This is a manuscript of an article from *Mathematical Programming* (2016): The final publication is available at Springer via http://dx.doi.org/10.1007/s10107-016-1057-8.

Iteration Complexity Analysis of Block Coordinate Descent Methods

Mingyi Hong[*], Xiangfeng Wang[†], Meisam Razaviyayn[‡] and Zhi-Quan Luo[§]

April 29, 2015

Abstract

In this paper, we provide a unified iteration complexity analysis for a family of general block coordinate descent (BCD) methods, covering popular methods such as the block coordinate gradient descent (BCGD) and the block coordinate proximal gradient (BCPG), under various different coordinate update rules. We unify these algorithms under the so-called Block Successive Upper-bound Minimization (BSUM) framework, and show that for a broad class of multi-block nonsmooth convex problems, all algorithms covered by the BSUM framework achieve a global sublinear iteration complexity of $\mathcal{O}(1/r)$, where r is the iteration index. Moreover, for the case of block coordinate minimization (BCM) where each block is minimized exactly, we establish the sublinear convergence rate of $O(1/r)$ without per block strong convexity assumption. Further, we show that when there are only two blocks of variables, a special BSUM algorithm with Gauss-Seidel rule can be accelerated to achieve an improved rate of $\mathcal{O}(1/r^2)$.

[*]Department of Industrial and Manufacturing Systems Engineering, Iowa State University, IA, USA. Email: mingyi@iastate.edu

[†]Shanghai Key Lab. of Trustworthy Computing, Software Engineering Institute, East China Normal University, Shanghai 200062, China. Email: xfwang@sei.ecnu.edu.cn

[‡]Department of Electrical Engineering, Stanford University, Palo Alto, CA, USA. Email: meisam@stanford.edu

[§]Department of Electrical and Computer Engineering, University of Minnesota, Minneapolis, MN, USA. Email: luozq@umn.edu

1 Introduction

Consider the problem of minimizing a nonsmooth convex function $f(x)$ of the form:

$$\begin{aligned} \text{minimize} \quad & f(x) := g\left(x_1, \cdots, x_K\right) + \sum_{k=1}^{K} h_k(x_k) \\ \text{subject to} \quad & x_k \in X_k, \quad k = 1, ..., K \end{aligned} \tag{1.1}$$

where $g(\cdot)$ is a smooth convex function; h_k is a nonsmooth convex function (possibly with extended values); $x = (x_1^T, ..., x_K^T)^T \in \mathbb{R}^n$ is a partition of the optimization variable x, with $x_k \in X_k \subseteq \mathbb{R}^{n_k}$. Let $X := \prod_{k=1}^{K} X_k$ denote the feasible set for x.

A well known family of algorithms for solving (1.1) is the block coordinate descent (BCD) type method whereby, at every iteration a single block of variables is optimized while the remaining blocks are held fixed. One of the best known algorithms in the BCD family is the block coordinate minimization (BCM) algorithm, where at iteration r, the blocks are updated by solving the following problem exactly [1]

$$x_k^r \in \arg\min_{x_k \in X_k} \quad g(x_1^r, \ldots, x_{k-1}^r, x_k, x_{k+1}^{r-1}, \ldots, x_K^{r-1}) + h_k(x_k), \ k = 1, \cdots, K. \tag{1.2}$$

When problem (1.2) is not easily solvable, a popular variant is to solve an approximate version of problem (1.2), yielding the so-called block coordinate gradient descent (BCGD) algorithm, or the block coordinate proximal gradient (BCPG) algorithm in the presence of nonsmooth function [2–5]. In particular, at a given iteration r, the following problem is solved for each block k:

$$x_k^r = \arg\min_{x_k \in X_k} \langle \nabla_k g(x_1^r, \ldots, x_{k-1}^r, x_k^{r-1}, \ldots, x_K^{r-1}), x_k - x_k^{r-1} \rangle + \frac{L_k}{2} \|x_k - x_k^{r-1}\|^2 + h_k(x_k) \tag{1.3}$$

where $L_k > 0$ is some appropriately chosen constant. Other variants of the BCD-type algorithm include those that solve different subproblems [6], or those with different block selection rules, such as the Gauss-Seidel (G-S) rule, the Gauss-Southwell (G-So) rule [7], the randomized rule [8], the essentially cyclic (E-C) rule [9], or the maximum block improvement (MBI) rule [10].

In all the above mentioned variants of BCD method, each step involves solving a simple subproblem of small size, therefore the BCD method can be quite effective for solving large-scale problems; see e.g., [4,6,8,11,12] and the references therein. The existing analysis of the BCD method [9,13–15] requires the uniqueness of the minimizer for each subproblem (1.2), or the quasi convexity of f [16]. Recently, a unified BCD-type framework, termed the block successive upper-bound minimization (BSUM) method, is proposed in [6]. At each iteration of the BSUM method, certain approximate function of the per-block subproblem (1.2) is constructed and optimized. Due to the flexibility in choosing the approximate function, the BSUM includes many BCD-type algorithms as special cases. It is shown in [6] that the method converges to stationary solutions for nonconvex problems and to global optimal solutions for convex problems, as long as certain regularity conditions are satisfied for the per-block subproblems.

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 [17]. When the objective function is smooth and not strongly convex, Luo and Tseng have shown that the BCD method with the classic G-S/G-So update rules converges linearly, provided that a certain local error bound is satisfied around the solution set [17–20]. In addition, such linear rate is global when the feasible set is compact. This line of analysis has recently been extended to allow certain class of nonsmooth functions in the objective [3, 21]. For more general problems where the objective is not strongly convex and the error bound condition does not hold, several recent studies have established the $\mathcal{O}(1/r)$ iteration complexity for various BCD-type algorithms including the randomized BCGD algorithm [8], and for more general settings with nonsmooth objective as well [4,22,23]. When the coordinates are updated according to the traditional G-S/G-So/E-C rule, however, the literature on the iteration complexity for the BCD-type algorithm is scarce. In [12], Saha and Tewari have proven the $\mathcal{O}(1/r)$ rate for the G-S BCPG algorithm when applied to certain special ℓ_1 minimization problem. In [5], Beck and Tetruashvili have shown the $\mathcal{O}(1/r)$ sublinear convergence for the G-S BCGD algorithm for constrained smooth problems. In [24], Beck has shown the sublinear convergence for the G-S BCM algorithm (termed Alternating Minimization method therein) when the number of blocks is two. Although the BCD-type algorithm with G-S rule sometimes has been found to perform better than its randomized counterpart (see, e.g., [12]), establishing its iteration complexity in a general multi-block nonsmooth setting is challenging [8]. To the best of our knowledge, the iteration complexity of the BCD-type algorithm with the classic G-S update rule has not yet been characterized for multi-block nonsmooth problems, not to mention other types of deterministic coordinate selection rules such as G-So, E-C or MBI. Further, there has been no iteration complexity analysis for the classic BCM iteration (1.2) when the number of variable blocks is more than two (i.e., $K \geq 3$).

In this paper, we provide a unified iteration complexity analysis for K-block BCD-type algorithm by utilizing the BSUM framework [6]. Our result covers many different BCD-type algorithms such as BCM, BCPG, and BCGD under a number of deterministic coordinate update rules. First, for a broad class of nonsmooth convex problems, we show that the BSUM algorithm achieves a global sublinear convergence rate of $\mathcal{O}(1/r)$, provided that *each subproblem* is strongly convex. Second, when the number of variable blocks is two, we establish an improved $\mathcal{O}(1/r^2)$ rate for a particular version of the BSUM algorithm, without the strong convexity of the subproblems, or the gradient Lipschitz continuity of one of the subproblems. Third, for the BCM algorithm (1.2), we show the global convergence rate of $\mathcal{O}(1/r)$ without the per-block strong convexity assumption. The main results of this paper are summarized in the following table[1].

[1]We have used the following abbreviations: NS=**N**on**s**mooth, C=**C**onstrained, K=**K**-block, BSC=**B**lock-wise **S**trongly **C**onvex, G-So=**G**auss-**So**uthwell, G-S=**G**auss-**S**eidel, E-C=**E**ssentially **C**yclic, MBI=**M**aximum **B**lock **I**mprovement. The notion of *valid upper-bound* as well as the function u_k will be introduced in Section 2.

Table 1: Summary of the Results

Method	Update Rule	Problem	Assumptions	Rate
BSUM	**G-S/E-C**	NS+C+K	u_k valid upper-bound	$\mathcal{O}(1/r)$
BSUM	**G-So/MBI**	NS+C+K	u_k valid upper-bound, h Lipchitz	$\mathcal{O}(1/r)$
BSUM	**G-S**	NS+C+2	u_1 valid upper-bound without BSC, $u_2 = g$	$\mathcal{O}(1/r)$
BSUM	N/A	NS+C+1	u_1 valid upper-bound without BSC	$\mathcal{O}(1/r)$
BCM	**MBI**	NS+C+K	h Lipchitz, without BSC	$\mathcal{O}(1/r)$
BCM	**G-S/E-C**	NS+C+K	$u_k = g$, without BSC	$\mathcal{O}(1/r)$
Accelerated BSUM	**G-S**	NS+C+2	u_1 valid upper-bound, $u_2 = g$, BSC	$\mathcal{O}(1/r^2)$

Notations: For a given matrix A, we use $A[i,j]$ to denote its (i,j)th element. For a symmetric matrix A use $\rho_{\max}(A)$ to denote its spectral norm. For a given vector x, we use $x[j]$ to denote its jth component; use $\|x\|$ to denote its ℓ_2 norm. We use $I_X(\cdot)$ to denote the indicator function for a given set X, i.e., $I_X(y) = 1$ if $y \in X$, and $I_X(y) = \infty$ if $y \notin X$. Let x_{-k} denote the vector x with x_k removed. For a given function $f(x_1, \cdots, x_K)$ which contains K block variables, we use $\nabla_k f(x_1, \cdots, X_k)$ to denote the partial gradient with respect to its kth block variable. We use ∂f to denote a subgradient of a function f. For a given convex nonsmooth function $\ell(\cdot)$, we define the proximity operator $\text{prox}_\ell(\cdot) : \mathbb{R}^n \mapsto \mathbb{R}^n$ as

$$\text{prox}_\ell^\beta(x) = \underset{u \in \mathbb{R}^n}{\text{argmin}} \ \ell(u) + \frac{\beta}{2}\|x-u\|^2.$$

Similarly, for a given convex set X, the projection operator $\text{proj}_X(\cdot) : \mathbb{R}^n \mapsto X$ is defined as

$$\text{proj}_X(x) = \underset{u \in X}{\text{argmin}} \ \frac{1}{2}\|x-u\|^2.$$

2 The BSUM Algorithm and Preliminaries

2.1 The BSUM Algorithm

In this paper, we consider a family of block coordinate descent methods (BCD) for solving problem (1.1). The family of the algorithms we consider falls in the general category of block successive upper-bound minimization (BSUM) method, in which certain *approximate version* of the objective function is optimized one block variable at a time, while fixing the rest of the block variables [6]. In particular, at iteration $r+1$, we first pick an index set $\mathcal{C}^{r+1} \subseteq \{1, \cdots, K\}$. Then the kth block

variable is updated by

$$x_k^{r+1}\begin{cases}\in \min_{x_k\in X_k}\ u_k\left(x_k; x_1^{r+1},\cdots,x_{k-1}^{r+1},x_k^r,\cdots,x_K^r\right)+h_k(x_k), & \text{if } k\in\mathcal{C}^{r+1};\\ =x_k^r, & \text{if } k\notin\mathcal{C}^{r+1},\end{cases}\tag{2.1}$$

where $u_k(\cdot;x_1^{r+1},\cdots,x_{k-1}^{r+1},x_k^r,\cdots,x_K^r)$ is an approximation of $g(x)$ at a given iterate $(x_1^{r+1},\cdots,x_{k-1}^{r+1},x_k^r,\cdots,x_K^r)$. We will see shortly that by properly specifying the approximation function $u_k(\cdot)$ as well as the index set $\mathcal{C}^{r+1}$, we can recover many popular BCD-type algorithms such as the BCM, the BCGD, the BCPG methods and so on.

To simplify notations, let us define a set of auxiliary variables

$$\begin{aligned} w_k^r &:= [x_1^r,\cdots,x_{k-1}^r,x_k^{r-1},x_{k+1}^{r-1},\cdots,x_K^{r-1}],\ k=1,\cdots,K,\\ w_{-k}^r &:= [x_1^r,\cdots,x_{k-1}^r,x_{k+1}^{r-1},\cdots,x_K^{r-1}],\ k=1,\cdots,K,\\ x_{-k} &:= [x_1,\cdots,x_{k-1},x_{k+1},\cdots,x_K].\end{aligned}$$

Clearly we have $w_{K+1}^r := x^r,\ w_1^r := x^{r-1}$. Moreover, at each iteration $r+1$, define a set of new variables $\{\hat{x}_k^{r+1}\}_{k=1}^K$ as follows

$$\hat{x}_k^{r+1}\in\min_{x_k\in X_k}\ u_k\left(x_k;x^r\right)+h_k(x_k),\ k=1,\cdots,K.\tag{2.2}$$

Clearly $\{\hat{x}_k^{r+1}\}_{k=1}^K$ represents a "virtual" update where all variables are optimized in a Jacobi manner based on x^r.

The BSUM algorithm is described formally in the following table.

The Block Successive Upper-Bound Minimization (BSUM) Algorithm

At each iteration $r+1$, pick an index set $\mathcal{C}^{r+1}$;

For $k=1,\cdots,K$, do:

$$x_k^{r+1}\begin{cases}\in \min_{x_k\in X_k}\ u_k\left(x_k;w_k^{r+1}\right)+h_k(x_k), & \text{if } k\in\mathcal{C}^{r+1};\\ =x_k^r, & \text{if } k\notin\mathcal{C}^{r+1}\end{cases}.$$

End For.

