quad (4.21)$$

Then, adding (4.19) and (4.20) gives

$$\frac{1}{\beta} \langle \lambda^k - \hat{\lambda},\ \hat{\lambda} - \lambda^* - \beta \mathcal{A}(x^k - x^{k+1}) \rangle + \langle x^{k+1} - x^*,\ P(x^k - x^{k+1}) \rangle \geq 0 \quad (4.22)$$

which can be simplified as

$$(\hat{u} - u^*)^T G_1(u^k - \hat{u}) \geq \langle \mathcal{A}(x^k - \hat{x}),\ \lambda^k - \hat{\lambda} \rangle. \tag{4.23}$$

By rearranging the terms in (4.23), we immediately get (4.17). □

Lemma 4.2. *If $(2-\gamma)P \succ \beta \mathcal{A}^T\mathcal{A}$, then there exists $\eta > 0$, such that*

$$\|u^k - u^*\|_G^2 - \|u^{k+1} - u^*\|_G^2 \geq \eta \|u^k - u^{k+1}\|_G^2. \tag{4.24}$$

Proof. Since $u^{k+1} = u^k - G_0(u^k - \hat{u})$, we have

$$\begin{aligned}
&\|u^k - u^*\|_G^2 - \|u^{k+1} - u^*\|_G^2 \\
&= 2(u^k - u^*)^T G_1(u^k - \hat{u}) - \|G_0(u^k - \hat{u})\|_G^2. \\
&\geq 2\|u^k - \hat{u}\|_{G_1}^2 + 2\langle \mathcal{A}(x^k - \hat{x}), \lambda^k - \hat{\lambda} \rangle - \|u^k - \hat{u}\|_{G_0 G G_0}^2 \text{ (by Lemma 4.1)} \\
&= \|x^k - \hat{x}\|_P^2 + \frac{2-\gamma}{\beta}\|\lambda^k - \hat{\lambda}\|^2 + 2\langle \mathcal{A}(x^k - \hat{x}), \lambda^k - \hat{\lambda} \rangle \\
&\geq \|x^k - x^{k+1}\|_{P - \frac{1}{\rho}\mathcal{A}^T\mathcal{A}}^2 + \left(\frac{2-\gamma}{\beta} - \rho\right)\frac{1}{\gamma^2}\|\lambda^k - \lambda^{k+1}\|^2,\ \forall \rho > 0
\end{aligned} \tag{4.25}$$

where the last inequality follows from the Cauchy-Schwarz inequality

$$2\langle \mathcal{A}(x^k - \hat{x}), \lambda^k - \hat{\lambda} \rangle \geq -\frac{1}{\rho}\|\mathcal{A}(x^k - \hat{x})\|^2 - \rho\|\lambda^k - \hat{\lambda}\|^2,\ \forall \rho > 0. \tag{4.26}$$

To prove such $\eta > 0$ exists for (4.24), we only need to show there exists some $\rho > 0$ such that $P - \frac{1}{\rho}\mathcal{A}^T\mathcal{A} \succ 0$ and $\frac{2-\gamma}{\beta} - \rho > 0$, which holds if and only if $(2-\gamma)P \succ \beta\mathcal{A}^T\mathcal{A}$. □

Lemma 4.2 provides the key inequality to prove the global convergence of the algorithm. The main convergence result is stated in the following theorem.

Theorem 4.2. *If $(2-\gamma)P \succ \beta\mathcal{A}^T\mathcal{A}$ and the matrix $\mathcal{B}$ has full column rank, then the sequence $\{(x^k, z^k, \lambda^k)\}$ generated by the inexact alternating direction method (Algorithm 8), from any initial point, converges to an optimal primal-dual solution of* (4.1).

Proof. From Lemma 4.2, it follows that

(a) $\|u^k - u^*\|_G^2$ is nonincreasing and thus converges;

(b) $\|u^k - u^{k+1}\|_G^2 \to 0$, i.e., $x^k - x^{k+1} \to 0$ and $\lambda^k - \lambda^{k+1} \to 0$;

(c) $\|u^k - u^*\|_G^2$ is bounded, so $\{u^k\}$ has a convergent subsequence $\{u^{k_j}\} \to \bar{u}$.

From (b) and $\lambda^{k+1} = \lambda^k - \gamma\beta(\mathcal{A}x^{k+1} + \mathcal{B}z^{k+1} - c)$, it follows that

$$\mathcal{A}x^{k+1} + \mathcal{B}z^{k+1} - c \to 0. \tag{4.27}$$

By (c), we have $\mathcal{B}z^{k_j} \to c - \mathcal{A}\bar{x}$. Since $\mathcal{B}$ has full column rank, we have $z^{k_j} \to \bar{z}$ where $\bar{z}$ satisfies

$$\mathcal{A}\bar{x} + \mathcal{B}\bar{z} - c = 0. \tag{4.28}$$

Next we will show that the limit point $\left\{\bar{x}, \bar{z}, \bar{\lambda}\right\}$ is an optimal solution of (4.1). Recall the optimality conditions for each iteration: $\forall x \in \mathcal{X},\ \forall z \in \mathcal{Z}$,

$$\langle x - x^k, \nabla f(x^k) - \mathcal{A}^T[\lambda^{k-1} - \beta(\mathcal{A}x^k + \mathcal{B}z^k - c)] - \hat{P}(x^{k-1} - x^k)\rangle \geq 0,$$
$$\langle z - z^k, \nabla g(z^k) - \mathcal{B}^T[\lambda^{k-1} - \beta(\mathcal{A}x^k + \mathcal{B}z^k - c)] + \beta\mathcal{B}^T\mathcal{A}(x^{k-1} - x^k)\rangle \geq 0.$$

Taking the limit over k_j, since $x^{k-1} - x^k \to 0$, $\lambda^{k-1} - \lambda^k \to 0$ and (4.27), we have

$$\langle x - \bar{x}, \nabla f(\bar{x}) - \mathcal{A}^T\bar{\lambda}\rangle \geq 0,\ \forall x \in \mathcal{X}, \tag{4.29}$$
$$\langle z - \bar{z}, \nabla g(\bar{z}) - \mathcal{B}^T\bar{\lambda}\rangle \geq 0,\ \forall z \in \mathcal{Z}. \tag{4.30}$$

Along with (4.28), $\left\{\bar{x}, \bar{z}, \bar{\lambda}\right\}$ satisfies the optimality conditions for (4.1) and thus is an optimal solution. Since (4.24) holds for any optimal solution, we can let $(x^*, z^*, \lambda^*) = (\bar{x}, \bar{z}, \bar{\lambda})$. Then (a) implies that $\|u^k - u^*\|_G^2 \to 0$, i.e., $(x^k, \lambda^k) \to (x^*, \lambda^*)$. By (4.27), it follows that $y^k \to y^*$. □

Remark 4.1. *When the x-subproblem is solved using linear proximal ($P = \frac{\beta}{\tau} I$), the condition $(2-\gamma)P \succ \beta \mathcal{A}^T \mathcal{A}$ is equivalent to*

$$\tau \|\mathcal{A}\|^2 + \gamma < 2. \tag{4.31}$$

When the x-subproblem is solved by one-step projected gradient descent ($P = \frac{1}{\alpha} I - H_f$), a sufficient condition for $(2-\gamma)P \succ \beta \mathcal{A}^T \mathcal{A}$ is

$$\frac{\beta \|\mathcal{A}\|^2}{\frac{1}{\alpha} - \|H_f\|} + \gamma < 2. \tag{4.32}$$

For the group sparse problems, the functions f and g are not necessarily smooth. For example, f (or g) is $\|\cdot\|_{w,2,1}$ and $-b^T x + \delta \|x\|_2$. In these cases, our convergence analysis still carries over by substituting the gradients by subgradients. Note that the matrix $\mathcal{B}$ (in Table 4.1) always satisfies the full column rank condition. Therefore, all the derived Algorithms 2-7 are guaranteed to converge to an optimal solution under the condition (4.31) or (4.32), when one subproblem is solved approximately by linear proximal method or one-step gradient descent.

