2018

Alternating randomized block coordinate descent

Alternating Randomized Block Coordinate Descent

Jelena Diakonikolas [1] **Lorenzo Orecchia** [1]

Abstract

Block-coordinate descent algorithms and alternating minimization methods are fundamental optimization algorithms and an important primitive in large-scale optimization and machine learning. While various block-coordinate-descent-type methods have been studied extensively, only alternating minimization – which applies to the setting of only two blocks – is known to have convergence time that scales independently of the least smooth block. A natural question is then: is the setting of two blocks special? We show that the answer is "no" as long as the least smooth block can be optimized exactly – an assumption that is also needed in the setting of alternating minimization. We do so by introducing a novel algorithm AR-BCD, whose convergence time scales independently of the least smooth (possibly non-smooth) block. The basic algorithm generalizes both alternating minimization and randomized block coordinate (gradient) descent, and we also provide its accelerated version – AAR-BCD.

1. Introduction

First-order methods for minimizing smooth convex functions are a cornerstone of large-scale optimization and machine learning. Given the size and heterogeneity of the data in these applications, there is a particular interest in designing iterative methods that, at each iteration, only optimize over a subset of the decision variables (Wright, 2015).

This paper focuses on two classes of methods that constitute important instantiations of this idea. The first class is that of *block-coordinate descent methods*, i.e., methods that partition the set of variables into $n \geq 2$ blocks and perform a gradient descent step on a single block at every iteration, while leaving the remaining variable blocks fixed. A paradigmatic example of this approach is the randomized Kaczmarz algorithm of (Strohmer & Vershynin, 2009) for linear systems and its generalization (Nesterov, 2012). The second class is that of *alternating minimization methods*, i.e., algorithms that partition the variable set into only $n = 2$ blocks and alternate between *exactly optimizing* one block or the other at each iteration (see, e.g., (Beck, 2015) and references therein).

Besides the computational advantage in only having to update a subset of variables at each iteration, methods in these two classes are also able to exploit better the structure of the problem, which, for instance, may be computationally expensive only in a small number of variables. To formalize this statement, assume that the set of variables is partitioned into $n \leq N$ mutually disjoint blocks, where the i^{th} block of variable $\mathbf{x}$ is denoted by $\mathbf{x}^i$, and the gradient corresponding to the i^{th} block is denoted by $\nabla_i f(\mathbf{x})$. Each block i will be associated with a smoothness parameter L_i, I.e., $\forall \mathbf{x}, \mathbf{y} \in \mathbb{R}^N$:

$$\|\nabla_i f(\mathbf{x} + I_N^i \mathbf{y}) - \nabla_i f(\mathbf{x})\|_* \leq L_i \|\mathbf{y}^i\|, \qquad (1.1)$$

where I_N^i is a diagonal matrix whose diagonal entries equal one for coordinates from block i, and are zero otherwise.

In this setting, the convergence time of standard randomized block-coordinate descent methods, such as those in (Nesterov, 2012), scales as $O\left(\frac{\sum_i L_i}{\epsilon}\right)$, where ϵ is the desired additive error. By contrast, when $n = 2$, the convergence time of the alternating minimization method (Beck, 2015) scales as $O\left(\frac{L_{\min}}{\epsilon}\right)$, where $L_{\min}$ is the minimum smoothness parameter of the two blocks. This means that one of the two blocks can have arbitrarily poor smoothness (including ∞), as long as it is easy to optimize over it. Some important examples with a nonsmooth block (with smoothness parameter equal to infinity) can be found in (Beck, 2015). Additional examples of problems for which exact optimization over the least smooth block can be performed efficiently are provided in Appendix B.

In this paper, we address the following open question, which was implicitly raised by (Beck & Tetruashvili, 2013): can we design algorithms that combine the features of randomized block-coordinate descent and alternating minimization? In particular, assuming we can perform ex-

[1]Department of Computer Science, Boston University, Boston, MA, USA. Correspondence to: Jelena Diakonikolas <jelenad@bu.edu>, Lorenzo Orecchia <orecchia@bu.edu>.

Proceedings of the 35^{th} International Conference on Machine Learning, Stockholm, Sweden, PMLR 80, 2018.

act optimization on block n, can we construct a block-coordinate descent algorithm whose running time scales with $O(\sum_{i=1}^{n-1} L_i)$, i.e., independently of the smoothness L_n of the n^{th} block? This would generalize both existing block-coordinate descent methods, by allowing one block to be optimized exactly, and existing alternating minimization methods, by allowing n to be larger than 2 and requiring exact optimization only on a single block.

We answer these questions in the affirmative by presenting a novel algorithm: alternating randomized block coordinate descent (AR-BCD). The algorithm alternates between an exact optimization over a fixed, possibly non-smooth block, and a gradient descent or exact optimization over a randomly selected block among the remaining blocks. For two blocks, the method reduces to the standard alternating minimization, while when the non-smooth block is empty (not optimized over), we get randomized block coordinate descent (RCDM) from (Nesterov, 2012).

Our second contribution is AAR-BCD, an accelerated version of AR-BCD, which achieves the accelerated rate of $\frac{1}{k^2}$ without incurring any dependence on the smoothness of block n. Furthermore, when the non-smooth block is empty, AAR-BCD recovers the fastest known convergence bounds for block-coordinate descent (Qu & Richtárik, 2016; Allen-Zhu et al., 2016; Nesterov, 2012; Lin et al., 2014; Nesterov & Stich, 2017). Another conceptual contribution is our extension of the approximate duality gap technique of (Diakonikolas & Orecchia, 2017), which leads to a general and more streamlined analysis. Finally, to illustrate the results, we perform a preliminary experimental evaluation of our methods against existing block-coordinate algorithms and discuss how their performance depends on the smoothness and size of the blocks.

Related Work Alternating minimization and cyclic block coordinate descent are old and fundamental algorithms (Ortega & Rheinboldt, 1970) whose convergence (to a stationary point) has been studied even in the non-convex setting, in which they were shown to converge asymptotically under the additional assumptions that the blocks are optimized exactly and their minimizers are unique (Bertsekas, 1999). However, even in the non-smooth convex case, methods that perform exact minimization over a fixed set of blocks may converge arbitrarily slowly. This has lead scholars to focus on the case of smooth convex minimization, for which nonasymptotic convergence rates were obtained recently in (Beck & Tetruashvili, 2013; Beck, 2015; Sun & Hong, 2015; Saha & Tewari, 2013). However, prior to our work, convergence bounds that are independent of the largest smoothness parameter were only known for the setting of two blocks.

Randomized coordinate descent methods, in which steps over coordinate blocks are taken in a non-cyclic randomized order (i.e., in each iteration one block is sampled with replacement) were originally analyzed in (Nesterov, 2012). The same paper (Nesterov, 2012) also provided an accelerated version of these methods. The results of (Nesterov, 2012) were subsequently improved and generalized to various other settings (such as, e.g., composite minimization) in (Lee & Sidford, 2013; Allen-Zhu et al., 2016; Nesterov & Stich, 2017; Richtárik & Takáč, 2014; Fercoq & Richtárik, 2015; Lin et al., 2014). The analysis of the different block coordinate descent methods under various sampling probabilities (that, unlike in our setting, are non-zero over all the blocks) was unified in (Qu & Richtárik, 2016) and extended to a more general class of steps within each block in (Gower & Richtárik, 2015; Qu et al., 2016).