In this paper, we consider four well-known block selection rules, described below:

1. *Gauss-Seidel (G-S) rule*: At each iteration $r+1$ all the indices are chosen, i.e., $\mathcal{C}^{r+1}=\{1,\cdots,K\}$. Using this rule, the blocks are updated cyclically with fixed order.

2. *Essentially cyclic (E-C) rule*: There exists a given period $T\geq 1$ during which each index is updated at least once, i.e.,

$$\bigcup_{i=1}^{T}\mathcal{C}^{r+i}=\{1,\cdots,K\},\ \forall\ r.\tag{2.3}$$

We call this update rule a *period-T* essentially cyclic update rule. Clearly when $T = 1$ we recover the G-S rule.

3. *Gauss-Southwell (G-So) rule*: At each iteration $r + 1$, $\mathcal{C}^{r+1}$ contains a single index k^* that satisfies:

$$k^* \in \left\{ k \;\middle|\; \|\hat{x}_k^{r+1} - x_k^r\| \geq q \max_j \|\hat{x}_j^{r+1} - x_j^r\| \right\} \tag{2.4}$$

for some constant $q \in (0,\ 1]$.

4. *Maximum block improvement (MBI) rule*: At each iteration $r+1$, $\mathcal{C}^{r+1}$ contains a single index k^* that satisfies:

$$k^* \in \arg\max_k -f(\hat{x}_k^{r+1}, x_{-k}^r). \tag{2.5}$$

2.2 Main Assumptions

Suppose f is a closed proper convex function in $\mathbb{R}^n$. Let dom f denote the effective domain of f and let int(dom f) denote the interior of dom f. We make the following standing assumptions regarding problem (1.1):

Assumption A.

(a) Problem (1.1) is a convex problem, ant its global minimum is attained. The intersection $X \cap \text{int}(\text{dom } f)$ is nonempty.

(b) The gradient of $g(\cdot)$ is block-coordinate-wise uniformly Lipschitz continuous

$$\|\nabla_k g([x_{-k}, x_k]) - \nabla_k g([x_{-k}, x'_k])\| \leq M_k \|x_k - x'_k\|, \quad \forall\ x_k, x'_k \in X_k,\ \forall\ x \in X,\ \forall\ k \tag{2.6}$$

where $M_k > 0$ is a constant. Define $M_{\max} = \max_k M_k$.

The gradient of $g(\cdot)$ is also uniformly Lipschitz continuous

$$\|\nabla g(x) - \nabla g(x')\| \leq M\|x - x'\|, \qquad \forall\ x, x' \in X \tag{2.7}$$

where $M > 0$ is a constant.

Next we make the following assumptions regarding the approximation function $u_k(\cdot;\cdot)$ in (2.1).

Assumption B.

(a) $u_k(x_k; x) = g(x), \quad \forall\ x \in X,\ \forall\ k,$

(b) $u_k(v_k; x) \geq g(v_k, x_{-k}), \quad \forall\, v_k \in X_k,\ \forall\, x \in X,\ \forall\, k,$

(c) $\nabla u_k(x_k; x) = \nabla_k g(x), \quad \forall\, x \in X,\ \forall\, k,$

(d) $u_k(v_k; x)$ is continuous in v_k and x. Further, for any given x, it is strongly convex in v_k

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

where $\gamma_k > 0$ is independent of the choice of x.

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

$$\|\nabla u_k(v_k; x) - \nabla u_k(\hat{v}_k; x)\| \leq L_k\|v_k - \hat{v}_k\|,\ \forall\, \hat{v}_k,\ v_k \in X_k,\ \forall\, k,\ \forall\, x \in X, \tag{2.8}$$

where $L_k > 0$ is some constant. Further, we have

$$\|\nabla u_k(v_k; x) - \nabla u_k(v_k; y)\| \leq G_k\|x - y\|,\ \forall\, v_k \in X_k,\ \forall\, k,\ \forall\, x, y \in X. \tag{2.9}$$

Define $L_{\max} := \max_k L_k$; $G_{\max} := \max_k G_k$.

We refer to the u_k's that satisfy Assumption B as a *valid upper-bound*.

A few remarks are in order regarding to the assumptions made above.

First of all, Assumption B indicates that for any given x, each $u_k(\cdot; x)$ is a locally tight upper bound for $g(x)$. When the approximation function is chosen as the original function $g(x)$, then we recover the classic BCM algorithm; cf. (1.2). In many practical applications especially nonsmooth problems, minimizing the approximation functions often leads to much simpler subproblems than directly minimizing the original function; see e.g., [25–29]. For example, if $h_k(\cdot) = 0$ for all k, and u_k takes the following form

$$u_k(x_k; w_k^{r+1}) = g(w_k^{r+1}) + \langle \nabla_k g(w_k^{r+1}), x_k - x_k^r \rangle + \frac{M_k}{2}\|x_k - x_k^r\|^2, \tag{2.10}$$

then we recover the well known BCGD method [5, 8, 17], in which x_k is updated by

$$x_k^{r+1} = \mathrm{proj}_{X_k}\left[x_k^r - \frac{1}{M_k}\nabla_k g(w_k^{r+1})\right]. \tag{2.11}$$

When the nonsmooth components h_k's are present, the above choice of $u_k(\cdot;\cdot)$ in (2.10) leads to the so-called BCPG method [3, 6, 30], in which x_k is updated by

$$x_k^{r+1} = \mathrm{prox}_{h_k + I_{X_k}}^{M_k}\left[x_k^r - \frac{1}{M_k}\nabla_k g(w_k^{r+1})\right]. \tag{2.12}$$

For other possible choices of the approximation function, we refer the readers to [6, 31].

Secondly, the strong convexity requirement on $u_k(\cdot;x)$ in Assumption B(d) is quite mild, see the examples given in the previous remark (e.g., BCPG and BCGD). When u_k is chosen as the original function $g(x)$, this requirement says that $g(x)$ must be *block-wise strongly convex* (BSC). The BSC condition is in fact satisfied in many practical engineering problems. The following are two interesting examples.

Example 2.1 *Consider the rate maximization problem in an uplink wireless communication network, where K users transmit to a single base station (BS) in the network. Suppose each user has n_t transmit antennas, and the BS has n_r receive antennas. Let $C_k \in \mathbb{R}^{n_t \times n_t}$ denote user k's transmit covariance matrix, P_k denote the maximum transmit power for user k, and $H_k \in \mathbb{R}^{n_r \times n_t}$ denote the channel matrix between user k and the BS. Then the uplink channel capacity optimization problem is given by the following convex program [32, 33]*

$$\min_{\{C_k\}_{k=1}^K} \; -\log\det\left|\sum_{k=1}^K H_k C_k H_k^T + I_{n_r}\right|, \quad \text{s.t.} \quad C_k \succeq 0,\ \mathrm{Tr}[C_k] \le P_k,\ k=1,\cdots,K, \tag{2.13}$$

where I_{n_r} is the $n_r \times n_r$ identity matrix. The celebrated iterative water-filling algorithm (IWFA) [33] for solving this problem is simply the BSUM algorithm with exact block minimization (i.e. the BCM algorithm) and G-S update rule. It is easy to verify that when $n_t \le n_r$ (i.e., the number of transmit antenna is smaller than that of the receive antenna), and when the channels are generated randomly, then with probability one $H_k^T H_k$ is of full rank, implying that the BSC condition is satisfied. We note that there has been no iteration complexity analysis of the IWFA algorithm for any type of block selection rules.

Example 2.2 *Consider the following LASSO problem:*

$$\min \|Ax-b\|^2 + \lambda\|x\|_1,$$

where $A \in \mathbb{R}^{M\times K}$, $b \in \mathbb{R}^M$, and $x = [x_1,\cdots,x_K]^T$, with $x_k \in \mathbb{R}$ for all k. That is, each block consists of a single scalar variable. In this case, as long as none of A's columns are zero (in which case we simply remove that column and the corresponding block variable), the problem satisfies the BSC property. Prior to our work, there is no iteration complexity analysis for applying BCD with deterministic block selection rules such as G-S and E-C for LASSO (with general data matrix A).

Note that the BSC property, or more generally the strong convexity assumption on the approximate function u_k, is reasonable as it ensures that each step of the BSUM algorithm is well-defined and has a unique solution. In the ensuing analysis of the BSUM algorithm, we assume that either the BSC property holds true, or u_k is a valid upper-bound. Later in Sections 4 - 6, we will consider the case where the BSC assumption is absent.

3 Convergence Analysis for BSUM

In this section, we show that under assumptions A and B, the BSUM algorithm with flexible update rules achieves global sublinear rate of convergence.

Let us define X^* as the optimal solution set, and let $x^* \in X^*$ be one of the optimal solutions. For the BSUM algorithm, define the optimality gap as

$$\Delta^r := f(x^r) - f(x^*). \tag{3.1}$$

Despite the generality of the BSUM algorithm, our analysis of BSUM only consists of three simple steps: S1) estimate the amount of successive decrease of the optimality gaps; S2) estimate the cost yet to be minimized after each iteration; S3) estimate the rate of convergence.

We first characterize the successive difference of the optimality gaps before and after one iteration of the BSUM algorithm, with different update rules.

Lemma 3.1 (Sufficient Descent) *Suppose Assumption A and Assumption B hold. Then*

1. *For BSUM with either G-S rule or the E-C rule, we have that for all $r \geq 1$*

$$\Delta^r - \Delta^{r+1} \geq \sum_{k=1}^{K} \frac{\gamma_k}{2} \|x_k^r - x_k^{r+1}\|^2 \geq \gamma \|x^r - x^{r+1}\|^2, \tag{3.2}$$

   where the constant $\gamma := \frac{1}{2} \min_k \gamma_k > 0$.

2. *For BSUM with G-So rule and MBI rule, we have that for all $r \geq 1$*

$$\Delta^r - \Delta^{r+1} \geq \frac{c_1}{K} \gamma \|x^r - \hat{x}^{r+1}\|^2, \tag{3.3}$$

   where the constant $\gamma := \frac{1}{2} \min_k \gamma_k > 0$; For G-So rule, $c_1 = q$, and for MBI rule, $c_1 = 1$.

Proof. We first show part (1) of the proof. Suppose that $k \notin \mathcal{C}^{r+1}$, then we have the following trivial inequality

$$f(w_k^{r+1}) - f(w_{k+1}^{r+1}) \geq \frac{\gamma_k}{2} \|x_k^{r+1} - x_k^r\|^2 \tag{3.4}$$

as both sides of the inequality are zero.

Suppose $k \in \mathcal{C}^{r+1}$. Then using Assumption B, we have that

$$\begin{aligned} f(w_k^{r+1}) - f(w_{k+1}^{r+1}) &\geq u_k(x_k^r; w_k^{r+1}) + h_k(x_k^r) - \left(u_k(x_k^{r+1}; w_k^{r+1}) + h_k(x_k^{r+1})\right) \\ &\geq \langle \nabla u_k(x_k^{r+1}; w_k^{r+1}), x_k^r - x_k^{r+1} \rangle + h_k(x_k^r) - h_k(x_k^{r+1}) + \frac{\gamma_k}{2}\|x_k^{r+1} - x_k^r\|^2 \\ &\geq \langle \nabla u_k(x_k^{r+1}; w_k^{r+1}) + \zeta_k^{r+1}, x_k^r - x_k^{r+1} \rangle + \frac{\gamma_k}{2}\|x_k^{r+1} - x_k^r\|^2 \\ &\geq \frac{\gamma_k}{2}\|x_k^{r+1} - x_k^r\|^2 \end{aligned} \tag{3.5}$$

where the first inequality is due to Assumption B(a)–B(b); the second inequality is due to Assumption B(d); in the third inequality we have defined $\zeta_k^{r+1} \in \partial h_k(x_k^{r+1})$; the last inequality is due to the fact that x_k^{r+1} is the optimal solution for the strongly convex problem

$$\arg\min_{x_k \in X_k} u_k(x_k; w_k^{r+1}) + h_k(x_k).$$

Summing over k, we have

$$f(x^r) - f(x^{r+1}) \geq \gamma \|x^r - x^{r+1}\|^2, \tag{3.6}$$

where $\gamma := \frac{1}{2}\min_k \gamma_k$.

We then show part (2) of the claim. Suppose $k \in \mathcal{C}^{r+1}$, then we have the following series of inequalities for the G-So rule

$$\begin{aligned} f(x^r) - f(x^{r+1}) &= f(x^r) - f(x_{-k}^r, \hat{x}_k^{r+1}) \\ &\geq u_k(x_k^r; x^r) + h_k(x_k^r) - u_k(\hat{x}_k^{r+1}; x^r) - h_k(\hat{x}_k^{r+1}) \\ &\geq \frac{1}{2}\gamma_k \|x_k^r - \hat{x}_k^{r+1}\|^2 \\ &\geq \frac{q \min_j \gamma_j}{2K} \sum_{j=1}^{K} \|x_j^r - \hat{x}_j^{r+1}\|^2 \\ &= \frac{q}{K}\gamma \|x^r - \hat{x}^{r+1}\|^2. \end{aligned} \tag{3.7}$$

Similar steps lead to the result for the MBI rule. **Q.E.D.**

Next we show the second step of the proof, which estimates the gap yet to be minimized after each iteration of the algorithm. Let us define the following constants:

$$R := \max_{x \in X} \max_{x^* \in X^*} \left\{ \|x - x^*\| \ : \ f(x) \leq f(x^1) \right\}, \quad Q := \max_{x \in X} \left\{ \|\nabla g(x)\| : \ f(x) \leq f(x^1) \right\}. \tag{3.8}$$

When assuming that the level set $\{x : f(x) \leq f(x^1)\}$ is compact, then all the above constants are finite. Clearly we have

$$\|x^r - x^*\| \leq R, \quad \|\nabla g(x^r)\| \leq Q, \ \forall \ r = 1, \cdots . \tag{3.9}$$

Occasionally we need to further make the assumption that the nonsmooth part $h(x)$ is Lipchitz continuous:

$$\|h(x) - h(y)\| \leq L_h \|x - y\|, \ \forall \ x, y \in X, \tag{3.10}$$

with some $L_h > 0$. Note that such assumption is satisfied by most of the popular nonsmooth regularizers such as the ℓ_1 norm, the ℓ_2 norm and so on. Also note that even with this assumption, our considered problem is still a *constrained* one, as the convex constraints $x_k \in X_k$ have not been moved to the objective as nonsmooth indicator functions.