Chapter 5

Numerical Experiments

In this chapter, we first conduct a simple numerical experiment to demonstrate the benefit of group sparsity. Then we present various numerical results to evaluate the performance of our proposed ADM algorithms in comparison with the state-of-the-art algorithms SPGL1 [26], SpaRSA [28] and SLEP [19].

5.1 Experiment settings

In the experiments, group sparse signals are randomly generated as follows. First, we divide an vector $x \in \mathbb{R}^n$ evenly into s groups. Then we randomly pick k of them as active groups and let the other $s-k$ groups be all zeros. The components of the active groups are drawn randomly from i.i.d. Gaussian distribution, or other distributions if specified.

Two types of matrices $A \in \mathbb{R}^{m\times n}$ are used to generate the measurement data $b \in \mathbb{R}^m$. One is the set of random i.i.d. Gaussian matrices. For better scaling, each row is normalized to have unit length, but rows are not orthogonal in general. The other type of matrices is the randomized partial Walsh-Hadamard matrix. Rows are randomly chosen from a $n \times n$ Walsh-Hadamard matrix, and columns are randomly permuted. Such matrices have orthonormal rows so that $AA^T = I$. In addition, the matrix-vector multiplications with A and A^T can be computed by fast Walsh-Hadamard transforms, which are implemented in C with a MATLAB mex-interface

available to all codes compared. Therefore, such transform matrices are suitable for large-scale computation.

In some of the experiments, random Gaussian noise is added to the measurements $b \in \mathbb{R}^m$. The magnitude of the noise is chosen to be 0.5% of the magnitude of the measurements, i.e., the standard deviation of the Gaussian noise is set to be $0.5\% \cdot \|b\|_2$, where the signal-to-noise ratio (SNR) is about 46dB. For other noise levels, the numerical results are similar.

We use the default parameter setting for all the compared algorithms. In particular, the default parameter setting for the ADM algorithms are listed in Table 5.1 and Table 5.2, where $\beta_0 = \frac{1}{m}\|b\|_1$ is used for better scaling of the penalty parameter β. Note that $\gamma = 1.618 \approx \frac{\sqrt{5}+1}{2}$ is the upper bound in theoretical convergence guarantee for exact ADM. The primal-based ADM algorithms are denoted by PADM, and the dual-based ADM algorithms by DADM. In addition, we use "Exact", "LProx" and "GD" to distinguish between the different variants of the ADM algorithms. Specifically, "Exact" denotes the exact version, whereas "LProx" and "GD" represent those inexact versions that solve one subproblem by linear proximal (LProx) method or one-step gradient descent (GD) method. All the algorithms are initialized at zero. The weights in the $\ell_{w,2,1}$ are set to 1.

Table 5.1 : Parameter setting of ADM algorithms for BP model

	PADM-Exact	PADM-GD	DADM-Exact	DADM-GD
β	$[0.3, 3]/\beta_0$	$[0.2, 0.1]/\beta_0$	$2\beta_0$	$2\beta_0$
γ	1.618	1.618	1.618	1.618

Table 5.2 : Parameter setting of ADM algorithms for BP_δ and BP_μ models

	BP_δ		BP_μ		
	PADM-LProx	DADM-LProx	PADM-LProx	DADM-Exact	DADM-GD
β	$1/\beta_0$	$2\beta_0$	$1/\beta_0$	$2\beta_0$	$2\beta_0$
γ	1.1	1.1	1.1	1.618	1.618
τ	0.8	0.8	0.8	-	-

All 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.

5.2 Recoverability Test

This experiment tests the recoverability of the $\ell_{2,1}$-regularized group sparsity model, in comparison with the ℓ_1-regularized standard sparsity model. We randomly generate group sparse signals of size $n = 8192$ with $s = 1024$ groups, where each group has 8 components. Randomized partial Walsh-Hadamard matrices are used to generate the measurements. From each generated measurement data, we reconstruct the signal by solving the group sparse BP model (2.2) and the ℓ_1-minimization BP model, respectively. We use the dual-based Algorithm 5 for solving the group sparsity problem, and the YALL1 package [30] for the ℓ_1-minimization problem. The stopping criterion for both solvers are set to be $\|x^{k+1} - x^k\|/\|x^k\| < 10^{-6}$, i.e., the relative change of two consecutive iterates being smaller than the tolerance. We regard a recovery as successful if the relative error between the recovered solution x_r and the true signal x^* is less than 1×10^{-3}, i.e., $\|x_r - x^*\|/\|x^*\| < 10^{-3}$.

In the experiment, we fix the number of measurements to be $m = 2048$, and vary the group sparsity level from 50 nonzero groups to 120 nonzero groups. At each sparsity level, we run 50 trials. The result is shown in Figure 5.1. We also do the same experiment except that 0.5% Gaussian noise is added to the measurements, and the criterion for successful recovery is then adjusted to be relative error less than 2×10^{-2}. A similar result is obtained as Figure 5.1, and thus is omitted.

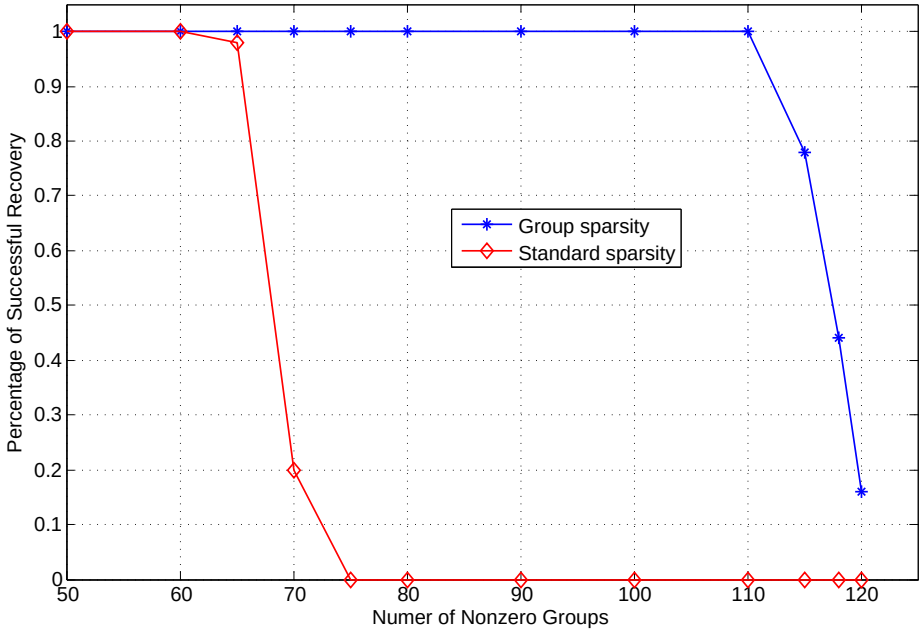


Figure 5.1 : Recoverability comparison: group sparsity ($\ell_{2,1}$-regularization) v.s. standard sparsity (ℓ_1-regularization).