Our results should be carefully compared to a number of proximal block-coordinate methods that rely on different assumptions (Tseng & Yun, 2009; Richtárik & Takáč, 2014; Lin et al., 2014; Fercoq & Richtárik, 2015). In this setting, the function f is assumed to have the structure $f_0(\mathbf{x}) + \Psi(\mathbf{x})$, where f_0 is smooth, the non-smooth convex function Ψ is separable over the blocks, i.e., $\Psi(\mathbf{x}) = \sum_{i=1}^{n} \Psi_i(\mathbf{x}_i)$, and we can efficiently compute the proximal operator of each Ψ_i. This strong assumption allows these methods to make use of the standard proximal optimization framework. By contrast, in our paper, the convex objective can be taken to have an arbitrary form, where the non-smoothness of a block need not be separable, though the function is assumed to be differentiable.

2. Preliminaries

We assume that we are given oracle access to the gradients of a continuously differentiable convex function $f : \mathbb{R}^N \to \mathbb{R}$, where computing gradients over only a subset of coordinates is computationally much cheaper than computing the full gradient. We are interested in minimizing $f(\cdot)$ over $\mathbb{R}^N$, and we denote $\mathbf{x}_* = \operatorname{argmin}_{\mathbf{x}\in\mathbb{R}^N} f(\mathbf{x})$. We let $\|\cdot\|$ denote an arbitrary (but fixed) norm, and $\|\cdot\|_*$ denote its dual norm, defined in the standard way: $\|\mathbf{z}\|_* = \sup_{\mathbf{x}\in\mathbb{R}^N:\|\mathbf{x}\|=1} \langle \mathbf{z}, \mathbf{x}\rangle$.[1]

Let I_N be the identity matrix of size N, I_N^i be a diagonal matrix whose diagonal elements j are equal to one if variable j is in the i^{th} block, and zero otherwise. Notice that $I_N = \sum_{i=1}^{n} I_N^i$. Let $S_i(\mathbf{x}) = \{\mathbf{y} \in \mathbb{R}^N : (I_N - I_N^i)\mathbf{y} = (I_N - I_N^i)\mathbf{x}\}$, that is, S_i contains all the points from $\mathbb{R}^N$ whose coordinates differ from those of $\mathbf{x}$ only over block i.

We denote the smoothness parameter of block i by L_i, as

[1] Note that the analysis extends in a straightforward way to the case where each block is associated with a different norm (see, e.g., (Nesterov, 2012)); for simplicity of presentation, we take the same norm over all blocks.

defined in Equation (1.1). Equivalently, $\forall \mathbf{x}, \mathbf{y} \in \mathbb{R}^N$:

$$f(\mathbf{x} + I_N^i \mathbf{y}) \leq f(\mathbf{x}) + \langle \nabla_i f(\mathbf{x}), \mathbf{y}^i \rangle + \frac{L_i}{2} \|\mathbf{y}^i\|^2. \quad (2.1)$$

The gradient step over block i is then defined as:

$$T_i(\mathbf{x}) = \underset{\mathbf{y} \in S_i(\mathbf{x})}{\operatorname{argmin}} \left\{ \langle \nabla f(\mathbf{x}), \mathbf{y} - \mathbf{x} \rangle + \frac{L_i}{2} \|\mathbf{y} - \mathbf{x}\|^2 \right\}. \quad (2.2)$$

By standard arguments (see, e.g., Exercise 3.27 in (Boyd & Vandenberghe, 2004)):

$$f(T_i(\mathbf{x})) - f(\mathbf{x}) \leq -\frac{1}{2L_i} \|\nabla_i f(\mathbf{x})\|_*^2. \quad (2.3)$$

Without loss of generality, we will assume that the n^{th} block has the largest smoothness parameter and is possibly non-smooth (i.e., it can be $L_n = \infty$). The standing assumption is that exact minimization over the n^{th} block is "easy", meaning that it is computationally inexpensive and possibly solvable in closed form; for some important examples that have this property, see Appendix B. Observe that when block n contains a small number of variables, it is often computationally inexpensive to use second-order optimization methods, such as, e.g., interior point method.

We assume that $f(\cdot)$ is strongly convex with parameter $\mu \geq 0$, where it could be $\mu = 0$ (in which case $f(\cdot)$ is not strongly convex). Namely, $\forall \mathbf{x}, \mathbf{y}$:

$$f(\mathbf{y}) \geq f(\mathbf{x}) + \langle \nabla f(\mathbf{x}), \mathbf{y} - \mathbf{x} \rangle + \frac{\mu}{2} \|\mathbf{y} - \mathbf{x}\|^2. \quad (2.4)$$

When $\mu > 0$, we take $\|\cdot\| = \|\cdot\|_2$, which is customary for smooth and strongly convex minimization (Bubeck, 2014).

Throughout the paper, whenever we take unconditional expectation, it is with respect to all randomness in the algorithm.

2.1. Alternating Minimization

In (standard) alternating minimization (AM), there are only two blocks of coordinates, i.e., $n = 2$. The algorithm is defined as follows.

$$\begin{aligned} \hat{\mathbf{x}}_k &= \underset{\mathbf{x} \in S_1(\mathbf{x}_{k-1})}{\operatorname{argmin}} f(\mathbf{x}), \\ \mathbf{x}_k &= \underset{\mathbf{x} \in S_2(\hat{\mathbf{x}}_k)}{\operatorname{argmin}} f(\mathbf{x}), \\ \mathbf{x}_1 &\in \mathbb{R}^N \text{ is an arbitrary initial point.} \end{aligned} \quad \text{(AM)}$$

We note that for the standard analysis of alternating minimization (Beck, 2015), the exact minimization step over the smoother block can be replaced by a gradient step (Equation (2.2)), while still leading to convergence that is only dependent on the smaller smoothness parameter.

2.2. Randomized Block Coordinate (Gradient) Descent

The simplest version of randomized block coordinate (gradient) descent (RCDM) can be stated as (Nesterov, 2012):

$$\begin{aligned} &\text{Select } i_k \in \{1, \dots, n\} \text{ w.p. } p_{i_k} > 0, \\ &\mathbf{x}_k = T_{i_k}(\mathbf{x}_{k-1}), \\ &\mathbf{x}_1 \in \mathbb{R}^N \text{ is an arbitrary initial point,} \end{aligned} \quad \text{(RCDM)}$$

where $\sum_{i=1}^n p_i = 1$. A standard choice of the probability distribution is $p_i \sim L_i$, leading to the convergence rate that depends on the sum of block smoothness parameters.

3. AR-BCD

The basic version of alternating randomized block coordinate descent (AR-BCD) is a direct generalization of (AM) and (RCDM): when $n = 2$, it is equivalent to (AM), while when the size of the n^{th} block is zero, it reduces to (RCDM). The method is stated as follows:

$$\begin{aligned} &\text{Select } i_k \in \{1, \dots, n-1\} \text{ w.p. } p_{i_k} > 0, \\ &\hat{\mathbf{x}}_k = T_{i_k}(\mathbf{x}_{k-1}), \\ &\mathbf{x}_k = \underset{\mathbf{x} \in S_n(\hat{\mathbf{x}}_k)}{\operatorname{argmin}} f(\mathbf{x}), \\ &\mathbf{x}_1 \in \mathbb{R}^N \text{ is an arbitrary initial point,} \end{aligned} \quad \text{(AR-BCD)}$$

where $\sum_{i=1}^{n-1} p_i = 1$. We note that nothing will change in the analysis if the step $\hat{\mathbf{x}}_k = T_{i_k}(\mathbf{x}_{k-1})$ is replaced by $\hat{\mathbf{x}}_k = \operatorname{argmin}_{\mathbf{x} \in S_{i_k}(\mathbf{x}_{k-1})} f(\mathbf{x})$, since $\min_{\mathbf{x} \in S_{i_k}(\mathbf{x}_{k-1})} f(\mathbf{x}) \leq f(T_{i_k}(\mathbf{x}_{k-1}))$.

