

A Block Successive Upper Bound Minimization Method of Multipliers for Linearly Constrained Convex Optimization

Mingyi Hong*, Tsung-Hui Chang[†] Xiangfeng Wang[‡]
Meisam Razaviyayn*, Shiqian Ma[§] Zhi-Quan Luo*

January 27, 2014

Abstract

Consider the problem of minimizing the sum of a smooth convex function and a separable nonsmooth convex function subject to linear coupling constraints. Problems of this form arise in many contemporary applications including signal processing, wireless networking and smart grid provisioning. Motivated by the huge size of these applications, we propose a new class of first order primal-dual algorithms called the *block successive upper-bound minimization method of multipliers* (BSUM-M) to solve this family of problems. The BSUM-M updates the primal variable blocks successively by minimizing locally tight upper-bounds of the augmented Lagrangian of the original problem, followed by a gradient type update for the dual variable in closed form. We show that under certain regularity conditions, and when the primal block variables are updated in either a deterministic or a random fashion, the BSUM-M converges to the set of optimal solutions. Moreover, in the absence of linear constraints, we show that the BSUM-M, which reduces to the *block successive upper-bound minimization* (BSUM [1]) method, is capable of linear convergence without strong convexity.

KEY WORDS: Block successive upper-bound minimization, alternating direction method of multipliers, randomized block coordinate descent.

AMS(MOS) Subject Classifications: 49, 90.

*University of Minnesota, Minneapolis, USA, email: {mhong, meisam, luozq}@umn.edu.

[†]National Taiwan University of Science and Technology, Taiwan, R.O.C., email: tsunghui.chang@ieee.org

[‡]Nanjing University, Nanjing, P.R. China, email: wangxf@mail.nju.edu.cn

[§]The Chinese University of Hong Kong, Hong Kong, P.R. China, email: sqma@se.cuhk.edu.hk

[¶]The research of M. Hong, M. Razaviyayn and Z.-Q. Luo is supported in part by the AFOSR, grant number FA9550-12-1-0340. The research of S. Ma is supported by the Hong Kong Research Grants Council (RGC) Early Career Scheme (ECS) (Project ID: CUHK 439513). T.-H. Chang is supported by the National Science Council, Taiwan (R.O.C.), under grant NSC 102-2221-E-011-005-MY3.

1 Introduction

Consider the problem of minimizing a convex function $f(x)$ subject to linear equality constraints:

$$\begin{aligned} & \text{minimize} && f(x) := g(x_1, \dots, x_K) + \sum_{k=1}^K h_k(x_k) \\ & \text{subject to} && E_1 x_1 + E_2 x_2 + \dots + E_K x_K = q, \\ & && x_k \in X_k, \quad k = 1, 2, \dots, K, \end{aligned} \tag{1.1}$$

where $g(\cdot)$ is a smooth convex function; h_k is a nonsmooth convex function; $x = (x_1^T, \dots, x_K^T)^T \in \mathfrak{R}^n$ is a partition of the optimization variable x , $x_k \in \mathfrak{R}^{n_k}$; $X = \prod_{k=1}^K X_k$ is the feasible set for x ; $q \in \mathfrak{R}^m$ is a vector. Let $E := (E_1, \dots, E_K)$ and $h(x) := \sum_{k=1}^K h_k(x_k)$. Many contemporary problems in signal processing, machine learning and smart grid systems can be formulated in the form (1.1). To motivate our work, we discuss several examples of the form (1.1) below.

1.1 Motivating Examples

The first example is the basis pursuit (BP) problem which solves the following nonsmooth problem

$$\min_x \|x\|_1 \quad \text{s.t.} \quad Ex = q, \quad x \in X. \tag{1.2}$$

One important application of this model is in compressive sensing, where a sparse signal (say an image) x needs to be recovered using a small number of observations q (i.e., $m \ll n$) [2]. Let us partition the signal vector as $x = [x_1^T, \dots, x_K^T]^T$ where $x_k \in \mathfrak{R}^{n_k}$, and partition E and X accordingly. Then the BP problem can be written in the form of (1.1)

$$\min_x \sum_{k=1}^K \|x_k\|_1 \quad \text{s.t.} \quad \sum_{k=1}^K E_k x_k = q, \quad x_k \in X_k, \quad \forall k. \tag{1.3}$$

The second example has to do with the control of a smart grid system. Consider a power grid system in which a utility company buys power from an electricity market to serve a neighborhood with K customers. The total cost for the utility includes the cost of purchasing the electricity from a day-ahead wholesale market and a real-time market. In the envisioned smart grid system, the utility will have the ability to control the power consumption of some appliances (e.g., controlling the charging rate of electrical vehicles) in a way to minimize its total cost. This problem, known as the demand response (DR) control problem, is central to the success of the smart grid system [3–5].

To formulate this problem, let us divide each day into L periods and let p_ℓ denote the amount of power the utility company bids for the ℓ -th period from a day-ahead market, $\ell = 1, \dots, L$. Let $\Psi_k \mathbf{x}_k$ denote the load profile of a customer $k = 1, \dots, K$, where $\mathbf{x}_k \in \mathbb{R}^{n_k}$ contains some control

variables for the equipments of customer k , and $\Psi_k \in \mathbb{R}^{L \times n_k}$ contains the information related to the appliance load model [6]. The retailer aims at minimizing the bidding cost as well as the cost incurred by power imbalance in the next day [3–5]

$$\begin{aligned} \min \quad & C_p \left(\left(\sum_{k=1}^K \Psi_k \mathbf{x}_k - \mathbf{p} \right)^+ \right) + C_s \left(\left(\mathbf{p} - \sum_{k=1}^K \Psi_k \mathbf{x}_k \right)^+ \right) + C_d(\mathbf{p}) \\ \text{s.t.} \quad & \mathbf{x}_k \in X_k, \quad k = 1, \dots, K, \quad \mathbf{x} \geq 0, \quad \mathbf{p} \geq 0 \end{aligned} \quad (1.4)$$

where $C_p(\cdot)$ and $C_s(\cdot)$ are increasing functions which model the cost incurred by insufficient and excessive power bids, respectively; $C_d(\cdot)$ represents the bidding cost function; $(x)^+ := \max\{x, 0\}$; X_k is some compact set; see [4]. Upon introducing a new variable $\mathbf{z} = \left(\sum_{k=1}^K \Psi_k \mathbf{x}_k - \mathbf{p} \right)^+$, the above problem can be equivalently transformed into the form of (1.1):

$$\begin{aligned} \min \quad & C_p(\mathbf{z}) + C_s(\mathbf{z} + \mathbf{p} - \sum_{k=1}^K \Psi_k \mathbf{x}_k) + C_d(\mathbf{p}) \\ \text{s.t.} \quad & \sum_{k=1}^K \Psi_k \mathbf{x}_k - \mathbf{p} - \mathbf{z} \leq 0, \quad \mathbf{z} \geq 0, \quad \mathbf{p} \geq 0, \quad \mathbf{x}_k \in X_k, \quad \forall k. \end{aligned} \quad (1.5)$$

The third example is related to the optimization of the so-called cognitive radio network (CRN) [7,8]. The CRN is an emerging wireless communication technology that promises significant improvement in radio spectrum utilization. The basic idea is to allow secondary (unlicensed) users to opportunistically access the spectrum that is not used by primary (licensed) users. Specifically, consider a network with K secondary users (SUs) and a single secondary base station (SBS) operating on M parallel frequency tones. The SUs are interested in transmitting their messages to the SBS. Let s_k^m denote user k 's transmit power on m th channel; let h_k^m denote the channel between user k and the SBS on tone m ; let P_k denote SU k 's total power budget. Also suppose that there are L primary users (PUs) in the system, and let $g_{k\ell}^m$ denote the channel between the k th SU to the ℓ th PU. The goal of the secondary network is to maximize the system throughput, subject to the requirement that certain interference temperature (IT) constraints measured at the receivers of the PUs are not violated [9,10]:

$$\begin{aligned} \max \quad & \sum_{m=1}^M \log \left(1 + \sum_{k=1}^K |h_k^m|^2 s_k^m \right) \\ \text{s.t.} \quad & s_k^m \geq 0, \quad \sum_{m=1}^M s_k^m \leq P_k, \quad \sum_{k=1}^K |g_{k\ell}^m|^2 s_k^m \leq I_\ell^m, \quad \forall \ell, k, m. \end{aligned} \quad (1.6)$$

In the objective, the term $\log \left(1 + \sum_{k=1}^K |h_k^m|^2 s_k^m \right)$ represents the sum-rate that all the users can jointly achieve on frequency tone m ; $I_\ell^m \geq 0$ denotes the IT threshold for PU ℓ on tone m . Clearly this problem is also in the form of (1.1).

1.2 Literature Review

When the linear coupling constraint is not present, a well known technique for solving (1.1) is to use the so-called block coordinate descent (BCD) method whereby, at every iteration, a single block of variables is optimized while the remaining blocks are held fixed. More specifically, at iteration r , the blocks are updated in a Gauss-Seidel fashion by

$$x_k^r \in \arg \min_{x_k \in \mathcal{X}_k} g(x_1^r, \dots, x_{k-1}^r, x_k, x_{k+1}^{r-1}, \dots, x_K^{r-1}) + h_k(x_k), \quad k = 1, \dots, K. \quad (1.7)$$

Since each step involves solving a simple subproblem of small size, the BCD method can be quite effective for solving large-scale problems; see e.g., [1, 11–14] and the references therein. The existing analysis of the BCD method [15–18] requires the uniqueness of the minimizer for each subproblem (1.7), or the quasi convexity of f [19]. When problem (1.7) is not easily solvable, a popular approach is to solve an approximate version of problem (1.7), yielding the block coordinate gradient decent (BCGD) algorithm (or the block coordinate proximal gradient algorithm in the presence of nonsmooth function h) [13, 20–22]. The global rate of convergence for BCD-type algorithm has been studied extensively. When the objective function is strongly convex, the BCD algorithm converges globally linearly [23]. When the objective function is smooth and not strongly convex, Luo and Tseng have shown that the BCD method and many of its variants can still converge linearly, provided that a certain local error bound condition is satisfied around the solution set [23–26]. This line of analysis has recently been extended to allow a certain class of nonsmooth functions in the objective [21, 27–29]. There are a few recent works characterizing the global sublinear convergence rate for the BCD-type algorithms [14, 22, 30, 31]. In particular, reference [30] shows that the BCD with Gauss-Seidel update rule converges sublinearly at the order of $\mathcal{O}(\frac{1}{r})$ for a large family of nonsmooth convex problems. Furthermore, a unified algorithmic framework called BSUM (block successive upper-bound minimization) and its convergence analysis is proposed in [1] whereby at each step a locally tight upper-bound of the objective function is minimized successively to update the variable blocks.

When the linear coupling constraint is present, it is well known that the BCD-type algorithm may fail to find any (local) optimal solution [32]. A popular algorithm for solving this type of problem is the so-called alternating direction method of multipliers (ADMM) [33–36]. In the ADMM method, instead of maintaining feasibility all the time, the constraint $Ex = q$ is dualized using the Lagrange multiplier y and a quadratic penalty term is added. The resulting *augmented Lagrangian function* is of the form:

$$L(x; y) = f(x) + \langle y, q - Ex \rangle + \frac{\rho}{2} \|q - Ex\|^2, \quad (1.8)$$

where $\rho > 0$ is a constant and $\langle \cdot, \cdot \rangle$ denotes the inner product operator. The ADMM method updates the primal block variables x_1, \dots, x_n by using a BCD type procedure to minimize $L(x; y)$.

The latter often leads to simple subproblems with closed form solutions. These primal updates are then followed by a gradient ascent update of the dual variable y .

Although the ADMM algorithm was introduced as early as in 1976 by Gabay, Mercier, Glowinski and Marrocco [35,37], it has become popular only recently due to its applications in modern large scale optimization problems arising from machine learning and computer vision [33, 38–42]. In practice, the algorithm is often computationally very efficient and exhibits much faster convergence than traditional algorithms such as the dual ascent algorithm [43–45] or the method of multipliers [46]. The convergence of ADMM has been established under the condition that the objective is separable and there are only two block variables, i.e., $g(x_1, \dots, x_K) = g_1(x_1) + \dots + g_K(x_K)$, and $K = 2$ [35,37]. For large scale problems such as those arising from compressive sensing, the optimal solution for the primal per-block subproblems may not be easily computable [47]. In these cases the classical ADMM can be modified to the one that performs a simple proximal gradient step for each subproblem [34, 40, 47–50]. When there are only two block variables, several recent works [51, 52] have shown that the ADMM method converges at a rate of $\mathcal{O}(\frac{1}{r})$ (and $\mathcal{O}(\frac{1}{r^2})$ for the accelerated version [53]). Moreover, references [53–55] have shown that the ADMM converges linearly when the objective function is strongly convex and there are only two blocks of variables. A recent study [56] has shown the global (linear) convergence of the ADMM for the case of $K \geq 3$ under the assumptions that: *a)* for each k , E_k is full column rank; *b)* the dual stepsize is sufficiently small; *c)* a certain error bound holds around the optimal solution set; and *d)* the objective is separable. If these conditions are not satisfied and when $K \geq 3$, it is shown in [57] that the ADMM can indeed diverge in general. Some other recent works have attempted to modify the original ADMM for $K \geq 3$ case [58–60].

Unfortunately, neither BCD nor ADMM can be used to solve problem (1.1). In fact, due to its multi-block structure as well as the variable coupling in *both* the objective and the constraints, this problem cannot be handled by many other methods for big data including SpARSA [61], FPC-BB [62], FISTA [63], ALM [64], HOGWILD [65], FPA [66]. The main contribution of this paper is to propose and analyze a novel block successive upper bound minimization method of multipliers (BSUM-M) and its randomized version, that can solve problem (1.1) efficiently. The BSUM-M algorithm integrates the BSUM and ADMM algorithm in a way that optimizes an *approximate* augmented Lagrangian of the original problem one block variable each time, and then updates the dual variable by using a gradient ascent step. The resulting algorithm is flexible because we can choose suitable approximations of the augmented Lagrangian function that allow convenient updates of the primal variable blocks (say in closed form). In the absence of linear coupling constraints, the randomized BSUM-M algorithm reduces to the randomized BCD algorithm. In this case, we show that the randomized BCD algorithm in fact converges linearly (in expectation) for a family of problems without strongly convex objectives. To the best of our knowledge, this is the first result that shows the linear rate of convergence for the randomized BCD algorithm in the

absence of strong convexity.

1.3 The BSUM-M Algorithm

Define $E := (E_1, E_2, \dots, E_K) \in \mathfrak{R}^{m \times n}$, and $h(x) := \sum_{k=1}^K h_k(x_k)$. The augmented dual function is given by

$$d(y) = \min_x g(x) + \langle y, q - Ex \rangle + \frac{\rho}{2} \|q - Ex\|^2 \quad (1.9)$$

and the dual problem (equivalent to (1.1) under mild conditions) is

$$\max_y d(y). \quad (1.10)$$

In the following, we detail the proposed BSUM-M algorithm. In its simplest form, the BSUM-M algorithm updates the dual variable using a gradient ascent step, followed by a BCD step for a certain approximate version of the augmented Lagrangian (1.8). In particular, at iteration $r + 1$, the block variable x_k is updated by solving the following subproblem

$$\min_{x_k \in X_k} u_k(x_k; x_1^{r+1}, \dots, x_{k-1}^{r+1}, x_k^r, \dots, x_K^r) + \langle y^{r+1}, q - E_k x_k \rangle + h_k(x_k) \quad (1.11)$$

where the function $u_k(\cdot; x_1^{r+1}, \dots, x_{k-1}^{r+1}, x_k^r, \dots, x_K^r)$ is an *upper-bound* of $g(x) + \frac{\rho}{2} \|q - Ex\|^2$ at a given iterate $(x_1^{r+1}, \dots, x_{k-1}^{r+1}, x_k^r, \dots, x_K^r)$. To simplify notations, let us define a new set of auxiliary variables

$$\begin{aligned} w_k^r &:= (x_1^r, \dots, x_{k-1}^r, x_k^{r-1}, x_{k+1}^{r-1}, \dots, x_K^{r-1}), \quad k = 1, \dots, K, \\ w_{K+1}^r &:= x^r, \quad w_1^r := x^{r-1}. \end{aligned}$$

The basic form of the BSUM-M algorithm is described in the following table.