Figure 5.1 shows that the group sparsity model exhibits much stronger recoverability than the standard sparsity model, which is not surprising since group sparsity encodes more prior information. As we can see, the ℓ_1-minimization fails completely when the number of nonzero groups is more than 70. In contrast, the $\ell_{2,1}$-minimization achieve 100% successful recovery for up to 110 nonzero groups. This experiment demonstrates the benefit of using group sparsity when we have prior information

about the grouping structure of the signals.

5.3 Convergence Rate Test

This set of experiments compare the decreasing behavior of recovery errors as each algorithm proceeds for a prescribed number of iterations. Our proposed different variants of ADM algorithms are compared with the state-of-the-art algorithms SPGL1, SpaRSA and SLEP. Since the main computational cost is roughly 2-4 matrix-vector multiplications for all compared algorithms, the decreasing of recovery error with respect to the number of iterations is a good measure of the efficiency of the algorithms.

In the experiments, we randomly generate group sparse signals of size $n = 2048$ with $s = 256$ groups, where each group has 8 components. The number of nonzero groups are set to be 25, unless otherwise specified. Random Gaussian matrices are used to generate the measurements, and the number of measurements is fixed to be $m = 512$. The BP model (2.2), BP_δ model (2.3) and BP_μ model (2.4) are solved to recover the signals. The relative error $\|x^{k+1} - x^*\|/\|x^*\|$ is plotted at each iteration, where x^* is the true signal. All the results are average of 50 runs.

5.3.1 BP Model

We compare the primal-based ADM (Algorithm 2), dual-based ADM (Algorithm 5) and SPGL1 for solving the BP model (2.2). Both the primal-based ADM (PADM) and the dual-based ADM (DADM) have an exact version and an inexact version. The exact version solves each subproblem exactly, whereas the inexact version solves a quadratic subproblem approximately by one-step gradient descent (GD). The decreasing of relative errors for these algorithms are shown in Figure 5.2 for noiseless data, and Figure 5.3 for noisy data where 0.5% Gaussian noise is added to the mea-

surements.

For the noiseless case, the true signal is exactly the optimal solution of the BP problem in our test. As we can see in Figure 5.2, all the algorithms will eventually converge to the optimal solution, thereby decreasing the error to machine precision. The three ADM solvers: PADM-Exact, DADM-Exact and DADM-GD perform almost identically and are the fastest ones, roughly three times faster than PADM-GD. The convergence rate of these ADM algorithms appear almost linear. SPGL1 is the slowest one.

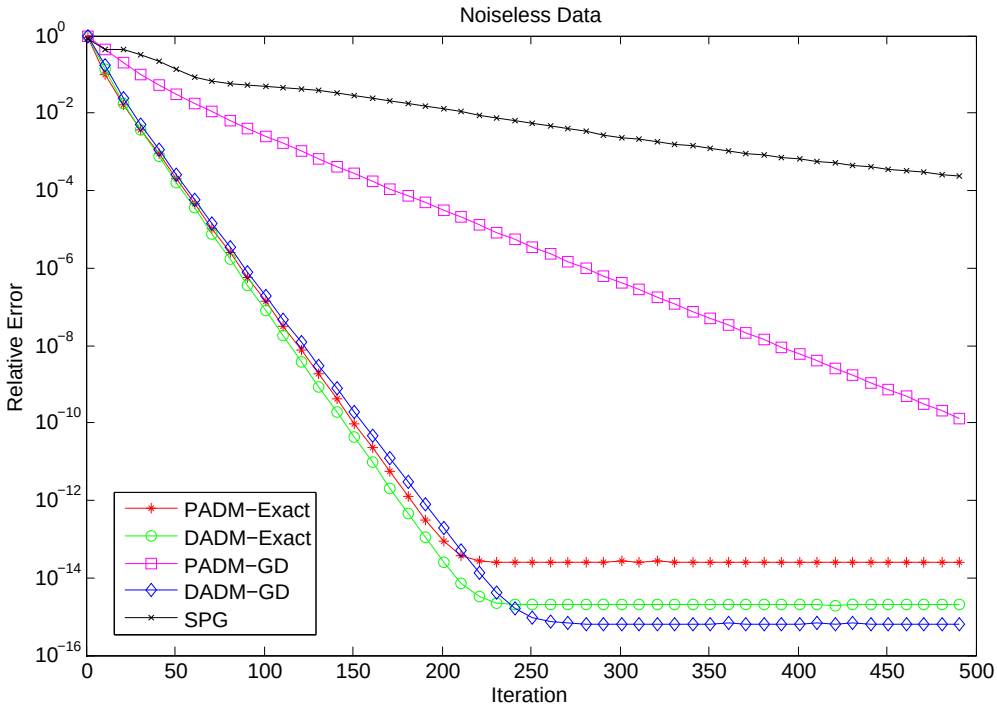


Figure 5.2 : BP model with noiseless data: comparison of the ADM algorithms and SPGL1 on the decreasing of recovery errors over iterations.

With 0.5% additive Gaussian noise in the measurements, 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}, and then start to go up a bit until convergence. This "down-then-up" phenomenon is because the optimal solution of the BP problem with erroneous data may not necessarily yield the best solution quality. In fact, the ADM algorithms still keep decreasing the objective function values even though the relative errors start to increase. This "down-then-up" phenomenon suggests that the ADM algorithms may give a better solution if it is stopped properly prior to convergence. We can see that PADM-Exact, DADM-Exact, DADM-GD decrease the relative error very quickly at the beginning and reach the lowest level with no more than 50 iterations. PADM-GD is a bit slower, reaching the lowest relative error after around 100 iterations. SPGL1 takes more than 200 iterations to decrease the relative error to 2×10^{-2}.

To compare the efficiency of the algorithms, we should not only consider how fast the recovery error is decreased over iterations, but also how much computation is needed per iteration. PADM-GD and DADM-GD are the cheapest ones, since they only consume two matrix-vector multiplications per iteration. DADM-Exact also consumes two matrix-vector multiplications with A and A^T, but additionally it needs to calculate the multiplication with the pre-computed $m \times m$ matrix $(AA^T)^{-1}$. Besides the multiplication with a pre-computed $m \times m$ matrix, PADM-Exact consumes four multiplications with A and A^T due to the use of the Sherman-Morrison-Woodbury formula. For SPGL1, the number of matrix-vector multiplications may vary over iterations, usually more than three per iteration on average.

