An Improved Convergence Analysis of Cyclic Block Coordinate Descent-type Methods for Strongly Convex Minimization

Xingguo Li* University of Minnesota

Tuo Zhao* Johns Hopkins University

Raman Arora Johns Hopkins University

Han Liu Princeton University

Mingyi Hong Iowa State University

Abstract

The cyclic block coordinate descent-type (CBCD-type) methods have shown remarkable computational performance for solving strongly convex minimization problems. Typical applications include many popular statistical machine learning methods such as elastic-net regression, ridge penalized logistic regression, and sparse additive regression. Existing optimization literature has shown that the CBCD-type methods attain iteration complexity of $\mathcal{O}(p \cdot \log(1/\epsilon))$, where ϵ is a pre-specified accuracy of the objective value, and p is the number of blocks. However, such iteration complexity explicitly depends on p, and therefore is at least p times worse than those of gradient descent methods. To bridge this theoretical gap, we propose an improved convergence analysis for the CBCD-type methods. In particular, we first show that for a family of quadratic minimization problems, the iteration complexity of the CBCD-type methods matches that of the GD methods in term of dependency on p (up to a $\log^2 p$ factor). Thus our complexity bounds are sharper than the existing bounds by at least a factor of $p/\log^2 p$. We also provide a lower bound to confirm that our improved complexity bounds are tight (up to a $\log^2 p$ factor) if the largest and smallest eigenvalues of the Hessian matrix do not scale with p. Finally, we generalize our analysis to other strongly convex minimization problems beyond quadratic ones.

Appearing in Proceedings of the 19^{th} International Conference on Artificial Intelligence and Statistics (AISTATS) 2016, Cadiz, Spain. JMLR: W&CP volume 51. *XL and TZ contributed equally.

1 Introduction

We consider a class of convex minimization problems in statistical machine learning:

$$x^* = \operatorname*{argmin}_{x \in \mathbb{R}^d} \mathcal{L}(x) + \mathcal{R}(x), \tag{1}$$

where $\mathcal{L}(\cdot)$ is a twice differentiable convex loss function and $\mathcal{R}(\cdot)$ is a possibly nonsmooth and strongly convex penalty function. Typical applications of (1) include elastic-net regression (Zou and Hastie, 2005), ridge penalized logistic regression (Hastie et al., 2009), support vector machine (Vapnik and Vapnik, 1998) and many other statistical machine learning problems (Hastie et al., 2009). The penalty function $\mathcal{R}(x)$ in these applications is block coordinate decomposable. For notational simplicity, we assume that there exists a partition of d coordinates such that

$$x = [x_1^\top, \ldots, x_p^\top]^\top \in \mathbb{R}^d,$$

where $x_j \in \mathbb{R}^{d_j}$, $d = \sum_{j=1}^p d_j$, and $d_j \ll p$. Then we can rewrite the objective function in (1) as

$$\mathcal{F}(x) = \mathcal{L}(x_1, \ldots, x_p) + \sum_{j=1}^p \mathcal{R}_j(x_j).$$

Many algorithms such as gradient decent (GD) methods (Nesterov, 2004, 2007), cyclic block coordinate descent-type (CBCD-type) methods (Tseng, 1993, 2001; Friedman et al., 2007; Liu et al., 2009; Tseng and Yun, 2009; Saha and Tewari, 2013; Schmidt and Friedlander, 2015; Zhao and Liu, 2015; Zhao et al., 2014b,a, 2012; Li et al., 2015b), and alternating direction method of multipliers (ADMM, Gabay and Mercier (1976); Boyd et al. (2011); He and Yuan (2012); Zhao and Liu (2012); Liu et al. (2014, 2015); Li et al. (2015a)) have been proposed to solve (1). Among these algorithms, the CBCD-type methods have been immensely successful (Friedman et al., 2007, 2010; Mazumder et al., 2011; Zhao et al., 2014a).

One popular instance of the CBCD-type methods is the cyclic block coordinate minimization (CBCM) method, which minimizes (1) with respect to a single block of variables while holding the rest fixed. Particularly, at the $(t+1)$-th iteration, given $x^{(t)}$, we choose to solve a collection of optimization problems: For $j = 1, \ldots, p$,

$$x_j^{(t+1)} = \underset{x_j}{\operatorname{argmin}} \ \mathcal{L}\left(x_{1:(j-1)}^{(t+1)}, x_j, x_{(j+1):p}^{(t)}\right) + \mathcal{R}_j(x_j), \quad (2)$$

where $x_{1:(j-1)}^{(t+1)} = [x_1^{(t+1)\top}, \ldots, x_{j-1}^{(t+1)\top}]^\top$ and $x_{(j+1):p}^{(t)} = [x_{j+1}^{(t)\top}, \ldots, x_p^{(t)\top}]^\top$. For some applications (e.g. elastic-net penalized linear regression), we can obtain a simple closed form solution to (2), but for many other applications (e.g. ridge-penalized logistic regression), (2) does not admit a closed form solution and requires more sophisticated optimization procedures.

A popular alternative is to solve a quadratic approximation of (2) using the cyclic block coordinate gradient descent (CBCGD) method. For notational simplicity, we denote the partial gradient $\nabla_{x_j}\mathcal{L}(x)$ by $\nabla_j\mathcal{L}(x)$. Then the CBCGD method solves a collection of optimization problems: For $j = 1, \ldots, p$,

$$x_j^{(t+1)} = \underset{x_j}{\operatorname{argmin}} \ (x_j - x_j^{(t)})^\top \nabla_j \mathcal{L}\left(x_{1:(j-1)}^{(t+1)}, x_{(j):p}^{(t)}\right) + \frac{\eta_j}{2}\|x_j - x_j^{(t)}\|^2 + \mathcal{R}_j(x_j), \quad (3)$$

where $\eta_j > 0$ is a step-size parameter for the j-th block.

There have been many results on iteration complexity[1] of block coordinate descent-type (BCD-type) methods, but most of them focus on the randomized BCD-type methods, where blocks are randomly chosen with replacement in each iteration (Shalev-Shwartz and Tewari, 2011; Richtárik and Takáč, 2012; Lu and Xiao, 2015). In contrast, existing literature on cyclic BCD-type methods is rather limited. Specifically, one line of research focuses on minimizing smooth objective functions, and has shown that given a pre-specified accuracy ϵ for the objective value, the CBCGD method attains linear iteration complexity of $\mathcal{O}(\log(1/\epsilon))$ for minimizing smooth and strongly convex problems, and sublinear iteration complexity of $\mathcal{O}(1/\epsilon)$ for smooth and nonstrongly convex problems (Beck and Tetruashvili, 2013). Another line of research focuses on minimizing nonsmooth composite objective functions such as (1), and has shown that the CBCM and CBCGD methods attain sublinear iteration complexity of $\mathcal{O}(1/\epsilon)$, when the objective function is nonstronlgy convex (Hong et al., 2013).