In the rest of the section, we show that (AR-BCD) leads to a convergence bound that interpolates between the convergence bounds of (AM) and (RCDM): it depends on the sum of the smoothness parameters of the first $n-1$ blocks, while the dependence on the remaining problem parameters is the same for all these methods.

3.1. Approximate Duality Gap

To analyze (AR-BCD), we extend the approximate duality gap technique (Diakonikolas & Orecchia, 2017) to the setting of randomized block coordinate descent methods. The approximate duality gap G_k is defined as the difference of an upper bound U_k and a lower bound L_k to the minimum function value $f(\mathbf{x}_*)$. For (AR-BCD), we choose the upper bound to simply be $U_k = f(\mathbf{x}_{k+1})$.

The generic construction of the lower bound is as follows. Let $\mathbf{x}_1, \mathbf{x}_2, ..., \mathbf{x}_k$ be any sequence of points from $\mathbb{R}^N$ (in fact we will choose them to be exactly the sequence constructed by (AR-BCD)). Then, by (strong) convexity of $f(\cdot)$, $f(\mathbf{x}_*) \geq f(\mathbf{x}_j) + \langle \nabla f(\mathbf{x}_j), \mathbf{x}_* - \mathbf{x}_j \rangle + \frac{\mu}{2} \|\mathbf{x}_* - \mathbf{x}_j\|^2$, $\forall j \in \{1, \dots, k\}$. In particular, if $a_j > 0$ is a sequence

of (deterministic, independent of i_j) positive real numbers and $A_k = \sum_{j=1}^k a_j$, then:

$$f(\mathbf{x}_*) \geq \frac{\sum_{j=1}^k a_j f(\mathbf{x}_j) + \sum_{j=1}^k a_j \langle \nabla f(\mathbf{x}_j), \mathbf{x}_* - \mathbf{x}_j \rangle}{A_k} + \frac{\frac{\mu}{2}\sum_{j=1}^k a_j \|\mathbf{x}_* - \mathbf{x}_j\|^2}{A_k} \stackrel{\text{def}}{=} L_k. \tag{3.1}$$

3.2. Convergence Analysis

The main idea in the analysis is to show that $\mathbb{E}[A_k G_k - A_{k-1}G_{k-1}] \leq E_k$, for some deterministic E_k. Then, using linearity of expectation, $\mathbb{E}[f(\mathbf{x}_{k+1})] - f(\mathbf{x}_*) \leq \mathbb{E}[G_k] \leq \frac{\mathbb{E}[A_1 G_1]}{A_k} + \frac{\sum_{j=2}^k E_j}{A_k}$. The bound in expectation can then be turned into a bound in probability, using well-known concentration bounds. The main observation that allows us not to pay for the non-smooth block is:

Observation 3.1. *For $\mathbf{x}_k$'s constructed by (AR-BCD), $\nabla_n f(\mathbf{x}_k) = \mathbf{0}$, $\forall k$, where $\mathbf{0}$ is the vector of all zeros.*

This observation is essentially what allows us to sample i_k only from the first $n-1$ blocks, and holds due to the step $\mathbf{x}_k = \text{argmin}_{\mathbf{x} \in S_n(\hat{\mathbf{x}}_k)} f(\mathbf{x})$ from (AR-BCD).

Denote $R_{\mathbf{x}_*^i} = \max_{\mathbf{x} \in \mathbb{R}^N} \{\|I_N^i(\mathbf{x}_* - \mathbf{x})\|^2 : f(\mathbf{x}) \leq f(\mathbf{x}_1)\}$, and let us bound the initial gap $A_1 G_1$.

Proposition 3.2. $\mathbb{E}[A_1 G_1] \leq E_1$, *where* $E_1 = a_1 \sum_{i=1}^{n-1} \left(\frac{L_i}{2p_i} - \frac{\mu}{2}\right) R_{\mathbf{x}_*^i}$.

Proof. By linearity of expectation, $\mathbb{E}[A_1 G_1] = \mathbb{E}[A_1 U_1] - \mathbb{E}[A_1 L_1]$. The initial lower bound is deterministic, and, by $\nabla_n f(\mathbf{x}_1) = \mathbf{0}$ and duality of norms, is bounded as:

$$\mathbb{E}[A_1 L_1] \geq a_1 f(\mathbf{x}_1) - a_1 \sum_{i=0}^{n-1} \|\nabla_i f(\mathbf{x}_1)\|_* \|\mathbf{x}_*^i - \mathbf{x}_1^i\| + a_1 \frac{\mu}{2}\|\mathbf{x}_* - \mathbf{x}_1\|^2.$$

Using (2.3), if $i_2 = i$, then:

$$U_1 = f(\mathbf{x}_2) \leq f(\hat{\mathbf{x}}_2) \leq f(\mathbf{x}_1) - \frac{1}{2L_i}\|\nabla_i f(\mathbf{x}_1)\|_*^2.$$

Since block i is selected with probability p_i and $A_1 = a_1$:

$$\mathbb{E}[A_1 U_1] \leq a_1 f(\mathbf{x}_1) - \sum_{i=1}^{n-1} \frac{a_1 p_i}{2L_i}\|\nabla_i f(\mathbf{x}_i)\|_*^2.$$

Since the inequality $2ab - a^2 \leq b^2$ holds $\forall a, b$, we have:

$$a_1 \|\nabla_i f(\mathbf{x}_1)\|_* \|\mathbf{x}_*^i - \mathbf{x}_1^i\| - \frac{a_1 p_i}{2L_i}\|\nabla_i f(\mathbf{x}_i)\|_*^2 \leq \frac{a_1 L_i}{2p_i}\|\mathbf{x}_*^i - \mathbf{x}_1^i\|^2, \ \forall i \in \{1, \ldots, n-1\}$$

Hence, when $\mu = 0$, $\mathbb{E}[A_1 G_1] \leq \sum_{i=1}^{n-1} \frac{a_1 L_i}{2p_i}\|\mathbf{x}_*^i - \mathbf{x}_1^i\|^2$. When $\mu > 0$, since in that case we are assuming $\|\cdot\| = \|\cdot\|_2$ (Section 2), $\|\mathbf{x}_* - \mathbf{x}_1\|^2 \geq \sum_{i=1}^{n-1} \|\mathbf{x}_*^i - \mathbf{x}_1^i\|^2$, leading to $\mathbb{E}[A_1 G_1] \leq a_1 \sum_{i=1}^{n-1} \left(\frac{L_i}{2p_i} - \frac{\mu}{2}\right)\|\mathbf{x}_*^i - \mathbf{x}_1^i\|^2$. □

We now show how to bound the error in the decrease of the scaled gap $A_k G_k$.