Block Successive Upper-bound Minimization Method of Multipliers (BSUM-M)

At each iteration $r \geq 1$:

$$\begin{cases} y^{r+1} = y^r + \alpha^r (q - Ex^r) = y^r + \alpha^r \left(q - \sum_{k=1}^K E_k x_k^r \right), \\ x_k^{r+1} = \operatorname{argmin}_{x_k \in X_k} u_k(x_k; w_k^{r+1}) - \langle y^{r+1}, E_k x_k \rangle + h_k(x_k), \quad \forall k \end{cases} \quad (1.12)$$

where $\alpha^r > 0$ is the step size for the dual update.

In this paper, we also consider a randomized version of the BSUM-M algorithm whereby at each iteration either a single randomly chosen primal variable block or the dual variable is updated.

Randomized BSUM-M (RBSUM-M)

Select a probability vector $\{p_k\}_{k=0}^K$ such that $p_k > 0$ and $\sum_{k=0}^K p_k = 1$.

At each iteration $t \geq 1$, pick an index $k \in \{0, \dots, K\}$, with probability p_k , and

If $k = 0$

$$y^{t+1} = y^t + \alpha^t(q - Ex^t),$$

$$x_k^{t+1} = x_k^t, \quad k = 1, \dots, K.$$

Else If $k \in \{1, \dots, K\}$

$$x_k^{t+1} = \operatorname{argmin}_{x_k \in X_k} u_k(x_k; x^t) - \langle y^t, E_k x_k \rangle + h_k(x_k), \quad (1.13)$$

$$x_j^{t+1} = x_j^t, \quad \forall j \neq k,$$

$$y^{t+1} = y^t.$$

End

where $\alpha^t > 0$ is the step size for the dual update.

As explained in [14, 31], the randomized version of the BCD-type algorithm is useful under many practical scenarios, for example when not all data is available at all times. We refer the readers to the aforementioned references for detailed discussions. Note that here we have used the index “ t ” to differentiate the iteration of RBSUM-M with that of the BSUM-M. The reason is that in RBSUM-M, at each iteration only a single block variable (primal or dual) is updated, while in BSUM-M all primal and dual variables are updated once.

2 Convergence Analysis

2.1 Main Assumptions

Suppose f is a closed proper convex function in \mathfrak{R}^n . Let $\operatorname{dom} f$ denote the effective domain of f and let $\operatorname{int}(\operatorname{dom} f)$ denote the interior of $\operatorname{dom} f$. Let x_{-k} (and similarly E_{-k}) denote the vector x with x_k removed. We make the following standing assumptions regarding problem (1.1):

Assumption A.

- (a) Problem (1.1) is a convex problem, its global minimum is attained and so is its dual optimal value. The intersection $X \cap \operatorname{int}(\operatorname{dom} f) \cap \{x \mid Ex = q\}$ is nonempty.

- (b) The function $g(x)$ can be decomposed as $g(x) = \ell(Ax) + \langle x, b \rangle$, where $\ell(\cdot)$ is a strictly convex and continuously differentiable function on $\text{int}(\text{dom } \ell)$, and A is some given matrix (not necessarily full column rank).

Each nonsmooth function h_k , if present, takes the form

$$h_k(x_k) = \lambda_k \|x_k\|_1 + \sum_J w_J \|x_{k,J}\|_2,$$

where $x_k = (\cdots, x_{k,J}, \cdots)$ is a partition of x_k with J being the partition index; $\lambda_k \geq 0$ and $w_J \geq 0$ are some constants.

- (c) The feasible sets X_k , $k = 1, \cdots, K$ are compact polyhedral sets, and are given by $X_k := \{x_k \mid C_k x_k \leq c_k\}$, for some matrix $C_k \in \mathfrak{R}^{m_k \times n_k}$ and $c_k \in \mathfrak{R}^{m_k}$.

Next we make the following assumptions regarding the approximation function $u_k(\cdot; \cdot)$ in (1.11).

Assumption B.

- (a) $u_k(x_k; x) = g(x) + \frac{\rho}{2} \|Ex - q\|^2$, $\forall x \in X, \forall k$,
(b) $u_k(v_k; x) \geq g(v_k, x_{-k}) + \frac{\rho}{2} \|E_k v_k - q + E_{-k} x_{-k}\|^2$, $\forall v_k \in X_k, \forall x \in X, \forall k$,
(c) $\nabla u_k(x_k; x) = \nabla_k (g(x) + \frac{\rho}{2} \|Ex - q\|^2)$, $\forall k, \forall x \in X$,
(d) For any given x , $u_k(v_k; x)$ is continuous in v_k and x . It is also strongly convex in v_k , that is

$$u_k(v_k; x) \geq u_k(\hat{v}_k; x) + \langle \nabla u_k(\hat{v}_k; x), v_k - \hat{v}_k \rangle + \frac{\gamma_k}{2} \|v_k - \hat{v}_k\|^2, \forall v_k, \hat{v}_k \in X_k, \forall x \in X$$

where γ_k is independent of the choice of x .

- (e) For given x , $u_k(v_k; x)$ has Lipschitz continuous gradient, that is

$$\|\nabla u_k(v_k; x) - \nabla u_k(\hat{v}_k; x)\| \leq L_k \|v_k - \hat{v}_k\|, \forall \hat{v}_k, v_k \in X_k, \forall k, \forall x \in X, \quad (2.1)$$

where $L_k > 0$ is some constant. Define $L_{\max} := \max_k L_k$.

Below we give a few remarks about the assumptions made above.

Remark 2.1 *The form of $g(\cdot)$ assumed in Assumption A(b) is fairly general. For example it includes the cases like $g(\cdot) = \sum_{k=1}^K \ell_k(A_k x_k)$, or $g(\cdot) = \ell(\sum_{k=1}^K A_k x_k)$, or the combination of these two, where $\ell_k(\cdot)$'s are strictly convex functions and A_k 's are matrices not necessarily with full rank. Moreover, since the matrix A is not required to have full rank, $g(x)$ (hence $f(x)$) is not necessarily strongly convex with respect to x . Note that all three examples mentioned in Section 1.1 satisfy Assumption A(b). Moreover, this assumption requires that the nonsmooth function $h_k(\cdot)$ is in the form of mixed ℓ_1 and ℓ_2 norm.*

Remark 2.2 *Assumption B indicates that for any x , each $u_k(\cdot; x)$ is an upper-bound, locally tight up to the first order, for $g(x) + \frac{\rho}{2}\|q - Ex\|^2$ (the latter function itself satisfies Assumption B trivially). In many practical applications especially for nonsmooth problems, optimizing such functions often leads to much simpler subproblems than working directly with the original function; see e.g., [1, 40, 50, 64]. As an example, suppose the augmented Lagrangian is given by:*

$$L(x; y) = \sum_{k=1}^K \|x_k\|_2 + \langle y, q - Ax \rangle + \rho \|Ax - q\|^2.$$

Then at $(r + 1)$ -th iteration, the subproblem for x_k is given by

$$x_k^{r+1} = \arg \min_{x_k \in X_k} \|x_k\|_2 + \langle y^{r+1}, q - A_k x_k \rangle + \rho \|A_k x_k - d_k^{r+1}\|^2,$$

for some constant $d_k^{r+1} = q - \sum_{j < k} A_j x_j^{r+1} - \sum_{j > k} A_j x_j^r$. This problem does not have closed form solution. A well-known strategy is to perform a proximal gradient step [67], that is, to solve the following approximate problem instead

$$\min_{x_k \in X_k} \|x_k\|_2 + \langle y^{r+1}, q - A_k x_k \rangle + \langle 2\rho A_k^T (A_k x_k^r - d_k^{r+1}), x_k \rangle + \frac{\tau}{2} \|x_k - x_k^r\|^2 \quad (2.2)$$

This problem readily admits a closed form solution; see e.g. [21, 68]. Moreover, when choosing $\tau \geq \|A_k^T A_k\|$, the strongly convex function $\langle 2\rho A_k^T (A_k x_k^r - d_k^{r+1}), x_k \rangle + \frac{\tau}{2} \|x_k - x_k^r\|^2$ is an approximation function that satisfies Assumption B (up to some constant).

Remark 2.3 *The strong convexity assumption for the approximation function $u_k(\cdot; \cdot)$ in $B(d)$ is quite mild, see the example given in the previous remark. This assumption ensures the iterates of (randomized) BSUM-M are well defined.*

2.2 Preliminaries

We first provide two important results that characterize the augmented dual function $d(y)$ and the augmented Lagrangian function $L(x; y)$. Let $X(y)$ denote the set of optimal solutions for the primal problem, that is

$$X(y) := \arg \min_x L(x; y).$$

Let f^* denote the optimal value for (1.1). For any given $x \in X$ and any set $Y \subseteq X$, let $\text{dist}(x, Y)$ denote the distance between x and the set Y , that is, $\text{dist}(x, Y) := \min_{\hat{x} \in Y} \|x - \hat{x}\|$.

The following lemma shows the differentiability and Lipschitz continuity of $d(y)$.

Lemma 2.1 ([56, Lemma 2.1, 2.2]) *Suppose Assumption A holds. Then for any $y \in \mathfrak{R}^m$, both Ex and $A_k x_k$, $k = 1, 2, \dots, K$, are constant over $X(y)$. Moreover, the dual function $d(y)$ is differentiable*

everywhere and

$$\nabla d(y) = q - Ex(y),$$

where $x(y) \in X(y)$. Moreover, fix any scalar $\eta \leq f^*$ and let $\mathcal{U} = \{ y \in \mathbb{R}^m \mid d(y) \geq \eta \}$. Then there holds

$$\|\nabla d(y') - \nabla d(y)\| \leq \frac{1}{\rho} \|y' - y\|, \quad \forall y' \in \mathcal{U}, y \in \mathcal{U}.$$

We then introduce the notion of a proximal gradient, which will serve as a measure of optimality.

Definition 2.1 (Proximal Gradient) *Suppose a convex function $f(x)$ can be written as $f(x) = g(x) + h(x)$ where g is convex and differentiable, h is a convex (possibly nonsmooth) function. Then we can define the proximal gradient of f with respect to h as*

$$\tilde{\nabla} f(x) := x - \text{prox}_h(x - \nabla g(x)),$$

where $\text{prox}_h(\cdot)$ is the proximity operator defined by

$$\text{prox}_h(x) = \underset{u \in \mathbb{R}^n}{\text{argmin}} \quad h(u) + \frac{1}{2} \|x - u\|^2.$$

Using the above definition, the proximal gradient for the augmented Lagrangian function can be expressed as

$$\tilde{\nabla}_x L(x; y) := x - \text{prox}_h(x - \nabla_x(L(x; y) - h(x))). \quad (2.3)$$

Lemma 2.2 ([56, Lemma 2.3]) *Suppose Assumptions A(a)—A(b) hold. Then*

1. *If in addition X is a polyhedral set (not necessarily compact), then there exist a positive scalars τ and δ such that the following error bound holds*

$$\text{dist}(x, X(y)) \leq \tau \|\tilde{\nabla}_x L(x; y)\|, \quad (2.4)$$

for all (x, y) such that $\|\tilde{\nabla}_x L(x; y)\| \leq \delta$, where the proximal gradient $\tilde{\nabla}_x L(x; y)$ is given by (2.3).

2. *If X is also a compact set, then there exists some $\tau > 0$ such that the error bound (2.4) holds for all $x \in X \cap \text{dom } h$.*

Moreover, in both cases the constant τ is independent of the choice of y and x .

If either the objective function f is strongly convex (i.e., A is full column rank in Assumptions A(b)), or if E is full row rank, then the augmented Lagrangian function $L(x; y)$ is strongly convex. In this case, the error bound in Lemma 2.2 holds automatically and globally with Assumption A(b) or the requirement that X_k being polyhedral or compact.

2.3 Convergence Analysis

In this section, we analyze the convergence of BSUM-M and its randomized version RBSUM-M.

Under Assumption B(d), each function $u_k(v_k; x)$ is strongly convex with respect to $v_k \in X_k$. As a result, the primal update steps in (1.12) and (1.13) are both well defined and have unique optimal solutions. For RBSUM-M, let us define a new vector $\hat{x}^{t+1} = [\hat{x}_1^{t+1}, \dots, \hat{x}_K^{t+1}]$ where

$$\hat{x}_k^{t+1} := \arg \min_{x_k \in X_k} u_k(x_k; x^t) + \langle y^t, q - E_k x_k \rangle + h_k(x_k), \quad k = 1, \dots, K. \quad (2.5)$$

Define \hat{y}^{t+1} as

$$\hat{y}^{t+1} = y^t + \alpha^t (q - E x^t). \quad (2.6)$$

Define $z^t := [x_1^t, \dots, x_K^t, y^t]$, then we can write $L(x^t; y^t)$ equivalently as $L(z^t)$.

We first characterize the successive difference of the augmented Lagrangian before and after one primal update (resp. one update) for the BSUM-M (resp. RBSUM-M).

Lemma 2.3 *Suppose Assumption B holds. Then*

1. *For BSUM-M, we have*

$$L(x^r; y^{r+1}) - L(x^{r+1}; y^{r+1}) \geq \gamma \|x^r - x^{r+1}\|^2, \quad (2.7)$$

where the constant $\gamma > 0$ is independent of r and y^{r+1} .

2. *For RBSUM-M, we have*

$$\mathbb{E}[L(z^t) - L(z^{t+1}) \mid z^t] \geq \hat{\gamma} \|x^t - \hat{x}^{t+1}\|^2 - \alpha^t p_0 \|q - E x^t\|^2, \quad (2.8)$$

where the expectation is taken over the algorithm's random choice of the update index; the constant $\hat{\gamma} > 0$ is independent of t and y^t .

Proof. We first show part (1) of the claim. Using Assumption B, we have that

$$\begin{aligned} & L(w_k^{r+1}; y^{r+1}) - L(w_{k+1}^{r+1}; y^{r+1}) \\ & \geq u_k(x_k^r; w_k^{r+1}) - \langle y^{r+1}, E_k x_k^r \rangle + h_k(x_k^r) \\ & \quad - (u_k(x_k^{r+1}; w_k^{r+1}) - \langle y^{r+1}, E_k x_k^{r+1} \rangle + h_k(x_k^{r+1})) \\ & \geq \gamma_k \|x_k^{r+1} - x_k^r\|^2 \end{aligned} \quad (2.9)$$

where the first inequality is due to Assumption B(a)–B(b), the second inequality is due to the strong convexity Assumption B(d), and the fact that x_k^{r+1} is the optimal solution for the convex problem

$$\arg \min_{x_k \in X_k} u_k(x_k; w_k^{r+1}) - \langle y^{r+1}, E_k x_k \rangle + h_k(x_k).$$

Summing over k and letting $\gamma := \min_k \gamma_k$, we obtain

$$L(x^r; y^{r+1}) - L(x^{r+1}; y^{r+1}) \geq \gamma \|x^r - x^{r+1}\|^2. \quad (2.10)$$

We then show part (2) of the claim. We have the following

$$\begin{aligned} & \mathbb{E}[L(z^t) - L(z^{t+1}) \mid z^t] \\ &= \sum_{k=1}^K p_k [L(x^t, y^t) - L(x_{-k}^t, \hat{x}_k^{t+1}; y^t)] + p_0 [L(x^t; y^t) - L(x^t; \hat{y}^{t+1})] \\ &\geq \sum_{k=1}^K p_k [u_k(x_k^t; x^t) - \langle y^t, E_k x_k^t \rangle + h_k(x_k^t) - u(\hat{x}_k^{t+1}; x^t) + \langle y^t, E_k \hat{x}_k^{t+1} \rangle - h_k(\hat{x}_k^{t+1})] \\ &\quad + p_0 [L(x^t; y^t) - L(x^t; \hat{y}^{t+1})] \\ &\geq \sum_{k=1}^K p_k \gamma_k \|x_k^t - \hat{x}_k^{t+1}\|^2 - \alpha^t p_0 \|q - E x^t\|^2 \\ &\geq \hat{\gamma} \|x^t - \hat{x}^{t+1}\|^2 - \alpha^t p_0 \|q - E x^t\|^2 \end{aligned} \quad (2.11)$$

where $\hat{\gamma} := \min_k p_k \gamma_k$ is independent of t and y^t .