[1]Each iteration considers one update of all blocks.

Here we are interested in establishing an improved iteration complexity of the CBCM and CBCGD methods, when the nonsmooth composite objective function is strongly convex. Particularly, Beck and Tetruashvili (2013) has shown that for smooth minimization, the CBCGD method attains linear iteration complexity of

$$\mathcal{O}\left(\mu^{-1} p L^2 \log(1/\epsilon)\right), \quad (4)$$

where L is the Lipschitz constant of the gradient mapping $\nabla\mathcal{L}(x)$ and μ is the strongly convex coefficient of the objective function. However, such iteration complexity explicitly depends on p (the number of blocks), and therefore is at least p times worse than those of the gradient descent (GD) methods. To bridge this theoretical gap, we propose an improved convergence analysis for the CBCD-type methods. Specifically, we show that for a family of quadratic minimization problems, the iteration complexity of the CBCD-type methods matches that of the GD methods in term of dependency on p (up to a $\log^2 p$ factor). More precisely, when $\mathcal{L}(x)$ is quadratic, the iteration complexity of the CBCD-type methods is

$$\mathcal{O}\left(\mu^{-1} L^2 \log^2 p \log(1/\epsilon)\right). \quad (5)$$

As can be seen easily, (5) is better than (4) by a factor of $p/\log^2 p$. We also provide a lower bound analysis that confirms that our improved iteration complexity is tight (up to a $\log^2 p$ factor) if the largest and smallest eigenvalues of the Hessian matrix do not scale with p. Finally, we provide the analysis of other strongly convex minimization problems beyond quadratic ones. Specifically, for general minimization problems with both smooth and nonsmooth regularizations, the iteration complexity of the CBCD-type method matches with the result in Beck and Tetruashvili (2013) that only analyzed smooth $\mathcal{R}(\cdot)$; for more details refer to Table 1[2]. It is worth mentioning that all the above results on the CBCD-type methods can be used to establish the iteration complexity for popular permuted BCM (PBCM) and permuted BCGD (PBCGD) methods, in which the blocks are randomly sampled without replacement.

2 Notations and Assumptions

We start with some notations used in this paper. Given a vector $\boldsymbol{v} = (v_1, \ldots, v_d)^\top \in \mathbb{R}^d$, we define vector norms: $\|\boldsymbol{v}\|_1 = \sum_j |v_j|$, $\|\boldsymbol{v}\|^2 = \sum_j v_j^2$, and $\|\boldsymbol{v}\|_\infty = \max_j |v_j|$. Let $\{\mathcal{A}_1, \ldots, \mathcal{A}_p\}$ be a partition of all d coordinates with $|\mathcal{A}_j| = d_j$ and $\sum_{j=1}^p d_j = d$. We

[2]When $\mathcal{R}(\cdot)$ is nonsmooth, the optimization problem is actually solved by the cyclic block coordinate proximal gradient (CBCPGD) method. For notational convenience in this paper, however, we simply call it the CBCGD method.

Table 1: Compared with Beck and Tetruashvili (2013), our contribution contains many folds: (1) Developing the iteration complexity bounds of the CBCM methods for different specifications on $\mathcal{L}(\cdot)$ and $\mathcal{R}(\cdot)$; (2) Developing the iteration complexity bound of CBCGD for quadratic $\mathcal{L}(\cdot)$ + nonsmooth $\mathcal{R}(\cdot)$; (3) Provide the iteration complexity bound of CBCGD for both smooth and nonsmooth $\mathcal{R}(\cdot)$.

	Method	$\mathcal{L}(\cdot)$	$\mathcal{R}(\cdot)$	Improved Iteration Complexity	Beck and Tetruashvili (2013)
[a]	CBCGD	Quadratic	Smooth	$\mathcal{O}\left(\mu^{-1}\log^2 pL^2\log(1/\epsilon)\right)$	$\mathcal{O}\left(\mu^{-1}pL^2\log(1/\epsilon)\right)$
[b]	CBCGD	Quadratic	Nonsmooth	$\mathcal{O}\left(\mu^{-1}\log^2 pL^2\log(1/\epsilon)\right)$	N/A
[c]	CBCGD	General Convex	Smooth	$\mathcal{O}\left(\mu^{-1}pL^2\log(1/\epsilon)\right)$	$\mathcal{O}\left(\mu^{-1}pL^2\log(1/\epsilon)\right)$
[d]	CBCGD	General Convex	Nonsmooth	$\mathcal{O}\left(\mu^{-1}pL^2\log(1/\epsilon)\right)$	N/A
[e]	CBCM	Quadratic	Smooth	$\mathcal{O}\left(\mu^{-1}\log^2 pL^2\log(1/\epsilon)\right)$	N/A
[f]	CBCM	Quadratic	Nonsmooth	$\mathcal{O}\left(\mu^{-1}\log^2 pL^2\log(1/\epsilon)\right)$	N/A
[g]	CBCM	General Convex	Smooth	$\mathcal{O}\left(\mu^{-1}pL^2\log(1/\epsilon)\right)$	N/A
[h]	CBCM	General Convex	Nonsmooth	$\mathcal{O}\left(\mu^{-1}pL^2\log(1/\epsilon)\right)$	N/A

Remark: Results [a] and [b] are presented in Theorem 3; Results [e] and [f] are presented in Theorem 4; Results [c], [d], [g], and [h] are presented in Theorem 5.

use $\boldsymbol{v}_j$ to denote the subvector of $\boldsymbol{v}$ with all indices in $\mathcal{A}_j$. Given a matrix $A \in \mathbb{R}^{d\times d}$, we use $\lambda_{\max}(A)$ and $\lambda_{\min}(A)$ to denote the largest and smallest eigenvalues of A. We denote $\|A\|$ as the spectral norm of A (i.e., the largest singular value). We denote $\otimes$ and $\odot$ as the Kronecker product and Hadamard (entrywise) product for two matrices respectively.

Before we proceed with our convergence analysis, we introduce some assumptions on $\mathcal{L}(\cdot)$ and $\mathcal{R}(\cdot)$.