In conclusion, the ADM algorithms are more efficient than SPGL1 in this test. The

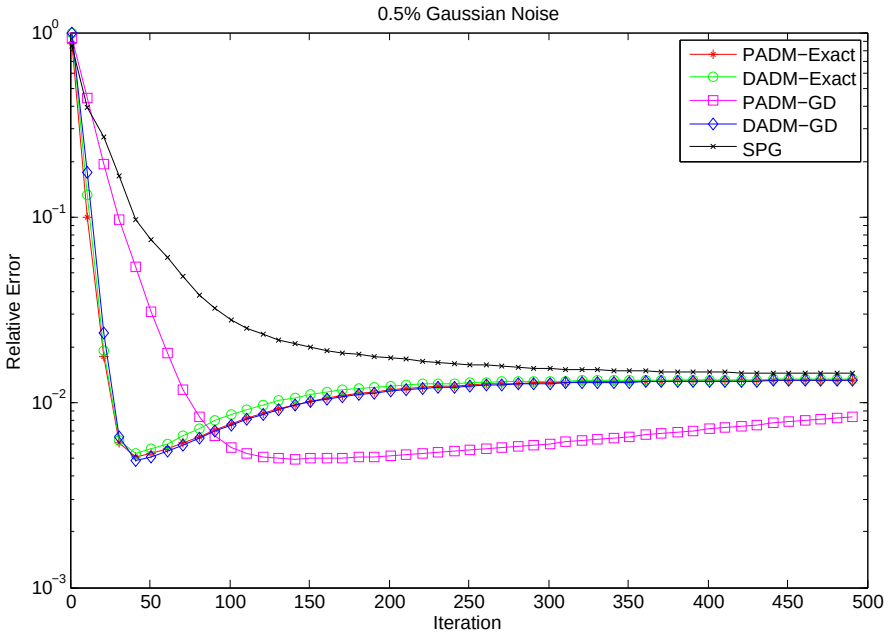


Figure 5.3 : BP model with 0.5% Gaussian noise: comparison of the ADM algorithms and SPGL1 on the decreasing of recovery errors over iterations.

two dual-based ADM algorithms: DADM-GD and DADM-Exact are the most efficient ones. Compared with PADM-Exact, PADM-GD trades off a little bit converge speed for the cheap computational cost. Overall, the efficiency of these two variants of PADM is similar in our test. However, we emphasize that PADM-Exact and DADM-Exact are not practical for large problems, since they need to compute a matrix inverse at the beginning.

5.3.2 BP_δ Model

Similarly, we compare the primal-based ADM (Algorithm 3), the dual-based ADM (Algorithm 6) and SPGL1 for solving the BP_δ model (2.3). Recall that both the primal-based ADM (PADM) and the dual-based ADM (DADM) use the linear proximal (LProx) approach to solve one subproblem approximately. Since 0.5% Gaussian

noise is added to the measurements b, the parameter δ is set to be $\delta = 0.5\% \cdot \|b\|_2$.

Figure 5.4 shows similar result as Figure 5.3. The relative errors of the ADM algorithms fall quickly below the SPGL1 curve, and exhibit a "down-then-up" curve. Eventually, the three curves converge to the same relative error level around 10^{-2}. However, due to the different solution paths, the ADM algorithms decrease the relative errors to a lower level with much fewer iterations than SPGL1. In addition, the main per-iteration computational cost for the ADM algorithms is two matrix-vector multiplications, whereas SPGL1 usually consumes more than three matrix-vector multiplications on average. Therefore, the ADM algorithms show better efficiency than SPGL1 in this test. Between the ADM algorithms, the dual-based ADM is slightly more efficient.

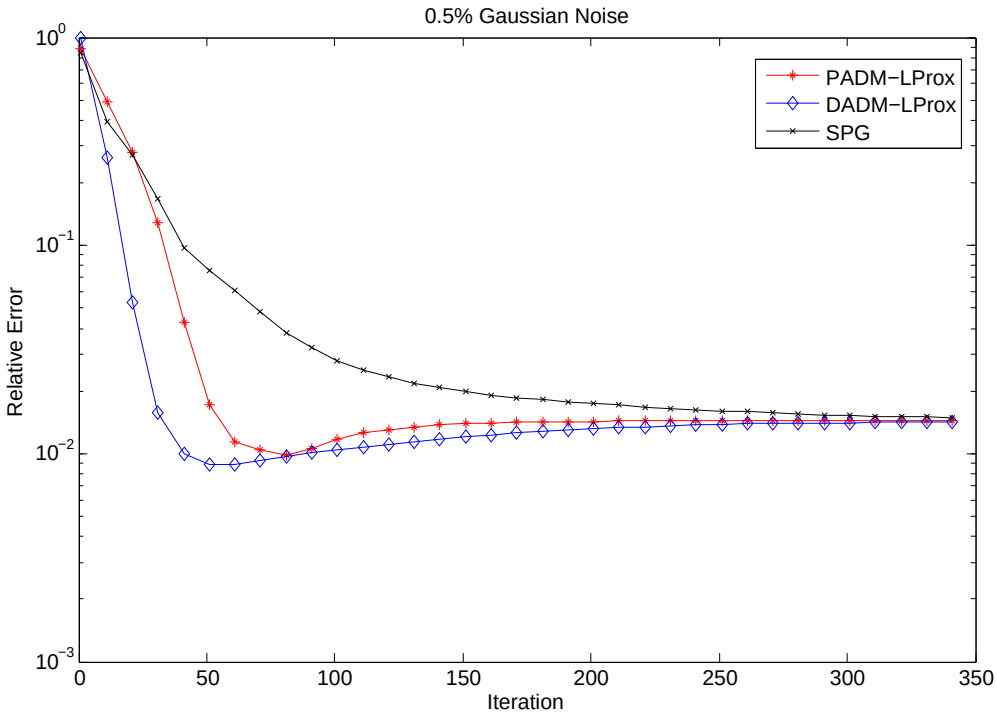


Figure 5.4 : BP_δ model with 0.5% Gaussian noise: comparison of the ADM algorithms and SPGL1 on the decreasing of recovery errors over iterations.

5.3.3 BP_μ Model

We also compare the primal-based ADM (Algorithm 4), the dual-based ADM (Algorithm 7), SpaRSA and SLEP for solving the BP_μ model (2.4). Recall that the primal-based ADM algorithm use the linear proximal method to solve one subproblem approximately. The dual-based ADM algorithm have an exact version (DADM-Exact), as well as an inexact version (DADM-LProx) which solves one subproblem approximately by the linear proximal method. The main computational cost for PADM-LProx, DADM-LProx and SLEP is two matrix-vector multiplications per iteration. DADM-Exact needs one more matrix-vector multiplications to solve a linear system at each iteration. For SpaRSA, the number of matrix-vector multiplications varies over iterations and is usually more than two.

In the experiment, 0.5% Gaussian noise is added to the measurements. In Figure 5.5, the number of nonzero groups is set to be 15, and the parameter μ is set to be 5×10^{-3} and 1×10^{-3}. In Figure 5.6, the group sparsity level is increased to 25 and the parameter μ is set to be 1×10^{-3}.

From Figure 5.5, we can see that the ADM algorithms decrease the relative errors to the lowest level within 50 iterations, much faster than SLEP and SpaRSA. It is worth noting that the performance of SLEP and SpaRSA is significantly affected by the value of μ. As we can see, the smaller value $\mu = 10^{-3}$ yields better recovery quality. However, as μ decreases, the convergence of these two algorithms becomes much slower, especially for SpaRSA. For small values of μ, continuation or other heuristic techniques may be needed to speed up these two algorithms. However, the value of μ almost has no impact on the the performance of the ADM algorithms.