Lemma 3.3. $\mathbb{E}[A_k G_k - A_{k-1}G_{k-1}] \leq E_k$, *where* $E_k = a_k \sum_{i=1}^{n-1} \left(\frac{a_k L_i}{2A_k p_i} - \frac{\mu}{2}\right) R_{\mathbf{x}_*^i}$.

Proof. Let $\mathcal{F}_k$ denote the natural filtration up to iteration k. By linearity of expectation and $A_k L_k - A_{k-1}L_{k-1}$ being measurable w.r.t. $\mathcal{F}_k$,

$$\mathbb{E}[A_k G_k - A_{k-1}G_{k-1}|\mathcal{F}_k] = \mathbb{E}[A_k U_k - A_{k-1}U_{k-1}|\mathcal{F}_k] - (A_k L_k - A_{k-1}L_{k-1}).$$

With probability p_i and as $f(\mathbf{x}_{k+1}) \leq f(\hat{\mathbf{x}}_{k+1})$, the change in the upper bound is:

$$\begin{aligned} A_k U_k - A_{k-1}U_{k-1} &\leq A_k f(\hat{\mathbf{x}}_{k+1}) - A_{k-1} f(\mathbf{x}_k) \\ &\leq a_k f(\mathbf{x}_k) - \frac{A_k}{2L_i}\|\nabla_i f(\mathbf{x}_k)\|_*^2, \end{aligned}$$

where the second line follows from $\hat{\mathbf{x}}_{k+1} = T_{i_k}(\mathbf{x}_k)$ and Equation (2.3). Hence:

$$\mathbb{E}[A_k U_k - A_{k-1}U_{k-1}|\mathcal{F}_k] \leq a_k f(\mathbf{x}_k) - A_k \sum_{i=1}^{n-1} \frac{p_i}{2L_i}\|\nabla_i f(\mathbf{x}_k)\|_*^2.$$

On the other hand, using the duality of norms, the change in the lower bound is:

$$\begin{aligned} A_k L_k - A_{k-1}L_{k-1} &\geq a_k f(\mathbf{x}_k) - a_k \sum_{i=1}^{n-1} \|\nabla_i f(\mathbf{x}_k)\|_* \|\mathbf{x}_*^i - \mathbf{x}_k^i\| + a_k \frac{\mu}{2}\|\mathbf{x}_* - \mathbf{x}_k\|^2 \\ &\geq a_k f(\mathbf{x}_k) - a_k \sum_{i=1}^{n-1} \|\nabla_i f(\mathbf{x}_k)\|_* \sqrt{R_{\mathbf{x}_*^i}} + a_k \frac{\mu}{2}\|\mathbf{x}_* - \mathbf{x}_k\|^2. \end{aligned}$$

By the same argument as in the proof of Proposition 3.2, it follows that: $\mathbb{E}[A_k G_k - A_{k-1}G_{k-1}|\mathcal{F}_k] \leq a_k \sum_{i=1}^{n-1} \left(\frac{L_i a_k}{2A_k p_i} - \frac{\mu}{2}\right) R_{\mathbf{x}_*^i} = E_k$. Taking expectations on both sides, as E_k is deterministic, the proof follows. □

We are now ready to prove the convergence bound for (AR-BCD), as follows.

Theorem 3.4. *Let $\mathbf{x}_k$ evolve according to (AR-BCD). Then, $\forall k \geq 1$:*

1. *If* $\mu = 0 : \mathbb{E}[f(\mathbf{x}_{k+1})] - f(\mathbf{x}_*) \leq \frac{2\sum_{i=1}^{n-1} \frac{L_i}{p_i} R_{\mathbf{x}_*^i}}{k+3}$. *In particular, for* $p_i = \frac{L_i}{\sum_{i'=1}^{n-1} L_{i'}}$, $1 \leq i \leq n-1$*:*

$$\mathbb{E}[f(\mathbf{x}_{k+1})] - f(\mathbf{x}_*) \leq \frac{2(\sum_{i'=1}^{n-1} L_{i'}) \sum_{i=1}^{n-1} R_{\mathbf{x}_*^i}}{k+3}.$$

Similarly, for $p_i = \frac{1}{n-1}$, $1 \leq i \leq n-1$:

$$\mathbb{E}[f(\mathbf{x}_{k+1})] - f(\mathbf{x}_*) \leq \frac{2(n-1)\sum_{i=1}^{n-1} L_i R_{\mathbf{x}_*^i}}{k+3}$$

2. *If* $\mu > 0$, $p_i = \frac{L_i}{\sum_{i'=1}^{n-1} L_{i'}}$ *and* $\|\cdot\| = \|\cdot\|_2$:

$$\mathbb{E}[f(\mathbf{x}_{k+1})] - f(\mathbf{x}_*) \leq \left(1 - \frac{\mu}{\sum_{i'=1}^{n-1} L_{i'}}\right)^k \cdot \frac{(\sum_{i'=1}^{n-1} L_{i'})\|(I_N - I_N^n)(\mathbf{x}_* - \mathbf{x}_1)\|^2}{2}.$$

Proof. From Proposition 3.2 and Lemma 3.3, by linearity of expectation and the definition of G_k:

$$\mathbb{E}[f(\mathbf{x}_{k+1})] - f(\mathbf{x}_*) \leq \mathbb{E}[G_k] \leq \frac{\sum_{j=1}^{k} E_j}{A_k}, \qquad (3.2)$$

where $E_j = \frac{a_j{}^2}{A_j} \sum_{i=1}^{n-1} \frac{L_i}{2p_i} R_{\mathbf{x}_*^i}$.

Notice that the algorithm does not depend on the sequence $\{a_j\}$ and thus we can choose it arbitrarily. Suppose that $\mu = 0$. Let $a_j = \frac{j+1}{2}$. Then $\frac{a_j{}^2}{A_j} = \frac{(j+1)^2}{j(j+3)} \leq 1$, and thus: $\frac{\sum_{j=1}^{k} E_j}{A_k} \leq \frac{2\sum_{i=1}^{n-1} \frac{L_i}{p_i} R_{\mathbf{x}_*^i}}{k+3}$,which proves the first part of the theorem, up to concrete choices of p_i's, which follow by simple computations.

For the second part of the theorem, as $\mu > 0$, we are assuming that $\|\cdot\| = \|\cdot\|_2$, as discussed in Section 2. From Lemma 3.3, $E_j = a_j \sum_{i=1}^{n-1} \left(\frac{a_j L_i}{2A_j p_i} - \frac{\mu}{2}\right) R_{\mathbf{x}_*^i}$, $\forall j \geq 2$. As $p_i = \frac{L_i}{\sum_{i'=1}^{n-1} L_{i'}}$, if we take $\frac{a_j}{A_j} = \frac{\mu}{\sum_{i'=1}^{n-1} L_{i'}}$, it follows that $E_j = 0$, $\forall j \geq 2$. Let $a_1 = A_1 = 1$ and $\frac{a_j}{A_j} = \frac{\mu}{\sum_{i'=1}^{n-1} L_{i'}}$ for $j \geq 2$. Then: $\mathbb{E}[f(\mathbf{x}_{k+1})] - f(\mathbf{x}_*) \leq \mathbb{E}[G_k] \leq \frac{\mathbb{E}[A_1 G_1]}{A_k}$. As $\frac{A_1}{A_k} = \frac{A_1}{A_2} \cdot \frac{A_2}{A_3} \cdot \ldots \cdot \frac{A_{k-1}}{A_k}$ and $\frac{A_{j-1}}{A_j} = 1 - \frac{a_j}{A_j}$: $\mathbb{E}[f(\mathbf{x}_{k+1})] - f(\mathbf{x}_*) \leq \left(1 - \frac{\mu}{\sum_{i'=1}^{n-1} L_{i'}}\right)^{k-1} \mathbb{E}[G_1]$. It remains to observe that, from Proposition 3.2, $\mathbb{E}[G_1] \leq \left(1 - \frac{\mu}{\sum_{i'=1}^{n-1} L_{i'}}\right) \frac{(\sum_{i'=1}^{n-1} L_{i'})\|(I_N - I_N^n)(\mathbf{x}_* - \mathbf{x}_1)\|^2}{2}$. □