Q.E.D.

Next we bound the size of the proximal gradient at any given iterate.

Lemma 2.4 *Suppose Assumption B holds. Then*

1. *For the iterates $\{(x^r, y^r)\}$ generated by the BSUM-M, there exists some constant $\sigma > 0$ (independent of y^r) such that*

$$\|\tilde{\nabla} L(x^r; y^r)\| \leq \sigma \|x^{r+1} - x^r\| \quad (2.12)$$

for all $r \geq 1$.

2. *For the iterates $\{(x^t, y^t)\}$ generated by the RBSUM-M, there exist some constants $\hat{\sigma}_1 > 0$ and $\hat{\sigma}_2 > 0$ (independent of y^t) such that*

$$\|\tilde{\nabla} L(x^t; \hat{y}^{t+1})\| \leq \hat{\sigma}_1 \|\hat{x}^{t+1} - x^t\| + \hat{\sigma}_2 \|\hat{y}^{t+1} - y^t\| \quad (2.13)$$

for all $t \geq 1$.

Proof. The proof of two cases follow similar steps, thus we only prove the second case here. Fix any $t \geq 1$ and any $1 \leq k \leq K$. According to the definition of \hat{x}_k^{t+1} in (2.5), we have

$$\hat{x}_k^{t+1} = \text{prox}_{h_k} [\hat{x}_k^{t+1} - \nabla u_k(\hat{x}_k^{t+1}; x^t) + E_k^T y^t]. \quad (2.14)$$

Therefore, we have

$$\begin{aligned} & \|\hat{x}_k^{t+1} - \text{prox}_{h_k} [x_k^t - \nabla u_k(x_k^t; x^t) + E_k^T \hat{y}^{t+1}]\| \\ &= \|\text{prox}_{h_k} [\hat{x}_k^{t+1} - \nabla u_k(\hat{x}_k^{t+1}; x^t) + E_k^T y^t] - \text{prox}_{h_k} [x_k^t - \nabla u_k(x_k^t; x^t) + E_k^T \hat{y}^{t+1}]\| \\ &\leq (L_k + 1)\|\hat{x}_k^{t+1} - x_k^t\| + \|E_k\|\|\hat{y}^{t+1} - y^t\| \end{aligned}$$

where the inequality follows from the nonexpansive property of the prox operator, as well as the Lipschitz continuity property of the gradient vector ∇u_k (cf. Assumption B(e)). Using this relation and the definition of the proximal gradient $\tilde{\nabla}_k L(x^t; \hat{y}^{t+1})$, we have

$$\begin{aligned} \|\tilde{\nabla}_k L(x^t; \hat{y}^{t+1})\| &= \|x_k^t - \text{prox}_{h_k} (x_k^t - \nabla_k g(x^t) + E_k^T \hat{y}^{t+1})\| \\ &\leq \|x_k^t - \hat{x}_k^{t+1}\| + \|\hat{x}_k^{t+1} - \text{prox}_{h_k} (x_k^t - \nabla_k g(x^t) + E_k^T \hat{y}^{t+1})\| \\ &= \|x_k^t - \hat{x}_k^{t+1}\| + \|\hat{x}_k^{t+1} - \text{prox}_{h_k} (x_k^t - \nabla_k u_k(x_k^t; x^t) + E_k^T \hat{y}^{t+1})\| \\ &\leq (L_k + 2)\|\hat{x}_k^{t+1} - x_k^t\| + \|E_k\|\|\hat{y}^{t+1} - y^t\|, \quad \forall k = 1, 2, \dots, K. \end{aligned}$$

This further implies that the full proximal gradient vector can be bounded by $\|\hat{x}^{t+1} - x^t\|$:

$$\|\tilde{\nabla} L(x^t; \hat{y}^{t+1})\| \leq \left(\max_k \{L_k\} + 2 \right) \sqrt{K} \|\hat{x}^{t+1} - x^t\| + \sqrt{K} \max_k \|E_k\| \|\hat{y}^{t+1} - y^t\|.$$

Setting $\hat{\sigma}_1 = (\max_k \{L_k\} + 2)\sqrt{K}$ and $\hat{\sigma}_2 = \sqrt{K} \max_k \|E_k\|$ (both of which are independent of y^t) completes the proof. **Q.E.D.**

To analyze the convergence of the algorithms, we need to make use of a certain ‘‘potential function’’ that measures the algorithm progress. Similar to [56], we will adopt the combined primal and dual optimality gap (to be defined shortly) as the ‘‘potential function’’.

Let d^* denote the dual optimal value. Due to Assumption A(a), d^* also equals to the primal optimal value. For each algorithm, define the *dual optimality gap* by

$$\Delta_d^r = d^* - d(y^r), \quad \Delta_d^t = d^* - d(y^t), \quad (2.15)$$

each of which represents the gap to the dual optimality at the current iteration. Similarly, define the *primal optimality gap* at each iteration by

$$\Delta_p^r = L(x^r; y^r) - d(y^r), \quad \Delta_p^t = L(x^t; y^t) - d(y^t). \quad (2.16)$$

Clearly, we have both $\Delta_d^r \geq 0$ and $\Delta_p^r \geq 0$ for all $r \geq 1$ (resp. $\Delta_d^t \geq 0$ and $\Delta_p^t \geq 0$ for all $t \geq 1$).

Let $X(y^r)$ denote the set of optimal solutions for the following optimization problem

$$\min_x L(x; y^r) = \min_x g(x) + \langle y^r, q - Ex \rangle + \frac{\rho}{2} \|Ex - q\|^2.$$

We denote

$$\bar{x}^r = \operatorname{argmin}_{\bar{x} \in X(y^r)} \|\bar{x} - x^r\|, \quad \bar{x}^t = \operatorname{argmin}_{\bar{x} \in X(\hat{y}^t)} \|\bar{x} - x^t\|.$$

We then bound the decrease of the dual optimality gap for BSUM-M as well as the conditional expected decrease of the dual optimality gap for the RBSUM-M.

Lemma 2.5 *1. For the BSUM-M algorithm, there holds*

$$\Delta_d^r - \Delta_d^{r-1} \leq -\alpha^{r-1} (Ex^r - q)^T (E\bar{x}^r - q). \quad (2.17)$$

2. For the RBSUM-M algorithm, there holds

$$\mathbb{E}[\Delta_d^t - \Delta_d^{t-1} \mid z^{t-1}] \leq -\alpha^{t-1} p_0 (Ex^{t-1} - q)^T (E\bar{x}^t - q). \quad (2.18)$$

Proof. The proof for the first case is similar to [56, Lemma 3.2]. We outline the proof here for completeness. We have the following series of inequalities

$$\begin{aligned} \Delta_d^r - \Delta_d^{r-1} &= d(y^{r-1}) - d(y^r) \\ &= L(\bar{x}^{r-1}; y^{r-1}) - L(\bar{x}^r; y^r) \\ &= [L(\bar{x}^r; y^{r-1}) - L(\bar{x}^r; y^r)] + [L(\bar{x}^{r-1}; y^{r-1}) - L(\bar{x}^r; y^{r-1})] \\ &= (y^{r-1} - y^r)^T (q - E\bar{x}^r) + [L(\bar{x}^{r-1}; y^{r-1}) - L(\bar{x}^r; y^{r-1})] \\ &= -\alpha^{r-1} (Ex^{r-1} - q)^T (E\bar{x}^r - q) + [L(\bar{x}^{r-1}; y^{r-1}) - L(\bar{x}^r; y^{r-1})] \\ &\leq -\alpha^{r-1} (Ex^{r-1} - q)^T (E\bar{x}^r - q), \quad \forall r \geq 1, \end{aligned} \quad (2.19)$$

where the last equality follows from the update of the dual variable y^{r-1} ; the last inequality is due to the fact that \bar{x}^{r-1} minimizes $L(\cdot, y^{r-1})$.

The proof for the second case is straightforward, as we can readily observe that

$$\begin{aligned} \mathbb{E}[\Delta_d^t - \Delta_d^{t-1} \mid z^{t-1}] &= \mathbb{E}[d(y^{t-1}) - d(y^t) \mid z^{t-1}] \\ &= p_0 (d(y^{t-1}) - d(\hat{y}^t)) \\ &\leq -\alpha^{t-1} p_0 (Ex^{t-1} - q)^T (E\bar{x}^t - q), \end{aligned}$$

where the last inequality has utilized the result in (2.19). This concludes the proof. \square

Q.E.D.

Next we proceed to bound the decrease (resp. conditional expected decrease) of the primal gap for the BSUM-M (resp. RBSUM-M).

Lemma 2.6 *Suppose Assumption B holds. Then*

1. *For the BSUM-M, the following bound holds true for each $r \geq 1$*

$$\Delta_p^{r+1} - \Delta_p^r \leq \alpha^r \|Ex^r - q\|^2 - \gamma \|x^{r+1} - x^r\|^2 - \alpha^r (Ex^r - q)^T (E\bar{x}^{r+1} - q) \quad (2.20)$$

for some γ independent of y^r .

2. *For the RBSUM-M, the following bound holds true for each $t \geq 1$*

$$\mathbb{E} [\Delta_p^{t+1} - \Delta_p^t \mid z^t] \leq p_0 \alpha^r \|Ex^t - q\|^2 - \hat{\gamma} \|\hat{x}^{t+1} - x^t\|^2 - \alpha^t p_0 (Ex^t - q)^T (E\bar{x}^{t+1} - q) \quad (2.21)$$

for some $\hat{\gamma}$ independent of y^r .

Proof. We first show part (1) of the claim. This result is similar to [56, Lemma 3.3], and we include its derivation here for completeness. Fix any $r \geq 1$, by using the dual update rule (cf. (1.12)), we have

$$\begin{aligned} L(x^r; y^{r+1}) &= f(x^r) + \langle y^r, q - Ex^r \rangle + \frac{\rho}{2} \|Ex^r - q\|^2 + \alpha^r \|Ex^r - q\|^2 \\ &= L(x^r; y^r) + \alpha^r \|Ex^r - q\|^2. \end{aligned}$$

Combined with the first part of Lemma 2.3, we obtain

$$L(x^{r+1}; y^{r+1}) - L(x^r; y^r) \leq \alpha^r \|Ex^r - q\|^2 - \gamma \|x^{r+1} - x^r\|^2, \quad \forall r \geq 1.$$

Hence, we have the following bound on the reduction of primal optimality gap

$$\begin{aligned} \Delta_p^{r+1} - \Delta_p^r &= (L(x^{r+1}; y^{r+1}) - d(y^{r+1})) - (L(x^r; y^r) - d(y^r)) \\ &= L(x^{r+1}; y^{r+1}) - L(x^r; y^r) - (d(y^{r+1}) - d(y^r)) \\ &\leq \alpha^r \|Ex^r - q\|^2 - \gamma \|x^{r+1} - x^r\|^2 - \alpha^r (Ex^r - q)^T (E\bar{x}^{r+1} - q), \quad \forall r \geq 1, \end{aligned}$$

where the last step is due to the first part of Lemma 2.5.

We then show part (2) of the claim. We have that for all $t \geq 1$

$$\begin{aligned} \mathbb{E} [\Delta_p^{t+1} - \Delta_p^t \mid z^t] &= \mathbb{E} [(L(z^{t+1}) - d(y^{t+1})) - (L(z^t) - d(y^t)) \mid z^t] \\ &= \mathbb{E}[L(z^{t+1}) - L(z^t) \mid z^t] - \mathbb{E}[d(y^{t+1}) - d(y^t) \mid z^t] \\ &\leq -\hat{\gamma} \|\hat{x}^{t+1} - x^t\|^2 + \alpha^t p_0 \|Ex^t - q\|^2 - \alpha^t p_0 (Ex^t - q)^T (E\bar{x}^{t+1} - q) \end{aligned}$$

where the last step is due to Lemma 2.5 and Lemma 2.3.

Q.E.D.

Next we present the first main result regarding the convergence of the BSUM-M and RBSUM-M.

Theorem 2.1 *Suppose that the error bound in Lemma 2.2 and Assumption B hold. Assume that one of the following stepsize rules is used: i) for all r , the stepsize $\alpha^r = \alpha$ is sufficiently small, or ii) α^r satisfies*

$$\sum_{r=1}^{\infty} \alpha^r = \infty, \quad \lim_{r \rightarrow \infty} \alpha^r = 0. \quad (2.22)$$

Then we have the following:

1. For the BSUM-M, we have $\lim_{r \rightarrow \infty} \|Ex^r - q\| = 0$, $\lim_{r \rightarrow \infty} \|x^r - x^{r+1}\| = 0$ and $\lim_{r \rightarrow \infty} \|x^r - \bar{x}^r\| = 0$. Further, every limit point of $\{x^r, y^r\}$ is a primal and dual optimal solution.
2. For the RBSUM-M, we have $\lim_{t \rightarrow \infty} \|Ex^t - q\| = 0$, $\lim_{t \rightarrow \infty} \|x^t - x^{t+1}\| = 0$, and $\lim_{t \rightarrow \infty} \|x^t - \bar{x}^t\| = 0$ w.p.1. Further, every limit point of $\{x^t, y^t\}$ is a primal and dual optimal solution w.p.1.

Proof. We focus on showing part (2). The proof for part (1) is easier and follows similar steps.