Lemma 3.2 (Cost-to-go Estimate) *Suppose Assumptions A and B are satisfied. Then*

1. *For the BSUM with G-S update rule, we have*

$$(\Delta^{r+1})^2 \leq R^2 K G_{\max}^2 \|x^{r+1} - x^r\|^2, \ \forall \ x^* \in X^*.$$

2. *For the BSUM with period-T E-C update rule, we have*

$$(\Delta^{r+T})^2 \leq T R^2 K G_{\max}^2 \sum_{t=1}^{T} \|x^{r+t} - x^{r+t-1}\|^2, \ \forall \ x^* \in X^*.$$

3. *For the BSUM with G-So and MBI rules, further assume that $h(\cdot)$ is Lipchitz continuous (cf.* (3.10)*). Then we have*

$$\Delta^r = f(x^r) - f(x^*) \leq 2\left((Q + L_h)^2 + L_{\max}^2 K R^2\right) \|\hat{x}^{r+1} - x^r\|^2, \ \forall \ x^* \in X^*.$$

Proof. We first show part (1) of the claim. We have the following sequence of inequalities

$$\begin{aligned} f(x^{r+1}) - f(x^*) &= g(x^{r+1}) - g(x^*) + h(x^{r+1}) - h(x^*) \\ &\leq \langle \nabla g(x^{r+1}), x^{r+1} - x^* \rangle + h(x^{r+1}) - h(x^*) \\ &= \sum_{k=1}^{K} \langle \nabla_k g(x^{r+1}) - \nabla u_k(x_k^{r+1}; w_k^{r+1}), x_k^{r+1} - x_k^* \rangle \\ &\quad + \sum_{k=1}^{K} \langle \nabla u_k(x_k^{r+1}; w_k^{r+1}), x_k^{r+1} - x_k^* \rangle + h(x^{r+1}) - h(x^*). \end{aligned} \tag{3.11}$$

Notice that x_k^{r+1} is the optimal solution for problem: $\operatorname{argmin}_{x_k \in X_k} u_k(x_k; w_k^{r+1}) + h_k(x_k)$. It follows from the optimality condition of this problem that there exists some $\zeta_k^{r+1} \in \partial\left(h_k(x_k^{r+1})\right)$ such that

$$\begin{aligned} 0 &\geq \langle \nabla u_k(x_k^{r+1}; w_k^{r+1}) + \zeta_k^{r+1}, x_k^{r+1} - x_k^* \rangle \\ &\geq \langle \nabla u_k(x_k^{r+1}; w_k^{r+1}), x_k^{r+1} - x_k^* \rangle + h_k(x_k^{r+1}) - h_k(x_k^*), \end{aligned} \tag{3.12}$$

where in the last inequality we have used the definition of subgradient

$$h_k(x_k) - h_k(v_k) \geq \langle \zeta_k^{r+1}, x_k - v_k \rangle, \ \forall \ x_k, v_k \in X_k. \tag{3.13}$$

Combining (3.11) and (3.12), we obtain

$$\begin{aligned}
&\big(f(x^{r+1}) - f(x^*)\big)^2 \\
&\overset{\text{(i)}}{\leq} \left(\sum_{k=1}^{K} \|\nabla_k g(x^{r+1}) - \nabla u_k(x_k^{r+1}; w_k^{r+1})\| \|x_k^{r+1} - x_k^*\| \right)^2 \\
&\overset{\text{(ii)}}{\leq} \left(\sum_{k=1}^{K} G_k \|x^{r+1} - w_k^{r+1}\| \|x_k^{r+1} - x_k^*\| \right)^2 \\
&\leq R^2 K G_{\max}^2 \|x^{r+1} - x^r\|^2
\end{aligned}$$

where in (i) we have used the Cauchy-Schwarz inequality and the Lipchitz continuity of $u_k(\cdot;\cdot)$ in (2.8); in (ii) we have used the Lipchitz continuity of $\nabla g(\cdot)$ in (2.7), and that $\nabla_k g(x^{r+1}) = \nabla_k u_k(x_k^{r+1}; x^{r+1})$ (cf. Assumption B(c)).

Next we show part (2) of the claim. Let us define a new index set $\{r_k\}$ as follows:

$$r_k := \arg\max_t \{x_k^t \neq x_k^{r+T}\} + 1, \ k = 1, \cdots, K. \tag{3.14}$$

That is, r_k is the latest iteration index (up until $r+T$) in which the kth variable has been updated. From this definition we have $x_k^{r_k} = x_k^{r+T}$, for all k.

We have the following sequence of inequalities

$$\begin{aligned}
&f(x^{r+T}) - f(x^*) \\
&= g(x^{r+T}) - g(x^*) + \sum_{k=1}^{K} \big(h_k(x_k^{r_k}) - h_k(x_k^*)\big) \\
&\leq \langle \nabla g(x^{r+T}), x^{r+T} - x^* \rangle + \sum_{k=1}^{K} \big(h_k(x_k^{r_k}) - h_k(x_k^*)\big) \\
&\overset{\text{(i)}}{=} \sum_{k=1}^{K} \Big(\langle \nabla_k g(x^{r+T}) - \nabla u_k(x_k^{r_k}; w_k^{r_k}), x_k^{r+T} - x_k^* \rangle + \langle \nabla u_k(x_k^{r_k}; w_k^{r_k}), x_k^{r_k} - x_k^* \rangle \Big) \\
&\qquad + \sum_{k=1}^{K} \big(h_k(x_k^{r_k}) - h_k(x_k^*)\big) \\
&\overset{\text{(ii)}}{\leq} \sum_{k=1}^{K} \langle \nabla_k g(x^{r+T}) - \nabla u_k(x_k^{r_k}; w_k^{r_k}), x_k^{r+T} - x_k^* \rangle
\end{aligned}$$

where in (i) we have used the fact that $x_k^{r+T} = x_k^{r_k}$, for all k; in (ii) we have used the optimality of $x_k^{r_k}$. Taking the square on both sides, we obtain

$$\begin{aligned}
&(f(x^{r+T}) - f(x^*))^2 \\
&\leq \left(\sum_{k=1}^{K} \|\nabla_k g(x^{r+T}) - \nabla u_k(x_k^{r_k}; w_k^{r_k})\| \|x_k^{r+T} - x_k^*\| \right)^2 \\
&\leq \left(\sum_{k=1}^{K} G_k \|x^{r+T} - w_k^{r_k}\| \|x_k^{r+T} - x_k^*\| \right)^2 \\
&\leq \left(\sum_{k=1}^{K} G_k \left(\|x^{r+T} - x^{r_k}\| + \|x^{r_k} - w_k^{r_k}\| \right) \|x_k^{r+T} - x_k^*\| \right)^2 \\
&\leq TKG_{\max}^2 R^2 \sum_{t=1}^{T} \|x^{r+t-1} - x^{r+t}\|^2.
\end{aligned}$$

Finally we show part (3) of the claim. We have the following sequence of inequalities

$$\begin{aligned}
&f(x^r) - f(x^*) \\
&= g(x^r) - g(x^*) + h(x^r) - h(x^*) \\
&\overset{\text{(i)}}{\leq} \langle \nabla g(x^r), x^r - x^* \rangle + L_h \|x^r - \hat{x}^{r+1}\| + h(\hat{x}^{r+1}) - h(x^*) \\
&= \langle \nabla g(x^r), x^r - \hat{x}^{r+1} \rangle + \langle \nabla g(x^r), \hat{x}^{r+1} - x^* \rangle + L_h \|x^r - \hat{x}^{r+1}\| + h(\hat{x}^{r+1}) - h(x^*) \\
&\leq (L_h + Q)\|x^r - \hat{x}^{r+1}\| + \sum_{k=1}^{K} \langle \nabla_k g(x^r) - \nabla u_k(\hat{x}_k^{r+1}; x^r), \hat{x}_k^{r+1} - x_k^* \rangle \\
&\quad + \sum_{k=1}^{K} \left\langle \nabla u_k(\hat{x}_k^{r+1}; x^t), \hat{x}_k^{r+1} - x_k^* \right\rangle + h(\hat{x}^{r+1}) - h(x^*)
\end{aligned} \tag{3.15}$$

where step (i) follows from the Lipchitz continuity assumption (3.10) as well as the convexity of $g(\cdot)$. Similar to the proof of (3.12) in part (1), we can show that

$$\sum_{k=1}^{K} \langle \nabla u_k(\hat{x}_k^{r+1}; x^r), \hat{x}_k^{r+1} - x_k^* \rangle + h(\hat{x}^{r+1}) - h(x^*) \leq 0. \tag{3.16}$$

Moreover, it follows from Assumption B(c) and B(e) that

$$\begin{aligned}
&\left(\sum_{k=1}^{K}\left\langle \nabla_k g(x^r) - \nabla u_k(\hat{x}_k^{r+1}; x^r), x_k^{r+1} - x_k^*\right\rangle\right)^2 \\
&= \left(\sum_{k=1}^{K}\left\langle \nabla u_k(x_k^r; x^r) - \nabla u_k(\hat{x}_k^{r+1}; x^r), x_k^{r+1} - x_k^*\right\rangle\right)^2 \\
&\le K\sum_{k=1}^{K} L_k^2 \|x_k^r - \hat{x}_k^{r+1}\|^2 \|x_k^{r+1} - x_k^*\|^2 \\
&\le K L_{\max}^2 \|x^r - \hat{x}^{r+1}\|^2 R^2. \qquad (3.17)
\end{aligned}$$

Putting the above three inequalities together, we have

$$f(x^r) - f(x^*) \le 2\left((Q + L_h)^2 + K L_{\max}^2 R^2\right)\|x^r - \hat{x}^{r+1}\|^2. \qquad (3.18)$$

This completes the proof. **Q.E.D.**

We are now ready to prove the $\mathcal{O}(1/r)$ iteration complexity for the BSUM algorithm when applied to problem (1.1). Our results below are more general than the recent analysis on the iteration complexity for BCD-type algorithms. The generality of our results can be seen from several fronts: 1) The family of algorithms we analyze is broad; it includes the classic BCD, the BCGD method, the BCPG methods as well as their variants based on different coordinate selection rules as special cases, while the existing works only focus on one particular algorithm; 2) When the coordinates are updated in a G-S fashion, our result covers the general multi-block nonsmooth case, where $h_k(x)$ can take any proper closed convex nonsmooth function, while existing works only cover some special cases [5, 12, 24]; 3) When the coordinates are updated using other update rules such as G-So, MBI, E-C fashion, our convergence results appear to be new.

Theorem 3.1 *Suppose Assumption A(a) and Assumption B hold true. We have the following.*

1. *Let $\{x^r\}$ be the sequence generated by the BSUM algorithm with G-S rule. Then we have*

$$\Delta^r = f(x^r) - f^* \le \frac{c_1}{\sigma_1}\frac{1}{r},\ \forall\ r \ge 1, \qquad (3.19)$$

where the constants are given below

$$\begin{aligned}
c_1 &= \max\{4\sigma_1 - 2, f(x^1) - f^*, 2\}, \\
\sigma_1 &= \frac{\gamma}{K G_{\max}^2 R^2}, \qquad (3.20)
\end{aligned}$$

2. *Let $\{x^r\}$ be the sequence generated by the BSUM algorithm with E-C rule. Then we have*

$$\Delta^r = f(x^r) - f^* \leq \frac{c_2}{\sigma_2}\frac{1}{r-T},\ \forall\ r > T, \tag{3.21}$$

where the constants are given below

$$\begin{aligned} c_2 &= \max\{4\sigma_2 - 2, f(x^1) - f^*, 2\}, \\ \sigma_2 &= \frac{\gamma}{KTR^2G_{\max}^2}. \end{aligned} \tag{3.22}$$

3. *Suppose the Lipchitz continuity assumption* (3.10) *holds true. Let $\{\mathbf{x}^r\}$ be the sequence generated by the BSUM algorithm with G-So and MBI rule. Then we have*

$$\Delta^r = f(x^r) - f^* \leq \frac{1}{\sigma_3 r} \tag{3.23}$$

where

$$\sigma_3 = \begin{cases} \frac{\gamma q}{2K((Q+L_h)^2+L_{\max}^2KR^2)}, & \text{(G-So rule)} \\ \frac{\gamma}{2K((Q+L_h)^2+L_{\max}^2KR^2)}, & \text{(MBI rule)} \end{cases}. \tag{3.24}$$

Proof. We first show part (1) of the claim by mathematical induction on r. From Lemma 3.2 and Lemma 3.1, we have that for the G-S rule, we have

$$\Delta^r - \Delta^{r+1} \geq \frac{\gamma}{KG_{\max}^2R^2}(\Delta^{r+1})^2 := \sigma_1(\Delta^{r+1})^2,\ \forall\ r \geq 1, \tag{3.25}$$

or equivalently

$$\sigma_1(\Delta^{r+1})^2 + \Delta^{r+1} \leq \Delta^r,\ \forall\ r \geq 1. \tag{3.26}$$

By definition, we have $\Delta^1 = f(x^1) - f^*$. We first argue that

$$\Delta^2 \leq \frac{c_1}{2\sigma_1},\ \text{with } c_1 := \max\{4\sigma_1 - 2, f(\mathbf{x}^1) - f^*, 2\}. \tag{3.27}$$