We note that when $n = 2$, the asymptotic convergence of AR-BCD coincides with the convergence of alternating minimization (Beck, 2015). When n^{th} block is empty (i.e., when all blocks are sampled with non-zero probability and there is no exact minimization over a least-smooth block), we obtain the convergence bound of the standard randomized coordinate descent method (Nesterov, 2012).

4. Accelerated AR-BCD

In this section, we show how to accelerate (AR-BCD) when $f(\cdot)$ is smooth. We believe it is possible to obtain similar results in the smooth and strongly convex case, which we defer to a future version of the paper. Denote:

$$\Delta_k = I_N^{i_k} \nabla f(\mathbf{x}_k)/p_{i_k},$$
$$\mathbf{v}_k = \operatorname*{argmin}_{\mathbf{u}} \Big\{ \sum_{j=1}^{k} a_j \langle \Delta_j, \mathbf{u} \rangle + \sum_{i=1}^{n} \frac{\sigma_i}{2} \|\mathbf{u}^i - \mathbf{x}_1^i\|^2 \Big\}, \qquad (4.1)$$

where $\sigma_i > 0$, $\forall i$, will be specified later. Accelerated AR-BCD (AAR-BCD) is defined as follows:

Select i_k from $\{1, \ldots, n-1\}$ w.p. p_{i_k},
$$\hat{\mathbf{x}}_k = \frac{A_{k-1}}{A_k}\mathbf{y}_{k-1} + \frac{a_k}{A_k}\mathbf{v}_{k-1},$$
$$\mathbf{x}_k = \operatorname*{argmin}_{\mathbf{x} \in S_n(\hat{\mathbf{x}}_k)} f(\mathbf{x}), \qquad \text{(AAR-BCD)}$$
$$\mathbf{y}_k = \mathbf{x}_k + \frac{a_k}{p_{i_k} A_k} I_N^{i_k}(\mathbf{v}_k - \mathbf{v}_{k-1}),$$
$\mathbf{x}_1$ is an arbitrary initial point,

where $\sum_{i=1}^{n-1} p_i = 1$, $p_i > 0$, $\forall i \in \{1, \ldots, n-1\}$, and $\mathbf{v}_k$ is defined by (4.1). To seed the algorithm, we further assume that $\mathbf{y}_1 = \mathbf{x}_1 + I_N^{i_1} \frac{1}{p_{i_1}}(\mathbf{v}_1 - \mathbf{x}_1)$.

Remark 4.1. Iteration complexity of (AAR-BCD) is dominated by the computation of $\hat{\mathbf{x}}_k$, which requires updating an entire vector. This type of an update is not unusual for accelerated block coordinate descent methods, and in fact appears in all such methods we are aware of (Nesterov, 2012; Lee & Sidford, 2013; Lin et al., 2014; Fercoq & Richtárik, 2015; Allen-Zhu et al., 2016). In most cases of practical interest, however, it is possible to implement this step efficiently (using that $\mathbf{v}_k$ changes only over block i_k in iteration k). More details are provided in Appendix B.

To analyze the convergence of AAR-BCD, we will need to construct a more sophisticated duality gap than in the previous section, as follows.

4.1. Approximate Duality Gap

We define the upper bound to be $U_k = f(\mathbf{y}_k)$. The constructed lower bound L_k from previous subsection is not directly useful for the analysis of (AAR-BCD). Instead, we

$$\Lambda_k = \frac{\sum_{j=1}^k a_j f(\mathbf{x}_j) + \min_{\mathbf{u}\in\mathbb{R}^N} \left\{ \sum_{j=1}^k a_j \langle \Delta_j, \mathbf{u} - \mathbf{x}_j \rangle + \sum_{i=1}^{n-1} \frac{\sigma_i}{2} \|\mathbf{u}^i - \mathbf{x}_1^i\|^2 \right\} - \sum_{i=1}^{n-1} \frac{\sigma_i}{2} \|\mathbf{x}_*^i - \mathbf{x}_1^i\|^2}{A_k}. \quad (4.2)$$

will construct a random variable Λ_k, which in expectation is upper bounded by $f(\mathbf{x}^*)$. The general idea, as in previous subsection, is to show that some notion of approximate duality gap decreases in expectation.

Towards constructing Λ_k, we first prove the following technical proposition, whose proof is in Appendix A.

Proposition 4.2. *Let $\mathbf{x}_k$ be as in (AAR-BCD). Then:*

$$\mathbb{E}[\sum_{j=1}^k a_j \langle \Delta_j, \mathbf{x}_* - \mathbf{x}_j \rangle] = \mathbb{E}[\sum_{j=1}^k a_j \langle \nabla f(\mathbf{x}_j), \mathbf{x}_* - \mathbf{x}_j \rangle].$$

Define the randomized lower bound as in Eq. (4.2), and observe that (4.1) defines $\mathbf{v}_k$ as the argument of the minimum from Λ_k. The crucial property of Λ_k is that it lower bounds $f(\mathbf{x}_*)$ in expectation, as shown in the following lemma.

Lemma 4.3. *Let $\mathbf{x}_k$ be as in (AAR-BCD). Then $f(\mathbf{x}_*) \geq \mathbb{E}[\Lambda_k]$.*

Proof. By convexity of $f(\cdot)$, for any sequence $\{\tilde{\mathbf{x}}_j\}$ from $\mathbb{R}^N$, $f(\mathbf{x}_*) \geq \frac{\sum_{j=1}^k a_j (f(\tilde{\mathbf{x}}_j) + \langle \nabla f(\tilde{\mathbf{x}}_j), \mathbf{x}_* - \tilde{\mathbf{x}}_j \rangle)}{A_k}$. Since the statement holds for any sequence $\{\tilde{\mathbf{x}}_j\}$, it also holds if $\{\tilde{\mathbf{x}}_j\}$ is selected according to some probability distribution. In particular, for $\{\tilde{\mathbf{x}}_j\} = \{\mathbf{x}_j\}$:

$$f(\mathbf{x}_*) \geq \mathbb{E}\Big[\frac{\sum_{j=1}^k a_j (f(\mathbf{x}_j) + \langle \nabla f(\mathbf{x}_j), \mathbf{x}_* - \mathbf{x}_j \rangle)}{A_k}\Big].$$

By linearity of expectation and Proposition 4.2:

$$f(\mathbf{x}_*) \geq \mathbb{E}\Big[\frac{\sum_{j=1}^k a_j (f(\mathbf{x}_j) + \langle \Delta_j, \mathbf{x}_* - \mathbf{x}_j \rangle)}{A_k}\Big]. \quad (4.3)$$