From Assumption A(c) we have that each X_k is compact, which implies the boundedness of x^t . Thus, we obtain from Lemma 2.2 that

$$\|x^t - \bar{x}^{t+1}\| \leq \tau \|\tilde{\nabla} L(x^t; \hat{y}^{t+1})\| \quad (2.23)$$

for some $\tau > 0$ (independent of y^t). Combining the two estimates (2.18) and (2.21), we obtain

$$\begin{aligned} & \mathbb{E}[\Delta_p^{t+1} + \Delta_d^{t+1} | z^t] - \mathbb{E}[\Delta_p^t + \Delta_d^t | z^t] \\ &= \mathbb{E}[\Delta_p^{t+1} - \Delta_p^t | z^t] + \mathbb{E}[\Delta_d^{t+1} - \Delta_d^t | z^t] \\ &\leq \alpha^t p_0 \|Ex^t - q\|^2 - \hat{\gamma} \|\hat{x}^{t+1} - x^t\|^2 - 2\alpha^t p_0 (Ex^t - q)^T (E\bar{x}^{t+1} - q) \\ &= \alpha^t p_0 \|Ex^t - E\bar{x}^{t+1}\|^2 - \alpha^t p_0 \|E\bar{x}^{t+1} - q\|^2 - \hat{\gamma} \|\hat{x}^{t+1} - x^t\|^2. \end{aligned} \quad (2.24)$$

Now we invoke (2.23) and Lemma 2.4 to upper bound $\|x^t - \bar{x}^{t+1}\|$:

$$\|x^t - \bar{x}^{t+1}\| \leq \tau \|\tilde{\nabla} L(x^t; \hat{y}^{t+1})\| \leq \tau (\hat{\sigma}_1 \|\hat{x}^{t+1} - x^t\| + \hat{\sigma}_2 \|\hat{y}^{t+1} - y^t\|). \quad (2.25)$$

Therefore, defining $\tilde{\sigma}_i^2 = 2\tau^2 \hat{\sigma}_i^2$ ($i = 1, 2$), we have

$$\begin{aligned} \|x^t - \bar{x}^{t+1}\|^2 &\leq 2\tau^2 (\hat{\sigma}_1^2 \|\hat{x}^{t+1} - x^t\|^2 + \hat{\sigma}_2^2 \|\hat{y}^{t+1} - y^t\|^2) \\ &:= \tilde{\sigma}_1^2 \|\hat{x}^{t+1} - x^t\|^2 + \tilde{\sigma}_2^2 \|\hat{y}^{t+1} - y^t\|^2 \\ &= \tilde{\sigma}_1^2 \|\hat{x}^{t+1} - x^t\|^2 + \tilde{\sigma}_2^2 \alpha_t^2 \|q - Ex^t\|^2, \end{aligned} \quad (2.26)$$

where the last step follows from (2.6). Using this result, we can bound the size of the constraint violation as follows

$$\begin{aligned}
\|q - Ex^t\|^2 &= \|q - E\bar{x}^{t+1} + E\bar{x}^{t+1} - Ex^t\|^2 \\
&\leq 2\|q - E\bar{x}^{t+1}\|^2 + 2\|E\|^2\|\bar{x}^{t+1} - x^t\|^2 \\
&\leq 2\|q - E\bar{x}^{t+1}\|^2 + 2\|E\|^2(\tilde{\sigma}_1^2\|\hat{x}^{t+1} - x^t\|^2 + \tilde{\sigma}_2^2\|\hat{y}^{t+1} - y^t\|^2) \\
&= 2\|q - E\bar{x}^{t+1}\|^2 + 2\|E\|^2(\tilde{\sigma}_1^2\|\hat{x}^{t+1} - x^t\|^2 + \tilde{\sigma}_2^2(\alpha^t)^2\|q - Ex^t\|^2).
\end{aligned}$$

Rearranging terms, we obtain (assuming α^t is small enough such that $1 - 2\tilde{\sigma}_2^2(\alpha^t)^2\|E\|^2 > 0$)

$$\|q - Ex^t\|^2 \leq \frac{2\|q - E\bar{x}^{t+1}\|^2 + 2\|E\|^2\tilde{\sigma}_1^2\|\hat{x}^{t+1} - x^t\|^2}{1 - 2\tilde{\sigma}_2^2(\alpha^t)^2\|E\|^2}. \quad (2.28)$$

Substituting (2.27) into (2.25) and using (2.28) yields

$$\begin{aligned}
&\mathbb{E}[(\Delta_p^{t+1} + \Delta_d^{t+1}) - (\Delta_p^t + \Delta_d^t) \mid z^t] \\
&\leq (\alpha^t p_0 \|E\|^2 \tilde{\sigma}_1^2 - \hat{\gamma}) \|\hat{x}^{t+1} - x^t\|^2 - \alpha^t p_0 \|E\bar{x}^{t+1} - q\|^2 + (\alpha^t)^3 p_0 \|E\|^2 \tilde{\sigma}_2^2 \|q - Ex^t\|^2 \\
&\leq \left(\alpha^t p_0 \|E\|^2 \tilde{\sigma}_1^2 + \frac{2\|E\|^4 \tilde{\sigma}_1^2 (\alpha^t)^3 p_0 \tilde{\sigma}_2^2}{1 - 2\tilde{\sigma}_2^2 (\alpha^t)^2 \|E\|^2} - \hat{\gamma} \right) \|\hat{x}^{t+1} - x^t\|^2 \\
&\quad + \left(\frac{2(\alpha^t)^3 p_0 \|E\|^2 \tilde{\sigma}_2^2}{1 - 2\tilde{\sigma}_2^2 (\alpha^t)^2 \|E\|^2} - \alpha^t p_0 \right) \|E\bar{x}^{t+1} - q\|^2.
\end{aligned} \quad (2.29)$$

Case 1): If we choose the constant stepsize $\alpha^t = \alpha$, and let α be sufficiently small. Then the constants in front of $\|\hat{x}^{t+1} - x^t\|^2$ and $\|E\bar{x}^{t+1} - q\|^2$ in (2.29) become negative. By applying the convergence theorem of non-negative almost supermartingale [69, Theorem 1], we have that

$$\lim_{t \rightarrow \infty} \Delta_p^{t+1} + \Delta_d^{t+1} \quad \text{exists and is finite, w.p.1,}$$

$$\lim_{t \rightarrow \infty} \|\hat{x}^{t+1} - x^t\| = 0 \quad \text{w.p.1,} \quad (2.30)$$

$$\lim_{t \rightarrow \infty} \|E\bar{x}^{t+1} - q\| = \lim_{t \rightarrow \infty} \|\nabla d(y^{t+1})\| = 0 \quad \text{w.p.1.} \quad (2.31)$$

We conclude that every limit point of the sequence $\{y^t\}$ is a dual optimal solution. Further, by (2.28), the constraint violation vanishes in the limit, i.e.,

$$\lim_{t \rightarrow \infty} \|Ex^t - q\| = \lim_{t \rightarrow \infty} \|y^t - \hat{y}^{t+1}\| = 0, \quad \text{w.p.1.} \quad (2.32)$$

Using (2.30) and the fact that $x^{t+1} - x^t$ has only one nonzero block which equals the corresponding block of $\hat{x}^{t+1} - x^t$ (c.f. (2.5)), we have

$$\lim_{t \rightarrow \infty} \|x^{t+1} - x^t\| \leq \lim_{t \rightarrow \infty} \|\hat{x}^{t+1} - x^t\| = 0, \quad \text{w.p.1.} \quad (2.33)$$

Substituting (2.30) and (2.32) into (2.26), we obtain $\lim_{t \rightarrow \infty} \|x^t - \bar{x}^{t+1}\| = 0$, w.p.1. Combining this with (2.33) further implies

$$\lim_{t \rightarrow \infty} \|x^{t+1} - \bar{x}^{t+1}\| = 0, \text{ w.p.1.} \quad (2.34)$$

Since $\bar{x}^{t+1} \in X(y^{t+1})$, we have $L(\bar{x}^{t+1}, y^{t+1}) \leq L(x, y^{t+1})$ for all $x \in X$. Passing limit, we have $L(x^\infty, y^\infty) \leq L(x, y^\infty)$ for all $x \in X$ w.p.1, where (x^∞, y^∞) is a limit point of $\{x^t, y^t\}$. Combining this with (2.32), we conclude that x^∞ is a primal optimal solution satisfying $x^\infty \in X(y^\infty)$ w.p.1.

Case 2): Suppose the stepsize is chosen according to (2.22). Then similar to Case 1, we have the descent estimate given in (2.29).

The assumption $\alpha^t \rightarrow 0$ implies that there must exist an index t_0 such that for all $t > t_0$, the constants in front of $\|\hat{x}^{t+1} - x^t\|^2$ and $\|E\bar{x}^{t+1} - q\|^2$ in (2.29) become negative. By applying the convergence theorem of non-negative almost supermartingale again, we conclude that

$$\sum_{t=1}^{\infty} \|\hat{x}^{t+1} - x^t\|^2 < \infty, \text{ w.p.1,} \quad (2.35)$$

$$\sum_{t=1}^{\infty} \alpha^t \|E\bar{x}^{t+1} - q\|^2 < \infty, \text{ w.p.1.} \quad (2.36)$$

Eq. (2.35) implies that

$$\|\hat{x}^{t+1} - x^t\| \rightarrow 0 \text{ w.p.1.} \quad (2.37)$$

while Eqs. (2.36) and (2.22) imply that

$$\liminf_{t \rightarrow \infty} \|E\bar{x}^{t+1} - q\|^2 = 0, \text{ w.p.1.} \quad (2.38)$$

To complete the proof, we show below that in fact $\lim_{t \rightarrow \infty} \|E\bar{x}^{t+1} - q\|^2 = 0$ w.p.1. Assume the contrary, so that there exists a $\delta > 0$ such that

$$\limsup_{t \rightarrow \infty} \|E\bar{x}^{t+1} - q\|^2 = \delta > 0. \quad (2.39)$$

Here and in what follows, all the statements hold in the almost sure sense, but we will omit the qualification ‘‘w.p.1’’ for simplicity. Using the Lipchitz continuity property of $\nabla d(y)$ (c.f., Lemma 2.1), we have

$$\begin{aligned} \|E\bar{x}^{t+1} - q\| - \|E\bar{x}^t - q\| &\leq \|(E\bar{x}^{t+1} - q) - (E\bar{x}^t - q)\| = \|\nabla d(\hat{y}^{t+1}) - \nabla d(\hat{y}^t)\| \\ &\leq \frac{1}{\rho} \|\hat{y}^{t+1} - \hat{y}^t\| \leq \frac{1}{\rho} (\|\hat{y}^{t+1} - y^t\| + \|y^t - \hat{y}^t\|) \\ &\leq \frac{1}{\rho} (\|\hat{y}^{t+1} - y^t\| + \|y^{t-1} - \hat{y}^t\|) \\ &= \frac{\alpha^t}{\rho} \|q - Ex^t\| + \frac{\alpha^{t-1}}{\rho} \|q - Ex^{t-1}\|. \end{aligned} \quad (2.40)$$

Note that the second to the last inequality is true because if a primal variable is updated at iteration t , we have $\|y^t - \hat{y}^t\| = \|y^{t-1} - \hat{y}^t\|$; else we have $0 = \|y^t - \hat{y}^t\| \leq \|y^{t-1} - \hat{y}^t\|$.

Now by (2.38) and under the assumption (2.39), there must exist two infinite subsequences $\{t(n)\}$ and $\{t(p)\}$ such that

$$\begin{aligned} \|E\bar{x}^{t(n)} - q\| &< \frac{\delta}{4}, & \|E\bar{x}^{t(n)+1} - q\| &> \frac{\delta}{4} \\ \frac{\delta}{2} &< \|E\bar{x}^{t(p)} - q\|, & \frac{\delta}{4} &< \|E\bar{x}^t - q\| < \frac{\delta}{2}, \quad \forall t \in [t(n) + 1, t(p) - 1]. \end{aligned} \quad (2.41)$$

It follows from (2.41) and (2.40) that

$$\begin{aligned} \frac{1}{4}\delta &< \|E\bar{x}^{t(p)} - q\| - \|E\bar{x}^{t(n)} - q\| \\ &= \sum_{t=t(n)}^{t(p)-1} (\|E\bar{x}^{t+1} - q\| - \|E\bar{x}^t - q\|) \\ &\leq \sum_{t=t(n)}^{t(p)-1} \left(\frac{\alpha^t}{\rho} \|q - Ex^t\| + \frac{\alpha^{t-1}}{\rho} \|q - Ex^{t-1}\| \right). \end{aligned} \quad (2.42)$$

Due to the fact that x^t lies in a compact set, there must exist a finite constant $\zeta > 0$ such that $\|q - Ex^t\| \leq \frac{\zeta}{16}\delta$ for all t . Combining this inequality with Eqs. (2.41)-(2.42), we conclude that for all p and n large enough,

$$\frac{2\rho}{\zeta} < \sum_{t=t(n)}^{t(p)} \alpha^{t-1}.$$

Since $\alpha^t \rightarrow 0$, this further implies

$$\frac{\rho}{\zeta} < \sum_{t=t(n)+1}^{t(p)} \alpha^{t-1}, \quad \text{for large } p \text{ and } n. \quad (2.43)$$

From $\sum_t \alpha^t \|E\bar{x}^{t+1} - q\|^2 < \infty$, we know that for any given $c > 0$, there exist p and n large enough such that

$$\sum_{t=t(n)+1}^{t(p)} \alpha^{t-1} \|E\bar{x}^t - q\|^2 \leq c.$$

Since $\|E\bar{x}^t - q\| \geq \delta/4$ for $t \in [t(n) + 1, t(p) - 1]$ (see (2.41)), it follows that

$$\frac{\delta^2}{16} \sum_{t=t(n)+1}^{t(p)} \alpha^{t-1} < c.$$

Let us set $c = \frac{\delta^2 \rho}{16\zeta}$, then we have

$$\sum_{t=t(n)+1}^{t(p)} \alpha^{t-1} < \frac{\delta^2 \rho}{16\zeta} \times \frac{16}{\delta^2} = \frac{\rho}{\zeta}, \quad \forall p, n \text{ large enough,}$$

which is a contradiction to (2.43). Hence, we must have

$$\limsup_{t \rightarrow \infty} \|E\bar{x}^t - q\| = 0.$$

Combining with (2.38), we have

$$\lim_{t \rightarrow \infty} \|E\bar{x}^t - q\| = 0, \text{ w.p.1.} \quad (2.44)$$

Similar to the proof for part (1), using (2.37) and (2.44), we conclude that with probability 1, every limit point of $\{x^t, y^t\}$ is a primal-dual optimal solution. **Q.E.D.**

Remark 2.4 *If either the objective function is strongly convex (i.e., A is full column rank in Assumptions A(b)), or if E is full column rank, then the augmented Lagrangian function $L(x; y)$ is strongly convex, implying that the error bound in Lemma 2.2 holds automatically and globally. In this case, Theorem 2.1 holds without Assumption A(b) nor the requirement of X_k being polyhedral and compact.*

3 Unconstrained Convex Optimization

In this section, we specialize the BSUM-M and the RBSUM-M methods to the unconstrained case. Since the linear coupling constraints are absent, the (randomized) BSUM-M reduces to the (randomize) BSUM algorithm, and stronger convergence results can be obtained.

3.1 The BSUM Algorithm

Consider the following special case of problem (1.1)

$$\begin{aligned} & \text{minimize} && f(x) := g(x_1, \dots, x_K) + \sum_{k=1}^K h_k(x_k) \\ & \text{subject to} && x_k \in X_k, \quad k = 1, 2, \dots, K. \end{aligned} \quad (3.45)$$

For each component k , define $u_k(\cdot; x): X_k \mapsto \mathfrak{R}$ as a locally tight upper-bound for the smooth function $g(\cdot)$ at a given point $x \in X$. Below we will assume that $u_k(\cdot; x)$ satisfies Assumption B (with $\rho = 0$).

The BSUM and R-BSUM algorithms are outlined in the following tables. Note that the BSUM-type algorithms described in this section are more general than the conventional BCD, in the sense that an approximation function $u_k(\cdot; \cdot)$ is used to update each component.

Block Successive Upper-bound Minimization (BSUM)

At each iteration $r \geq 1$:

$$x_k^{r+1} = \arg \min_{x_k \in X_k} u_k(x_k; w_k^{r+1}) + h_k(x_k), \quad k = 1, \dots, K. \quad (3.46)$$

Randomized BSUM (R-BSUM)

Select a probability vector $\{p_k\}_{k=1}^K$ such that $p_k > 0$ and $\sum_{k=1}^K p_k = 1$.

At each iteration $t \geq 1$, pick an index $k \in \{1, \dots, K\}$, with probability p_k

$$\begin{aligned} x_k^{t+1} &= \arg \min_{x_k \in X_k} u_k(x_k; x^t) + h_k(x_k), \\ x_j^{t+1} &= x_j^t, \quad \forall j \neq k. \end{aligned} \quad (3.47)$$

3.2 Linear Convergence of the BSUM Algorithm

In this section, we show that under similar assumptions given in Section 2.1, both BSUM and R-BSUM converge linearly. For the BSUM algorithm, define the optimality gap as $\Delta^r := f(x^{r+1}) - f(x^*)$, where $x^* \in X^*$ is an optimal solution. Similarly, for the R-BSUM algorithm, define $\Delta^t := f(x^{t+1}) - f(x^*)$.