From (3.26) and the fact that $\Delta^1 \leq c_1$, we have

$$\Delta^2 \leq \frac{-1+\sqrt{1+4\sigma_1c_1}}{2\sigma_1} = \frac{2c_1}{1+\sqrt{1+4\sigma_1c_1}} \leq \frac{2c_1}{1+|4\sigma_1-1|}$$

where in the last inequality we have used the fact that $c_1 \geq 4\sigma_1 - 2$. Suppose $4\sigma_1 - 1 \geq 0$, then we immediately have $\Delta^2 \leq \frac{c_1}{2\sigma_1}$. Suppose $4\sigma_1 - 1 < 0$, then

$$\Delta^2 \leq \frac{2c_1}{2-4\sigma_1} \leq \frac{2c_1}{8\sigma_1 - 4\sigma_1} = \frac{c_1}{2\sigma_1}. \tag{3.28}$$

Next we argue that if $\Delta^r \leq \frac{c_1}{r\sigma_1}$, then we must have

$$\Delta^{r+1} \leq \frac{c_1}{(r+1)\sigma_1}. \tag{3.29}$$

Using the condition (3.26) and the inductive hypothesis $\Delta^r \leq \frac{c_1}{r\sigma_1}$, we have

$$\begin{aligned}\Delta^{r+1} &\leq \frac{-1+\sqrt{1+\frac{4c_1}{r}}}{2\sigma_1} = \frac{2c_1}{r\sigma_1\left(1+\sqrt{1+\frac{4c_1}{r}}\right)} \\ &\leq \frac{2c_1}{\sigma_1\left(r+\sqrt{r^2+4r+4}\right)} = \frac{c_1}{\sigma_1(r+1)}\end{aligned} \tag{3.30}$$

where the last inequality is due to the fact that $c_1 \geq 2$, and $r \geq 2$. Consequently, we have shown that for all $r \geq 1$

$$\Delta^r = f(x^r) - f^* \leq \frac{c_1}{\sigma_1}\frac{1}{r}. \tag{3.31}$$

For the E-C rule, first note that from Lemma 3.1, we have

$$\Delta^r - \Delta^{r+T} \geq \frac{\gamma}{TKR^2G_{\max}^2}(\Delta^{r+T})^2 := \sigma_2(\Delta^{r+T})^2, \ \forall\ r \geq 1. \tag{3.32}$$

Then using the similar argument as for the G-S rule, we can obtain the desired result.

Next we show part (3) of the claim. For the G-So rule, we have from Lemma 3.2, the second part of Lemma 3.1, that for all $r \geq 1$

$$\Delta^r - \Delta^{r+1} \geq \frac{q}{K}\gamma\|\hat{x}^{r+1} - x^r\|^2 \geq \frac{\gamma q}{2K\left((Q+L_h)^2 + L_{\max}^2 KR^2\right)}(\Delta^r)^2 := \sigma_3(\Delta^r)^2. \tag{3.33}$$

Similar relation can be shown for the MBI rule as well. The rest of the proof follows standard argument, see for example [8, Theorem 1]. **Q.E.D.**

Below we provide further remarks on some special cases of the BSUM algorithm.

1. One popular choice of the upper bound function $u_k(\cdot,\cdot)$ is [4, 5, 8, 25–28]

$$u_k(z_k; x) := g(x) + \langle \nabla_k g(x), z_k - x_k\rangle + \frac{L_k}{2}\|z_k - x_k\|^2 \tag{3.34}$$

where the constant $L_k \geq \rho_{\max}(\nabla^2 g(x))$, is often chosen to be largest eigenvalue of the Hessian of $g(x)$. In this case, evidently we have $\gamma_k = L_k = M_k \leq M$, for all k, and $G_{\max} \leq M$. We can also verify that $G_k \leq 2M$ for all k. Using this choice of $u_k(\cdot;\cdot)$ and L_k, the first result in Theorem 3.1 reduces to

$$\Delta^r \leq 2\frac{c_1 KM^2R^2}{M_{\min}}\frac{1}{r} \tag{3.35}$$

where $M_{\min} := \min_k M_k$. Let us compare the order given in (3.35) with the one stated in [5, Theorem 6.1], which is the best known complexity bound for the G-S BCD algorithm for *smooth* problems (i.e., when h_k is not present). The bound derived in [5] for smooth constrained problem (resp. smooth unconstrained problem) is in the order of $\frac{KM^2R^2}{M_{\min}}\frac{1}{r}$ (resp. $\frac{M_{\max}KM^2R^2}{M_{\min}^2}\frac{1}{r}$). These orders are approximately the same as (3.35). However, our proof covers the general nonsmooth cases, and is simpler. Similarly, when $u_k(\cdot;\cdot)$ takes the form (3.34), the bounds for the BSUM with the E-C/G-So/MBI rules shown in Theorem 3.1 can also be simplified.

2. The results derived in Theorem 3.1 is equally applicable to the BCM scheme (1.2) with various block selection rules discussed above. In particular, we can specialize the upper-bound function u_k to be the original smooth function g. As long as $g(x_1, \cdots, x_K)$ satisfies the BSC property, Theorem 3.1 carries over. As mentioned in Section 2.2, the BSC property is fairly mild and is satisfied in many engineering applications. Nevertheless, we will further relax the BSC condition in the subsequent sections.

4 The BSUM for Single Block Problem

4.1 The SUM Algorithm

In this section, we consider the following single-block problem with $K = 1$:

$$\begin{aligned} \min \quad & f(x) := g(x) + h(x) \\ \text{s.t.} \quad & x \in X. \end{aligned} \tag{4.1}$$

In this case the BSUM algorithm reduces to to the so-called successive upper-bound minimization (SUM) algorithm [6], listed in the following table.

The Successive Upper-Bound Minimization (SUM) Algorithm

At each iteration $r+1$, do:

$$x^{r+1} \in \min_{x \in X} \; u\left(x; x^r\right) + h(x). \tag{4.2}$$

Let us make the following assumptions on the function $u(v; x)$.

Assumption C.

(a) $u(x; x) = g(x), \quad \forall\, x \in X.$

(b) $u(v;x) \geq g(v), \quad \forall\, v \in X,\ \forall\, x \in X.$

(c) $\nabla u(x;x) = \nabla g(x), \quad \forall\, x \in X.$

(d) For any given x, $u(v;x)$ has Lipschitz continuous gradient, that is

$$\|\nabla u(v;x) - \nabla u(\hat{v};x)\| \leq L\|v - \hat{v}\|,\ \forall\ \hat{v},\ v \in X, \forall\ x \in X, \tag{4.3}$$

where $L > 0$ is some constant.

Compared to Assumption B, Assumption C does not require $u(v;x)$ to be strongly convex in v, nor $\nabla u(v;x)$ to be Lipschitz continuous over x. Notice that the Lipschitz continuity of ∇u given in (4.3) implies the Lipschitz continuity of ∇g.

Proposition 4.1 *Suppose $g(x)$ is convex, and $u(v;x)$ satisfies Assumption C. Then we must have*

$$\|\nabla g(v) - \nabla g(x)\| \leq L\|v - x\|, \quad \forall\ x, v \in X. \tag{4.4}$$

That is, ∇g is Lipschitz continuous with the coefficient no larger than L.

Proof. Utilizing Assumption C, we must have

$$\begin{aligned} g(v) - g(x) &\leq u(v;x) - u(x;x) \\ &\leq \langle \nabla u(x;x), v - x\rangle + \frac{L}{2}\|x - v\|^2 \\ &= \langle \nabla g(x), v - x\rangle + \frac{L}{2}\|x - v\|^2,\ \forall\ x, v \in X. \end{aligned}$$

Further, using the convexity of g we have

$$g(v) - g(x) \geq \langle \nabla g(x), v - x\rangle,\ \forall\ x, v \in X.$$

Combining these two inequalities we obtain

$$0 \leq g(v) - g(x) - \langle \nabla g(x), v - x\rangle \leq \frac{L}{2}\|x - v\|^2,\ \forall\ x, v \in X. \tag{4.5}$$

Similar to [34, Theorem 2.1.5], we construct the following function

$$\phi(x) = g(x) - \langle \nabla g(v), x\rangle.$$

Clearly $v \in \arg\min \phi(x)$. We have

$$\phi(v) \leq \phi\left(x - \frac{1}{L}\nabla g(x)\right) \leq \phi(x) - \frac{1}{2L}\|\nabla \phi(x)\|^2 \tag{4.6}$$

where the first inequality is due to the optimality of v and the second inequality uses (4.5). Plugging in the definition of $\phi(x)$ and $\phi(v)$ we have

$$g(v) - \langle \nabla g(v), v\rangle \leq g(x) - \langle \nabla g(v), x\rangle - \frac{1}{2L}\|\nabla g(v) - \nabla g(x)\|^2.$$

Since the above inequality is true for any $x, v \in X$, we can interchange x and v and obtain

$$g(x) - \langle \nabla g(x), x\rangle \leq g(v) - \langle \nabla g(x), v\rangle - \frac{1}{2L}\|\nabla g(v) - \nabla g(x)\|^2.$$

Adding these two inequalities we obtain

$$\frac{1}{L}\|\nabla g(x) - \nabla g(v)\|^2 \leq \langle \nabla g(x) - \nabla g(v), x - v\rangle \leq \|\nabla g(x) - \nabla g(v)\|\|x - v\|.$$

Cancelling $\|\nabla g(x) - \nabla g(v)\|$ we arrive at the desired results. **Q.E.D.**

We remark that this result is only true when both $g(\cdot)$ and $u(\cdot;\cdot)$ are convex functions.

Our main result is that the SUM algorithm converges sublinearly under Assumption C, *without* the strong convexity of the upper-bound function $u(v;x)$ in v. The proof of this claim is an extension of Theorem 3.1, therefore we will only provide its key steps. Observe that the following is true

$$\begin{aligned} f(x^r) - f(x^{r+1}) &\overset{\text{(i)}}{\geq} f(x^r) - \big(u(x^{r+1};x^r) + h(x^{r+1})\big) \\ &\overset{\text{(ii)}}{\geq} f(x^r) - \big(u(\widetilde{x}^{r+1};x^r) + h(\widetilde{x}^{r+1})\big) \overset{\text{(iii)}}{\geq} \frac{\gamma}{2}\|x^r - \widetilde{x}^{r+1}\|^2 \end{aligned} \tag{4.7}$$

where $\widetilde{x}^{r+1}$ is the iterate obtained by solving the following auxiliary problem for any $\gamma > 0$

$$\widetilde{x}^{r+1} = \arg\min_{x\in X} u(x;x^r) + h(x) + \frac{\gamma}{2}\|x - x^r\|^2. \tag{4.8}$$

In (4.7), (i) is true because $u(x;y)$ is an upper-bound function for $g(x)$ satisfying Assumption C(b); (ii) is true because x^{r+1} is a minimizer of problem (4.2); (iii) is true due to the fact that $\widetilde{x}^{r+1}$ is the optimal solution of (4.8) while x^r is a feasible solution.

Then we bound $f(x^{r+1})$ using $f(\widetilde{x}^{r+1})$. We have

$$\begin{aligned} f(x^{r+1}) &\leq u(x^{r+1};x^r) + h(x^{r+1}) \\ &\overset{\text{(i)}}{\leq} u(\widetilde{x}^{r+1};x^r) + h(\widetilde{x}^{r+1}) \\ &\overset{\text{(ii)}}{\leq} u(x^r;x^r) + \langle \nabla u(x^r;x^r), \widetilde{x}^{r+1} - x^r\rangle + \frac{L}{2}\|\widetilde{x}^{r+1} - x^r\|^2 + h(\widetilde{x}^{r+1}) \\ &\overset{\text{(iii)}}{\leq} g(\widetilde{x}^{r+1}) + \langle \nabla u(x^r;x^r), \widetilde{x}^{r+1} - x^r\rangle + \langle \nabla g(\widetilde{x}^{r+1}), x^r - \widetilde{x}^{r+1}\rangle + L\|\widetilde{x}^{r+1} - x^r\|^2 + h(\widetilde{x}^{r+1}) \\ &\overset{\text{(iv)}}{=} g(\widetilde{x}^{r+1}) + \langle \nabla g(\widetilde{x}^{r+1}) - \nabla g(x^r), x^r - \widetilde{x}^{r+1}\rangle + L\|\widetilde{x}^{r+1} - x^r\|^2 + h(\widetilde{x}^{r+1}) \\ &\overset{\text{(v)}}{\leq} f(\widetilde{x}^{r+1}) + L\|\widetilde{x}^{r+1} - x^r\|^2 \end{aligned}$$

where (i) is due to the optimality of x^{r+1} for problem (4.2); (ii) uses the gradient Lipschitz continuity of $u(\cdot; x^r)$; (iii) uses the fact that $u(x^r; x^r) = g(x^r)$, the gradient Lipschitz continuity of $g(\cdot)$ derived in Proposition 4.1; (iv) uses the fact that $\nabla u(x^r; x^r) = \nabla g(x^r)$ (cf. Assumption C(c)); (v) uses the convexity of $g(\cdot)$.