Adding and subtracting (deterministic) $\sum_{i=1}^{n-1} \frac{\sigma_i}{2} \|\mathbf{x}_*^i - \mathbf{x}_1^i\|^2$ to/from (4.3) and using that:

$$\begin{aligned}
&\sum_{j=1}^k a_j \langle \Delta_j, \mathbf{x}_* - \mathbf{x}_j \rangle + \sum_{i=1}^{n-1} \frac{\sigma_i}{2} \|\mathbf{x}_*^i - \mathbf{x}_1^i\|^2 \\
&\quad \geq \min_{\mathbf{u}} \Big\{ \sum_{j=1}^k a_j \langle \Delta_j, \mathbf{u} - \mathbf{x}_j \rangle + \sum_{i=1}^{n-1} \frac{\sigma_i}{2} \|\mathbf{u}^i - \mathbf{x}_1^i\|^2 \Big\} \\
&\quad = \min_{\mathbf{u}} m_k(\mathbf{u}),
\end{aligned}$$

where $m_k(\mathbf{u}) = \sum_{j=1}^k a_j \langle \Delta_j, \mathbf{u} - \mathbf{x}_j \rangle + \sum_{i=1}^{n-1} \frac{\sigma_i}{2} \|\mathbf{u}^i - \mathbf{x}_1^i\|^2$, it follows that:

$$f(\mathbf{x}_*) \geq \mathbb{E}\Big[\frac{\sum_{j=1}^k a_j f(\mathbf{x}_j) - \sum_{i=1}^{n-1} \frac{\sigma_i}{2} \|\mathbf{x}_*^i - \mathbf{x}_1^i\|^2}{A_k} + \frac{\min_{\mathbf{u}\in\mathbb{R}^N} m_k(\mathbf{u})}{A_k}\Big],$$

which is equal to $\mathbb{E}[\Lambda_k]$, and completes the proof. □

Similar as before, define the approximate gap as $\Gamma_k = U_k - \Lambda_k$. Then, we can bound the initial gap as follows.

Proposition 4.4. *If $a_1 = \frac{{a_1}^2}{A_1} \leq \frac{\sigma_i {p_i}^2}{L_i}$, $\forall i \in \{1, ..., n-1\}$, then $\mathbb{E}[A_1\Gamma_1] \leq \sum_{i=1}^{n-1} \frac{\sigma_i}{2} \|\mathbf{x}_* - \mathbf{x}_1\|^2$.*

Proof. As $a_1 = A_1$ and $\mathbf{y}_1$ differs from $\mathbf{x}_1$ only over block $i = i_1$, by smoothness of $f(\cdot)$:

$$\begin{aligned}
&A_1 U_1 = A_1 f(\mathbf{y}_1) \\
&\leq a_1 f(\mathbf{x}_1) + a_1 \left\langle \nabla_i f(\mathbf{x}_1), \mathbf{y}_1^i - \mathbf{x}_1^i \right\rangle + \frac{a_1 L_i}{2} \|\mathbf{y}_1^i - \mathbf{x}_1^i\|^2.
\end{aligned}$$

On the other hand, the initial lower bound is:

$$\begin{aligned}
A_1\Lambda_1 =& a_1 (f(\mathbf{x}_1) + \langle \Delta_1, \mathbf{v}_1 - \mathbf{x}_1 \rangle) \\
&+ \sum_{i=1}^{n-1} \frac{\sigma_i}{2} \|\mathbf{v}_1^i - \mathbf{x}_1^i\|^2 - \sum_{i=1}^{n-1} \frac{\sigma_i}{2} \|\mathbf{x}_*^i - \mathbf{x}_1^i\|^2.
\end{aligned}$$

Recall that $\mathbf{y}_1^i = \mathbf{x}_1^i + \frac{1}{p_i}(\mathbf{v}_1^i - \mathbf{x}_1^i)$. Using $A_1\Gamma_1 = A_1 U_1 - A_1\Lambda_1$ and the bounds on U_1, Λ_1 from the above: $A_1\Gamma_1 \leq \sum_{i=1}^{n-1} \frac{\sigma_i}{2} \|\mathbf{x}_*^i - \mathbf{x}_1^i\|^2$, as $a_1 \leq {p_i}^2 \frac{\sigma_i}{L_i}$, and, thus, $\mathbb{E}[A_1\Gamma_1] \leq \sum_{i=1}^{n-1} \frac{\sigma_i}{2} \|\mathbf{x}_*^i - \mathbf{x}_1^i\|^2$. □

The next part of the proof is to show that $A_k\Gamma_k$ is a supermartingale. The proof is provided in Appendix A.

Lemma 4.5. *If $\frac{{a_k}^2}{A_k} \leq \frac{{p_i}^2 \sigma_i}{L_i}$, $\forall i \in \{1, \ldots, n-1\}$, then $\mathbb{E}[A_k\Gamma_k | \mathcal{F}_{k-1}] \leq A_{k-1}\Gamma_{k-1}$.*

Finally, we bound the convergence of (AAR-BCD).

Theorem 4.6. *Let $\mathbf{x}_k$, $\mathbf{y}_k$ evolve according to (AAR-BCD), for $\frac{{a_k}^2}{A_k} = \min_{1\leq i\leq n-1} \frac{\sigma_i {p_i}^2}{L_i} = \text{const}$. Then, $\forall k \geq 1$:*

$$\mathbb{E}[f(\mathbf{y}_k)] - f(\mathbf{x}_*) \leq \frac{\sum_{i=1}^{n-1} \sigma_i \|\mathbf{x}_*^i - \mathbf{x}_a^i\|^2}{2A_k}.$$

In particular, if $p_i = \frac{\sqrt{L_i}}{\sum_{i'=1}^{n-1} \sqrt{L_{i'}}}$, $\sigma_i = (\sum_{i'=1}^{n-1} \sqrt{L_{i'}})^2$, and $a_1 = 1$, then:

$$\mathbb{E}[f(\mathbf{y}_k)] - f(\mathbf{x}_*) \leq \frac{2(\sum_{i'=1}^{n-1} \sqrt{L_{i'}})^2 \sum_{i=1}^{n-1} \|\mathbf{x}_*^i - \mathbf{x}_1^i\|^2}{k(k+3)}.$$

Alternatively, if $p_i = \frac{1}{n-1}$, $\sigma_i = L_i$, and $a_1 = \frac{1}{(n-1)^2}$:

$$\mathbb{E}[f(\mathbf{y}_k)] - f(\mathbf{x}_*) \leq \frac{2(n-1)^2 \sum_{i=1}^{n-1} L_i \|\mathbf{x}_*^i - \mathbf{x}_1^i\|^2}{k(k+3)}.$$

Proof. The first part of the proof follows immediately by applying Proposition 4.4 and Lemma 4.5. The second part follows by plugging in the particular choice of parameters and observing that a_j grows faster than $\frac{j+1}{2}$ in the former, and faster than $\frac{j+1}{2(n-1)^2}$ in the latter case. □

Finally, we make a few remarks regarding Theorem 4.6. In the setting without a non-smooth block (when n^{th} block is empty), (AAR-BCD) with sampling probabilities $p_i \sim \sqrt{L_i}$ has the same convergence bound as the NU_ACDM algorithm (Allen-Zhu et al., 2016) and the ALPHA algorithm for smooth minimization (Qu & Richtárik, 2016). Further, when the sampling probabilities are uniform, (AAR-BCD) converges at the same rate as the ACDM algorithm (Nesterov, 2012) and the APCG algorithm applied to non-composite functions (Lin et al., 2014).