In Figure 5.6, the speed advantage of the ADM algorithms becomes more significant as the group sparsity level increases from 15 to 25. While the performance of

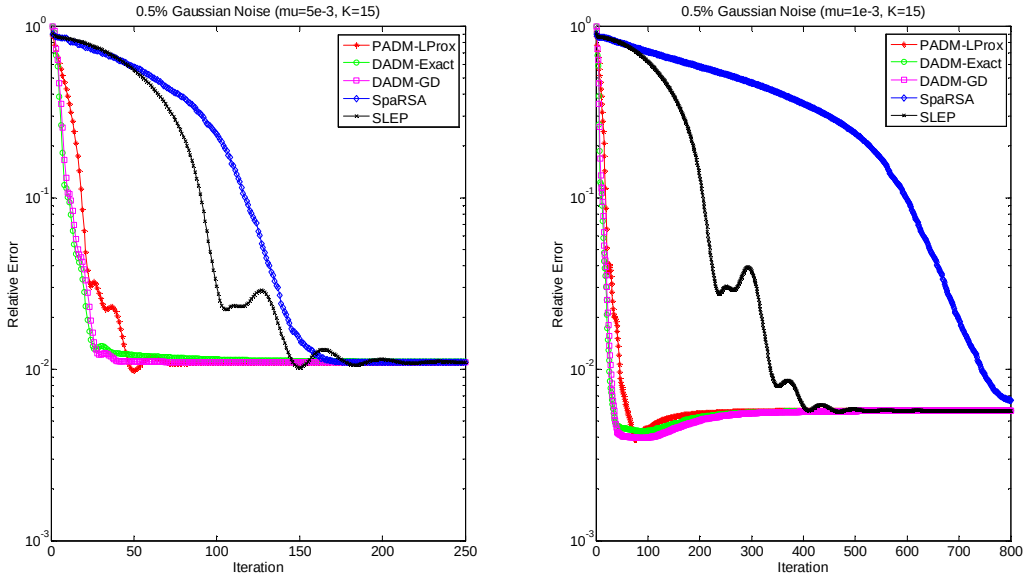


Figure 5.5 : BP_μ model with 0.5% Gaussian noise: comparison of the ADM algorithms, SpaRSA and SLEP on the decreasing of recovery errors over iterations. Parameter μ is set to be 5×10^{-3} (left) and 1×10^{-3} (right), and group sparsity is $K = 15$.

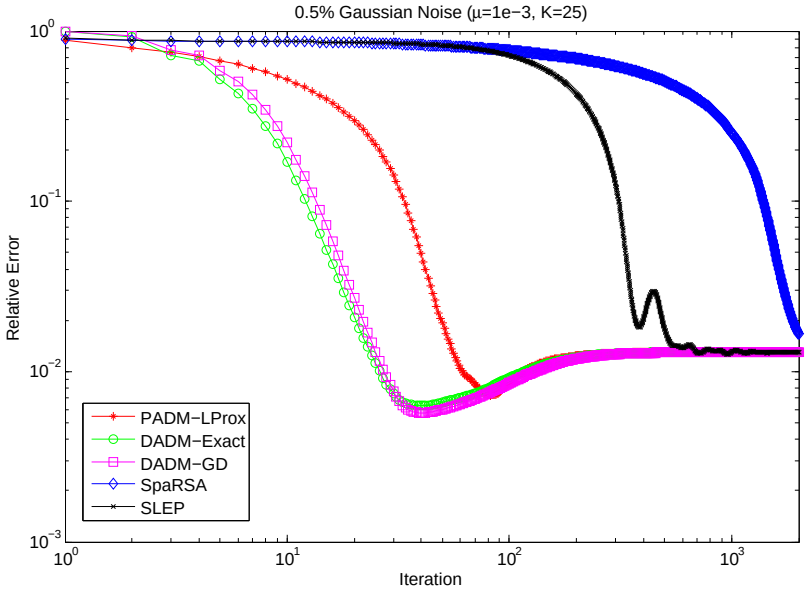


Figure 5.6 : BP_μ model with 0.5% Gaussian noise: comparison of the ADM algorithms, SpaRSA and SLEP on the decreasing of recovery errors over iterations. Parameter μ is set to be 1×10^{-3} and group sparsity is $K = 25$.

the ADM algorithms almost remain the same, SLEP and SpaRSA take substantially more iterations to converge. In addition, the solution paths of the ADM algorithms also suggest that better recovery quality may be attained when stopped properly prior to convergence.

5.3.4 On Other Types of Signals

In the previous experiments, we tested on random Gaussian group-sparse signals and have shown that our ADM algorithms outperform the other compared algorithms. In this section, we test on two other types of signals: one is random Bernoulli group-spares signals with ± 1 nonzero entries, and the other is power-law decaying group-sparse signals. While the Bernoulli signals have zero decay, the power-law decaying signals are fast decaying signals whose (sorted) nonzero entries are $\pm i^{-1/\lambda}$ $(i = 1, 2, \ldots, k)$ for some $\lambda \in (0, 1)$. In fact, the algorithms may have different performances on different types of signals.

For the Bernoulli signals, the performances of all the compared algorithms are similar to our previous results on Gaussian signals, and the results are thus omitted. For the power-law decaying signals, we find that the ADM algorithms still exhibit a clear speed advantage over SLEP and SpaRSA, but their speed advantage over SPGL1 begins to diminish as the signal decaying rate becomes faster, i.e., λ becomes smaller. For small λ, the ADM algorithms can no longer outperform SPGL1 under the current parameter setting.

To accelerate the ADM algorithms, we apply a continuation scheme on the penalty parameter β following the rule in [7]. The basic idea is that we increase the penalty parameter if the constraint violation does not decrease much during the iterations.

For example, for PADM-Exact, we perform continuation as follows:

$$\beta^{k+1} = \begin{cases} \eta\beta^k, & \text{if } \|r_1^{k+1}\| \geq \alpha\|r_1^k\|, \text{ and } \|r_2^{k+1}\| \geq \alpha\|r_2^k\|; \\ \beta^k, & \text{otherwise,} \end{cases} \tag{5.1}$$

where $\beta = [\beta_1,\ \beta_2]$ is the penalty parameter, $r_1 = x - z$ and $r_2 = Ax - b$ are the constraint violations, $0 < \alpha < 1$ and $\eta > 1$ are some constant parameters. Similar continuation schemes can be applied to the other variants of ADM algorithms.

Figure 5.7 shows the comparison result with SPGL1 on recovering power-law decaying signals ($\lambda = 0.6$) by solving the BP model with noiseless data, and Figure 5.8 shows the comparison result when the data is contaminated by 0.5% Gaussian noise.

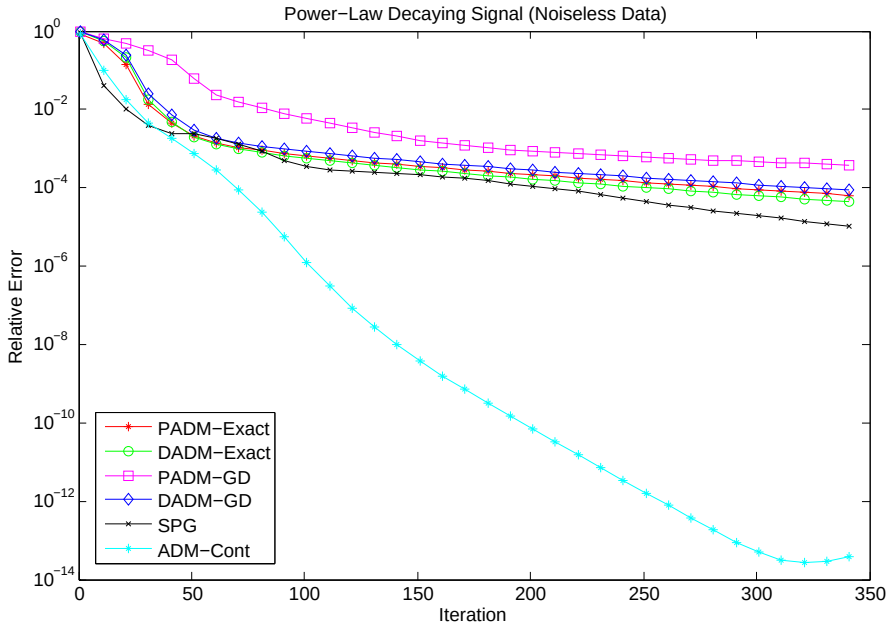


Figure 5.7 : BP model with power-law decaying signals and noiseless data: comparison of the ADM algorithms and SPGL1 on the decreasing of recovery errors over iterations, where ADM-Cont applies continuation on the penalty parameters to PADM-Exact.