Utilizing this bound, we derive the estimate of the cost-to-go

$$\begin{aligned}
f(x^{r+1}) - f(x^*) &\le f(\widetilde{x}^{r+1}) - f(x^*) + L\|\widetilde{x}^{r+1} - x^r\|^2 \\
&\le \left\langle \nabla g(\widetilde{x}^{r+1}), \widetilde{x}^{r+1} - x^* \right\rangle + h(\widetilde{x}^{r+1}) - h(x^*) + L\|\widetilde{x}^{r+1} - x^r\|^2 \\
&= \left\langle \nabla g(\widetilde{x}^{r+1}) - \nabla g(x^r), \widetilde{x}^{r+1} - x^* \right\rangle + L\|\widetilde{x}^{r+1} - x^r\|^2 \\
&\quad + \left\langle \nabla g(x^r) - \nabla \left(u(\widetilde{x}^{r+1}; x^r) + \frac{\gamma}{2}\|\widetilde{x}^{r+1} - x^r\|^2 \right), \widetilde{x}^{r+1} - x^* \right\rangle \\
&\quad + h(\widetilde{x}^{r+1}) - h(x^*) + \left\langle \nabla \left(u(\widetilde{x}^{r+1}; x^r) + \frac{\gamma}{2}\|\widetilde{x}^{r+1} - x^r\|^2 \right), \widetilde{x}^{r+1} - x^* \right\rangle \\
&\overset{\text{(i)}}{\le} \left\langle \nabla g(\widetilde{x}^{r+1}) - \nabla g(x^r), \widetilde{x}^{r+1} - x^* \right\rangle + L\|\widetilde{x}^{r+1} - x^r\|^2 \\
&\quad + \left\langle \nabla u(x^r; x^r) - \nabla u(\widetilde{x}^{r+1}; x^r), \widetilde{x}^{r+1} - x^* \right\rangle - \gamma \left\langle \widetilde{x}^{r+1} - x^r, \widetilde{x}^{r+1} - x^* \right\rangle \\
&\overset{\text{(ii)}}{\le} (2L + \gamma)\|\widetilde{x}^{r+1} - x^r\| R + L\|\widetilde{x}^{r+1} - x^r\| \|\widetilde{x}^{r+1} - x^* + x^* - x^r\| \\
&\le (4L + \gamma)\|\widetilde{x}^{r+1} - x^r\| R.
\end{aligned}$$

Here (i) is due to the optimality of $\widetilde{x}^{r+1}$ to the problem (4.8); in (ii) we have used (4.4), Cauchy-Schwartz inequality and the definition of R (it is easy to show that $f(\widetilde{x}^{r+1}) \le f(x^r) \le f(x^0)$, hence $\|\widetilde{x}^{t+1} - x^*\| \le R$ for all t).

Combining the above two inequalities, we obtain

$$\Delta^r - \Delta^{r+1} \ge \frac{\gamma}{2R^2(4L + \gamma)^2}(\Delta^{r+1})^2, \quad \forall \gamma > 0. \tag{4.9}$$

Maximizing over γ (with $\gamma = 4L$), we have

$$\Delta^r - \Delta^{r+1} \ge \frac{1}{32R^2L}(\Delta^{r+1})^2 := \sigma_4(\Delta^{r+1})^2. \tag{4.10}$$

Using the same derivation as in Theorem 3.1, we obtain

$$\Delta^{r+1} \le \frac{c_4}{\sigma_4}\frac{1}{r}, \quad \text{with} \quad \sigma_4 = \frac{1}{32R^2L}, \quad c_4 := \max\{4\sigma_4 - 2, f(x^1) - f^*, 2\}. \tag{4.11}$$

4.2 Application

To see the importance of the above result, consider the well-known method of Iterative Reweighted Least Squares (IRLS) [24, 35]. The IRLS is a popular algorithm used for solving problems such as

sparse recovery and Fermat-Weber problem; see [24, Section 4] for a few applications. Consider the following problem

$$\min_x \quad h(x) + \sum_{j=1}^{\ell} \|A_j x + b_j\|_2, \quad \text{s.t.} \quad x \in X \tag{4.12}$$

where $A_j \in \mathbb{R}^{k_i \times m}$, $b_j \in \mathbb{R}^{k_i}$, $X \subseteq \mathbb{R}^m$, and $h(x)$ is some convex function not necessarily smooth. Let us introduce a constant $\eta > 0$ and consider a *smooth approximation* of problem (4.12):

$$\min_x \quad h(x) + g(x) := h(x) + \sum_{j=1}^{\ell} \sqrt{\|A_j x + b_j\|_2^2 + \eta^2}, \quad \text{s.t.} \quad x \in X. \tag{4.13}$$

The IRLS algorithm generates the following iterates

$$x^{r+1} = \arg\min_{x \in X} \left\{ h(x) + \frac{1}{2} \sum_{j=1}^{\ell} \frac{\|A_j x + b_j\|^2 + \eta^2}{\sqrt{\|A_j x^r + b_j\|^2 + \eta^2}} \right\}. \tag{4.14}$$

It is known that the IRLS iteration is equivalent to a BCM method applied to the following two-block problem (i.e., the first block is x and the second block is $\{z_j\}_{j=1}^{\ell}$)

$$\begin{aligned} \min \quad & h(x) + \frac{1}{2} \sum_{j=1}^{\ell} \left(\frac{\|A_j x + b_j\|^2 + \eta^2}{z_j} + z_j \right) \\ \text{s.t.} \quad & x \in X, \quad z_j \in [\eta/2, \infty), \ \forall\ j. \end{aligned} \tag{4.15}$$

Utilizing such two-block BCM interpretation, the author of [24] shows that the IRLS converges sublinearly when $h(x)$ has Lipschitz continuous gradient; see [24, Theorem 4.1].

Differently from [24], here we take a new perspective. We argue that the IRLS is in fact the SUM algorithm in disguise, therefore our simple iteration complexity analysis given in Section 4.1 for SUM can be directly applied.

Let us consider the following function:

$$u(x; x^r) = \frac{1}{2} \sum_{j=1}^{\ell} \left(\frac{\|A_j x + b_j\|^2 + \eta^2}{\sqrt{\|A_j x^r + b_j\|^2 + \eta^2}} + \sqrt{\|A_j x^r + b_j\|^2 + \eta^2} \right). \tag{4.16}$$

It is clear that $g(x^r) = u(x^r; x^r)$, so Assumption C(a) is satisfied. To verify Assumption C(b), we apply the arithmetic-geometric inequality, and have

$$\begin{aligned} u(x; x^r) &= \frac{1}{2} \sum_{j=1}^{\ell} \left(\frac{\|A_j x + b_j\|^2 + \eta^2}{\sqrt{\|A_j x^r + b_j\|^2 + \eta^2}} + \sqrt{\|A_j x^r + b_j\|^2 + \eta^2} \right) \\ &\geq \sum_{j=1}^{\ell} \sqrt{\|A_j x + b_j\|^2 + \eta^2} = g(x), \ \forall\ x \in X. \end{aligned}$$

Assumptions C(c)-(d) are also easy to verify. Note that the matrices A_j's do not necessarily have full column rank, so $u(x;x^r)$ may not be strongly convex over $x \in X$. Nevertheless, $u(x;x^r)$ defined in (4.16) is indeed an upper bound function for the smooth function $g(x)$, and we have shown that it satisfies Assumptions C. It follows that the iteration (4.14) corresponds to a single-block BSUM algorithm. Our analysis leading to (4.11) suggests that this algorithm converges in a sublinear rate, even when $h(x)$ is a nonsmooth function. To be more specific, for this problem we have

$$L = \frac{1}{\eta}\rho_{\max}\left(\sum_{j=1}^{\ell} A_j^T A_j\right)$$

Therefore the rate can be expressed as

$$\Delta^{r+1} \leq \max\{4\sigma_4 - 2, f(x^1) - f(x^*), 2\}\frac{32R^2\rho_{\max}\left(\sum_{j=1}^{\ell} A_j^T A_j\right)}{\eta r}. \tag{4.17}$$

Note that compared with the result derived in [24, Theorem 4.1] which is based on transforming the IRLS algorithm to the two-block BCM problem (4.15), our analysis is based on the key insight of the equivalence between IRLS and the single block BSUM, and it is significantly simpler. Further we do not require $h(x)$ to be smooth, while the result in [24, Theorem 4.1] additionally requires that the gradient of $h(x)$ is Lipschitz continuous [2].

5 The BSUM for Two Block Problem

5.1 Iteration Complexity for 2-Block BSUM

In this section, we consider the following two-block problem ($K = 2$), which is a special case of problem (1.1):

$$\begin{array}{ll} \min & f(x_1, x_2) := g(x_1, x_2) + h_1(x_1) + h_2(x_2) \\ \text{s.t.} & x_1 \in X_1,\ x_2 \in X_2. \end{array} \tag{5.1}$$

This problem has many applications, such as the special case of Example 2.1 with two users, the two-block formulation of the IRLS algorithm (4.15) or the example presented in [24, Section 5]. Throughout this section, we assume that Assumption A(a) is true. We make the following additional assumptions about problem (5.1).

Assumption D.

[2] It appears that the proof in [24, Theorem 4.1] can be modified to allow nonsmooth h, just that it is not explicitly mentioned in the paper. But as it stands, the bound in [24, Theorem 4.1] is explicitly dependent on the Lipschitz constant of the gradient of h, while the bound we derived here in (4.17) is not.

(a) The problem $\min_{x_2 \in X_2} f(x_1, x_2)$ has a unique solution.

(b) The gradient of $g(x_1, x_2)$ with respect to x_1 is Lipschitz continuous, i.e.,

$$\|\nabla_1 g(x_1, x_2) - \nabla_1 g(v_1, x_2)\| \leq M_1 \|x_1 - v_1\|.$$

Note that here we do not require that the gradient of $g(\cdot)$ with respect to the second block to be Lipschitz continuous.

We first show that for this problem BSUM with G-S update rule is able to achieve sublinear rate without the BSC condition or the Lipschitz continuity of $\nabla_2 g(x_1, x_2)$. Under the same assumption, we further show that it is possible to accelerate the BSUM method with G-S rule to get an $\mathcal{O}(1/r^2)$ iteration complexity.

In table given below we list the two-block BSUM algorithm with G-S update rule.

The G-S 2-block BSUM for problem (5.1)

At each iteration $r+1$, update the variable blocks by:

$$\begin{aligned} x_2^{r+1} &= \arg\min_{x_2 \in X_2} u_2(x_2; x_1^r, x_2^r) + h_2(x_2) \\ x_1^{r+1} &\in \arg\min_{x_1 \in X_1} u_1(x_1; x_1^r, x_2^{r+1}) + h_1(x_1). \end{aligned} \tag{5.2}$$

Unfortunately for the problem of interest here the rate analysis provided in Theorem 3.1 is no longer applicable because $\nabla_2 g(x_1, x_2)$ may not be Lipschitz continuous, and both subproblems may not be strongly convex. To analyze the convergence rate, let us consider the following special choices of the upper bound where $u_1(x_1; x)$ satisfies Assumption B(a)-(c) and the Lipschtiz continuous gradient condition (2.8), restated below for convenience

$$\|\nabla u_1(x_1; x) - \nabla u_1(v_1; x)\| \leq L_1 \|x_1 - v_1\|, \ \forall \ x_1, v_1 \in X_1, \ \forall \ x \in X. \tag{5.3}$$

By utilizing the argument in Proposition 4.1, we can show that $L_1 \geq M_1$, therefore the following is true as well

$$\|\nabla_1 g(x_1, x_2) - \nabla_1 g(v_1, x_2)\| \leq L_1 \|x_1 - v_1\|.$$

Further we do not use any upper bound for the second block, i.e., we let

$$u_2(v_2; x) = g(v_2, x_1), \ \forall \ x_1 \in X_1, \ v_2 \in X_2.$$

This suggests that the x_2-block is minimized exactly.

To analyze the algorithm, it is convenient to consider an equivalent *single-block* problem, which only takes x_1 as its variable:

$$\min_{x_1 \in X_1} \quad \ell(x_1) + h_1(x_1) := \min_{x_1 \in X_1} \min_{x_2 \in X_2} f(x_1, x_2), \tag{5.4}$$

where we have defined $\ell(x_1) := \min_{x_2 \in X_2} g(x_1, x_2) + h_2(x_2)$. Let us denote an optimal solution of the inner problem $\min_{x_2 \in X_2} f(x_1, x_2)$ by the mapping: $x_2^*(x_1) : X_1 \to X_2$, which is a singleton for any $x_1 \in X_1$ by Assumption D(a). Next we analyze problem (5.4).

Let us define a new function

$$u(v_1; x_1) := u_1(v_1; x_1, x_2^*(x_1)) + h_2(x_2^*(x_1)). \tag{5.5}$$

First we argue that for all $x_1, v_1 \in X_1$, $u(v_1; x_1)$ is an upper bound for $\ell(v_1)$, and it satisfies Assumption C given in Section 4.1. Clearly Assumption C(a) is true because

$$\ell(x_1) = g(x_1, x_2^*(x_1)) + h_2(x_2^*(x_1)) = u_1(x_1; x_1, x_2^*(x_1)) + h_2(x_2^*(x_1)) = u(x_1; x_1) \tag{5.6}$$

where the second equality is due to the fact that $u_1(x_1; x)$ is an upper bound function for $g(\cdot, x_2)$. The last equality is from the definition of $u(\cdot; \cdot)$.

Assumption C(b) is true because

$$u(v_1; x_1) = u_1(v_1; x_1, x_2^*(x_1)) + h_2(x_2^*(x_1)) \geq g(v_1, x_2^*(x_1)) + h_2(x_2^*(x_1)) \geq \min_{x_2} g(v_1, x_2) + h_2(x_2).$$

To verify Assumption C(c), recall that by Assumption D the inner problem $\min_{x_2 \in X_2} f(x_1, x_2)$ has a *unique* solution, or equivalently for any given $x_1 \in X_1$, the mapping $x_2^*(x_1)$ is a singleton. By applying [36, Corollary 4.5.2–4.5.3], we obtain

$$\nabla \ell(x_1) = \nabla_1 g\left(x_1, \widetilde{x}_2\right), \ \forall \ x_1 \in X_1 \tag{5.7}$$

where $\widetilde{x}_2 = \arg\min_{x_2 \in X_2} f(x_1, x_2)$. Therefore, we must have

$$\nabla \ell(x_1) = \nabla_1 g\left(x_1, \widetilde{x}_2\right) = \nabla u_1(x_1; x_1, \widetilde{x}_2) = \nabla u_1(x_1; x_1, x_2^*(x_1)) = \nabla u(x_1; x_1),$$

where the second equality comes from the fact that $u_1(\cdot; \cdot)$ satisfies Assumption B(c); the third inequality is because $\widetilde{x}_2 = x_2^*(x_1)$ by definition; the last equality is from (5.8). This verifies Assumption C(c).