We first note that these algorithms indeed converge. This is a consequence of Theorem 2.1 (just ignore the linear constraints).

Corollary 3.1 *Suppose Assumptions A(a) and B hold. Then we have the following:*

1. *For the BSUM, the sequences $\{\Delta^r\}$ and $\{\|x^r - x^{r+1}\|\}$ both converge to zero. Further, every limit point of $\{x^r\}$ is an optimal solution for problem (3.45).*
2. *For the R-BSUM, the sequences $\{\Delta^t\}$ and $\{\|x^t - x^{t+1}\|\}$ both converge to zero w.p.1. Further, every limit point of $\{x^t\}$ is an optimal solution for problem (3.45) w.p.1.*

Remark 3.1 *The conditions used in Corollary 3.1 are slightly stronger than those for the original BSUM algorithm [1]. In particular, here we require that the per-block upper-bound function $u_k(v_k; x)$*

is strongly convex with respect to v_k , while in [1, Theorem 2(a)], it is only assumed that $u_k(v_k; x)$ is quasi-convex, and that problem $\min_{v_k \in X_k} u_k(v_k; x)$ has a unique optimal solution. The per-block strong convexity is needed here to show part (2) of Corollary 3.1.

Remark 3.2 Different from the proof of Theorem 2.1, Corollary 3.1 does not require Assumption A(b)-(c). Such assumptions are needed in Theorem 2.1 to invoke the error bound property (2.23), which in turn is used to establish the key descent property of the combined primal and dual gaps (cf. (2.29)). In contrast, the analysis of BSUM/R-BSUM only involves the primal gaps, whose descent is guaranteed by the algorithms. The error bound, however, is needed below to establish linear convergence.

To show linear convergence of these algorithms, we need an additional result that bounds the size of the optimality gap.

Lemma 3.1 *We have the following estimate of the optimality gaps.*

1. For the BSUM, suppose Assumption A(a) and Assumption B hold. Then there exist positive scalars ζ and ζ' (independent of y^r) such that

$$\Delta^r \leq \zeta \|x^{r+1} - x^r\|^2 + \zeta' \|x^r - \bar{x}^r\|^2, \quad \text{for all } r \geq 1. \quad (3.48)$$

2. For the R-BSUM, suppose Assumptions A and B hold. Additionally, assume that $h_k(x_k) = \lambda_k \|x_k\|_1$ for any $\lambda_k \geq 0$, and that C_k is full row rank for each k . Then there exists a finite $t_0 > 0$ and a positive scalar $\hat{\zeta}'$ (independent of y^r) such that

$$\mathbb{E} [\Delta^t | x^t] \leq \hat{\zeta}' \|x^t - \bar{x}^t\|^2, \quad \text{for all } t \geq t_0. \quad (3.49)$$

Proof. We only show part (2) of the proof. Part (1) of the proof is much simpler, and can be found in [56, Lemma 3.1].

From the mean value theorem, for any k there exists some \tilde{x}^t in the line segment joining x^t and \bar{x}^t such that

$$g(x^t) - g(\bar{x}^t) = \langle \nabla g(\tilde{x}^t), x^t - \bar{x}^t \rangle.$$

The conditional expected value of Δ^t can be bounded above by

$$\begin{aligned}
\mathbb{E}[\Delta^t \mid x^t] &= \sum_{k=1}^K p_k (f(x_{-k}^t, \hat{x}_k^{t+1}) - f(\bar{x}^t)) \\
&\leq \sum_{k=1}^K p_k (f(x^t) - f(\bar{x}^t)) \\
&= \langle \nabla g(\tilde{x}^t), x^t - \bar{x}^t \rangle + h(x^t) - h(\bar{x}^t) \\
&= \langle \nabla g(\tilde{x}^t) - \nabla g(\bar{x}^t), x^t - \bar{x}^t \rangle + \langle \nabla g(\bar{x}^t), x^t - \bar{x}^t \rangle + h(x^t) - h(\bar{x}^t) \\
&\leq L \|\tilde{x}^t - \bar{x}^t\| \|x^t - \bar{x}^t\| + \langle \nabla g(\bar{x}^t), x^t - \bar{x}^t \rangle + h(x^t) - h(\bar{x}^t) \\
&\leq L \|x^t - \bar{x}^t\|^2 + \langle \nabla g(\bar{x}^t), x^t - \bar{x}^t \rangle + h(x^t) - h(\bar{x}^t)
\end{aligned}$$

where the last inequality comes from the fact that \tilde{x}^t lies in the line segment joining x^t and \bar{x}^t . In the following, we will show that when t is large enough, with probability 1 we have $\langle \nabla g(\bar{x}^t), x^t - \bar{x}^t \rangle + h(x^t) - h(\bar{x}^t) = 0$.

We first observe that \bar{x}^t satisfies

$$\langle \nabla g(\bar{x}^t) + \partial h(\bar{x}^t), x - \bar{x}^t \rangle \geq 0, \quad \forall x \in X.$$

This implies that

$$h(x) - h(\bar{x}^t) + \langle \nabla g(\bar{x}^t), x - \bar{x}^t \rangle \geq 0, \quad \forall x \in X.$$

Assumption A(b) implies that $\nabla g(x^*)$ takes the same value for any $x^* \in X^*$. Let us denote

$$\nabla g(x^*) = a^*, \quad \forall x^* \in X^*. \quad (3.50)$$

Then for any two optimal solutions $x^*, x^\infty \in X^*$, we have

$$\begin{aligned}
h(x^*) - h(x^\infty) + \langle \nabla g(x^\infty), x^* - x^\infty \rangle &\geq 0, \\
h(x^\infty) - h(x^*) + \langle \nabla g(x^*), x^\infty - x^* \rangle &\geq 0.
\end{aligned}$$

Using the fact that $\nabla g(x^*) = \nabla g(x^\infty)$, we conclude

$$\begin{aligned}
h(x^\infty) + \langle \nabla g(x^\infty), x^\infty \rangle &= h(x^*) + \langle \nabla g(x^\infty), x^* \rangle \\
&= h(x^*) + \langle \nabla g(x^*), x^* \rangle, \quad \forall x^*, x^\infty \in X^*.
\end{aligned} \quad (3.51)$$

The main part of the proof is to show that for t large enough, there exists an $x^\infty \in X^*$ such that

$$\langle \nabla g(x^\infty), x^t - x^\infty \rangle + h(x^t) - h(x^\infty) = 0. \quad (3.52)$$

If the above relation is true, then utilizing (3.51), we can conclude that $\langle \nabla g(\bar{x}^t), x^t - \bar{x}^t \rangle + h(x^t) - h(\bar{x}^t) = 0$.

For any given block k and any given iteration index $t > 0$, let $t(k)$ denote the last iteration such that x_k^t has been updated, i.e., $t(k) := \max\{j \mid j < t, x_k^j \neq x_k^t\}$. Then according to the way that x_k is updated, we have

$$\begin{aligned} x_k^t &= x_k^{t(k)+1} = \text{prox} \left[x_k^{t(k)+1} - \nabla u_k \left(x_k^{t(k)+1}; x^{t(k)} \right) \right] \\ &= \text{prox} \left[x_k^{t-1} - \nabla_k g \left(x^{t-1} \right) + e_k^t \right] \end{aligned} \quad (3.53)$$

where we have defined e_k^t as

$$e_k^t := \nabla_k g \left(x^{t-1} \right) - \nabla u_k \left(x_k^{t(k)+1}; x^{t(k)} \right) + x_k^{t(k)+1} - x_k^{t-1}.$$

Clearly the norm of the error term e_k^t is bounded by

$$\begin{aligned} \|e_k^t\| &= \left\| \nabla_k g \left(x^{t-1} \right) - \nabla u_k \left(x_k^{t(k)+1}; x^{t(k)} \right) + x_k^{t(k)+1} - x_k^{t-1} \right\| \\ &= \left\| \nabla u_k \left(x_k^{t(k)}; x^{t(k)} \right) - \nabla u_k \left(x_k^{t(k)+1}; x^{t(k)} \right) + \nabla_k g \left(x^{t-1} \right) - \nabla u_k \left(x_k^{t(k)}; x^{t(k)} \right) + x_k^t - x_k^{t-1} \right\| \\ &\leq L_k \|x_k^{t(k)} - x_k^{t(k)+1}\| + \left\| \nabla_k g \left(x^{t-1} \right) - \nabla u_k \left(x_k^{t(k)}; x^{t(k)} \right) \right\| + \|x_k^t - x_k^{t-1}\| \\ &= L_k \|x_k^{t(k)} - x_k^{t(k)+1}\| + \left\| \nabla_k g \left(x^{t-1} \right) - \nabla_k g \left(x^{t(k)} \right) \right\| + \|x_k^t - x_k^{t-1}\|. \end{aligned}$$

The fact that p_k is bounded away from 0 for all k implies that if $t \rightarrow \infty$, then $t(k) \rightarrow \infty$ for all k w.p.1. Thus, using the results in the second part of Corollary 3.1, we have that both $x^{t(k)}$ and x^{t-1} converge to the set of X^* w.p.1 (though they may have different limit points), and that $\|x_k^{t(k)} - x_k^{t(k)+1}\| \rightarrow 0$ and $\|x_k^t - x_k^{t-1}\| \rightarrow 0$ w.p.1. These results imply that

$$\begin{aligned} \lim_{t \rightarrow \infty} \|e_k^t\| &= \lim_{t \rightarrow \infty} L_k \|x_k^{t(k)} - x_k^{t(k)+1}\| + \left\| \nabla_k g \left(x^{t-1} \right) - \nabla_k g \left(x^{t(k)} \right) \right\| + \|x_k^t - x_k^{t-1}\| \\ &= 0 + \|a_k^* - a_k^*\| + 0 = 0, \end{aligned} \quad (3.54)$$

where a^* is defined by (3.50).

Next we show (3.52). To proceed, we need a few new definitions. Let $C_k[j]$ denote the j th row of the matrix C_k , let $c_k[j]$ denote the j th element of the vector c_k . Define \mathcal{I}_k as the set of indices contained in x_k ; \mathcal{J}_k^t as the set of indices of active constraints for block k at iteration t : $\mathcal{J}_k^t := \{j : C_k[j]x_k^t = c_k[j]\}$. Eq. (3.53) and the fact $h_k(x_k) = \lambda_k \|x_k\|_1$ imply that the optimality condition for block variable k at iteration t is given by

$$\begin{cases} -\lambda_k \leq [x_k^t - x_k^{t-1} + \nabla_k g(x^{t-1}) + C_k^T \mu_k^t - e_k^t]_i \leq \lambda_k, \forall i \in \mathcal{I}_k \\ x_k^t[i] \geq 0 \text{ if } [x_k^t - x_k^{t-1} + \nabla_k g(x^{t-1}) + C_k^T \mu_k^t - e_k^t]_i = -\lambda_k \\ x_k^t[i] \leq 0 \text{ if } [x_k^t - x_k^{t-1} + \nabla_k g(x^{t-1}) + C_k^T \mu_k^t - e_k^t]_i = \lambda_k \\ x_k^t[i] = 0 \text{ if } -\lambda_k < [x_k^t - x_k^{t-1} + \nabla_k g(x^{t-1}) + C_k^T \mu_k^t - e_k^t]_i < \lambda_k \\ C_k[j]x_k^t = c_k[j], \forall i \in \mathcal{J}_k^t, \mu_k^t[j] = 0, \forall j \notin \mathcal{J}_k^t, \mu_k^t \geq 0 \end{cases} \quad (3.55)$$

where the notation $[\cdot]_i$ denotes the i th element of a vector.

Define \mathcal{J}^t as the index set of active constraints at iteration t :

$$\mathcal{J}^t := \bigcup_{k=1}^K \mathcal{J}_k^t = \bigcup_{k=1}^K \{j : j \in \mathcal{I}_k, C_k[j]x_k^t = c_k[j], k = 1, \dots, K\}.$$

Since there are only a finite number of distinct choices of \mathcal{J}^t , it follows that there exists some \mathcal{J}^∞ such that $\mathcal{J}^t = \mathcal{J}^\infty$ for an infinite number of t . Due to the compactness of X_k , the full rankness of C_k^T , and the fact that $e^t \rightarrow 0$, it follows that for sufficiently large t , x_k^t and μ_k^t are both bounded. By further passing to a subsequence \mathcal{T} if necessary, we can assume that

$$\begin{aligned} \lim_{t \in \mathcal{T}, t \rightarrow \infty} x_k^t &= \lim_{t \in \mathcal{T}, t \rightarrow \infty} x_k^{t-1} = x_k^\infty, & \lim_{t \in \mathcal{T}, t \rightarrow \infty} \mu_k^t &= \mu_k^\infty, \\ \lim_{t \in \mathcal{T}, t \rightarrow \infty} e_k^t &= 0, & \lim_{t \in \mathcal{T}, t \rightarrow \infty} (x_k^t - x_k^{t-1}) &= 0, \quad \forall k. \end{aligned} \quad (3.56)$$

Taking limit $t \rightarrow \infty$ along \mathcal{T} , we obtain from (3.55) the following for all $k = 1, \dots, K$:

$$\begin{cases} -\lambda_k \leq [\nabla_k g(x^\infty) + C_k^T \mu_k^\infty]_i \leq \lambda_k, \quad \forall i \in \mathcal{I}_k \\ x_k^\infty[i] \geq 0 \text{ if } [\nabla_k g(x^\infty) + C_k^T \mu_k^\infty]_i = -\lambda_k \\ x_k^\infty[i] \leq 0 \text{ if } [\nabla_k g(x^\infty) + C_k^T \mu_k^\infty]_i = \lambda_k \\ x_k^\infty[i] = 0 \text{ if } -\lambda_k < [\nabla_k g(x^\infty) + C_k^T \mu_k^\infty]_i < \lambda_k \\ C_k[j]x_k^\infty = c_k[j], \quad \forall j \in \mathcal{J}_k^\infty, \quad \mu_k^\infty[j] = 0, \quad \forall j \notin \mathcal{J}_k, \quad \mu_k^\infty \geq 0 \end{cases} \quad (3.57)$$

where $\mu_k^\infty, \mathcal{J}_k^\infty$ are the corresponding components of μ^∞ and \mathcal{J}^∞ respectively.