Among the compared algorithms, *ADM-Cont* represents the one that applies the above continuation scheme (5.1) to PADM-Exact, where we used the default penalty

parameters as the initial values of β and set the continuation parameters $\alpha = 0.9$ and $\eta = 1.2$.

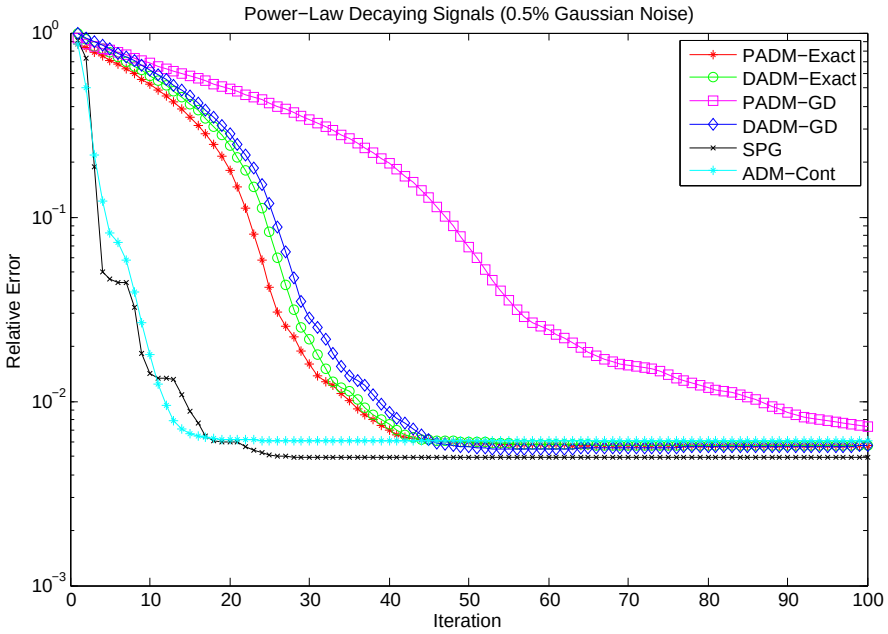


Figure 5.8 : BP model with power-law decaying signals and 0.5% Gaussian noise: comparison of the ADM algorithms and SPGL1 on the decreasing of recovery errors over iterations, where ADM-Cont applies continuation on the penalty parameters to PADM-Exact.

As shown in the figures, the ADM algorithms using fixed penalty parameters can no longer outperform SPGL1. Especially when data contains noise, SPGL1 decreases the relative error very fast. However, we can see that using continuation on the penalty parameters significantly speed up the convergence of the ADM algorithms. On noiseless data, ADM-Cont decreases the relative error much faster than the other compared algorithms, reaching machine precision with substantially fewer number of iterations. On noisy data, ADM-Cont converges with no more than 20 iterations, which is comparable with SPGL1.

In conclusion, using continuation to update the penalty parameters has shown to be an effective way to accelerate the ADM algorithms. However, it is theoretically not clear whether these continuation schemes can guarantee the convergence of the algorithms. How to develop an effective and robust way to adaptively adjust the penalty parameters remains to be further investigated.

Chapter 6

Conclusions

In this thesis, we have developed efficient algorithms for solving a variety of $\ell_{2,1}$-regularized optimization problems with applications in recovering data with group sparsity. The proposed algorithms are based on a variable splitting strategy and the alternating direction methodology, coupled with the linear proximal and one-step gradient descent approaches to solve some subproblems approximately. The per-iteration computational cost of our algorithms is reasonably low, which is roughly two matrix-vector multiplications. The proposed algorithms are efficient first-order methods and are suitable for large-scale computation. In addition, several important extensions of the algorithms have been made to enforce nonnegativity in the data and allow arbitrary grouping structures such as overlapping groups.

The global convergence of our algorithms are guaranteed by the existing ADM theory under certain parameter restrictions. We also extend the convergence theory to allow more generality on the choices of γ, the step-length for updating the Lagrangian multipliers.

We have carried out various numerical experiments on synthetic data to justify the benefit of group sparsity and demonstrate the efficiency, stability and robustness of the proposed algorithms. In particular, our algorithms exhibit a clear and significant speed advantage over the state-of-the-art solvers SPGL1, SLEP and SpaRSA on recovering group-sparse signals of either Gaussian or Bernoulli type. For power-law decaying signals, our algorithms still outperform SLEP and SpaRSA, but are

not necessarily better than SPGL1 under the default parameter setting. However, our algorithms can be well accelerated by applying continuation techniques on the penalty parameters, thereby achieving competitive or even better performance against SPGL1. Moreover, it has been observed that at least on random problems our algorithms are capable of achieving a higher solution quality than the other compared algorithms can, when data contains noise. More comprehensive numerical experiments and applications of the algorithms on real data will be conducted in the future.

Chapter 7

Future Work

In this thesis, we have applied the alternating direction methodology to the group sparse optimization problems and obtain outstandingly efficient algorithms. In fact, the alternating direction methods have proven to be effective for solving a wide range of optimization problems, such as the ℓ_1-regularized problems [30], total variation (TV) problems [27, 29] and matrix completion problems [5]. However, there are still many open questions in the ADM theory that are worth further investigation.

7.1 Convergence Rate of Alternating Direction Methods

The convergence rate of alternating direction methods has not been well established in the literature. Until very recently, He and Yuan [15] proved $O\left(\frac{1}{k}\right)$ convergence rate for problem (2.5). However, empirical evidence leads us to believe that better convergence rate can be achieved under certain conditions. For example, our numerical results in Section 5.3 show that the convergence rate of the ADM algorithms is almost linear. Therefore, we are interested to establish a linear convergence rate for the alternating direction methods.

7.1.1 Preliminary Result

We consider the following generalized alternating direction method (Algorithm 9) for solving (2.5), where $\hat{P}$ and $\hat{Q}$ are in general positive semidefinite matrices.