Assumption 1. *$\mathcal{L}(\cdot)$ is convex, and its gradient mapping $\nabla\mathcal{L}(\cdot)$ is Lipschitz continuous and also blockwise Lipschitz continuous, i.e., there exist positive constants L and L_j's such that for any $x, x' \in \mathbb{R}^d$ and $j = 1, \ldots, p$, we have*

$$\|\nabla\mathcal{L}(x') - \nabla\mathcal{L}(x)\| \leq L\|x - x'\| \text{ and}$$
$$\|\nabla_j\mathcal{L}\left(x_{1:(j-1)}, x'_j, x_{(j+1):p}\right) - \nabla_j\mathcal{L}(x)\| \leq L_j\|x_j - x'_j\|.$$

Moreover, we define $L_{\max} = \max_j L_j$ and $L_{\min} = \min_j L_j$.

Assumption 2. *$\mathcal{R}(\cdot)$ is strongly convex and also blockwise strongly convex, i.e., there exist positive constants μ and μ_j's such that for any $x, x' \in \mathbb{R}^d$ and $j = 1, \ldots, p$, we have*

$$\mathcal{R}(x) \geq \mathcal{R}(x') + (x - x')^\top\xi' + \frac{\mu}{2}\|x - x'\|^2 \text{ and}$$
$$\mathcal{R}_j(x_j) \geq \mathcal{R}_j(x'_j) + (x_j - x'_j)^\top\xi'_j + \frac{\mu_j}{2}\|x_j - x'_j\|^2,$$

for any ξ' in the sub-differential of $\mathcal{R}(x')$, i.e. $\xi' \in \partial\mathcal{R}(x')$. Moreover, we define $\mu_{\min} = \min_j \mu_j$.

For notational simplicity, we define auxiliary variables

$$L^\mu_{\min} = \min_j L_j + \mu_j \quad \text{and}$$
$$y^{(t,j)} = [x^{(t)\top}_{1:(j-1)}, x^{(t-1)\top}_{j:p}]^\top, \; j = 1, \ldots, p.$$

Our analysis considers $L_{\min}$, $L_{\max}$, $L^\mu_{\min}$, $\mu_{\min}$, μ, and $d_{\max} = \max_j d_j$ as constants, which do not scale with the block size p.

3 Improved Convergence Analysis

Our analysis consists of the following three steps:

(1) Characterize the successive descent after each CBCD iteration;

(2) Characterize the gap towards the optimal objective value after each CBCD iteration;

(3) Combine (1) and (2) to establish the iteration complexity bound.

We present our analysis under different specifications on $\mathcal{L}(\cdot)$ and $\mathcal{R}(\cdot)$.

3.1 Quadratic Minimization

We first consider a scenario, where $\mathcal{L}(\cdot)$ is a quadratic function. Particularly, we solve

$$x^* = \operatorname*{argmin}_{x\in\mathbb{R}^d} \mathcal{L}(x) + \mathcal{R}(x)$$
$$= \operatorname*{argmin}_{\substack{x_j\in\mathbb{R}^{d_j}\\ j=1,\ldots,p}} \frac{1}{2}\left\|\sum_{j=1}^p A_{*j}x_j - b\right\|^2 + \sum_{j=1}^p \mathcal{R}_j(x_j), \quad (6)$$

where $A_{*j} \in \mathbb{R}^{n\times d_j}$ for $j = 1, ..., p$. Typical applications of (6) in statistical machine learning include ridge regression, elastic-net penalized regression, and sparse additive regression.

We first characterize the successive descent of the CBCGD method.

Lemma 1. *Suppose that Assumptions 1 and 2 hold. We choose $\eta_j = L_j$ for the CBCGD method. Then for all $t \geq 1$, we have*

$$\mathcal{F}(x^{(t)}) - \mathcal{F}(x^{(t+1)}) \geq \frac{L^{\mu}_{\min}}{2}\|x^{(t)} - x^{(t+1)}\|^2.$$

Proof. At t-th iteration, there exists a $\xi_j^{(t+1)} \in \partial\mathcal{R}_j(x_j^{(t+1)})$ satisfying the optimality condition:

$$\nabla_j\mathcal{L}(y^{(t+1,j+1)}) + \eta_j(x_j^{(t+1)} - x_j^{(t)}) + \xi_j^{(t+1)} = 0. \quad (7)$$

Then by definition of CBCGD (3), we have

$$\begin{aligned}&\mathcal{F}(y^{(t+1,j)}) - \mathcal{F}(y^{(t+1,j+1)}) \\ &= (x_j^{(t)} - x_j^{(t+1)})^\top\nabla_j\mathcal{L}(y^{(t+1,j+1)}) + \mathcal{R}_j(x_j^{(t)}) \\ &\quad - \frac{L_j}{2}\|x_j^{(t+1)} - x_j^{(t)}\|^2 - \mathcal{R}_j(x_j^{(t+1)}). \end{aligned} \quad (8)$$

By Assumptions 2, we have

$$\begin{aligned}\mathcal{R}_j(x_j^{(t)}) - \mathcal{R}_j(x_j^{(t+1)}) &\geq (x_j^{(t)} - x_j^{(t+1)})^\top\xi_j^{(t+1)} \\ &\quad + \frac{\mu_j}{2}\|x_j^{(t)} - x_j^{(t+1)}\|^2. \end{aligned} \quad (9)$$

Combining (7), (8) and (9), we have

$$\begin{aligned}&\mathcal{F}(y^{(t+1,j)}) - \mathcal{F}(y^{(t+1,j+1)}) \\ &\qquad \geq \frac{L_j + \mu_j}{2}\|x_j^{(t)} - x_j^{(t+1)}\|^2. \end{aligned} \quad (10)$$

We complete the proof via summation of (10) over $j = 1, \ldots, p$. $\square$

Next, we characterize the gap towards the optimal objective value.

Lemma 2. *Suppose that Assumptions 1 and 2 hold. Then for all $t \geq 1$, we have*

$$\mathcal{F}(x^{(t+1)}) - \mathcal{F}(x^*) \leq \frac{L^2\log^2(2p\cdot d_{\max})}{2\mu}\|x^{(t+1)} - x^{(t)}\|^2.$$

Due to space limit, we only provide a proof sketch of Lemma 2, and the detailed proof can be found in Appendix A.

Proof sketch. Since $\mathcal{L}(x)$ is quadratic, its second order Taylor expansion is tight, i.e.

$$\begin{aligned}\mathcal{L}(x^*) = \mathcal{L}(x^{(t+1)}) &+ \langle\nabla\mathcal{L}(x^{(t+1)}), x^* - x^{(t+1)}\rangle \\ &+ \frac{1}{2}\|A(x^{(t+1)} - x^*)\|^2, \end{aligned} \quad (11)$$

where $A = [A_{*1}, \ldots, A_{*p}] \in \mathbb{R}^{n\times d}$.