In the following, we compare the two systems (3.55), (3.57) and show that when t becomes large enough, x^t and x^∞ will have the same sign pattern. To make this statement precise, let us define three index sets below

$$\begin{cases} \mathcal{I}_k^+ = \{i : [\nabla_k g(x^\infty) + C_k^T \mu_k^\infty]_i = -\lambda_k, i \in \mathcal{I}_k\} \\ \mathcal{I}_k^- = \{i : [\nabla_k g(x^\infty) + C_k^T \mu_k^\infty]_i = \lambda_k, i \in \mathcal{I}_k\} \\ \mathcal{I}_k^\pm = \{i : |[\nabla_k g(x^\infty) + C_k^T \mu_k^\infty]_i| < \lambda_k, i \in \mathcal{I}_k\} \end{cases} \quad (3.58)$$

We claim that for t large enough, the following *identifiability condition* is true for each $k = 1, \dots, K$:

$$x_k^t[i] \geq 0, \quad \forall i \in \mathcal{I}_k^+, \quad x_k^t[i] \leq 0, \quad \forall i \in \mathcal{I}_k^-, \quad x_k^t[i] = 0, \quad \forall i \in \mathcal{I}_k^\pm. \quad (3.59)$$

Suppose $i \in \mathcal{I}_k^+$, then we have $[\nabla_k g(x^\infty) + C_k^T \mu_k^\infty]_i = -\lambda_k$. The fact that $x_k^\infty[i] \geq 0$ implies

$$[x_k^\infty - \nabla_k g(x^\infty) - C_k^T \mu_k^\infty]_i \geq \lambda_k.$$

By (3.56), we have

$$\lim_{t \in \mathcal{T}, t \rightarrow \infty} [x_k^{t-1} - \nabla_k g(x^{t-1}) - C_k^T \mu_k^t + e_k^t]_i = [x_k^\infty - \nabla_k g(x^\infty) - C_k^T \mu_k^\infty]_i \geq \lambda_k,$$

which further implies that there exists some t_1 such that for all $t > t_1$

$$[x_k^{t-1} - \nabla_k g(x^{t-1}) - C_k^T \mu_k^t + e_k^t]_i \geq \frac{1}{2} \lambda_k > -\lambda_k.$$

If $x_k^t[i] < 0$, then it follows from the above inequality that

$$[x_k^t - x_k^{t-1} + \nabla_k g(x^{t-1}) + C_k^T \mu_k^t - e_k^t]_i < \lambda_k$$

which by (3.55) would imply $x_k^t[i] \geq 0$, a contradiction. Thus, we must have $x_k^t[i] \geq 0$ for all $t > t_1$.

Using a similar argument, we can show that there exists some t_2 such that $x_k^t[i] \leq 0$ for any $i \in \mathcal{I}_k^-$ and for all $t > t_2$. For any $i \in \mathcal{I}_k^-$, then there holds

$$x_k^\infty[i] = 0, \quad |[\nabla_k g(x^\infty) + C_k^T \mu_k^\infty]_i| < \lambda_k.$$

It follows from (3.56) that

$$\lim_{t \in \mathcal{T}, t \rightarrow \infty} |[-\nabla_k g(x^{r-1}) + x_k^{r-1} - C_k^T \mu_k^t + e_k^t]_i| = |[\nabla_k g(x^\infty) + C_k^T \mu_k^\infty]_i| < \lambda_k,$$

which further implies that there exists some t_3 such that

$$|[-\nabla_k g(x^{r-1}) + x_k^{r-1} - C_k^T \mu_k^t + e_k^t]_i| < \lambda_k, \quad \forall t \geq t_3.$$

We prove by contradiction that $x_k^t[i] = 0$ for all $t > t_3$. Specifically, if $x_k^t[i] > 0$, then the above inequality implies

$$[x_k^t - x_k^{r-1} + \nabla_k g(x^{r-1}) + C_k^T \mu_k^t + e_k^t]_i > x_k^t[i] - \lambda_k > -\lambda_k,$$

which by (3.55) further implies that $x_k^t[i] \leq 0$, a contradiction. Similarly, $x_k^t[i]$ cannot be negative either. Thus, we have $x_k^t[i] = 0$ for all $i \in \mathcal{I}_k^-$ and all $t \geq t_3$. This completes the proof of the identifiability property (3.59) for all k .

Now we are ready to show (3.52). Assume that t is large enough such that the identifiability condition (3.59) is true. Suppose $i \in \mathcal{I}_k^+$, we have

$$\begin{aligned} [\nabla_k g(x^\infty)]_i [x_k^t - x_k^\infty]_i &= (-\lambda_k - [C_k^T \mu_k^\infty]_i) [x_k^t - x_k^\infty]_i \\ &= (-\lambda_k - C_k^T [i] \mu_k^\infty) [x_k^t - x_k^\infty]_i \\ &= -\lambda_k [x_k^t - x_k^\infty]_i - \langle \mu_k^\infty, (C_k^T [i])^T ([x_k^t - x_k^\infty]_i) \rangle. \end{aligned}$$

Similarly, we have

$$\begin{aligned} [\nabla_k g(x^\infty)]_i [x_k^t - x_k^\infty]_i &= \lambda_k [x_k^t - x_k^\infty]_i - \langle \mu_k^\infty, (C_k^T [i])^T ([x_k^t - x_k^\infty]_i) \rangle, \quad \forall i \in \mathcal{I}_k^-, \\ [\nabla_k g(x^\infty)]_i [x_k^t - x_k^\infty]_i &= -\langle \mu_k^\infty, (C_k^T [i])^T ([x_k^t - x_k^\infty]_i) \rangle = 0, \quad \forall i \in \mathcal{I}_k^-. \end{aligned}$$

Using the above relations, we obtain for t large enough,

$$\begin{aligned}
\langle \nabla g(x^\infty), x^t - x^\infty \rangle + h(x^t) - h(x^\infty) &= \sum_{k=1}^K \sum_{i \in \mathcal{I}_k} ([\nabla_k g(x^\infty)]_i [x_k^t - x_k^\infty]_i + \lambda_k (|x_k^t[i]| - |x_k^\infty[i]|)) \\
&= \sum_{k=1}^K \sum_{i \in \mathcal{I}^+} -\lambda_k [x_k^t - x_k^\infty]_i + \sum_{k=1}^K \sum_{i \in \mathcal{I}^-} \lambda_k [x_k^t - x_k^\infty]_i \\
&\quad - \sum_{k=1}^K \langle \mu_k^\infty, C_k(x_k^t - x_k^\infty) \rangle + \lambda_k (|x_k^t[i]| - |x_k^\infty[i]|) \\
&= \sum_{k=1}^K \sum_{i \in \mathcal{I}^+} -\lambda_k [x_k^t - x_k^\infty]_i + \sum_{k=1}^K \sum_{i \in \mathcal{I}^-} \lambda_k [x_k^t - x_k^\infty]_i \\
&\quad + \sum_{k=1}^K \sum_{i \in \mathcal{I}^+} \lambda_k [x_k^t - x_k^\infty]_i + \sum_{k=1}^K \sum_{i \in \mathcal{I}^-} -\lambda_k [x_k^t - x_k^\infty]_i \\
&= 0.
\end{aligned}$$

Note that the second to the last equality is due to $\mathcal{J}^t = \mathcal{J}^\infty$ for all $t \in \mathcal{T}$, which implies

$$C_k[j]x_k^\infty = c_k, \quad \forall j \in \mathcal{J}_k, \quad \mu_k^\infty[j] = 0, \quad \forall j \notin \mathcal{J}_k,$$

so that $\sum_{k=1}^K \langle \mu_k^\infty, C_k(x_k^t - x_k^\infty) \rangle = 0$. Using (3.51), we obtain the desired result. This completes the proof for part (2) of the lemma. **Q.E.D.**

We remark that the identifiability property (3.59) has been observed numerically by Richtárik and Takáč in [31, Section 6.1.7] when using a randomized block coordinate descent method to solve a certain ℓ_1 -minimization problem. Here in the proof of Lemma 2.3, we have established this property theoretically.

Next we use Lemma 2.3 to show that both the BSUM and R-BSUM converge linearly.

Theorem 3.1 *Suppose Assumptions A and B hold. Then we have the following:*

1. *For the BSUM algorithm, the sequence $\{\Delta^r\}$ vanishes Q -linearly. The same conclusion is true if the compactness assumption A(c) is replaced with the compactness of the level set $X^1 := \{x \mid f(x) \leq f(x^1)\}$.*
2. *For the R-BSUM algorithm, assume that the nonsmooth part has the form $h_k(x_k) = \lambda_k \|x_k\|_1$ for some $\lambda_k \geq 0$, and that C_k has full row rank for each k . Then $\{\mathbb{E}[\Delta^t]\}$ vanishes Q -linearly.*

Proof. We first show part (1) of the claim. By directly adapting the proof of Lemma 2.3-(1), we

can show the following sufficient descent

$$\Delta^r - \Delta^{r-1} \leq -\gamma \|x^{r+1} - x^r\|^2,$$

where $\gamma = \min_k \gamma_k$. This implies that $x^r \in X^1$ for all $r \geq 1$. By (3.48) in Lemma 3.1, we have that for all $r \geq 1$

$$\begin{aligned} \Delta^r &\leq \zeta \|x^{r+1} - x^r\|^2 + \zeta' \|x^r - \bar{x}^r\|^2 \\ &\leq \zeta \|x^{r+1} - x^r\|^2 + \zeta' \tau^2 \|\tilde{\nabla} f(x^r)\|^2 \\ &\leq (\zeta + \zeta' \tau^2 \sigma^2) \|x^{r+1} - x^r\|^2 \end{aligned} \quad (3.60)$$

where the last inequality is obtained by specializing Lemma 2.4-(1) to the BSUM algorithm. Note that due to the compactness of either the feasible set X or the level set X^1 , the second inequality, which uses the error bound condition in Lemma 2.2, holds true for all $r \geq 1$. Combining the previous two results, we have

$$\Delta^r - \Delta^{r-1} \leq -\frac{\gamma}{(\zeta + \zeta' \tau^2 \sigma^2)} \Delta^r, \quad \text{for all } r \geq 1.$$

Define $\lambda := \frac{\gamma}{(\zeta + \zeta' \tau^2 \sigma^2)}$, we conclude that Δ^r converges Q -linearly, that is,

$$0 \leq \Delta^r \leq \frac{1}{1 + \lambda} \Delta^{r-1} \quad \text{for all } r \geq 1.$$

It remains to show part (2) of the theorem. By adapting the proof of Lemma 2.3-(2) and taking full expectation, we have

$$\mathbb{E}[\Delta^t - \Delta^{t-1}] \leq -\hat{\gamma} \mathbb{E}[\|\hat{x}^{t+1} - x^t\|^2]. \quad (3.61)$$

where $\hat{\gamma} = \min_k p_k \gamma_k$, Using (2.8) in Lemma 3.1, there exists a $t_0 > 0$ such that for all $t > t_0$, the following is true

$$\mathbb{E}[\Delta^t | x^t] \leq \hat{\zeta} \|x^t - \bar{x}^t\|^2 \leq \hat{\zeta} \tau^2 \|\tilde{\nabla} f(x^t)\|^2 \leq \hat{\zeta} \tau^2 \hat{\sigma}^2 \|\hat{x}^{t+1} - x^t\|^2, \quad \text{w.p.1.},$$

where the last inequality is obtained by specializing Lemma 2.4-(2) to the R-BCD algorithm. Taking full expectation, we obtain

$$\mathbb{E}[\Delta^t] \leq \hat{\zeta} \tau^2 \hat{\sigma}^2 \mathbb{E}[\|\hat{x}^{t+1} - x^t\|^2], \quad \forall t \geq t_0.$$

Combining this with (3.61) yields

$$\mathbb{E}[\Delta^t - \Delta^{t-1}] \leq -\frac{\hat{\gamma}}{\hat{\zeta} \tau^2 \hat{\sigma}^2} \mathbb{E}[\Delta^t], \quad \forall t \geq t_0.$$

Define $\hat{\lambda} := \frac{\hat{\gamma}}{\zeta\tau^2\delta^2}$, we conclude that there exists a $t_0 > 0$ such that

$$0 \leq \mathbb{E}[\Delta^t] \leq \frac{1}{1 + \hat{\lambda}} \mathbb{E}[\Delta^{t-1}], \quad \forall t \geq t_0.$$

implying that $\mathbb{E}[\Delta^t]$ vanishes Q-linearly.

Q.E.D.

Recently the authors of [28] have shown that the cyclic BCD algorithm converges R-linearly under assumptions similar to Assumption A, except that the compactness assumption (Assumption A(c)) is not required. Compared with [28], the new elements in part (1) of Theorem 3.1 are: (i) the cyclic BCD algorithm converges Q-linearly when the feasible set or the level set is compact; (ii) The same rate can be obtained when the per-block problem is minimized approximately by working with the approximate function $u_k(\cdot; \cdot)$.

4 Numerical Results

In this section, we report numerical results that illustrate the effectiveness of the BSUM-M for large practical problems.

4.1 Solving a Linear System of Equations

Recently, the authors of [57] have demonstrated via a counterexample, that the classic two-block ADMM algorithm could diverge when applied to solve problems with three or more blocks. In the counterexample, ADMM is used to solve the following linear systems of equations (which has a unique solution $x_1 = x_2 = x_3 = 0$)

$$E_1x_1 + E_2x_2 + E_3x_3 = 0, \tag{4.62}$$

$$\text{with } [E_1 \ E_2 \ E_3] = \begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 2 \\ 1 & 2 & 2 \end{bmatrix}. \tag{4.63}$$

It is shown in [57] that regardless of the starting point, the ADMM algorithm always diverges. However, we have shown in this paper that the BSUM-M is guaranteed to obtain the unique solution of the above linear system of equations¹. The following special version of the BSUM-M

¹Since the $[E_1 \ E_2 \ E_3]$ is full rank, it follows that the augmented Lagrangian function is strongly convex and therefore the global error bound condition holds and the compactness assumption is not needed.

iteration for solving (4.62) has the same iteration as the ADMM except for a different dual stepsize.

$$\begin{aligned} y^{r+1} &= y^r + \alpha^r (E_1 x_1^r + E_2 x_2^r + E_3 x_3^r) \\ x_1^{r+1} &= (E_1^T E_1)^{-1} (-E_1^T E_2 x_2^r - E_1^T E_3 x_3^r - E_1^T y^{r+1}/\rho) \\ x_2^{r+1} &= (E_2^T E_2)^{-1} (-E_2^T E_1 x_1^{r+1} - E_2^T E_3 x_3^r - E_2^T y^{r+1}/\rho) \\ x_3^{r+1} &= (E_3^T E_3)^{-1} (-E_3^T E_1 x_1^{r+1} - E_3^T E_2 x_2^{r+1} - E_3^T y^{r+1}/\rho). \end{aligned}$$

In our experiment, we choose $\rho = 1$ and $\alpha^r = \rho \times \frac{1}{\sqrt{r}}$. We run the BSUM-M and RBSUM-M for 1000 trials, and for each trial we initialize the components in x and y uniformly randomly from $[-10, 10]$. For the RBSUM-M algorithm, the primal and dual blocks are picked with equal probability at each iteration. We see from Figs. 1–2 below that in all the trials both algorithms converge nicely. The RBSUM-M takes longer time to converge, because at each iteration only a single primal or dual variable is updated.

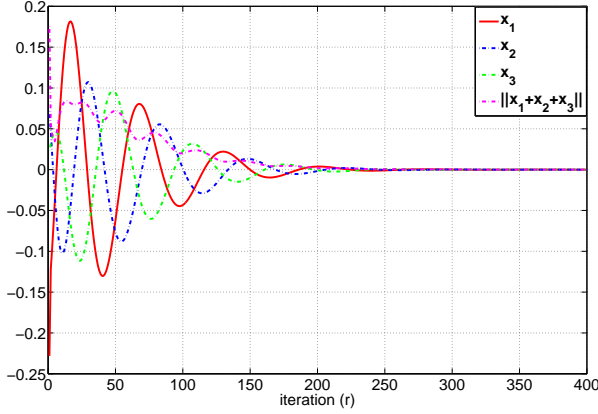


Figure 1: Iterates generated by the BSUM-M for solving (4.62). Each curve is averaged over 1000 runs (with random starting points).

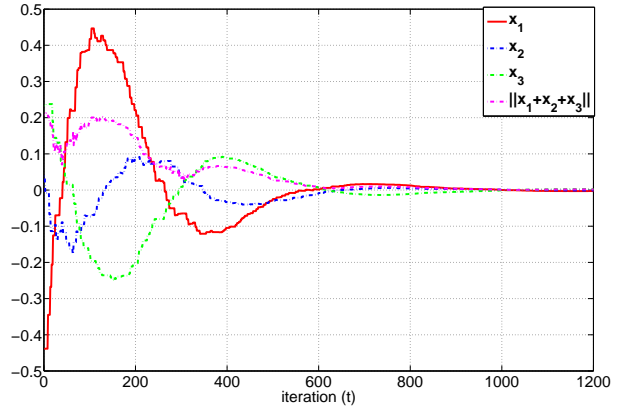


Figure 2: Iterates generated by the RBSUM-M algorithm for solving (4.62). Each curve is averaged over 1000 runs (with random starting points).