5. Numerical Experiments

To illustrate the results, we solve the least squares problem on the BlogFeedback Data Set (Buza, 2014) obtained from UCI Machine Learning Repository (Lichman, 2013). The data set contains 280 attributes and 52,396 data points. The attributes correspond to various metrics of crawled blog posts. The data is labeled, and the labels correspond to the number of comments that were posted within 24 hours from a fixed basetime. The goal of a regression method is to predict the number of comments that a blog post receives.

What makes linear regression with least squares on this dataset particularly suitable to our setting is that the smoothness parameters of individual coordinates in the least squares problem take values from a large interval, even when the data matrix $\mathbf{A}$ is scaled by its maximum absolute value (the values are between 0 and ∼354).[2] The minimum eigenvalue of $\mathbf{A}^T\mathbf{A}$ is zero (i.e., $\mathbf{A}^T\mathbf{A}$ is not a full-rank matrix), and thus the problem is not strongly convex.

We partition the data into blocks as follows. We first sort the coordinates by their individual smoothness parameters. Then, we group the first N/n coordinates (from the sorted list of coordinates) into the first block, the second N/n coordinates into the second block, and so on. The chosen block sizes N/n are 5, 10, 20, 40, corresponding to $n = \{56, 28, 14, 7\}$ coordinate blocks, respectively.

The distribution of the smoothness parameters over blocks, for all chosen block sizes, is shown in Fig. 1(a)-1(d). Observe that as the block size increases (going from left to right in Fig. 1(a)-1(d)), the discrepancy between the two largest smoothness parameters increases.

[2]We did not compare AR-BCD and AAR-BCD to other methods on problems with a non-smooth block ($L_n = \infty$), as no other methods have any known theoretical guarantees in such a setting.

In all the comparisons between the different methods, we define an epoch to be equal to n iterations (this would correspond to a single iteration of a full-gradient method). The graphs plot the optimality gap of the methods over epochs, where the optimal objective value f^* is estimated via a higher precision method and denoted by $\hat{f}^*$. All the results are shown for 50 method repetitions, with bold lines representing the median[3] optimality gap over those 50 runs. The norm used in all the experiments is ℓ_2, i.e., $\|\cdot\| = \|\cdot\|_2$.

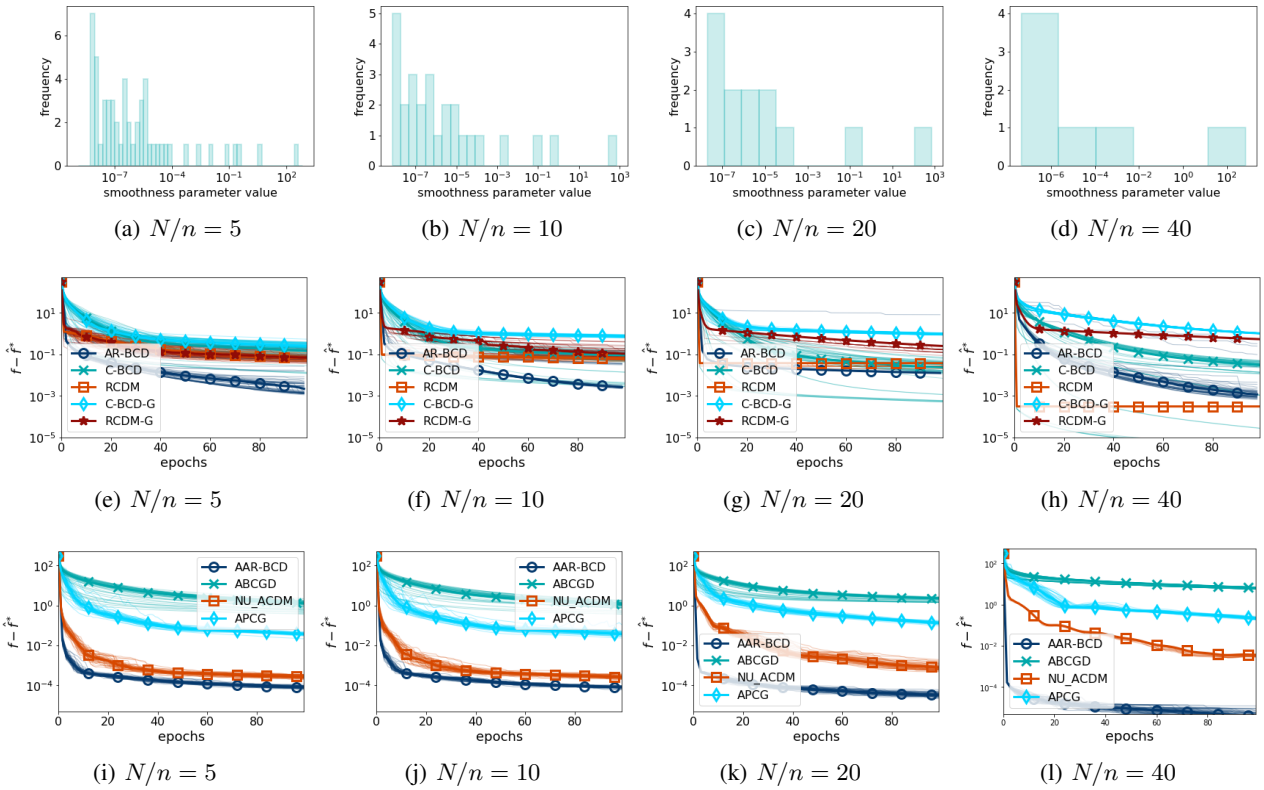
Non-accelerated methods We first compare AR-BCD with a gradient step to RCDM (Nesterov, 2012) and standard cyclic BCD – C-BCD (see, e.g., (Beck & Tetruashvili, 2013)). To make the comparison fair, as AR-BCD makes two steps per iteration, we slow it down by a factor of two compared to the other methods (i.e., we count one iteration of AR-BCD as two). In the comparison, we consider two cases for RCDM and C-BCD: (i) the case in which these two algorithms perform gradient steps on the first $n-1$ blocks and exact minimization on the n^{th} block (denoted by RCDM and C-BCD in the figure), and (ii) the case in which the algorithms perform gradient steps on all blocks (denoted by RCDM-G and C-BCD-G in the figure). The sampling probabilities for RCDM and AR-BCD are proportional to the block smoothness parameters. The permutation for C-BCD is random, but fixed in each method run.

Fig. 1(e)-1(h) shows the comparison of the described non-accelerated algorithms, for block sizes $N/n \in \{5, 10, 20, 40\}$. The first observation to make is that adding exact minimization over the least smooth block speeds up the convergence of both C-BCD and RCDM, suggesting that the existing analysis of these two methods is not tight. Second, AR-BCD generally converges to a lower optimality gap. While RCDM makes a large initial progress, it stagnates afterwards due to the highly non-uniform sampling probabilities, whereas AR-BCD keeps making progress.