Consider matrices $\widetilde{P}$ and $\widetilde{A}$, defined as

$$\widetilde{P} = \begin{bmatrix} L_1 & 0 & 0 & \ldots & 0 & 0 \\ 0 & L_2 & 0 & \ldots & 0 & 0 \\ \vdots & \vdots & \vdots & \ldots & \vdots & \vdots \\ 0 & 0 & & \ldots & 0 & L_p \end{bmatrix} \in \mathbb{R}^{p\times p},$$

$$\widetilde{A} = \begin{bmatrix} A_{*1} & 0 & 0 & \ldots & 0 & 0 \\ 0 & A_{*2} & 0 & \ldots & 0 & 0 \\ \vdots & \vdots & \vdots & \ldots & \vdots & \vdots \\ 0 & 0 & & \ldots & 0 & A_{*p} \end{bmatrix} \in \mathbb{R}^{np\times d},$$

which gives us the following inequality

$$\widetilde{P}\otimes I_m \succeq \widetilde{A}^\top\widetilde{A}. \quad (12)$$

To characterize the gap towards the optimal objective value based on the strong convexity of $\mathcal{R}(\cdot)$, we exploit the tightness of the second order Taylor expansion of quadratic $\mathcal{L}(\cdot)$ in (11), the optimality condition of subproblems and a symmetrization technique involving Kronecker product, to show that

$$\begin{aligned}\mathcal{F}(x^{(t+1)}) - \mathcal{F}(x^*) &\leq (x^{(t+1)} - x^{(t)})^\top B(x^{(t+1)} - x^*) \\ &\quad - \frac{\mu}{2}\|x^{(t+1)} - x^*\|^2, \end{aligned} \quad (13)$$

where $B = \left(A^\top A - \widetilde{A}^\top\widetilde{A}\right)\odot D_d + \widetilde{A}^\top\widetilde{A} - \widetilde{P}\otimes I_m$. By minimizing the R.H.S. of (13) w.r.t. x^*, we have

$$\begin{aligned}\mathcal{F}(x^{(t+1)}) - \mathcal{F}(x^*) &\leq \frac{1}{2\mu}\|B(x^{(t+1)} - x^{(t)})\|^2 \\ &\leq \frac{\lambda^2_{\max}(B)}{2\mu}\|x^{(t+1)} - x^{(t)}\|^2 \end{aligned}$$

Finally, we obtain the desired result by bounding the spectral norm of B using the result for triangular truncation operation, i.e.,

$$\begin{aligned}\lambda_{\max}(B) &\leq \lambda_{\max}\left(A^\top A - \widetilde{A}^\top\widetilde{A}\right)\log(2d) \\ &\leq L\log(2p\cdot d_{\max}), \end{aligned}$$

which completes the proof. $\square$

Using Lemmas 1 and 2, we establish the iteration complexity bound of the CBCGD method for minimizing (6) in the next theorem.

Theorem 3. *Suppose that Assumptions 1 and 2 hold. We choose $\eta_j = L_j$ for the CBCGD method. Given a pre-specified accuracy ϵ of the objective value, we need at most*

$$\left\lceil\frac{\mu L^{\mu}_{\min} + L^2\log^2(2p\cdot d_{\max})}{\mu L^{\mu}_{\min}}\log\left(\frac{\mathcal{F}(x^{(0)}) - \mathcal{F}(x^*)}{\epsilon}\right)\right\rceil$$

iterations for the CBCGD method such that $\mathcal{F}(x^{(t)}) - \mathcal{F}(x^*) \leq \epsilon$

Proof. Combining Lemmas 1 and 2, we obtain

$$\begin{aligned}&\mathcal{F}(x^{(t)})-\mathcal{F}(x^*)\\&=[\mathcal{F}(x^{(t)})-\mathcal{F}(x^{(t+1)})]+[\mathcal{F}(x^{(t+1)})-\mathcal{F}(x^*)]\\&\geq\frac{L^\mu_{\min}}{2}\|x^{(t)}-x^{(t+1)}\|^2+[\mathcal{F}(x^{(t+1)})-\mathcal{F}(x^*)]\\&\geq\left(1+\frac{L^\mu_{\min}\mu}{L^2\log^2(2p\cdot d_{\max})}\right)[\mathcal{F}(x^{(t+1)})-\mathcal{F}(x^*)].\end{aligned}$$

Recursively applying the above inequality for $t\geq 1$, we obtain

$$\frac{\mathcal{F}(x^{(t)})-\mathcal{F}(x^*)}{\mathcal{F}(x^{(0)})-\mathcal{F}(x^*)}\leq\left(1-\frac{\mu L^\mu_{\min}}{\mu L^\mu_{\min}+L^2\log^2(2p\cdot d_{\max})}\right)^t.$$

To secure $\mathcal{F}(x^{(t)})-\mathcal{F}(x^*)\leq\epsilon$, we only need a large enough t such that

$$\left(1-\frac{\mu L^\mu_{\min}}{\mu L^\mu_{\min}+L^2\log^2(2pd_{\max})}\right)^t[\mathcal{F}(x^{(0)})-\mathcal{F}(x^*)]\leq\epsilon.$$

We complete the proof by the above inequality, and the basic inequality $\kappa\geq\log^{-1}\left(\frac{\kappa}{\kappa-1}\right)$. □

As can be seen in Theorem 3, the iteration complexity depends on p only in the order of $\log^2 p$, which is generally mild in practice. The iteration complexity of the CBCM method can be established in a similar manner.

Theorem 4. *Suppose that Assumptions 1 and 2 hold. Given a pre-specified accuracy ϵ, we need at most*

$$\left\lceil\frac{\mu\mu_{\min}+4L^2\log^2(2p\cdot d_{\max})}{\mu\mu_{\min}}\log\left(\frac{\mathcal{F}(x^{(0)})-\mathcal{F}(x^*)}{\epsilon}\right)\right\rceil$$

iterations for the CBCM method such that $\mathcal{F}(x^{(t)})-\mathcal{F}(x^*)\leq\epsilon$

Proof. The overall proof also consists of three major steps: (i) successive descent, (ii) gap towards the optimal objective value, and (iii) iteration complexity.