4.2 The BP Problem

In the second experiment, we consider the BP problem (1.3), and fix each block variable x_k to be a *scalar*. Then the primal subproblem for the BSUM-M at the r -th iteration for the k -th variable is given by

$$\min_{x_k} \frac{1}{\rho \|e_k\|^2} |x_k| + \frac{1}{2} \left(x_k + \frac{e_k^T c_k^r}{\|e_k\|^2} \right)^2 \quad (4.64)$$

where e_k is the k -th column of E , $c_k^r = w_{-k}^r + y^{r+1}/\rho - q$. This problem can be solved in closed-form by the soft-thresholding operator. It is worth noting that the update for each component

n	m	p	BSUM-M	RBSUM-M	PALM	DALM	FISTA
10000	3000	0.06	226	796	948	840	768
10000	3000	0.01	74	418	370	374	584
10000	5000	0.06	144	670	542	604	618
10000	5000	0.01	64	422	188	234	582

Table 1: Average #MVM performance for different algorithms.

variable x_k only requires one piece of data e_k . Therefore in situations where the data matrix is only partially available at each update iteration [14, 31], the randomized BSUM-M algorithm can be very valuable.

We randomly generate the matrix $E \in \mathfrak{R}^{m \times n}$ and the true solutions \bar{x} with each of their nonzero components following standard Gaussian distribution. We let E be a dense matrix, and \bar{x} be a sparse vector, with each component having probability $p \in (0, 1)$ to be nonzero (see [70] for details). We normalize the columns of E to have norm 1. We have used the following stepsize rule for the BSUM-M and the RBSUM-M: $\rho = 10 \times m / \|q\|_1$, $\alpha^r = \rho \frac{10+1}{\sqrt{r+10}}$. Unless specified explicitly, the blocks in the RBSUM-M are chosen uniformly with $p_k = \frac{1}{K+1}$ for all $k = 0, \dots, K$. The BSUM-M and the RBSUM-M are compared with a number of well-known algorithms for BP such as DALM, PALM [47] and FISTA [63]; see [70] for a detailed review and implementation of these algorithms. In particular, for PALM, the primal and dual stepsizes are set equally to $10 \times m / \|q\|_1$; for DALM, the primal and dual stepsizes are set to $0.1 \times \|q\|_1 / m$; for FISTA, backtrack line search is used (these are the default settings in the package [70]).

We first consider a relatively small problem. The stopping criteria for all the algorithms is that either the iteration counter is larger than 1000, or the relative error $\|x^r - \bar{x}\| / \|\bar{x}\| \leq 10^{-10}$. Fig. 3 shows the convergence behavior of all the algorithms for one instance of the problem with $n = 10000$, $m = 3000$ and $p = 0.06$. For ease of exposition, in this figure each iteration of the RBSUM-M consists of 10000 random update steps. In Table 1, we show the averaged performance (over 100 problem realizations) for different algorithms. For a fair comparison of the computational cost, the algorithms are compared according to the number of matrix-vector multiplications, denoted by #MVM, which includes both Ex and $E^T y$ (see e.g., [47] for a similar definition). Clearly the BSUM-M approach exhibits superior performance over all other algorithms.

It is worth mentioning that except for BSUM-M and RBSUM-M, all the rest of algorithms suffer from pitfalls that prevent them from solving really large problems. For example the PALM requires the knowledge of $\rho(E^T E)$ (the largest eigenvalue of $E^T E$), the version of DALM with convergence guarantee requires the inversion of EE^T [47], both of which are difficult operations when E is large (say when n and m are larger than 10^6). The FISTA algorithm either needs $\rho(E^T E)$, or is required to perform backtrack line search within each iteration [63], both of which are again difficult to implement for large size problems. In contrast, each step of the BSUM-M and RBSUM-M is

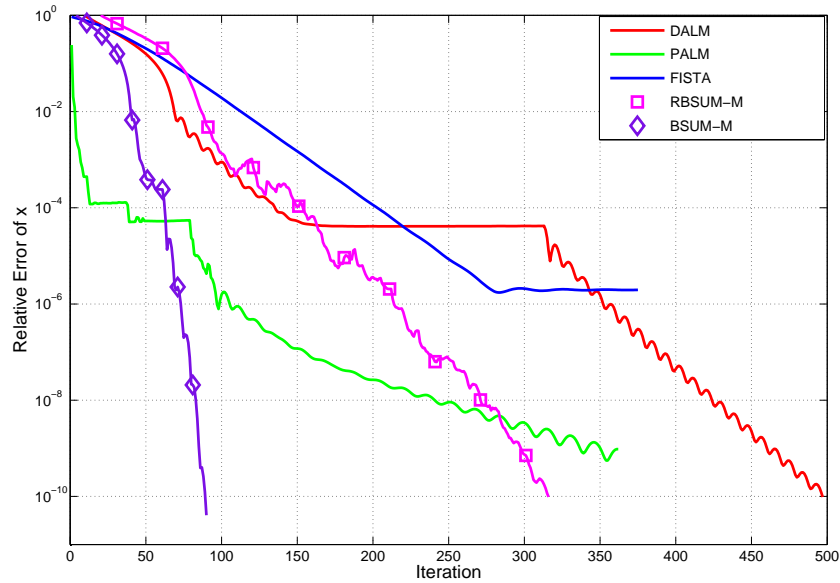


Figure 3: Relative error performance for all algorithms on a small-size problem. $n = 10000$, $m = 3000$, $p = 0.06$. The relative error is given by $\|\bar{x} - x^r\|/\|\bar{x}\|$.

# of iterations r	Exp. 1	Exp. 2
1	1	1
5	0.35	0.35
10	0.0012	0.16
15	7e-6	2e-3
20	N/A	1e-5
25	N/A	8e-7

Table 2: Relative error performance of BSUM-M for large-scale problem.

simple and in closed-form, which makes it easily scalable for large problems. We have also tested the BSUM-M on two large experiments ²: experiment 1 with $n = 10^6$, $m = 10^3$ and $\|\bar{x}\|_0 = 28$; experiment 2 with $n = 10^6$, $m = 2 \times 10^3$ and $\|\bar{x}\|_0 = 82$. It takes 7 GB and 14 GB of memory space to store the data of these problems, respectively. For both problems, the BSUM-M and RBSUM-M perform quite well: for the first (resp. the second) experiment they take around 15 iterations and about 60 seconds (resp. 25 iterations and 200 seconds) to reduce the relative error to about 10^{-6} .

4.3 The LASSO Problem

In this section, we solve the LASSO problem

$$\min_x \|Ax - b\| + \lambda \|x\|_1 \quad (4.65)$$

²We use a PC with 128 GB RAM and 24 Intel Xeon 2.67 GHz cores.

# of iterations t ($\times 10^6$)	Exp. 1	Exp. 2
1	1	1
5	0.05	0.18
10	1e-4	0.002
15	2e-7	0.0019
20	N/A	0.0028
25	N/A	6e-5
30	N/A	9e-7

Table 3: Relative error performance of RBSUM-M for large-scale problem.

p_b	p_A	R-BCD ($\alpha = 0.5$)	R-BCD ($\alpha = 0$)	BCD
0.01	0.1	68	212	30
0.1	0.1	386	784	334
0.1	0.01	376	444	1242
0.05	0.01	180	294	529

Table 4: Average #MVM performance for different algorithms, with $n = 2000$, $m = 1000$.

using R-BCD and BCD, which are special cases of RBSUM-M and BSUM-M, respectively. For both algorithms each block variable again consists of a single scalar (i.e., $n_k=1$), so that no approximation is needed, and the per-block subproblem has a closed-form solution. Here our goal is not to establish the superiority of BCD-based algorithms in solving this type of problem (we refer the interested readers to [66], [31] for comprehensive numerical studies for such purpose). Rather, we wish to demonstrate that R-BCD may sometimes outperform the cyclic BCD and vice versa.

We use the instance generator proposed in [71, Section 6] to generate the problem data. After choosing the sparsity level for A and b , the generator generates A , b , x^* . We use p_A (resp. p_b) to denote the probability for which each element of A (resp. b) is nonzero. We also use the following formula to choose the update probability p_k for each block k [14, 31]

$$p_k = \frac{L_k^\alpha}{\sum_k L_k^\alpha}, \quad 0 \leq \alpha \leq 1. \quad (4.66)$$

We first let $n = 2000$, $m = 1000$. The stopping criteria for both algorithms is that either the iteration counter is larger than 2000, or the relative error $\|x^r - \bar{x}\|/\|\bar{x}\| \leq 10^{-10}$. In Table 4, we show the performance for R-BCD and BCD with different combinations of p_b and p_A . Each entry in the table is an average of the results over 100 realizations of the problem data. First we observe that using $0 < \alpha \leq 1$ improves the convergence significantly compared with uniform sampling (i.e., $\alpha = 0$). Second, we see that R-BCD performs better when the data matrix is sparse ($p_A = 0.01$), while the cyclic BCD outperforms R-BCD for the rest of the cases.

Next we consider the scenario where $n = \{50000, 30000, 10000\}$, $m = 10000$, $p_A = \{0.01, 0.001\}$ and $p_b = 0.016$, and use the uniform sampling for the R-BCD algorithm. We plot the relative

errors for both algorithms in Figs. 4–5. In these figures we have again condensed n random update steps for the R-BCD into a single iteration, so that the iteration numbers of the two algorithms are comparable. Comparing these two figures, we observe that both algorithms achieve better performance when the data matrix is sparser. In particular, the R-BCD converges faster than the cyclic BCD when $p_A = 0.001$ and when n/m is relatively small. Its performance degrades when n/m becomes large. When $n = 50000$, the R-BCD does not show sign of convergence within the first few thousands of iterations. This observation was also noted in [31, Section 6.14] where the authors show that in a similar setting, it takes about 20000 iterations (in each iteration all variables are updated once) for the R-BCD to converge to a reasonable solution. What is probably surprising here is that when $n = 50000$, the cyclic BCD performs quite well compared to the randomized version.

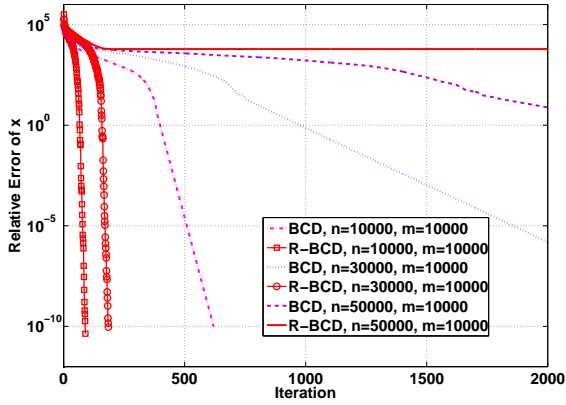


Figure 4: Relative error performance for R-BCD and BCD . $p_A = 0.001$, $p_b = 0.016$, $n = \{10000, 30000, 50000\}$, $m = 10000$.

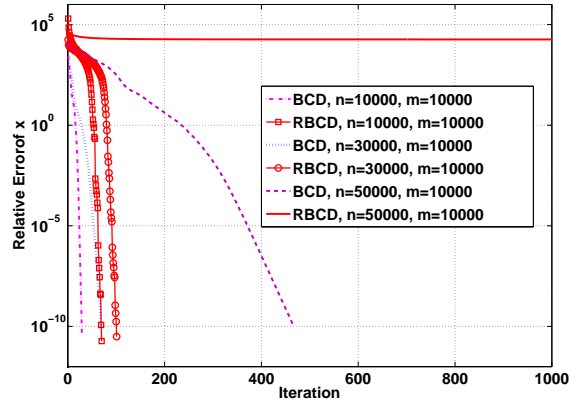


Figure 5: Relative error performance for R-BCD and BCD . $p_A = 0.01$, $p_b = 0.016$, $n = \{10000, 30000, 50000\}$, $m = 10000$.

4.4 The DR Problem

Let us now test the BSUM-M on the DR problem described in (1.5). Suppose that there are up to 3000 users in the system with each user having 4 controllable appliances; also assume that each day is divided into 96 time periods. That is, $m = 96$ and $n_k = 96 \times 4$. The load model is generated according to [6], and the detailed construction of the matrices $\{\Psi_k\}_{k=1}^K$ can be found in [4]. For simplicity, we assume that the day-ahead bidding is completed, with power supply \mathbf{p} determined by an average of 5 random generation of all the uncontrolled consumptions of the users. This reduces problem (1.5) to having only $\{\mathbf{x}_k\}_{k=1}^K$ and \mathbf{z} as optimization variables. Additionally, we let $C_p(\cdot)$ and $C_s(\cdot)$ take the form of quadratic functions.

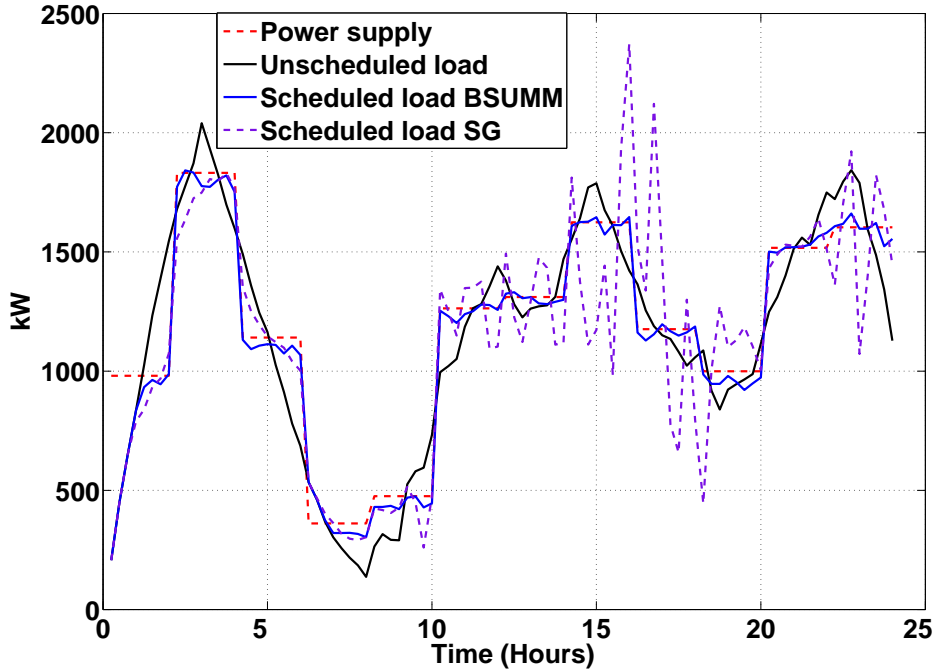


Figure 6: The unscheduled consumption, power supply and the scheduled consumption by the BSUM-M and the subgradient algorithm.

Algorithm	$K = 50$	$K = 100$	$K = 500$	$K = 1000$	$K = 3000$
BSUM-M	0.4860	0.8099	3.3964	4.648	14.827
SG	0.9519	1.5630	9.4835	16.595	60.896
Unscheduled	1.0404	1.7940	7.5749	14.389	45.900

Table 5: Total Cost Performance of Different Approaches (10^3 unit price).