Accelerated methods Finally, we compare AAR-BCD to NU_ACDM (Allen-Zhu et al., 2016), APCG (Lin et al., 2014), and accelerated C-BCD (ABCGD) from (Beck & Tetruashvili, 2013). As AAR-BCD makes three steps per iteration (as opposed to two steps normally taken by other methods), we slow it down by a factor 1.5 (i.e., we count one iteration of AAR-BCD as 1.5). We chose the sampling probabilities of NU_ACDM and AAR-BCD to be proportional to $\sqrt{L_i}$, while the sampling probabilities for APCG are uniform[4]. Similar as before, each full run of ABCGD is performed on a random but fixed permutation of the blocks.

[3]We choose to show the median as opposed to the mean, as it is well-known that in the presence of outliers the median is a robust estimator of the true mean (Hampel et al., 2011).

[4]The theoretical results for APCG were only presented for uniform sampling (Lin et al., 2014).



(a) $N/n = 5$ (b) $N/n = 10$ (c) $N/n = 20$ (d) $N/n = 40$

(e) $N/n = 5$ (f) $N/n = 10$ (g) $N/n = 20$ (h) $N/n = 40$

(i) $N/n = 5$ (j) $N/n = 10$ (k) $N/n = 20$ (l) $N/n = 40$

Figure 1. Comparison of different block coordinate descent methods: (a)-(d) distribution of smoothness parameters over blocks, (e)-(h) comparison of non-accelerated methods, and (i)-(l) comparison of accelerated methods. Block sizes N/n increase going left to right.

The results are shown in Fig. 1(i)-1(l). Compared to APCG (and ABCGD), NU_ACDM and AAR-BCD converge much faster, which is expected, as the distribution of the smoothness parameters is highly non-uniform and the meethods with non-uniform sampling are theoretically faster by factor of the order $\sqrt{n}$ (Allen-Zhu et al., 2016). As the block size is increased (going left to right), the discrepancy between the smoothness parameters of the least smooth block and the remaining blocks increases, and, as expected, AAR-BCD exhibits more dramatic improvements compared to the other methods.

6. Conclusion

We presented a novel block coordinate descent algorithm AR-BCD and its accelerated version for smooth minimization AAR-BCD. Our work answers the open question of (Beck & Tetruashvili, 2013) whether the convergence of block coordinate descent methods intrinsically depends on the largest smoothness parameter over all the blocks by showing that such a dependence is not necessary, as long as exact minimization over the least smooth block is possible. Before our work, such a result only existed for the setting of two blocks, using the alternating minimization method.

There are several research directions that merit further investigation. For example, we observed empirically that exact optimization over the non-smooth block improves the performance of RCDM and C-BCD, which is not justified by the existing analytical bounds. We expect that in both of these methods the dependence on the least smooth block can be removed, possibly at the cost of a worse dependence on the number of blocks. Further, AR-BCD and AAR-BCD are mainly useful when the discrepancy between the largest block smoothness parameter and the remaining smoothness parameters is large, while under uniform distribution of the smoothness parameters it can be slower than other methods by a factor 1.5-2. It is an interesting question whether there are modifications to AR-BCD and AAR-BCD that would make them uniformly better than the alternatives.

Acknowledgements

Part of this work was done while the authors were visiting the Simons Institute for the Theory of Computing. It was partially supported by NSF grant #CCF-1718342, by the DIMACS/Simons Collaboration on Bridging Continuous and Discrete Optimization through NSF grant #CCF-1740425 and by DHS-ALERT subaward 505035-78050.

References

Allen-Zhu, Z., Qu, Z., Richtárik, P., and Yuan, Y. Even faster accelerated coordinate descent using non-uniform sampling. In *Proc. ICML'16*, 2016.

Beck, A. On the convergence of alternating minimization for convex programming with applications to iteratively reweighted least squares and decomposition schemes. *SIAM J. Optimiz.*, 25(1):185–209, 2015.

Beck, A. and Tetruashvili, L. On the convergence of block coordinate descent type methods. *SIAM J. Optimiz.*, 23 (4):2037–2060, 2013.

Bertsekas, D. P. *Nonlinear programming*. Athena scientific Belmont, 1999.

Boyd, S. and Vandenberghe, L. *Convex optimization*. Cambridge university press, 2004.

Bubeck, S. *Theory of Convex Optimization for Machine Learning*. 2014. arXiv preprint, arXiv:1405.4980v1.

Buza, K. Feedback prediction for blogs. In *Data analysis, machine learning and knowledge discovery*, pp. 145–152. Springer, 2014.

Diakonikolas, J. and Orecchia, L. The approximate duality gap technique: A unified theory of first-order methods, 2017. arXiv preprint, arXiv:1712.02485.

Fercoq, O. and Richtárik, P. Accelerated, parallel, and proximal coordinate descent. *SIAM J. Optimiz.*, 25(4):1997–2023, 2015.

Gower, R. M. and Richtárik, P. Stochastic dual ascent for solving linear systems. *arXiv preprint arXiv:1512.06890*, 2015.

Hampel, F. R., Ronchetti, E. M., Rousseeuw, P. J., and Stahel, W. A. *Robust statistics: the approach based on influence functions*, volume 196. John Wiley & Sons, 2011.

Lee, Y. T. and Sidford, A. Efficient accelerated coordinate descent methods and faster algorithms for solving linear systems. In *Proc. IEEE FOCS'13*, 2013.

Lichman, M. UCI machine learning repository, 2013. URL http://archive.ics.uci.edu/ml.

Lin, Q., Lu, Z., and Xiao, L. An accelerated proximal coordinate gradient method. In *Proc. NIPS'14*, 2014.

Nesterov, Y. Efficiency of coordinate descent methods on huge-scale optimization problems. *SIAM J. Optimiz.*, 22 (2):341–362, 2012.

Nesterov, Y. and Stich, S. U. Efficiency of the accelerated coordinate descent method on structured optimization problems. *SIAM J. Optimiz.*, 27(1):110–123, 2017.

Ortega, J. M. and Rheinboldt, W. C. *Iterative solution of nonlinear equations in several variables*, volume 30. SIAM, 1970.

Qu, Z. and Richtárik, P. Coordinate descent with arbitrary sampling i: Algorithms and complexity. *Optimization Methods and Software*, 31(5):829–857, 2016.

Qu, Z., Richtárik, P., Takáč, M., and Fercoq, O. SDNA: Stochastic dual Newton ascent for empirical risk minimization. In *Proc. ICML'16*, 2016.

Richtárik, P. and Takáč, M. Iteration complexity of randomized block-coordinate descent methods for minimizing a composite function. *Math. Prog.*, 144(1-2):1–38, 2014.

Saha, A. and Tewari, A. On the nonasymptotic convergence of cyclic coordinate descent methods. *SIAM J. Optimiz.*, 23(1):576–601, 2013.

Strohmer, T. and Vershynin, R. A Randomized Kaczmarz Algorithm with Exponential Convergence. *J. Fourier Anal. Appl.*, 15(2):262–278, 2009.

Sun, R. and Hong, M. Improved iteration complexity bounds of cyclic block coordinate descent for convex problems. In *Proc. NIPS'15*, 2015.

Tseng, P. and Yun, S. A coordinate gradient descent method for nonsmooth separable minimization. *Math. Prog.*, 117(1-2):387–423, 2009.

Wright, S. J. Coordinate descent algorithms. *Math. Prog.*, 151(1):3–34, 2015.