Successive Descent: At t-th iteration, there exists a $\xi_j^{(t+1)}\in\partial\mathcal{R}_j(x_j^{(t+1)})$ satisfying the optimality condition:

$$\nabla_j\mathcal{L}(y^{(t+1,j+1)})+\xi_j^{(t+1)}=0. \qquad (14)$$

Then we have

$$\begin{aligned}&\mathcal{F}(y^{(t+1,j)})-\mathcal{F}(y^{(t+1,j+1)})\\&\overset{\text{(i)}}{\geq}(x_j^{(t)}-x_j^{(t+1)})^\top\nabla_j\mathcal{L}(y^{(t+1,j+1)})\\&\qquad\qquad+\mathcal{R}_j(x_j^{(t)})-\mathcal{R}_j(x_j^{(t+1)})\\&\overset{\text{(ii)}}{\geq}\left(\nabla_j\mathcal{L}(y^{(t+1,j+1)})+\xi_j^{(t+1)}\right)^\top\cdot(x_j^{(t)}-x_j^{(t+1)})\\&\qquad\qquad+\frac{\mu_j}{2}\|x_j^{(t)}-x_j^{(t+1)}\|^2\\&\overset{\text{(iii)}}{=}\frac{\mu_j}{2}\|x_j^{(t)}-x_j^{(t+1)}\|^2,\end{aligned} \qquad (15)$$

where (i) is from the convexity of $\mathcal{L}(\cdot)$, (ii) is from Assumptions 2, and (iii) is from (14). By summation of (15) over $j=1,\dots,p$, we have

$$\mathcal{F}(x^{(t)})-\mathcal{F}(x^{(t+1)})\geq\frac{\mu_{\min}}{2}\|x^{(t)}-x^{(t+1)}\|^2.$$

Gap towards the Optimal Objective Value: The proof follows the same arguments with the proof of Lemma 2, with a few differences.

First, with the optimality condition to the subproblem associated with x_j, $\langle\nabla_j\mathcal{L}(x^{(t+1)})+\xi_j^{(t+1)},x_j-x_j^{(t+1)}\rangle\geq 0$ for any $x_j\in\mathbb{R}^m$, we have

$$\begin{aligned}\mathcal{F}(x^{(t+1)})-\mathcal{F}(x^*)\leq{}&(x^{(t+1)}-x^{(t)})^\top B(x^{(t+1)}-x^*)\\&-\frac{\mu}{2}\|x^{(t+1)}-x^*\|^2,\end{aligned}$$

where $B=\left(A^\top A-\widetilde{A}^\top\widetilde{A}\right)\odot D_d+\widetilde{A}^\top\widetilde{A}$.

Then, using the same technique to bound the eigenvalues for matrices with Hadamard product, we have

$$\begin{aligned}\mathcal{F}(x^{(t+1)})-\mathcal{F}(x^*)&\leq\frac{L^2\log^2(2d)+L^2_{\max}}{\mu}\|x^{(t+1)}-x^{(t)}\|^2\\&\leq\frac{2L^2\log^2(2d)}{\mu}\|x^{(t+1)}-x^{(t)}\|^2.\end{aligned}$$

Iteration Complexity: The analysis follows from that of Theorem 3. □

Theorem 4 establishes that the iteration complexity of the CBCM method matches that of the CBCGD method. To the best of our knowledge, Theorems 3 and 4 are the sharpest iteration complexity analysis of the CBCD-type methods for minimizing (6).

3.2 The Tightness of the Iteration Complexity for Quadratic Problems

We next provide an example to establish the tightness of the above result. We consider the following optimization problem

$$\min_x\mathcal{H}(x):=\|Bx\|^2, \qquad (16)$$

where $B \in \mathbb{R}^{p\times p}$ is a tridiagonal Toplitz matrix defined as follows:

$$B = \begin{bmatrix} 3 & 1 & 0 & 0 & \dots & 0 & 0 & 0 \\ 1 & 3 & 1 & 0 & \dots & 0 & 0 & 0 \\ 0 & 1 & 3 & 1 & \dots & 0 & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots \\ 0 & 0 & 0 & 0 & \dots & 1 & 3 & 1 \\ 0 & 0 & 0 & 0 & \dots & 0 & 1 & 3 \end{bmatrix}.$$

Note that the minimizer to (16) is $x^* = [0, 0, \dots, 0]^\top$, and the eigenvalues of B are given by $3+2\cos(j\pi/(j+1))$ for $j = 1, \dots, p$. Since the Hessian matrix of (16) is $2B^\top B$, we have

$$L = \lambda_{\max}(2B^\top B) \leq 50, \mu = \lambda_{\min}(2B^\top B) \geq 2, \mu_{\min} = 10.$$

Clearly, for this problem the largest and smallest eigenvalues of the Hessian matrix, as well as L/γ do not scale with p. We consider each coordinate $x_j \in \mathbb{R}$ as a block. Then the problem can be rewritten as $\min \|\sum_{j=1}^p B_{*j}x_j\|$, where B_{*j} denotes the j-th column of B. Given an initial solution $x^{(0)}$, we can show that $x^{(1)}$ is generated by

$$x_1^{(1)} = -\frac{1}{4}\left(4x_2^{(0)} + x_3^{(0)}\right), \tag{17}$$

$$x_2^{(1)} = -\frac{1}{5}\left(4x_1^{(1)} + 4x_3^{(0)} + x_4^{(0)}\right) \tag{18}$$

$$x_3^{(1)} = -\frac{1}{5}\left(x_1^{(1)} + 4x_2^{(1)} + x_4^{(0)} + x_5^{(0)}\right), \tag{19}$$

$$x_j^{(1)} = -\frac{1}{5}\left(x_{j-2}^{(1)} + 4x_{j-1}^{(1)} + x_{j+1}^{(0)} + x_{j+2}^{(0)}\right) \tag{20}$$

$$x_{p-1}^{(1)} = -\frac{1}{5}\left(x_{p-3}^{(1)} + 4x_{p-2}^{(0)} + 4x_p^{(0)}\right), \tag{21}$$

$$x_p^{(1)} = -\frac{1}{4}\left(x_{p-2}^{(1)} + 4x_{p-1}^{(1)}\right). \tag{22}$$

Now we choose the initial solution

$$x^{(0)} = \left[1, \frac{9}{32}, \frac{7}{8}, 1, \dots, 1, 1\right]^\top.$$

Then by (17)–(22), we obtain

$$x^{(1)} = \left[-\frac{1}{2}, -\frac{1}{2}, \dots, -\frac{1}{2}, -\frac{3}{10}, -\frac{17}{40}\right]^\top,$$

which yields

$$\mathcal{H}(x^{(1)}) - \mathcal{H}(x^*) \geq \frac{25}{4}(p-3),$$

$$\|x^{(0)} - x^*\|^2 \leq p - 2 + \left(\frac{9}{32}\right)^2 + \left(\frac{7}{8}\right)^2 \leq p - 1.$$

Therefore, we have

$$\frac{\mathcal{H}(x^{(1)}) - \mathcal{H}(x^*)}{\|x^{(0)} - x^*\|^2} \geq \frac{25(p-3)}{4p} \geq \frac{22}{4}.$$

This implies that when the largest and smallest eigenvalues of the Hessian matrix do not scale with p (the number of blocks), the iteration complexity is independent of p, and cannot be further improved.