The Lipschitz continuous gradient condition (with constant L_1) in Assumption C(d) can be verified by combining (5.3) and the following equality

$$\nabla u_1(v_1; x_1, x_2^*(x_1)) = \nabla u(v_1; x_1), \ \forall \ v_1, x_1 \in X_1. \tag{5.8}$$

Now that we have verified that $u(v_1; x_1)$ given in (5.5) satisfies Assumption C, then Proposition 4.1 implies $\ell(\cdot)$ also has Lipschitz continuous gradient with constant L_1, that is

$$\|\nabla \ell(x_1) - \nabla \ell(v_1)\| \leq L_1 \|x_1 - v_1\|, \ \forall \ v_1, x_1 \in X.$$

At this point it is clear that the 2-block BSUM algorithm with G-S update rule is in fact the SUM algorithm given in Section 4.1, where the iterates are generated by

$$x_1^{r+1} \in \arg\min u(x_1; x_1^r). \tag{5.9}$$

By applying the argument leading to (4.11), we conclude that the 2-block BSUM in which the second block performs an exact minimization converges sublinearly. Also note that neither subproblems in (5.1) is required to be strongly convex, which suggests that the BCM applied to problem (5.1) converges sublinearly without block strong convexity. The precise statement is given in the following corollary.

Corollary 5.1 *Assume that Assumption A(a) and D hold for problem* (5.1)*. Then we have the following.*

1. *Suppose that* $u_2(v_2; x) = g(x_1, v_2)$ *for all* $v_2 \in X_2,\ x \in X$ *and that* $u_1(v_1; x)$ *satisfies Assumption B(a)-(c) and the Lipschtiz continuous gradient condition* (2.8)*. Then the 2-block BSUM algorithm with G-S rule is equivalent to the SUM algorithm and converges sublinearly, i.e.,*

$$\Delta^{r+1} \leq \frac{c_4}{\sigma_4}\frac{1}{r} \tag{5.10}$$

   where c_4 *and* σ_4 *is given in* (4.11)*, with* L *in* (4.11) *replaced by* L_1*.*

2. *The BCM algorithm applied to* (2.8) *converges sublinearly with the same rate, again with* L *in* (4.11) *replaced by* L_1*.*

5.2 Accelerating the 2-Block BSUM

Next we show that it is possible to accelerate the above G-S BSUM iterations (5.2) to obtain an improved rate. The main idea is again to use the *single-block* interpretation of the 2-block BSUM.

Let us pick the following upper bound function for x_1

$$u_1(v_1; x) = \langle \nabla_1 g(x_1, x_2), v_1 - x_1 \rangle + h_1(v_1) + \frac{M_1}{2}\|v_1 - x_1\|^2, \tag{5.11}$$

where M_1 is the Lipschitz constant for $\nabla_1 g(x_1, x_2)$. Then utilizing the single block interpretation of the 2-block BSUM (5.9) we must have

$$\begin{aligned} x_1^{r+1} &= \arg\min_{x_1 \in X_1} u(x_1; x_1^r) = \arg\min_{x_1 \in X_1} u_1(x_1; x_1^r, x_2^{r+1}) + h_1(x_1) \\ &= \mathrm{prox}_{h_1 + I_{X_1}}^{M_1}\left[x_1^r - \frac{1}{M_1}\nabla g(x_1^r, x_2^{r+1})\right] \\ &= \mathrm{prox}_{h_1 + I_{X_1}}^{M_1}\left[x_1^r - \frac{1}{M_1}\nabla \ell(x_1^r)\right] \end{aligned} \tag{5.12}$$

where the last inequality comes from (5.7).

This observation lends itself to a simple acceleration scheme by applying known Nesterov-type acceleration schemes. The scheme, named Accelerated 2-Block BSUM (A-2BSUM) Algorithm, is described in the following table. The $\mathcal{O}(1/r^2)$ iteration complexity of the algorithm can be obtained directly from existing analysis for accelerated proximal gradient; see, e.g., [34, 37, 38]. It is interesting to see that for the two block problem (5.1), the acceleration scheme developed here as well as the resulting rate are not dependent on the Lipschitz constant for the gradient of the second block, since we do not require $\nabla_2 g(x_1, x_2)$ to be Lipschitz continuous.

The A-2BSUM Algorithm

At any given iteration $r > 1$, do the following:
S1) Choose $\theta^r = \frac{2}{r+1}$;
S2) $v_1^r = (1 - \theta^{r-1})x_1^{r-1} + \theta^r(w_1^{r-1})$;
S3) $x_2^r = \arg\min_{x_2 \in X_2} f(v_1^r, x_2)$;
S4) $x_1^r = \arg\min_{x_1 \in X_1} u_1(x_1; v_1^r, x_2^r)$, where u_1 is given in (5.11);
S5) $w_1^r = x_1^{r-1} + \frac{1}{\theta^r}(x_1^r - x_1^{r-1})$.

To conclude this section, we note that the schemes and analysis developed in this section are special in the sense that they heavily rely on the fact that $K = 2$, and the resulting transformation to the single block problem. It is unclear whether the same sublinear iteration complexity holds for a general K without the BSC condition, or if the algorithm can be accelerated for any K; see [5,38] for related discussions.

6 Analysis of the BCM without Per-Block Strong Convexity

In this section, we consider the BCM algorithm below, which is the BSUM algorithm without using approximation for each block. We analyze its iteration complexity *without* the BSC assumption.

The Block Coordinate Minimization (BCM) Algorithm

At each iteration $r + 1$, pick an index set $\mathcal{C}^{r+1}$; update the variable blocks by:

$$x_k^{r+1} \begin{cases} \in \min_{x_k \in X_k} \; g\left(x_k, w_{-k}^{r+1}\right) + h_k(x_k), & \text{if } k \in \mathcal{C}^{r+1}; \\ = x_k^r, & \text{if } k \notin \mathcal{C}^{r+1}. \end{cases}$$

In the absence of the BSC property, there can be multiple optimal solutions for each subproblem. This makes it tricky to establish the convergence of BCM. Specifically, in the context of the three-step analysis framework presented herein, it is difficult to bound the sufficient descent of the

objective using the size of of the successive iterates (as per Lemma 3.1). In this section, we overcome this obstacle by developing several variants of the sufficient descent estimate step. We first show that BCM with MBI, G-S and E-C rules has an iteration complexity of $O(1/r)$ for problem (1.1) without the BSC condition. Further, we argue that for certain special classes of problem (1.1), this sublinear rate can be improved in terms of the dependence on K for the G-S/E-C rules. Throughout this section we will impose Assumption A.

We first consider the MBI rule. We notice that the following is true

$$f(x^r) - f(x^{r+1}) \overset{\text{(i)}}{\geq} f(x^r) - f(\bar{x}^{r+1}) \overset{\text{(ii)}}{\geq} \frac{\gamma}{K}\|x^r - \hat{x}^{r+1}\|^2, \tag{6.1}$$

where $\bar{x}^{r+1}$ is the iterates obtained by any BSUM algorithm with MBI rule; $\hat{x}^{r+1}$ is defined in (2.2). In the above expression (ii) can be obtained using Lemma 3.1, while (i) is true because we used the exact minimization in each step. Then it is straightforward to establish, using the additional assumption that h is Lipschitz continuous, the same rate stated in part (3) of Theorem 3.1.

Next we show that the BCM algorithm with the G-S and E-C rules also achieves an $\mathcal{O}(1/r)$ iteration complexity, without the BSC assumption.

6.1 A General Analysis for G-S and E-C rules

The main difficulty in analyzing the BCM without the BSC is that the size of the difference of the successive iterates is no longer a good measure of the "sufficient descent". Indeed, due to the lack of per-block strong convexity, it is possible that a block variable travels a long distance without changing the objective value (i.e., it stays in the per-block optimal solution set).

Below we analyze the iteration complexity of BCM. We need to make use of the following key inequality due to Nesterov [34]; also see (4.5) for a proof. From Assumption A we know that g is convex and has Lipschitz continuous gradient with constant M, then we must have have

$$g(x) - g(v) \geq \langle \nabla g(v), x - v\rangle + \frac{1}{2M}\|\nabla g(v) - \nabla g(x)\|^2, \ \forall\ v, x \in X. \tag{6.2}$$

Utilizing this inequality, the sufficient descent estimate is given by the following lemma.

Lemma 6.1 *Suppose Assumption A holds. Then for BCM with either G-S rule or the E-C rule, we have that for all* $r \geq 1$

$$\Delta^r - \Delta^{r+1} \geq \frac{1}{2M}\sum_{k=1}^{K}\|\nabla g(w_k^{r+1}) - \nabla g(w_{k+1}^{r+1})\|^2. \tag{6.3}$$

Proof. Suppose that $k \notin \mathcal{C}^{r+1}$, then we have the following trivial inequality

$$f(w_k^{r+1}) - f(w_{k+1}^{r+1}) \geq \frac{1}{2M}\|\nabla g(w_k^{r+1}) - \nabla g(w_{k+1}^{r+1})\|^2 \tag{6.4}$$

as both sides of the inequality are zero.

Suppose $k \in \mathcal{C}^{r+1}$. Then by (6.2), we have that

$$\begin{aligned}
&f(w_k^{r+1}) - f(w_{k+1}^{r+1}) \\
&\geq \langle \nabla g(w_{k+1}^{r+1}), w_k^{r+1} - w_{k+1}^{r+1}\rangle + h(x_k^r) - h(x_k^{r+1}) + \frac{1}{2M}\|\nabla g(w_k^{r+1}) - \nabla g(w_{k+1}^{r+1})\|^2 \\
&\overset{\text{(i)}}{\geq} \langle \nabla_k g(w_{k+1}^{r+1}), x_k^r - x_k^{r+1}\rangle + h_k(x_k^r) - h_k(x_k^{r+1}) + \frac{1}{2M}\|\nabla g(w_k^{r+1}) - \nabla g(w_{k+1}^{r+1})\|^2 \\
&\overset{\text{(ii)}}{\geq} \frac{1}{2M}\|\nabla g(w_k^{r+1}) - \nabla g(w_{k+1}^{r+1})\|^2
\end{aligned} \tag{6.5}$$

where (i) is because w_{k+1}^{r+1} and w_k^{r+1} only differs by a single block; (ii) is due to the optimality of x_k^{t+1}. Summing over k, we have

$$f(x^r) - f(x^{r+1}) \geq \sum_{k=1}^{K} \frac{1}{2M}\|\nabla g(w_k^{r+1}) - \nabla g(w_{k+1}^{r+1})\|^2. \tag{6.6}$$

This completes the proof of this lemma. **Q.E.D.**

Lemma 6.2 *Suppose Assumptions A is satisfied. Then*

1. *For the BCM with the G-S update rule, we have*

$$(\Delta^{r+1})^2 \leq 2K^2R^2 \sum_{k=1}^{K} \|\nabla g(w_{k+1}^{r+1}) - \nabla g(w_k^{r+1})\|^2, \ \forall x^* \ \in X^*.$$

2. *For the BCM with the period-T E-C update rule, we have*

$$(\Delta^{r+T})^2 \leq 2TK^2R^2 \sum_{k=1}^{K}\sum_{t=1}^{T} \|\nabla g(w_{k+1}^{r+t}) - \nabla g(w_k^{r+t})\|^2, \ \forall\, x^* \in X^*.$$

Proof. We only show the second part of the claim, as the proof for the first part is simply a special case. Define a new index set $\{r_k\}$ as in (3.14). Recall that we have $x_k^{r_k} = x_k^{r+T}$, for all k.

We have the following series of inequalities

$$
\begin{aligned}
& f(x^{r+T}) - f(x^*) \\
& \leq \sum_{k=1}^{K} \langle \nabla_k g(x^{r+T}), x_k^{r+T} - x_k^* \rangle + \sum_{k=1}^{K} h_k(x_k^{r_k}) - h_k(x_k^*) \\
& = \sum_{k=1}^{K} \langle \nabla_k g(x^{r+T}) - \nabla_k g(w_{k+1}^{r_k}), x_k^{r+T} - x_k^* \rangle + \langle \nabla_k g(w_{k+1}^{r_k}), x_k^{r+T} - x_k^* \rangle + h_k(x_k^{r_k}) - h_k(x_k^*) \\
& \overset{\text{(i)}}{\leq} \sum_{k=1}^{K} \langle \nabla_k g(x^{r+T}) - \nabla_k g(w_{k+1}^{r_k}), x_k^{r+T} - x_k^* \rangle \\
& \leq \sum_{k=1}^{K} \|\nabla g(x^{r+T}) - \nabla g(w_{k+1}^{r_k})\| \|x_k^{r+T} - x_k^*\| \\
& \leq \sum_{k=1}^{K} \sum_{t=1}^{T} \sum_{j=1}^{K} \|\nabla g(w_{j+1}^{r+t}) - \nabla g(w_j^{r+t})\| \|x_k^{r+T} - x_k^*\| \\
& \leq \sum_{t=1}^{T} \sum_{j=1}^{K} \|\nabla g(w_{j+1}^{r+t}) - \nabla g(w_j^{r+t})\| \sum_{k=1}^{K} \|x_k^{r+T} - x_k^*\|
\end{aligned}
$$

where in (i) we have used the optimality of $x_k^{r_k}$ and $x_k^{r_k} = x_k^{r+T}$, for all k. Then taking the square on both sides, we obtain

$$
\big(f(x^{r+T}) - f(x^*)\big)^2 \leq TK^2R^2 \sum_{t=1}^{T} \sum_{k=1}^{K} \|\nabla g(w_{k+1}^{r+t}) - \nabla g(w_k^{r+t})\|^2. \tag{6.7}
$$

The proof is complete. **Q.E.D.**

Combining these two results, and utilizing the technique in Theorem 3.1, we readily have the following main result for BCM.