Algorithm 9: Generalized Alternating Direction Method

Initialize $x^0,\ \lambda^0,\ \beta > 0, \gamma > 0$;

for $k = 0,\ 1, \ldots$ **do**

$z^{k+1} = \arg\min_{z \in \mathcal{Z}} \mathcal{L}_{\mathcal{A}}(x^k, z, \lambda^k) + \frac{1}{2}\|z - z^k\|^2_{\hat{Q}}$;

$x^{k+1} = \arg\min_{x \in \mathcal{X}} \mathcal{L}_{\mathcal{A}}(x, z^{k+1}, \lambda^k) + \frac{1}{2}\|x - x^k\|^2_{\hat{P}}$;

$\lambda^{k+1} = \lambda^k - \gamma\beta(Ax^{k+1} + Bz^{k+1} - c)$;

This framework has been studied in [13]. It generalizes many different variants of the alternating direction methods. For example,

- $\hat{P} = O$ and $\hat{Q} = O$ gives the classic alternating direction method;
- it is easy to see that $\hat{P} = \frac{\beta}{\tau} I - \beta A^T A$ (or $\hat{Q} := \frac{\beta}{\tau} I - \beta B^T B$) corresponds to the linear proximal method for solving the x-subproblem (or z-subproblem);
- When $\hat{P} = \frac{1}{\alpha} I - H_f - \beta A^T A$, it reduces to applying one-step projected gradient descent to the x-subproblem for a quadratic function f, where $H_f := \nabla^2 f(x) \succeq 0$ and $\alpha > 0$ is a constant step length. It is similar for $\hat{Q}$.

For convenience, we assume both functions f and g are differentiable and $\gamma = 1$. Let

$$u := \begin{pmatrix} x \\ z \\ \lambda \end{pmatrix},\ G := \begin{pmatrix} P & & \\ & Q & \\ & & \frac{1}{\beta} I_p \end{pmatrix}, \tag{7.1}$$

and $P = \hat{P} + \beta A^T A \succeq 0$, $Q = \hat{Q} \succeq 0$. Our preliminary analysis has established the following theorem.

Theorem 7.1. *Suppose*

- $\mathcal{X} = \mathbb{R}^n$;
- *matrix A has full row rank, and B has full column rank;*
- *f is strongly convex: for some $\nu_f > 0$,*

$$\langle \nabla f(x_1) - \nabla f(x_2), x_1 - x_2 \rangle \geq \nu_f \|x_1 - x_2\|^2, \quad \forall x_1, x_2 \in \mathcal{X}; \tag{7.2}$$

- *∇f is Lipschitz continuous: for some $L > 0$,*

$$\|\nabla f(x_1) - \nabla f(x_2)\| \leq L\|x_1 - x_2\|, \quad \forall x_1, x_2 \in \mathcal{X}; \tag{7.3}$$

- $\gamma = 1$;
- $P = \beta A^T A$ *or* $P \succ \beta A^T A$;

there exists $\delta > 0$, such that

$$\|u^k - u^*\|_G^2 \geq (1 + \delta)\|u^{k+1} - u^*\|_G^2. \tag{7.4}$$

Remark 7.1. *If $P \succ 0$ and $Q \succ 0$, Theorem 7.1 indicates that $\{u^k\}$ has a global Q-linear convergence rate, which implies that $\{x^k\}$, $\{y^k\}$ and $\{\lambda^k\}$ converge at least R-linearly.*

Remark 7.2. *In fact, the constant δ can be explicitly derived as follows.*

(i) For $P = \beta A^T A$:

$$\delta := \max\left\{ \min\left(\frac{2\nu_f}{\theta_1},\ \frac{\lambda_{\min}(P) + 2\beta\nu_f}{\theta_2},\ \frac{1}{\beta\theta_3} \right) : \mu_1 > 1, \mu_2 > 0 \right\}. \tag{7.5}$$

(ii) For $P \succ \beta A^T A$:

$$\delta := \max\left\{ \min\left(\frac{2\nu_f}{\theta_1},\ \frac{\lambda_{\min}\left(P - \frac{1}{\rho}A^T A\right)}{\theta_2},\ \frac{\frac{1}{\beta} - \rho}{\theta_3} \right) : \mu_1 > 1,\ \mu_2 > 0,\ \rho > 0 \right\}. \tag{7.6}$$

Here

$$\theta_1 = \|P\| + \frac{\mu_1 L^2}{\beta(\mu_1 - 1)\lambda_{\min}(AA^T)} + \frac{(\mu_2 + 1)\|Q\|\|A\|^2}{\mu_2 \lambda_{\min}(B^T B)}, \tag{7.7}$$

$$\theta_2 = \frac{\mu_1 \|P - \beta A^T A\|^2}{\beta \lambda_{\min}(AA^T)}, \tag{7.8}$$

$$\theta_3 = \frac{(1 + \mu_2)\|Q\|}{\beta^2 \lambda_{\min}(B^T B)}. \tag{7.9}$$

7.1.2 On the Exact ADM Scheme

As a special case, we consider the classic alternating direction method (Algorithm 1) in which both subproblems are solved exactly. In this case, $P = \beta A^T A,\ Q = O$. It follows that $\theta_2 = \theta_3 = 0$ and θ_1 is maximized when $\mu_1 \to \infty$. Therefore, we have

$$\delta = 2\nu_f \Big/ \left(\beta \|A\|^2 + \frac{L^2}{\beta \lambda_{\min}(AA^T)}\right). \tag{7.10}$$

We can choose β to be

$$\beta = \frac{L}{\|A\|\sqrt{\lambda_{\min}(AA^T)}}, \tag{7.11}$$

and get the largest δ:

$$\delta_{\max} = \frac{\nu_f}{L\sqrt{\kappa(AA^T)}}, \tag{7.12}$$

where $\kappa(AA^T) := \frac{\lambda_{\max}(AA^T)}{\lambda_{\min}(AA^T)}$ is the condition number of AA^T. We can see that the linear convergence rate is determined by the strongly convexity constant ν_f, the Lipschitz constant L and the condition number $\kappa(AA^T)$.

Since P is not necessarily positive definite and $Q = O$, Theorem 7.1 does not imply the Q-linear convergence of $\{u^k\}$. But it indicates that $\left\{(Ax^k; \lambda^k)\right\}$ converges Q-linearly, so $\{Ax^k\}$ and $\{\lambda^k\}$ converge at least R-linearly. When the matrix B has full column rank or the function g is also strongly convex, we can show that $\{z^k\}$ also converges R-linearly. Furthermore, we can derive Q-linear convergence rates for $\{x^k\}$ and $\{\lambda^k\}$ under certain conditions in the following Theorem.

Theorem 7.2. *When both subproblems are solved exactly, under the assumptions of Theorem 7.1, we have*

$$\|x^{k+1} - x^*\|_2^2 \leq \frac{\sigma_1}{1+\delta}\|x^k - x^*\|_2^2, \tag{7.13}$$

$$\|\lambda^{k+1} - \lambda^*\|_2^2 \leq \frac{\sigma_2}{1+\delta}\|\lambda^k - \lambda^*\|_2^2, \tag{7.14}$$

where

$$\sigma_1 := \left(\frac{\beta^2\|A\|^4}{\nu_f^2} + \frac{L^2\|A\|^2}{\nu_f^2\lambda_{\min}(AA^T)}\right) \Bigg/ \left(\frac{\beta^2\|A\|^2\lambda_{\min}(A^TA)}{\nu_f^2} + 1\right), \tag{7.15}$$

$$\sigma_2 := \left(\frac{\beta^2\|A\|^4}{\nu_f^2} + 1\right) \Bigg/ \left(\frac{\beta^2\lambda_{\min}(AA^T)\lambda_{\min}(A^TA)}{L^2} + 1\right). \tag{7.16}$$

Remark 7.3. *When $\frac{\sigma_1}{1+\delta} < 1$ and $\frac{\sigma_2}{1+\delta} < 1$, Theorem 7.2 indicates that $\{x^k\}$ and $\{\lambda^k\}$ converge Q-linearly.*

It is easy to show that there always exists $\bar{\beta} > 0$, such that $\frac{\sigma_2}{1+\delta} < 1$ for $\beta \in (0, \bar{\beta})$. There may not exist $\beta > 0$ such that $\frac{\sigma_1}{1+\delta} < 1$ in general. But in many cases, we do have $\frac{\sigma_1}{1+\delta} < 1$. For example, if $\lambda_{\min}(A^TA) = \lambda_{\max}(A^TA) = 1$, then $\frac{\sigma_1}{1+\delta} < 1$ holds as long as β is big enough. If additionally $\nu = L$, then $\frac{\sigma_1}{1+\delta} < 1$ holds for any $\beta > 0$.