3.3 General Minimization

We provide an iteration complexity bound of the CBCM and CBCGD methods for a general convex $\mathcal{L}(\cdot)$ and a potentially nonsmooth $\mathcal{R}(\cdot)$.

Theorem 5. *Suppose that Assumptions 1 and 2 hold. We choose $\eta_j = L_j$ for the CBCGD method. Then given a pre-specified accuracy ϵ of the objective value, we need at most*

$$\left\lceil \frac{\mu L_{\min}^\mu + 4pL^2}{\mu L_{\min}^\mu} \log\left(\frac{\mathcal{F}(x^{(0)}) - \mathcal{F}(x^*)}{\epsilon}\right)\right\rceil$$

iterations for the CBCGD method and at most

$$\left\lceil \frac{\mu \mu_{\min} + pL^2}{\mu \mu_{\min}} \log\left(\frac{\mathcal{F}(x^{(0)}) - \mathcal{F}(x^*)}{\epsilon}\right)\right\rceil$$

iterations for the CBCM method to guarantee $\mathcal{F}(x^{(t)}) - \mathcal{F}(x^*) \leq \epsilon$.

Proof. **Successive Descent**: For CBCGD, using the same analysis of Lemma 1, we have that for all $t \geq 1$,

$$\mathcal{F}(x^{(t)}) - \mathcal{F}(x^{(t+1)}) \geq \frac{L_{\min}^\mu}{2}\|x^{(t)} - x^{(t+1)}\|^2.$$

For CBCM, using the same analysis of Theorem 4, we have that for all $t \geq 1$,

$$\mathcal{F}(x^{(t)}) - \mathcal{F}(x^{(t+1)}) \geq \frac{\mu_{\min}}{2}\|x^{(t)} - x^{(t+1)}\|^2.$$

Gap towards the Optimal Objective Value: By the strong convexity of $\mathcal{R}(\cdot)$, we have

$$\begin{aligned} \mathcal{F}(x) - \mathcal{F}(x^{(t+1)}) \geq &\frac{\mu}{2}\|x - x^{(t+1)}\|^2 \\ &+ (x - x^{(t+1)})^\top(\nabla\mathcal{L}(x^{(t+1)}) + \xi^{(t+1)}), \end{aligned} \tag{23}$$

where $\xi_j^{(t+1)} \in \partial\mathcal{R}_j(x_j^{(t+1)})$. We then minimize both sides of (23) with respect to x and obtain

$$\mathcal{F}(x^{(t+1)}) - \mathcal{F}(x^*) \leq \frac{\|\nabla\mathcal{L}(x^{(t+1)}) + \xi^{(t+1)}\|^2}{2\mu}, \tag{24}$$

For CBCGD, we have from the optimality condition $\nabla_j\mathcal{L}(y^{(t+1,j+1)}) + L_j(x_j^{(t+1)} - x_j^{(t)}) + \xi_j^{(t+1)} = 0$,

$$\begin{aligned} &\|\nabla\mathcal{L}(x^{(t+1)}) + \xi^{(t+1)}\|^2 \\ &\leq \sum_{j=1}^p 2\|\nabla\mathcal{L}(x^{(t+1)}) - \nabla_j\mathcal{L}(y^{(t+1,j+1)})\|^2 \\ &\quad + 2L_j^2\|x_j^{(t+1)} - y_j^{(t+1,j)}\|^2 \\ &\leq 4pL^2\|x^{(t+1)} - x^{(t)}\|^2, \end{aligned} \tag{25}$$

Combining (24) and (25), we have

$$\mathcal{F}(x^{(t+1)}) - \mathcal{F}(x^*) \leq \frac{2pL^2\|x^{(t+1)} - x^{(t)}\|^2}{\mu}.$$

For CBCM, we have from the optimality condition $\nabla_j \mathcal{L}(y^{(t+1,j+1)}) + \xi_j^{(t+1)} = 0$,

$$\begin{aligned} &\|\nabla \mathcal{L}(x^{(t+1)}) + \xi^{(t+1)}\|^2 \\ &\leq \sum_{j=1}^{p} \|\nabla \mathcal{L}(x^{(t+1)}) - \nabla_j \mathcal{L}(y^{(t+1,j+1)})\|^2 \\ &\leq pL^2\|x^{(t+1)} - x^{(t)}\|^2, \end{aligned} \tag{26}$$

Combining (24) and (26), we have

$$\mathcal{F}(x^{(t+1)}) - \mathcal{F}(x^*) \leq \frac{pL^2\|x^{(t+1)} - x^{(t)}\|^2}{2\mu}.$$

Iteration Complexity: The analysis follows from that of Theorem 3. □

Theorem 5 is a general result for both smooth and nonsmooth minimizations. In contrast, Beck and Tetruashvili (2013) only covers general smooth minimization.

3.4 Extensions to Nonstrongly Convex Minimization

For nonstrongly convex minimization, we only need to add a strongly convex perturbation to the objective function

$$\widehat{x} = \operatorname{argmin} F(x) + \frac{\sigma}{2}\|x\|^2, \tag{27}$$

where $\sigma > 0$ is a perturbation parameter. Then, the results above can be used to analyze the CBCD-type methods for minimizing (27). Eventually, by setting σ as a reasonable small value, we can establish $\mathcal{O}(1/\epsilon)$-type iteration complexity bounds up to a $\log(1/\epsilon)$ factor. See Shalev-Shwartz and Zhang (2014) for more details.

4 Numerical Results

We consider two typical statistical machine learning problems as examples to illustrate our analysis.