Theorem 6.1 *Suppose Assumption A holds true. We have the following.*

1. *Let $\{x^r\}$ be the sequence generated by the BCM algorithm with G-S rule. Then we have*

$$
\Delta^r = f(x^r) - f^* \leq \frac{c_5}{\sigma_5} \frac{1}{r}, \ \forall \ r \geq 1, \tag{6.8}
$$

where the constants are given below

$$
\begin{aligned}
c_5 &= \max\{4\sigma_5 - 2, f(x^1) - f^*, 2\}, \\
\sigma_5 &= \frac{1}{2MK^2R^2},
\end{aligned} \tag{6.9}
$$

2. *Let $\{x^r\}$ be the sequence generated by the BCM algorithm with E-C rule. Then we have*

$$\Delta^r = f(x^r) - f^* \leq \frac{c_6}{\sigma_6}\frac{1}{r-T},\ \forall\ r > T, \tag{6.10}$$

where the constants are given below

$$\begin{aligned} c_6 &= \max\{4\sigma_6 - 2, f(x^1) - f^*, 2\}, \\ \sigma_6 &= \frac{1}{2K^2TR^2M}. \end{aligned} \tag{6.11}$$

6.2 Special Case: The Constrained Nonsmooth Composite Problem

The rate derived in the previous subsection is inversely proportional to K^2, which is worse than most of the rates derived so far for problems with the BSC assumption. In the following two subsections we sharpen the above results for two special problems of (1.1).

We first make the following additional assumption on the smooth part of the problem (1.1) (besides Assumption A). Suppose that $g(x)$ takes the following form

$$\begin{aligned} g(x) &= \sum_{i=1}^{I} g^i(x_1, x_2, \cdots, x_K) + \sum_{k=1}^{K} b_k^T x_k \\ &= \sum_{i=1}^{I} \ell^i(A_1^i x_1, A_2^i x_2, \cdots, A_k^i x_K) + \sum_{k=1}^{K} b_k^T x_k \end{aligned} \tag{6.12}$$

where the smooth function $\ell^i(y_1^i, \cdots, y_K^i)$ is a *composite* of a strongly convex function and a linear mapping. Specifically, $\ell^i(\cdot)$ satisfies the following conditions.

Assumption E.

(a) $\ell^i(\cdot)$ is strongly convex with respect to each block variable y_k^i, with η_k^i as the modulus.

(b) $\nabla_k \ell^i(y_k^i, y_{-k}^i)$ is Lipschitz continuous for all feasible y_k^i,

$$\|\nabla_k \ell^i(y_k^i, y_{-k}^i) - \nabla_k \ell^i(y_k^i, \tilde{y}_{-k}^i)\| \leq P_k^i \|y_{-k}^i - \tilde{y}_{-k}^i\|, \quad \forall\ y_k^i,\ y_{-k}^i,\ \tilde{y}_{-k}^i, \tag{6.13}$$

where P_k^i is the Lipschitz constant.

Note that the smooth part $g(x)$ may not be strongly convex with respect to any block x_k, as A_k^i's can be rank deficient. Two simple examples covered by this family of problems are provided below.

Example 6.1 *The sparse logistic regression (SLR) problem with a compact feasible set and the group LASSO problem are special cases of problem* (1.1) *with composite smooth function in the form*

of (6.12). *More specifically, these problems have the following objective functions, respectively:*

$$f^{SLR}(x) = \sum_{i=1}^{I} \log\left(1 + \exp(-y_i a_i^T x)\right) + \nu\|x\|_1,$$
$$f^{G\text{-}LASSO}(x) = \left\|\sum_{k=1}^{K} A_k x_k - b\right\|^2 + \sum_{k=1}^{K} \nu_k \|x_k\|_2.$$

For the SLR problem, $\nu \geq 0$ is the penalty coefficient; I is the total number of observations; $y_i \in \mathbb{R}$ is the i-th observation; $a_i \in \mathbb{R}^n$ is the i-th data point. For the group LASSO problem, $\{\nu_k \geq 0\}$ are the penalty coefficients; each $A_k \in \mathbb{R}^{m\times n}$ is a data matrix not necessarily having full column rank; and $b \in \mathbb{R}^m$ is the observation vector.

Our analysis consists of similar three main steps as before. To simplify presentation, below we only show the analysis and result for BCM with G-S update rule. We first show the sufficient descent property. By using the short-handed notation:

$$A^i_{-k} w^{r+1}_{-k} := [A^i_1 x^{r+1}_1, \cdots, A^i_{k-1} x^{r+1}_{k-1}, A^i_{k+1} x^r_{k+1}, \cdots A^i_K x^r_K],$$

we have the following series of inequalities

$$\begin{aligned}
&f(x^r_k, w^{r+1}_{-k}) - f(x^{r+1}_k, w^{r+1}_{-k}) \\
&\overset{\text{(i)}}{\geq} \sum_{i=1}^{I} \left(\langle \nabla_k \ell^i(A^i_k x^{r+1}_k, A^i_{-k} w^{r+1}_{-k}), A^i_k(x^r_k - x^{r+1}_k)\rangle + \frac{\eta^i_k}{2}\|A^i_k(x^{r+1}_k - x^r_k)\|^2 \right) \\
&\qquad + \langle b_k, x^r_k - x^{r+1}_k\rangle + h_k(x^r_k) - h_k(x^{r+1}_k) \\
&\overset{\text{(ii)}}{=} \langle \nabla_k g(w^{r+1}_{k+1}), x^r_k - x^{r+1}_k\rangle + h_k(x^r_k) - h_k(x^{r+1}_k) + \sum_{i=1}^{I} \frac{\eta^i_k}{2}\|A^i_k(x^{r+1}_k - x^r_k)\|^2 \\
&\overset{\text{(iii)}}{\geq} \sum_{i=1}^{I} \frac{\eta^i_k}{2}\|A^i_k(x^r_k - x^{r+1}_k)\|^2,
\end{aligned}$$

where in (i) we have used the strong convexity property of $\ell^i(\cdot)$; in (ii) we have used the property that $\nabla_k g^i(w^{r+1}_{k+1}) = (A^i_k)^T \nabla_k \ell^i(A^i_k x^{r+1}_k, A^i_{-k} w^{r+1}_{-k})$; in (iii) we have used the optimality of x^{r+1}_k.

As a result we have

$$f(x^r) - f(x^{r+1}) \geq \frac{1}{2} \sum_{i=1}^{I} \sum_{k=1}^{K} \eta^i_k \|A^i_k(x^r_k - x^{r+1}_k)\|^2. \tag{6.14}$$

It is important to note that the sufficient descent estimate described above is measured by the size of the linearly transformed version of the successive difference of the iterates, as opposed to the size of the successive difference of the iterates given in Lemma 3.1.

Next let us show the cost-to-go estimate. First note that when $g(x)$ is the composite function described above, we have

$$\begin{aligned}
&\|\nabla_k g(x^{r+1}) - \nabla_k g(x_k^{r+1}, w_{-k}^{r+1})\| \\
&= \left\| \sum_{i=1}^{I} (A_k^i)^T \left(\nabla_k \ell^i (A_1^i x_1^{r+1}, \cdots, A_K^i x_K^{r+1}) - \nabla_k \ell^i (A_k^i x_k^{r+1}, A_{-k}^i w_{-k}^{r+1}) \right) \right\| \\
&\le \sum_{i=1}^{I} \sqrt{\left\| A_k^i (A_k^i)^T \right\|} P_k^i \sqrt{\sum_{j=1}^{K} \|A_j^i (x_j^{r+1} - x_j^r)\|^2}.
\end{aligned}$$

We have the following series of inequalities

$$\begin{aligned}
f(x^{r+1}) - f(x^*) &\le \langle \nabla g(x^{r+1}), x^{r+1} - x^* \rangle + h(x^{r+1}) - h(x^*) \\
&= \sum_{k=1}^{K} \langle \nabla_k g(x^{r+1}) - \nabla_k g(x_k^{r+1}, w_{-k}^{r+1}), x_k^{r+1} - x_k^* \rangle \\
&\quad + \sum_{k=1}^{K} \langle \nabla_k g(x_k^{r+1}, w_{-k}^{r+1}), x_k^{r+1} - x_k^* \rangle + h(x^{r+1}) - h(x^*) \\
&\le \sum_{k=1}^{K} \langle \nabla_k g(x^{r+1}) - \nabla_k g(x_k^{r+1}, w_{-k}^{r+1}), x_k^{r+1} - x_k^* \rangle \\
&\le \sum_{i=1}^{I} \sqrt{\sum_{j=1}^{K} \|A_j^i (x_j^{r+1} - x_j^r)\|^2} \sum_{k=1}^{K} \sqrt{\left\| A_k^i (A_k^i)^T \right\|} P_k^i \|x_k^{r+1} - x_k^*\| \qquad (6.15)
\end{aligned}$$

where the second inequality is true due to the optimality of x_k^{r+1}.

Squaring both sides of (6.15) we obtain

$$(f(x^{r+1}) - f(x^*))^2 \le KIR^2 \max_{k,i} \|A_k^i (A_k^i)^T\|_2 (P_k^i)^2 \sum_{i=1}^{I} \sum_{k=1}^{K} \|A_k^i (x_k^{r+1} - x_k^r)\|^2. \qquad (6.16)$$

Then by the similar argument as in Theorem 3.1, we have the following result.

Corollary 6.1 *Suppose $g(\cdot)$ takes the composite form as expressed in* (6.12)*. Further suppose Assumption A and E hold true. Let $\{\mathbf{x}^r\}$ be the sequence generated by the BCM algorithm with G-S rule. Then we have*

$$\Delta^r = f(\mathbf{x}^r) - f^* \le \frac{c_7}{\sigma_7} \frac{1}{r} \qquad (6.17)$$

where

$$\sigma_7 := \frac{\min_{k,j} \eta_k^j}{2KIR^2 \max_{k,i} \|A_k^i (A_k^i)^T\|_2 (P_k^i)^2}, \quad c_7 := \max\{4\sigma_7 - 2, f(\mathbf{x}^1) - f^*, 2\}.$$

Remark 6.1 *Compared with what we have derived in Theorem 6.1, the rate here is explicitly dependent on various problem parameters, hence can be sharpened in certain cases. As an example, consider the simple case where $I = 1$ and $\ell(\cdot) = \frac{1}{2}\|\cdot\|^2$. For this problem the Lipschitz continuity constant for the entire smooth part is $M = \|AA^T\|_2$, where $A = [A_1, \cdots, A_K]$. Further we have $P_k^i = \sqrt{K}$, $\eta_k^i = 1$ for all k, i. When $\|A_k A_k^T\|$'s are approximately the same for all k, $\max_k \|A_k(A_k)^T\|_2$ is approximately $\frac{1}{K}\|AA^T\|_2$. This implies that σ_7 is upper bounded by $\frac{1}{2KR^2M}$, which is K times greater than σ_5 given in* (6.9).

Remark 6.2 *Our analysis above implies that when using the BCM (or equivalently the IWFA algorithm [33]) to solve the rate optimization problem given in Example 2.1, a sublinear rate can be obtained regardless of the rank of the channel matrices $\{H_k\}$. To see this, we first check Assumption E-(a). Denote $X_k := I_{n_r} + \sum_{j\neq k} H_j C_j H_j^T \succ 0$, then the kth subproblem can be reformulated as*

$$\min_{C_k} -\log\left(|X_k|\left|I_{n_t} + H_k^T X_k^{-1} H_k C_k\right|\right), \quad \text{s.t. } C_k \succeq 0,\ \mathrm{Tr}[C_k] \leq P_k. \tag{6.18}$$

Clearly for any feasible choice of $\{C_j\}_{j\neq k}$, we must have

$$|X_k| > 0, \quad 0 < \left\| H_k^T \left(I + \sum_{j\neq k} P_j H_j H_j^T \right)^{-1} H_k \right\|_2 \leq \|H_k^T X_k^{-1} H_k\|_2 \leq \|H_k^T H_k\|_2.$$

This says that the problem is strongly convex with respect to $H_k^T X_k^{-1} H_k C_k$. It is also easy to verify that the Lipschitz continuous assumption E-(b) is also satisfied; see for example a related discussion in [39, Section V-A]. Then Corollary 6.1 implies that IWFA converges in a rate $O(1/r)$, regardless of the rank of the channel matrices. Prior to our work, no convergence rate analysis has been done for the IWFA when solving problem (2.13).

6.3 Special Case: The Constrained Nonsmooth L2-SVM Problem

In this subsection, we assume that $g(x)$ takes the following form (besides Assumption A)

$$g(x) = \sum_{i=1}^{I} g_i(x) = \sum_{i=1}^{I} \left[(1 - x^T a_i)^+\right]^2 = \sum_{i=1}^{I} \left[(1 - \sum_{k=1}^{K} x_k^T a_{i,k})^+\right]^2 \tag{6.19}$$

where $(y)^+$ means $\max\{0, y\}$; $a_{i,k} \in \mathbb{R}^{n_k}$ denotes a subvector of a_i that corresponds to the block x_k. This objective is known as the L2 SVM loss. It is easy to observe that the problem is not strongly convex with respect to any block variable $x_k \in \mathbb{R}^{n_k}$. Moreover it is also not a special case of the problems considered in the previous subsection.

To proceed let us define the following short-handed notations:

$$\ell(y) := \|y\|^2, \quad q_i(x) := (1 - x^T a_i)^+ \geq 0.$$

Using these definitions, we have $g_i(x) = \ell(q_i(x))$.