7.2 Discussions

Our preliminary analysis establishes the global linear convergence rate of the alternating direction methods. It is still an ongoing work to extend our analysis to allow more generality and further improve the linear rate. For example, it may be possible to relax those assumptions that $\mathcal{X} = \mathbb{R}^n$, $\gamma = 1$, the full row rankness of A and full column rankness of B, as well as the differentiability of functions f and g.

In addition, the matrices $\hat{P}$ and $\hat{Q}$ in the added proximal terms of Algorithm 9 may vary over iterations. For instance, the linear proximal parameter τ and the

step-length α of the one-step gradient descent method can be chosen differently at each iteration, which may lead to faster convergence of the algorithms. Therefore, it is important to extend our framework to cover such situations.

Moreover, how to choose the penalty parameter β is always an important issue, which can largely affect the practical performance of the alternating direction methods. However, there is lack of theoretical guidance on the choice of β. Nowadays, it is mostly chosen based on empirical experience, or by some heuristic techniques such as continuation. Our convergence rate analysis can possibly give more insights about how the penalty parameter β affects the convergence speed, thereby providing some theoretical guidance for choosing β. Our preliminary analysis assumes β is a fixed parameter. Alternatively, a variable sequence of penalty parameters $\{\beta^k\}$ can be used and has shown to be effective in practice. It is worthwhile to extend our analysis to study the alternating direction methods with variable penalty parameters. There arise several questions: how to develop an adaptive way to adjust the penalty parameters based on the iterate information? Is it possible to obtain a superlinear convergence rate for a properly chosen sequence $\{\beta^k\}$, where β^k eventually goes to $+\infty$?

As discussed above, there are still many open questions in the theory and applications of alternating direction methods. It is of great importance to devote further efforts to a better understanding and more thorough analysis of the alternating direction methods.

Bibliography

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

[2] R. Baraniuk, V. Cevher, M. Duarte, and C. Hegde, *Model-based compressive sensing*, Information Theory, IEEE Transactions on, 56 (2010), pp. 1982–2001.

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

[4] A. Beck and M. Teboulle, *A fast iterative shrinkage-thresholding algorithm for linear inverse problems*, SIAM Journal on Imaging Sciences, 2 (2009), pp. 183–202.

[5] E. Candès and B. Recht, *Exact matrix completion via convex optimization*, Foundations of Computational Mathematics, 9 (2009), pp. 717–772.

[6] E. Candes, J. Romberg, and T. Tao, *Robust uncertainty principles: Exact signal reconstruction from highly incomplete frequency information*, Information Theory, IEEE Transactions on, 52 (2006), pp. 489–509.

[7] S. Chan, R. Khoshabeh, K. Gibson, P. Gill, and T. Nguyen, *An augmented lagrangian method for total variation video restoration*, Image Processing, IEEE Transactions on, (2011), pp. 1–1.

[8] D. Donoho, *Compressed sensing*, Information Theory, IEEE Transactions on, 52 (2006), pp. 1289–1306.

[9] D. Gabay and B. Mercier, *A dual algorithm for the solution of nonlinear variational problems via finite element approximation*, Computers & Mathematics with Applications, 2 (1976), pp. 17–40.

[10] R. Glowinski, *Numerical methods for nonlinear variational problems*, (1984).

[11] R. Glowinski and A. Marrocco, *Sur l'approximation, par éléments finis d'ordre un, et la résolution, par pénalisation-dualité, d'une classe de problèmes de Dirichlet non linéaires*, Laboria, 1975.

[12] E. Hale, W. Yin, and Y. Zhang, *Fixed-point continuation for ℓ_1-minimization: Methodology and convergence*, SIAM Journal on Optimization, 19 (2008), p. 1107.

[13] B. He, L. Liao, D. Han, and H. Yang, *A new inexact alternating directions method for monotone variational inequalities*, Mathematical Programming, 92 (2002), pp. 103–118.

[14] B. He and H. Yang, *Some convergence properties of a method of multipliers for linearly constrained monotone variational inequalities*, Operations research letters, 23 (1998), pp. 151–161.

[15] B. He and X. Yuan, *On non-ergodic convergence rate of douglas-rachford alternating direction method of multipliers*, (2012).

[16] J. Huang, T. Zhang, and D. Metaxas, *Learning with structured sparsity*, in Proceedings of the 26th Annual International Conference on Machine Learning,

ACM, 2009, pp. 417–424.

[17] 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, ACM, 2009, pp. 433–440.

[18] H. Krim and M. Viberg, *Two decades of array signal processing research: the parametric approach*, Signal Processing Magazine, IEEE, 13 (2002), pp. 67–94.

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

[20] S. Ma, X. Song, and J. Huang, *Supervised group Lasso with applications to microarray data analysis*, BMC bioinformatics, 8 (2007), p. 60.

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

[22] K. Pruessmann, M. Weiger, M. Scheidegger, and P. Boesiger, *SENSE: sensitivity encoding for fast MRI*, Magnetic Resonance in Medicine, 42 (1999), pp. 952–962.

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

[24] J. Tropp and A. Gilbert, *Signal recovery from random measurements via orthogonal matching pursuit*, Information Theory, IEEE Transactions on, 53 (2007), pp. 4655–4666.

[25] E. van den Berg and M. P. Friedlander, *Probing the pareto frontier for basis pursuit solutions*, SIAM Journal on Scientific Computing, 31 (2008),

pp. 890–912.

[26] 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).

[27] Y. Wang, J. Yang, W. Yin, and Y. Zhang, *A new alternating minimization algorithm for total variation image reconstruction*, SIAM Journal on Imaging Sciences, 1 (2008), pp. 248–272.

[28] S. Wright, R. Nowak, and M. Figueiredo, *Sparse reconstruction by separable approximation*, Signal Processing, IEEE Transactions on, 57 (2009), pp. 2479–2493.

[29] J. Yang, W. Yin, Y. Zhang, and Y. Wang, *A fast algorithm for edge-preserving variational multichannel image restoration*, SIAM Journal on Imaging Sciences, 2 (2009), pp. 569–592.

[30] J. Yang and Y. Zhang, *Alternating direction algorithms for ℓ_1-problems in compressive sensing*, SIAM journal on scientific computing, 33 (2011), pp. 250–278.

[31] M. Yuan and Y. Lin, *Model selection and estimation in regression with grouped variables*, Journal of the Royal Statistical Society: Series B (Statistical Methodology), 68 (2006), pp. 49–67.