(I) Elastic-net Penalized Linear Regression: Let $A \in \mathbb{R}^{n\times d}$ be the design matrix, and $b \in \mathbb{R}^n$ be the response vector. We solve the following optimization problem

$$\min_{x\in\mathbb{R}^d} \quad \frac{1}{2n}\|b - Ax\|^2 + \lambda_1\|x\|^2 + \lambda_2\|x\|_1,$$

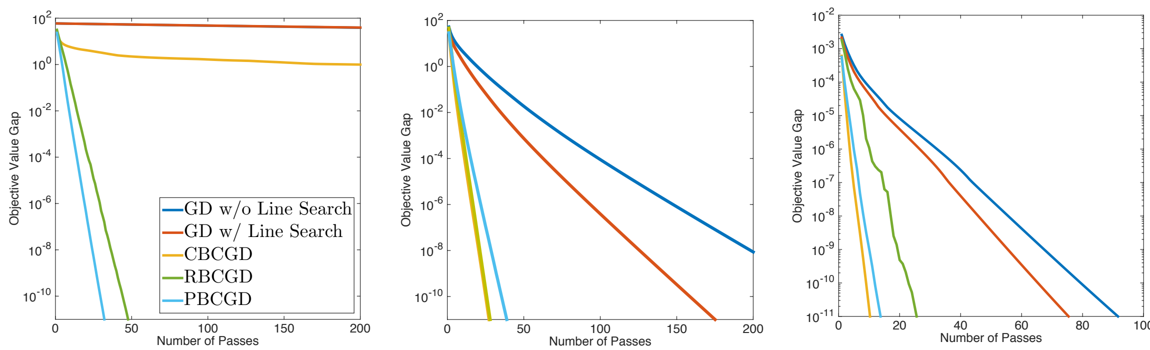
where λ is the regularization parameter. We set $n = 10{,}000$ and $d = 20{,}000$. We simply treat each coordinate as a block (i.e., $d_{\max} = 1$). Each row of A is independently sampled from a 20,000-dimensional Gaussian distribution with mean 0 and covariance matrix Σ. We randomly select 2,500 entries of x, each of which is independently sampled from a uniform distribution over support $(-2, +2)$. The response vector b is generated by the linear model $b = Ax + \epsilon$, where ϵ is sampled from an n-variate Gaussian distribution $N(0, I_n)$. We set $\lambda_1 = 1/\sqrt{n} = 0.01$ and $\lambda_2 = \sqrt{\log d/n} \approx 0.0315$. We normalize A to have $\|A_{*j}\| = \sqrt{n}$ for $j = 1, .., d$, where A_{*j} denotes the j-th column of A. For the BCGD method, we choose $\eta_j = 1$. For the gradient descent method, we either choose $\eta = \lambda_{\max}\left(\frac{1}{n}A^\top A\right)$, or adaptively select η by backtracking line search.

(II) Ridge Penalized Logistic Regression: We solve the following optimization problem

$$\min_{x\in\mathbb{R}^d} \quad \frac{1}{n}\sum_{i=1}^{n}\left[\log(1 + \exp(x^\top A_{i*})) - b_i x^\top A_{i*}\right] + \lambda\|x\|^2.$$

We generate the design matrix A and regression coefficient vector x using the same scheme as sparse linear regression. Again we treat each coordinate as a block (i.e., $d_{\max} = 1$). The response $b = [b_1, ..., b_n]^\top$ is generated by the logistic model $b_i = \text{Bernoulli}([1 + \exp(-x^\top A_{*i})]^{-1})$. We set $\lambda = \sqrt{1/n}$. For the BCGD method, we choose $\eta_j = \frac{1}{4}$. For gradient descent methods, we choose either the step size $\eta = \frac{1}{4}\lambda_{\max}\left(\frac{1}{n}A^\top A\right)$ or adaptively select η by backtracking line search.

We evaluate the computational performance using the number of passes over p blocks of coordinates (normalized iteration complexity). For the CBCGD method, we count one iteration as one pass (all p blocks). For the randomized BCGD (RBCGD) method, we count p iterations as one pass (since it only updates one block in each iteration). Besides the CBCGD and RBCGD methods, we also consider a variant of the CBCGD method named the permuted BCGD (PBCGD) method, which randomly permutes all indices for the p blocks in each iteration. Since the RBCGD and PBCGD methods are inherently stochastic, we report the objective values averaged over 20 different runs. Moreover, for the RBCGD method, the block of coordinates is selected uniformly at random in each iteration. We consider three different settings: Setting (I) is the sparse linear regression, where the covariance matrix for generating the design matrix has $\Sigma_{jj} = 1$ and $\Sigma_{jk} = 0.5$ for any $k \neq j$; Setting (II) is the sparse linear regression, where the covariance matrix for generating the design matrix has $\Sigma_{jk} = 0.5^{|j-k|}$ for any j and k; Setting (III) is the sparse logistic regression, where the covariance matrix for generating the



(a) Setting (I) (b) Setting (II) (c) Setting (III)

Figure 1: Comparison among different methods under different settings. "RBCGD" and "PBCGD" denote the randomized BCD-type and permuted BCD-type methods respectively. The vertical axis corresponds to the gap towards the optimal objective value, $\log[\mathcal{F}(x) - \mathcal{F}(x^*)]$; the horizontal axis corresponds to the number of passes over p blocks of coordinates. Though all methods attain linear iteration complexity, their empirical behaviors are different from each others. Note that in plot (b) the curves for the CBCGD method and the RBCGD methods overlap.

design matrix is the same as Setting (II). Note that the condition number of the Hessian matrix depends on Σ. Setting (I) tends to yield a badly conditioned Hessian matrix whereas Settings (II) and (III) tend to yield well-conditioned Hessian matrices.

Figure 1 plots the gap between the objective value and the optimal as a function of number of passes for different methods. Our empirical findings can be summarized as follows: (1) All BCD-type methods attain better performance than the GD methods; (2) When the Hessian matrix is badly conditioned (i.e., in Setting (I)), the CBCGD performs worse than the RBCGD and PBCGD methods. (3) When the Hessian matrix is well conditioned (e.g., in Settings (II) and (III)), all three BCD-type methods attain good performance, and the CBCGD method slightly outperforms the PBCGD method; (4) The CBCGD method outperforms the RBCGD method in Setting (III).

5 Discussions

Existing literature has established an iteration complexity of $\mathcal{O}(\mu^{-1}L \cdot \log(1/\epsilon))$ for the gradient descent methods when solving strongly convex composite problems. However, our analysis shows that the CBCD-type methods only attains an iteration complexity of $\mathcal{O}(\mu^{-1}pL^2 \cdot \log(1/\epsilon))$. Even though our analysis further shows that the iteration complexity of the CBCD-type methods can be further improved to $\mathcal{O}(\mu^{-1}\log^2 pL^2 \cdot \log(1/\epsilon))$ for a quadratic $\mathcal{L}(\cdot)$, there still exists a gap of factor $L \log^2 p$. As our numerical experiments show, however, the CBCD-type methods can actually attain a better computational performance than the gradient methods regardless of whether $\mathcal{L}(\cdot)$ is quadratic or not, thereby suggesting that perhaps there is still room for improvement in the iteration complexity analysis of the CBCD-type methods.