For simplicity, let us consider the BCM scheme with G-S update rule, in which the kth block is updated by

$$x_k^{r+1} \in \arg\min_{x_k \in X_k} \sum_{i=1}^{I} \ell\left(q_i(x_k, w_{-k}^{r+1})\right) + h_k(x_k). \tag{6.20}$$

Moreover, we note that

$$\nabla_k g_i(x) = -2a_{i,k} q_i(x), \quad \nabla \ell(y) = 2y. \tag{6.21}$$

We can obtain the following series of inequalities

$$\begin{aligned}
&f(x_k^r, w_{-k}^{r+1}) - f(x_k^{r+1}, w_{-k}^{r+1}) \\
&= f(w_k^{r+1}) - f(w_{k+1}^{r+1}) \\
&\geq \sum_{i=1}^{I} \nabla \ell\left(q_i(w_{k+1}^{r+1})\right)\left(q_i(w_k^{r+1}) - q_i(w_{k+1}^{r+1})\right)) + h_k(x_k^r) - h_k(x_k^{r+1}) + \frac{1}{2}\|q_i(w_{k+1}^{r+1}) - q_i(w_k^{r+1})\|^2 \\
&\overset{\text{(i)}}{\geq} \sum_{i=1}^{I} \langle 2q_i(w_{k+1}^{r+1}) \partial q_i(w_{k+1}^{r+1}), x_k^r - x_{k+1}^r \rangle + h_k(x_k^r) - h_k(x_k^{r+1}) + \frac{1}{2}\|q_i(w_{k+1}^{r+1}) - q_i(w_k^{r+1})\|^2 \\
&\overset{\text{(ii)}}{\geq} \frac{1}{2} \sum_{i=1}^{I} \|q_i(w_{k+1}^{r+1}) - q_i(w_k^{r+1})\|^2
\end{aligned} \tag{6.22}$$

where (i) is due to the fact that $\nabla \ell\left(q_i(w_{k+1}^{r+1})\right) = 2q_i(w_{k+1}^{r+1}) \geq 0$ and the fact that $q_i(\cdot)$ is a convex function (albeit nonsmooth); (ii) is due to the optimality condition for the x_k^{r+1} subproblem. Therefore we have the following sufficient descent estimate

$$f(x^r) - f(x^{r+1}) = f(w_1^{r+1}) - f(w_{K+1}^{r+1}) \geq \sum_{i=1}^{I} \sum_{j=1}^{K} \frac{1}{2}\|q_i(w_{j+1}^{r+1}) - q_i(w_j^{r+1})\|^2. \tag{6.23}$$

Next we proceed to estimate the cost-to-go. To this end, we first bound $\|\nabla_k g(x^{r+1}) - \nabla_k g(x_k^{r+1}, w_{-k}^{r+1})\|$. We have the following

$$\begin{aligned}
\|\nabla_k g(x^{r+1}) - \nabla_k g(x_k^{r+1}, w_{-k}^{r+1})\| &= 2 \left\| \sum_{i=1}^{I} a_{i,k} \left(q_i(x^{r+1}) - q_i(w_{k+1}^{r+1})\right) \right\| \\
&\leq 2 \max_i \|a_{i,k}\| \sum_{i=1}^{I} \left\| q_i(x^{r+1}) - q_i(w_{k+1}^{r+1}) \right\| \\
&\leq 2 \max_i \|a_{i,k}\| \sum_{i=1}^{I} \sum_{j=1}^{K} \left\| q_i(w_j^{r+1}) - q_i(w_{j+1}^{r+1}) \right\|.
\end{aligned}$$

Consequently the cost-to-go estimate can be expressed as

$$\begin{aligned}
f(x^{r+1}) - f(x^*) &\leq \langle \nabla g(x^{r+1}), x^{r+1} - x^* \rangle + h(x^{r+1}) - h(x^*) \\
&= \sum_{k=1}^{K} \langle \nabla_k g(x^{r+1}) - \nabla_k g(x_k^{r+1}, w_{-k}^{r+1}), x_k^{r+1} - x_k^* \rangle \\
&\quad + \sum_{k=1}^{K} \langle \nabla_k g(x_k^{r+1}, w_{-k}^{r+1}), x_k^{r+1} - x_k^* \rangle + h(x^{r+1}) - h(x^*) \\
&\leq \sum_{k=1}^{K} \|\nabla_k g(x^{r+1}) - \nabla_k g(x_k^{r+1}, w_{-k}^{r+1})\| \|x_k^{r+1} - x_k^*\| \\
&\leq 2 \sum_{k=1}^{K} \max_i \|a_{i,k}\| \sum_{i=1}^{I} \sum_{j=1}^{K} \left\| q_i(w_j^{r+1}) - q_i(w_{j+1}^{r+1}) \right\| \|x_k^{r+1} - x_k^*\|. \qquad (6.24)
\end{aligned}$$

Finally, we have

$$(f(x^{r+1}) - f(x^*))^2 \leq 4 \left(\sum_{k=1}^{K} \max_i \|a_{i,k}\| \right)^2 KIR^2 \sum_{i=1}^{I} \sum_{j=1}^{K} \|q_i(w_j^{r+1}) - q_i(w_{j+1}^{r+1})\|^2. \qquad (6.25)$$

Then we have the following result.

Corollary 6.2 *Suppose Assumption A holds true, and suppose $g(\cdot)$ takes the form as expressed in (6.19). Let $\{\mathbf{x}^r\}$ be the sequence generated by the BCM algorithm with G-S rule. Then we have*

$$\Delta^r = f(\mathbf{x}^r) - f^* \leq \frac{c_8}{\sigma_8} \frac{1}{r} \qquad (6.26)$$

where

$$\sigma_8 := \frac{1}{8 \left(\sum_{k=1}^{K} \max_i \|a_{i,k}\| \right)^2 KIR^2}, \quad c_8 := \max\{4\sigma_8 - 2, f(\mathbf{x}^1) - f^*, 2\}.$$

To close this section, we mention that the E-C rule also achieves a sublinear rate for both the composite case and the L2-SVM cases. The analysis follows a similar argument as those presented above, therefore it is not repeated here.

6.4 Extensions

We briefly discuss a few extensions of the results presented so far in this section.

The first extension is to the BSUM algorithm without strongly convex upper bounds. For example, to extend Corollary 6.1, suppose that $g(x)$ is given by (6.12). Further assume that

$q_k(y_k; y)$ is an upper bound function for

$$\ell(y_1, \cdots, y_K) := \sum_{i=1}^{I} \ell^i(y_1, y_2, \cdots, y_K) = \sum_{i=1}^{I} \ell^i(A_1^i x_1, A_2^i x_2, \cdots, A_K^i x_K)$$

which is not necessarily strongly convex with respect to x_k. If $q_k(y_k; y)$ and $\ell(y_1, \cdots, y_K)$ together satisfy Assumption B for each k, then the BSUM algorithm that successively minimizes the upper bounds q_k's achieves a sublinear rate $O(1/r)$.

Second, our analysis can be directly applied to the algorithm with random permutation of the coordinates between the iterations, a strategy that has been found to be effective in practice [40, Section 8.5]. Indeed, the analysis for both BSUM and BCM with the G-S rule only requires that within each iteration the coordinates are chosen cyclically. There is no need to maintain the same order across different iterations.

Third, if the smooth function $g(x)$ is given by the composite form expressed in (6.12), and that the following additional assumptions are satisfied, then the BCM algorithm is capable of linear convergence.

Assumption F.

1. Each h_k satisfies either one of the following conditions:
   (a) The epigraph of $h_k(x_k)$ is a polyhedral set.
   (b) $h_k(x_k) = \lambda_k \|x_k\|_1 + \sum_J w_J \|x_{k,J}\|_2$, where $x_k = (\cdots, x_{k,J}, \cdots)$ is a partition of x_k with J being the partition index.
   (c) Each $h_k(x_k)$ is the sum of the functions described in the previous two items.
2. The feasible sets X_k, $k = 1, \cdots, K$ are polyhedral sets.
3. Each A_k^i has full column rank.

The key for proving the linear convergence is to show certain error bound condition holds true for different types of problems. We refer the readers to [7, 18, 19, 41, 42] for detailed arguments.

7 Concluding Remarks

In this paper we have analyzed the iteration complexity of a family of BCD-type algorithms for solving general convex nonsmooth problems of the form (1.1). Using a three-step argument, we show that the family of BCD-type algorithms, which includes BCM, BCGD, BCPG algorithms

with G-S, E-C, G-So and MBI update rules, converges globally in a sublinear rate of $\mathcal{O}(1/r)$. It should be noted that in case of the classical BCM algorithm, such sublinear rate can be achieved even without the per-block strong convexity. As a future work, it will be interesting to see whether the three-step approach can be extended to establish the iteration complexity bounds for other first order methods.

References

[1] D. P. Bertsekas, *Nonlinear Programming, 2nd ed*, Athena Scientific, Belmont, MA, 1999.

[2] P. Tseng and S. Yun, "A coordinate gradient descent method for nonsmooth separable minimization," *Mathematical Programming*, vol. 117, pp. 387–423, 2009.

[3] 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.

[4] 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.

[5] A. Beck and L. Tetruashvili, "On the convergence of block coordinate descent type methods," *SIAM Journal on Optimization*, vol. 23, no. 4, pp. 2037–2060, 2013.

[6] 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, 2013.

[7] P. Tseng and S. Yun, "Block-coordinate gradient descent method for linearly constrained nonsmooth separable optimization," *Journal of Optimization Theory and Applications*, vol. 140, pp. 513–535, 2009.

[8] Y. Nesterov, "Efficiency of coordiate descent methods on huge-scale optimization problems," *SIAM Journal on Optimization*, vol. 22, no. 2, pp. 341–362, 2012.

[9] 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.

[10] B. Chen, Z. Li S. He, and S. Zhang, "Maximum block improvement and polynomial optimization," *SIAM Journal on Optimization*, vol. 22, no. 1, pp. 87–107, 2012.

[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] D. P. Bertsekas and J. N. Tsitsiklis, *Neuro-Dynamic Programming*, Athena Scientific, Belmont, MA, 1996.

[14] D. P. Bertsekas and J. N. Tsitsiklis, *Parallel and Distributed Computation: Numerical Methods, 2nd ed*, Athena Scientific, Belmont, MA, 1997.

[15] J. M. Ortega and W. C. Rheinboldt, *Iterative Solution of Nonlinear Equations in Several Variables*, Academic Press, 1972.

[16] 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.

[17] 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.

[18] 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.

[19] 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.

[20] 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.

[21] P. Tseng, "Approximation accuracy, gradient methods, and error bound for structured convex optimization," *Mathematical Programming*, vol. 125, no. 2, pp. 263–295, 2010.

[22] P. Richtarik and M. Takac, "Iteration complexity of randomized block-coordinate descent methods for minimizing a composite function," *Mathematical Programming*, vol. 144, pp. 1–38, 2014.

[23] Z. Lu and L. Xiao, "On the complexity analysis of randomized block-coordinate descent methods," 2013, accepted by Mathematical Programming.

[24] A. Beck, "On the convergence of alternating minimization with applications to iteratively reweighted least squares and decomposition schemes," *SIAM Journal on Optimization*, vol. 25, no. 1, pp. 185–209, 2015.

[25] 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.

[26] 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.

[27] 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.

[28] J. Yang, Y. Zhang, and W. Yin, "A fast alternating direction method for TVL1-L2 signal reconstruction from partial fourier data," *IEEE Journal of Selected Topics in Signal Processing*, vol. 4, no. 2, pp. 288–297, 2010.

[29] M. Hong, M. Razaviyayn, Z.-Q. Luo, and J.-S. Pang, "A unified algorithmic framework for block-structured optimization involving big data," 2015, submitted for publication.

[30] 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.

[31] J. Mairal, "Optimization with first-order surrogate functions," in *International Conference on Machine Learning (ICML)*., 2013.

[32] T. M. Cover and J. A. Thomas, *Elements of Information Theory, second edition*, Wiley, 2005.

[33] W. Yu, W. Rhee, S. Boyd, and J. M. Cioffi, "Iterative water-filling for Gaussian vector multiple-access channels," *IEEE Transactions on Information Theory*, vol. 50, no. 1, pp. 145–152, 2004.

[34] Y. Nesterov, *Introductory lectures on convex optimization: A basic course*, Springer, 2004.

[35] I. Daubechies, R. DeVore, M. Fornasier, and C. S. Gunturk, "Iteratively reweighted least squares minimization for sparse recovery," *Communications on Pure and Applied Mathematics*, vol. 63, no. 1, pp. 1–38, 2010.

[36] J.-B. Hiriart-Urruty and C. Lemarechal, *Convex Analysis and Minimization Algorithms I: Fundamentals*, Springer, 1996.

[37] A. Beck and M. Teboulle, "A fast iterative shrinkage-thresholding algorithm for linear inverse problems," *SIAM Journal on Imgaging Science*, vol. 2, no. 1, pp. 183–202, 2009.

[38] P. Tseng, "On accelerated proximal gradient methods for convex-concave optimization," 2008, preprint.

[39] G. Scutari, F. Facchinei, P. Song, D. P. Palomar, and J.-S. Pang, "Decomposition by partial linearization: Parallel optimization of multi-agent systems," *IEEE Transactions on Signal Processing*, vol. 63, no. 3, pp. 641–656, 2014.

[40] S. Shalev-Shwartz and T. Zhang, "Proximal stochastic dual coordinate ascent methods for regularzied loss minimization," *Journal of Machine Learning Rsearch*, vol. 14, pp. 567–599, 2013.

[41] M. Hong, T.-H. Chang, X. Wang, M. Razaviyayn, S. Ma, and Z.-Q. Luo, "A block successive upper bound minimization method of multipliers for linearly constrained convex optimization," 2013, Preprint, available online arXiv:1401.7079.

[42] M. Sanjabi, M. Kadkhodaei, and Z.-Q. Luo, "On the linear convergence of approximate proximal splitting methods for non-smooth convex minimization," 2012, manuscript.