We compare the BSUM-M with the dual subgradient (SG) algorithm [4]³. We let both algorithms run 200 iterations. Note that each iteration of the SG is computationally more expensive, as it involves solving a linear program [4], while each iteration of the BSUM-M is again in closed-form. In Table 5, we compare the total costs of the scheduled loading solutions generated by the BSUM-M and the SG with that of unscheduled loads. Clearly the BSUM-M is able to achieve about 50% of cost reduction, while the SG algorithm fails to converge within 200 iterations which results in significantly larger costs. In Fig. 6, we plot the power supply, the consumption levels of unscheduled loads as well as those scheduled by the BSUM-M and the SG. We can see that the BSUM-M can track the supply curve quite well, while the SG fails to do so within 200 iterations.

³Note that here the dual SG is applied to the DR with quadratic costs, whereas the reference [4] employed *linear* costs.

5 Conclusion

In this paper, we propose a first order primal-dual method for nonsmooth convex minimization problems subject to linear constraints. The new algorithm, which we call the block successive upper-bound minimization method of multipliers (BSUM-M), alternates between simple primal and dual steps either randomly or deterministically, and is well suited for large scale applications involving big data. In the primal steps, certain locally tight upper-bounds of the augmented Lagrangian function are successively minimized, while the dual step is in closed form and follows an approximate dual ascent step. The algorithm is a generalization of the ADMM method and the BCD method in that it offers greater flexibility both in choosing a suitable upper-bound function in place of the augmented Lagrangian function when performing the primal update, and in the order of primal-dual updates. We have established the convergence of the BSUM-M algorithm (for both the deterministic and randomized versions) and have demonstrated their strong numerical performance for large scale realistic applications. In future, it will be interesting to study if the BSUM-M can converge to a local stationary point for nonconvex problems, and if so, how effective it is in practical applications.

References

- [1] M. Razaviyayn, M. Hong, and Z.-Q. Luo, “A unified convergence analysis of block successive minimization methods for nonsmooth optimization,” *SIAM Journal on Optimization*, vol. 23, no. 2, pp. 1126–1153.
- [2] S. S. Chen, D. L. Donoho, and M. A. Saunders, “Atomic decomposition by basis pursuit,” *SIAM Journal on Scientific Computing*, vol. 20, no. 1, pp. 33–61, 1998.
- [3] M. Alizadeh, X. Li, Z. Wang, A. Scaglione, and R. Melton, “Demand side management in the smart grid: Information processing for the power switch,” *IEEE Signal Processing Magazine*, vol. 59, no. 5, pp. 55–67, 2012.
- [4] T.-H. Chang, M. Alizadeh, and A. Scaglione, “Coordinated home energy management for real-time power balancing,” in *Proc. IEEE PES General Meeting*, July 2012, pp. 1–8.
- [5] N. Li, L. Chen, and S. H. Low, “Optimal demand response based on utility maximization in power networks,” in *IEEE PES General Meeting*, 2011, pp. 1–8.
- [6] J. V. Paatero and P. D. Lund, “A model for generating household electricity load profiles,” *International Journal on Energy Research*, vol. 20, pp. 273–290, 2006.

- [7] Q. Zhao and B. M. Sadler, “A survey of dynamic spectrum access,” *IEEE Signal Processing Magazine*, , no. 5, pp. 79–89, 2007.
- [8] G. Scutari and D.P. Palomar, “MIMO cognitive radio: A game theoretical approach,” *IEEE Transactions on Signal Processing*, vol. 58, no. 2, pp. 761–780, 2010.
- [9] FCC, “In the matter of facilitating opportunities for flexible, efficient and reliable spectrum use employing Cognitive Radio technologies,” Dec 2003, ET Docket No. 03-108.
- [10] M. Hong and A. Garcia, “Equilibrium pricing of interference in cognitive radio networks,” *IEEE Transactions on Signal Processing*, vol. 59, no. 12, pp. 6058–6072, 2011.
- [11] Friedman J, Hastie T, and Tibshirani R., “Regularization paths for generalized linear models via coordinate descent,” *Journal of Statistical Software*, vol. 33, no. 1, pp. 1–22, 2010.
- [12] A. Saha and A. Tewari, “On the nonasymptotic convergence of cyclic coordinate descent method,” *SIAM Journal on Optimization*, vol. 23, no. 1, pp. 576–601, 2013.
- [13] S. Shalev-Shwartz and A. Tewari, “Stochastic methods for ℓ_1 regularized loss minimization,” *Journal of Machine Learning Research*, vol. 12, pp. 1865–1892, 2011.
- [14] Y. Nesterov, “Efficiency of coordinate descent methods on huge-scale optimization problems,” *SIAM Journal on Optimization*, vol. 22, no. 2, pp. 341–362, 2012.
- [15] P. Tseng, “Convergence of a block coordinate descent method for nondifferentiable minimization,” *Journal of Optimization Theory and Applications*, vol. 103, no. 9, pp. 475–494, 2001.
- [16] D. P. Bertsekas and J. N. Tsitsiklis, *Neuro-Dynamic Programming*, Athena Scientific, Belmont, MA, 1996.
- [17] D. P. Bertsekas and J. N. Tsitsiklis, *Parallel and Distributed Computation: Numerical Methods, 2nd ed*, Athena Scientific, Belmont, MA, 1997.
- [18] J. M. Ortega and W. C. Rheinboldt, *Iterative Solution of Nonlinear Equations in Several Variables*, Academic Press, 1972.
- [19] L. Grippo and M. Sciandrone, “On the convergence of the block nonlinear Gauss-Seidel method under convex constraints,” *Operations Research Letters*, vol. 26, pp. 127–136, 2000.
- [20] P. Tseng and S. Yun, “A coordinate gradient descent method for nonsmooth separable minimization,” *Mathematical Programming*, vol. 117, pp. 387–423, 2009.
- [21] H. Zhang, J. Jiang, and Z.-Q. Luo, “On the linear convergence of a proximal gradient method for a class of nonsmooth convex minimization problems,” *Journal of the Operations Research Society of China*, vol. 1, no. 2, pp. 163–186, 2013.

- [22] A. Beck and L. Tretuashvili, “On the convergence of block coordinate descent type methods,” *SIAM Journal on Optimization*, vol. 23, no. 4, pp. 2037–2060, 2013.
- [23] Z.-Q. Luo and P. Tseng, “Error bounds and convergence analysis of feasible descent methods: a general approach,” *Annals of Operations Research*, vol. 46-47, pp. 157–178, 1993.
- [24] Z.-Q. Luo and P. Tseng, “On the convergence of the coordinate descent method for convex differentiable minimization,” *Journal of Optimization Theory and Application*, vol. 72, no. 1, pp. 7–35, 1992.
- [25] Z.-Q. Luo and P. Tseng, “On the linear convergence of descent methods for convex essentially smooth minimization,” *SIAM Journal on Control and Optimization*, vol. 30, no. 2, pp. 408–425, 1992.
- [26] Z.-Q. Luo and P. Tseng, “On the convergence rate of dual ascent methods for strictly convex minimization,” *Mathematics of Operations Research*, vol. 18, no. 4, pp. 846–867, 1993.
- [27] P. Tseng, “Approximation accuracy, gradient methods, and error bound for structured convex optimization,” *Mathematical Programming*, vol. 125, no. 2, pp. 263–295, 2010.
- [28] M. Sanjabi, M. Kадkhodaei, and Z.-Q. Luo, “On the linear convergence of approximate proximal splitting methods for non-smooth convex minimization,” 2012, manuscript.
- [29] K. Hou, Z. Zhou, A. M.-S. So, and Z.-Q. Luo, “On the linear convergence of the proximal gradient method for trace norm regularization,” in *Neural Information Processing Systems (NIPS)*, 2014.
- [30] M. Hong, X. Wang, M. Razaviyayn, and Z.-Q. Luo, “Iteration complexity analysis for a family of block coordinate descent method for convex nonsmooth optimization,” 2013, Technical Report, University of Minnesota.
- [31] P. Richtarik and M. Takac, “Iteration complexity of randomized block-coordinate descent methods for minimizing a composite function,” *Mathematical Programming*, pp. 1–38, 2012.
- [32] M. V. Solodov, “On the convergence of constrained parallel variable distribution algorithms,” *SIAM J. on Optimization*, vol. 8, no. 1, pp. 187–196, 1998.
- [33] S. Boyd, N. Parikh, E. Chu, B. Peleato, and J. Eckstein, “Distributed optimization and statistical learning via the alternating direction method of multipliers,” *Foundations and Trends in Machine Learning*, vol. 3, 2011.
- [34] J. Eckstein and D. P. Bertsekas, “On the douglas-rachford splitting method and the proximal point algorithm for maximal monotone operators,” *Mathematical Programming*, vol. 55, no. 1, pp. 293–318, 1992.

- [35] R. Glowinski and A. Marrocco, “Sur l’approximation par éléments finis et la résolution par pénalisation-dualité d’une classe de problèmes de dirichlet non linéaires,” *RAIRO*, vol. R-2, pp. 41–76, 1975.
- [36] R. Glowinski and P. Le Tallec, “Augmented lagrangian and operator splitting methods in nonlinear mechanics,” *SIAM Studies in Applied Mathematics*, pp. 41–76, 1989.
- [37] D. Gabay and B. Mercier, “A dual algorithm for the solution of nonlinear variational problems via finite element approximation,” *Computers & Mathematics with Applications*, vol. 2, pp. 17–40, 1976.
- [38] W. Yin, S. Osher, D. Goldfarb, and J. Darbon, “Bregman iterative algorithms for l_1 -minimization with applications to compressed sensing,” *SIAM Journal on Imaging Science*, vol. 1, no. 1, pp. 143–168, Mar. 2008.
- [39] J. Yang, Y. Zhang, and W. Yin, “An efficient tvl1 algorithm for deblurring multichannel images corrupted by impulsive noise,” *SIAM Journal on Scientific Computing*, vol. 31, no. 4, pp. 2842–2865, 2009.
- [40] X. Zhang, M. Burger, and S. Osher, “A unified primal-dual algorithm framework based on bregman iteration,” *Journal of Scientific Computing*, vol. 46, no. 1, pp. 20–46, 2011.
- [41] K. Scheinberg, S. Ma, and D. Goldfarb, “Sparse inverse covariance selection via alternating linearization methods,” in *Twenty-Fourth Annual Conference on Neural Information Processing Systems (NIPS)*, 2010.
- [42] M. Tao and X.M. Yuan, “Recovering low-rank and sparse components of matrices from incomplete and noisy observations,” *SIAM Journal on Optimization*, vol. 21, no. 1, pp. 57–81, 2011.
- [43] D. P. Bertsekas, *Nonlinear Programming, 2nd ed*, Athena Scientific, Belmont, MA, 1999.
- [44] S. Boyd and L. Vandenberghe, *Convex Optimization*, Cambridge University Press, 2004.
- [45] A. Nedic and A. Ozdaglar, “Cooperative distributed multi-agent optimization,” in *Convex Optimization in Signal Processing and Communications*. Cambridge University Press, 2009.
- [46] D. P. Bertsekas, *Constrained Optimization and Lagrange Multiplier Method*, Academic Press, 1982.
- [47] J. Yang and Y. Zhang, “Alternating direction algorithms for l_1 -problems in compressive sensing,” *SIAM Journal on Scientific Computing*, pp. 250–278, 2011.

- [48] J. Eckstein, “Some saddle-function splitting methods for convex programming,” *Optimization Methods and Software*, vol. 4, no. 1, pp. 75–83, 1994.
- [49] B. He, L. Liao, D. Han, and H. Yang, “A new inexact alternating directions method for monotone variational inequalities,” *Mathematical Programming*, vol. 92, no. 1, pp. 103–118, 2002.
- [50] X. Wang and X. Yuan, “The linearized alternating direction method of multipliers for dantzig selector,” *SIAM Journal on Scientific Computing*, vol. 34, no. 5, pp. 2792–2811, 2012.
- [51] B. He and X. Yuan, “On the $o(1/n)$ convergence rate of the douglas-rachford alternating direction method,” *SIAM Journal on Numerical Analysis*, vol. 50, no. 2, pp. 700–709, 2012.
- [52] R. Monteiro and B. Svaiter, “Iteration-complexity of block-decomposition algorithms and the alternating direction method of multipliers,” *SIAM Journal on Optimization*, vol. 23, no. 1, pp. 475–507, 2013.
- [53] T. Goldstein, B. O’Donoghue, and S. Setzer, “Fast alternating direction optimization methods,” *UCLA CAM technical report*, 2012.
- [54] D. Boley, “Linear convergence of ADMM on a model problem,” *SIAM Journal on Optimization*, 2013, to appear.
- [55] W. Deng and W. Yin, “On the global linear convergence of alternating direction methods,” *Preprint*, 2012.
- [56] M. Hong and Z.-Q. Luo, “On the linear convergence of the alternating direction method of multipliers,” *arXiv preprint arXiv:1208.3922*, 2012, submitted to Mathematical Programming.
- [57] C. Chen, B. He, X. Yuan, and Y. Ye, “The direct extension of admm for multi-block convex minimization problems is not necessarily convergent,” 2013, preprint.
- [58] B. He, M. Tao, and X. Yuan, “Alternating direction method with gaussian back substitution for separable convex programming,” *SIAM Journal on Optimization*, pp. 313–340, 2012.
- [59] X. Wang, M. Hong, S. Ma, and Z.-Q. Luo, “Solving multiple-block separable convex minimization problems using two-block alternating direction method of multipliers,” 2013, submitted.
- [60] S. Ma, “Alternating proximal gradient method for convex minimization,” 2012, preprint.
- [61] S.J. Wright, R.D. Nowak, and M.A.T. Figueiredo, “Sparse reconstruction by separable approximation,” *IEEE Transactions on Signal Processing*, vol. 57, no. 7, pp. 2479–2493, 2009.
- [62] E. Hale, W. Yin, and Y. Zhang, “Fixed-point continuation for ℓ_1 -minimization: Methodology and convergence,” *SIAM Journal on Optimization*, vol. 19, no. 3, pp. 1107–1130, 2008.

- [63] A. Beck and M. Teboulle, “A fast iterative shrinkage-thresholding algorithm for linear inverse problems,” *SIAM Journal on Imaging Science*, vol. 2, no. 1, pp. 183–202, 2009.
- [64] D. Goldfarb, S. Ma, and K. Scheinberg, “Fast alternating linearization methods for minimizing the sum of two convex functions,” *Mathematical Programming*, vol. 141, no. 1-2, pp. 349–382, 2012.
- [65] F. Niu, B. Recht, C. Re, and S. J. Wright, “Hogwild!: A lock-free approach to parallelizing stochastic gradient descent,” 2011, online at arXiv:1106.5732v2.
- [66] F. Facchinei, S. Sagratella, and G. Scutari, “Flexible parallel algorithms for big data optimization,” 2013, Submitted to ICASSP 2014, Available at arXiv:1311.2444.
- [67] P. Combettes and J.-C. Pesquet, “Proximal splitting methods in signal processing,” in *Fixed-Point Algorithms for Inverse Problems in Science and Engineering*, Springer Optimization and Its Applications, pp. 185–212. Springer New York, 2011.
- [68] M. Yuan and Y. Lin, “Model selection and estimation in regression with grouped variables,” *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, vol. 68, no. 1, pp. 49–67, 2006.
- [69] H. Robbins and D. Siegmund, *A Convergence Theorem for Non-Negative Almost Supermartingales and Some Applications*, Optimizing Methods in Statistics. Academic Press, New York, 1971.
- [70] A. Y. Yang, S.S. Sastry, A. Ganesh, and Y. Ma, “Fast l_1 -minimization algorithms and an application in robust face recognition: A review,” in *IEEE International Conference on Image Processing (ICIP)*, 2010, pp. 1849–1852, online at <http://www.eecs.berkeley.edu/~yang/software/l1benchmark/>.
- [71] Y. Nesterov, “Gradient methods for minimizing composite objective function,” *Core Discussion Paper 2007/76*, Universite Catholique De Louvain, Center for Operations Research and Econometrics (CORE).