It is also worth mentioning that though some literature claims that the CBCD-type methods works as well as the randomized BCD-type methods in practice, there do exist some counter examples, e.g. our experiment in Setting (I), where the CBCD-type methods fail significantly. This suggests that the CBCD-type methods do have some possible disadvantages in practice. To the best of our knowledge, we are not aware of any similar experimental results reported in existing literature.

Furthermore, our numerical results show that the permuted BCD-type methods, which can be viewed as a hybrid of the cyclic and the randomized BCD-type (RBCD-type) methods, has a stable performance irrespective of the problem being well conditioned or not. But to the best of our knowledge, no iteration complexity result has been established for the permuted BCD-type (PBCD-type) methods. We leave these problems for future investigation.

Acknowledgements

This research is supported by NSF DMS1454377-CAREER; NSF IIS 1546482-BIGDATA; NIH R01MH102339; NSF IIS1408910; NSF IIS1332109; NIH R01GM083084.

References

Angelos, J. R., Cowen, C. C. and Narayan, S. K. (1992). Triangular truncation and finding the norm of a hadamard multiplier. *Linear algebra and its applications* **170** 117–135.

Beck, A. and Tetruashvili, L. (2013). On the convergence of block coordinate descent type methods. *SIAM Journal on Optimization* **23** 2037–2060.

Boyd, S., Parikh, N., Chu, E., Peleato, B. and Eckstein, J. (2011). Distributed optimization and statistical learning via the alternating direction method of multipliers. *Foundations and Trends® in Machine Learning* **3** 1–122.

Friedman, J., Hastie, T., Höfling, H. and Tibshirani, R. (2007). Pathwise coordinate optimization. *The Annals of Applied Statistics* **1** 302–332.

Friedman, J., Hastie, T. and Tibshirani, R. (2010). Regularization paths for generalized linear models via coordinate descent. *Journal of statistical software* **33** 1–13.

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

Hastie, T., Tibshirani, R., Friedman, J., Hastie, T., Friedman, J. and Tibshirani, R. (2009). *The elements of statistical learning*. Springer.

He, B. and Yuan, X. (2012). On non-ergodic convergence rate of douglas-rachford alternating direction method of multipliers. Tech. rep., Nanjing University.

Hong, M., Wang, X., Razaviyayn, M. and Luo, Z.-Q. (2013). Iteration complexity analysis of block coordinate descent methods. *arXiv preprint arXiv:1310.6957* .

Li, X., Zhao, T., Yuan, X. and Liu, H. (2015a). The flare package for high dimensional linear regression and precision matrix estimation in R. *The Journal of Machine Learning Research* **16** 553–557.

Li, X., Zhao, T., Zhang, T. and Liu, H. (2015b). The picasso package for nonconvex regularized m-estimation in high dimensions in R. *Technical Report* .

Liu, H., Palatucci, M. and Zhang, J. (2009). Blockwise coordinate descent procedures for the multi-task lasso, with applications to neural semantic basis discovery. In *Proceedings of the 26th Annual International Conference on Machine Learning*.

Liu, H., Wang, L. and Zhao, T. (2014). Sparse covariance matrix estimation with eigenvalue constraints. *Journal of Computational and Graphical Statistics* **23** 439–459.

Liu, H., Wang, L. and Zhao, T. (2015). Calibrated multivariate regression with application to neural semantic basis discovery. *Journal of Machine Learning Research* **16** 1579–1606.

Lu, Z. and Xiao, L. (2015). On the complexity analysis of randomized block-coordinate descent methods. *Mathematical Programming* **152** 615–642.

Mazumder, R., Friedman, J. H. and Hastie, T. (2011). Sparsenet: Coordinate descent with nonconvex penalties. *Journal of the American Statistical Association* **106** 1125–1138.

Nesterov, Y. (2004). *Introductory lectures on convex optimization: A basic course*, vol. 87. Springer.

Nesterov, Y. (2007). Gradient methods for minimizing composite objective function. Tech. rep., Université catholique de Louvain, Center for Operations Research and Econometrics (CORE).

Richtárik, P. and Takáč, M. (2012). Iteration complexity of randomized block-coordinate descent methods for minimizing a composite function. *Mathematical Programming* 1–38.

Saha, A. and Tewari, A. (2013). On the nonasymptotic convergence of cyclic coordinate descent methods. *SIAM Journal on Optimization* **23** 576–601.

Schmidt, M. and Friedlander, M. (2015). Coordinate descent converges faster with the gauss-southwell rule than random selection. *arXiv preprint arXiv:1506.00552* .

Shalev-Shwartz, S. and Tewari, A. (2011). Stochastic methods for ℓ_1-regularized loss minimization. *The Journal of Machine Learning Research* **12** 1865–1892.

Shalev-Shwartz, S. and Zhang, T. (2014). Accelerated proximal stochastic dual coordinate ascent for regularized loss minimization. *Mathematical Programming* 1–41.

Tseng, P. (1993). Dual coordinate ascent methods for non-strictly convex minimization. *Mathematical programming* **59** 231–247.

Tseng, P. (2001). Convergence of a block coordinate descent method for nondifferentiable minimization. *Journal of optimization theory and applications* **109** 475–494.

Tseng, P. and Yun, S. (2009). A coordinate gradient descent method for nonsmooth separable minimization. *Mathematical Programming* **117** 387–423.

Vapnik, V. N. and Vapnik, V. (1998). *Statistical learning theory*, vol. 1. Wiley New York.

Zhao, T. and Liu, H. (2012). Sparse additive machine. In *International Conference on Artificial Intelligence and Statistics*.

Zhao, T. and Liu, H. (2015). Accelerated path-following iterative shrinkage thresholding algorithm with application to semiparametric graph estimation. *Journal of Computational and Graphical Statistics* .

Zhao, T., Liu, H., Roeder, K., Lafferty, J. and Wasserman, L. (2012). The huge package for high-dimensional undirected graph estimation in R. *The Journal of Machine Learning Research* **13** 1059–1062.

Zhao, T., Liu, H. and Zhang, T. (2014a). A general theory of pathwise coordinate optimization. *arXiv preprint arXiv:1412.7477* .

Zhao, T., Yu, M., Wang, Y., Arora, R. and Liu, H. (2014b). Accelerated mini-batch randomized block coordinate descent method. In *Advances in neural information processing systems*.

Zou, H. and Hastie, T. (2005). Regularization and variable selection via the elastic net. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)* **67** 301–320.